

SINUMERIK

SINUMERIK 840D sl / 828D Grundfunktionen

Funktionshandbuch

Gültig für

Steuerung
SINUMERIK 840D sl / 840DE sl / 828D

CNC-Software Version 4.8 SP1

05/2017
A5E40870716

Vorwort

Grundlegende Sicherheitshinweise	1
A2: Diverse NC/PLC-Nahtstellensignale und Funktionen	2
A3: Achsüberwachungen	3
A5: Schutzbereiche	4
B1: Bahnsteuerbetrieb, Genauhalt, LookAhead	5
B2: Beschleunigung	6
F1: Fahren auf Festanschlag	7
G2: Geschwindigkeiten, Soll-/Istwertsysteme, Regelung	8
H2: Hilfsfunktionsausgaben an PLC	9
K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten	10
K2: Achsen, Koordinatensysteme, Frames	11
N2: Not-Halt	12
P1: Planachsen	13
P3: PLC-Grundprogramm für SINUMERIK 840D sl	14
P4: PLC für SINUMERIK 828D	15
R1: Referenzieren	16
S1: Spindeln	17
V1: Vorschübe	18
W1: Werkzeugkorrektur	19
Z1: NC/PLC-Nahtstellensignale	20
Anhang	A

Rechtliche Hinweise

Warnhinweiskonzept

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

GEFAHR

bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

WARNUNG

bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

VORSICHT

bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG

bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

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

Qualifiziertes Personal

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

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

WARNUNG

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

Marken

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

Haftungsausschluss

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

Vorwort

SINUMERIK-Dokumentation

Die SINUMERIK-Dokumentation ist in folgende Kategorien gegliedert:

- Allgemeine Dokumentation/Kataloge
- Anwender-Dokumentation
- Hersteller-/Service-Dokumentation

Weiterführende Informationen

Unter folgender Adresse (<https://support.industry.siemens.com/cs/de/de/view/108464614>) finden Sie Informationen zu den Themen:

- Dokumentation bestellen/Druckschriftenübersicht
- Weiterführende Links für den Download von Dokumenten
- Dokumentation online nutzen (Handbücher/Informationen finden und durchsuchen)

Bei Fragen zur technischen Dokumentation (z. B. Anregungen, Korrekturen) senden Sie eine E-Mail an folgende Adresse (<mailto:docu.motioncontrol@siemens.com>).

mySupport/Dokumentation

Unter folgender Adresse (<https://support.industry.siemens.com/My/ww/de/documentation>) finden Sie Informationen, wie Sie Dokumentation auf Basis der Siemens Inhalte individuell zusammenstellen und für die eigene Maschinendokumentation anpassen.

Training

Unter folgender Adresse (<http://www.siemens.de/sitrain>) finden Sie Informationen zu SITRAIN - dem Training von Siemens für Produkte, Systeme und Lösungen der Antriebs- und Automatisierungstechnik.

FAQs

Frequently Asked Questions finden Sie in den Service&Support-Seiten unter Produkt Support (<https://support.industry.siemens.com/cs/de/de/ps/faq>).

SINUMERIK

Informationen zu SINUMERIK finden Sie unter folgender Adresse (<http://www.siemens.de/sinumerik>).

Zielgruppe

Die vorliegende Druckschrift wendet sich an:

- Projektueure
- Technologen (von Maschinenherstellern)
- Inbetriebnehmer (von Systemen/Maschinen)
- Programmierer

Nutzen

Das Funktionshandbuch beschreibt die Funktionen, so dass die Zielgruppe die Funktionen kennt und auswählen kann. Es befähigt die Zielgruppe, die Funktionen in Betrieb zu nehmen.

Standardumfang

In der vorliegenden Dokumentation ist die Funktionalität des Standardumfangs beschrieben. Ergänzungen oder Änderungen, die durch den Maschinenhersteller vorgenommen werden, werden vom Maschinenhersteller dokumentiert.

Es können in der Steuerung weitere, in dieser Dokumentation nicht erläuterte Funktionen ablauffähig sein. Es besteht jedoch kein Anspruch auf diese Funktionen bei der Neulieferung bzw. im Servicefall.

Ebenso enthält diese Dokumentation aus Gründen der Übersichtlichkeit nicht sämtliche Detailinformationen zu allen Typen des Produkts und kann auch nicht jeden denkbaren Fall der Aufstellung, des Betriebes und der Instandhaltung berücksichtigen.

Technical Support

Landesspezifische Telefonnummern für technische Beratung finden Sie im Internet unter folgender Adresse (<https://support.industry.siemens.com/sc/ww/de/sc/2090>) im Bereich "Kontakt".

Informationen zu Struktur und Inhalt

Aufbau

Das vorliegende Funktionshandbuch ist wie folgt aufgebaut:

- Innentitel (Seite 3) mit dem Titel des Funktionshandbuchs, den SINUMERIK-Steuerungen sowie der Software und Version, für die diese Ausgabe des Funktionshandbuchs gültig ist, und der Übersicht der einzelnen Funktionsbeschreibungen.
- Funktionsbeschreibungen in der Reihenfolge ihrer alphanumerischen Kurzzeichen (z. B. A2, A3, B1 etc.)

- Anhang mit:
 - Abkürzungsverzeichnis
 - Dokumentationsübersicht
- Begriffsverzeichnis

Hinweis

Ausführliche Daten- und Alarm-Beschreibungen siehe:

- für Maschinen- und Settingdaten:
Ausführliche Maschinendaten-Beschreibung
 - für NC/PLC-Nahtstellensignale:
Listenhandbuch NC-Variablen und Nahtstellensignale
 - für Alarme:
Diagnosehandbuch
-

Schreibweise von Systemdaten

Für Systemdaten gelten in dieser Dokumentation folgende Schreibweisen:

Signal/Datum	Schreibweise	Beispiel
NC/PLC-Nahtstellensignale	... NC/PLC-Nahtstellensignal: <Signaladresse> (<Signalname>)	Ist die neue Getriebestufe eingelegt, dann werden vom PLC-Programm die folgenden NC/PLC-Nahtstellensignale gesetzt: DB31, ... DBX16.0-2 (Istgetriebestufe A bis C) DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
Maschinendatum	... Maschinendatum: <Typ><Nummer> <Vollständiger Bezeichner> (<Bedeutung>)	Masterspindel ist die abgelegte Spindel im Maschinendatum: MD20090 \$MC_SPIND_DEF_MASTER_SPIND (Löschstellung Masterspindel im Kanal)
Settingdatum	... Settingdatum: <Typ><Nummer> <Vollständiger Bezeichner> (<Bedeutung>)	Die logische Masterspindel ist enthalten im Settingdatum: SD42800 \$SC_SPIND_ASSIGN_TAB[0] (Spindelnummernumsetzer)

Hinweis

Signaladresse

Die Funktionsbeschreibungen enthalten als <Signaladresse> eines NC/PLC-Nahtstellensignals nur die für SINUMERIK 840D sl gültige Adresse. Die Signaladresse für SINUMERIK 828D ist den Datenlisten "Signale an/von ..." am Ende der jeweiligen Funktionsbeschreibung zu entnehmen.

Mengengerüst

Erläuterungen bezüglich der NC/PLC-Nahtstelle gehen von der absoluten maximalen Anzahl folgender Komponenten aus:

- Betriebsartengruppen (DB11)
- Kanäle (DB21, ...)
- Achsen/Spindeln (DB31, ...)

Datentypen

In der Steuerung stehen zur Programmierung in Teileprogrammen folgenden Datentypen zur Verfügung:

Typ	Bedeutung	Wertebereich
INT	Ganzzahlige Werte mit Vorzeichen	-2.147.483.648 ... +2.147.483.647
REAL	Zahlen mit Dezimalpunkt	$\approx \pm 5,0 \cdot 10^{-324} \dots \approx \pm 1,7 \cdot 10^{308}$
BOOL	Wahrheitswerte	TRUE ($\neq 0$) , FALSE (0)
CHAR	Zeichen ASCII bzw. Byte	0 ... 255 bzw. -128 ... 127
STRING	Zeichenkette, Null-terminiert	maximal 400 Zeichen + /0 (keine Sonderzeichen)
AXIS	Achsamen	alle in der Steuerung vorhandenen Achsamen
FRAME	geometrische Angaben für Verschieben, Drehen, Skalieren, Spiegeln	---

Felder

Felder können nur aus gleichen elementaren Datentypen gebildet werden. Es sind maximal 3-dimensionale Arrays möglich.

Beispiel: `DEF INT FELD[2, 3, 4]`

Zahlensysteme

Es stehen folgende Zahlensysteme zur Verfügung:

- Dezimal: `DEF INT Zahl = 1234` oder `DEF REAL Zahl = 1234.56`
- Hexadezimal: `DEF INT Zahl = 'H123ABC'`
- Binär: `DEF INT Zahl = 'B10001010010'`

Abfrage von REAL-Variablen

Es wird empfohlen, die Abfragen von REAL- bzw. DOUBLE-Variablen in NC-Programmen und Synchronaktionen als Grenzwertbetrachtung zu programmieren.

Beispiel: Abfrage des Istwerts einer Achse auf einen bestimmten Wert

Programmcode	Kommentar
<code>DEF REAL AXPOS = 123.456</code>	
<code>IF (\$VA_IM[<Achse>] - 1ex-6) <= AXPOS <= (\$VA_IM[<Achse>] + 1ex-6)</code>	<code>; Istposition</code>
<code>...</code>	<code>== AXPOS</code>

Programmcode	Kommentar
ELSE	
...	<> AXPOS
ENDIF	

Inhaltsverzeichnis

	Vorwort.....	3
1	Grundlegende Sicherheitshinweise.....	37
1.1	Allgemeine Sicherheitshinweise.....	37
1.2	Industrial Security.....	38
2	A2: Diverse NC/PLC-Nahtstellensignale und Funktionen.....	39
2.1	Kurzbeschreibung.....	39
2.2	NC/PLC-Nahtstellensignale.....	39
2.2.1	Allgemeine Informationen.....	39
2.2.2	Bereitschaftssignale an PLC.....	41
2.2.3	Statussignale an PLC.....	41
2.2.4	Signale an/von Bedientafelfront.....	42
2.2.5	Signale an Kanal.....	44
2.2.6	Signale an Achse/Spindel.....	44
2.2.7	Signale von Achse/Spindel.....	53
2.2.8	Signale an Achse/Spindel (Digitale Antriebe).....	54
2.2.9	Signale von Achse/Spindel (Digitale Antriebe).....	55
2.3	Funktionen.....	57
2.3.1	Verhalten der Bildaktualisierung bei Überlastung - nur 840D sl.....	57
2.3.2	Einstellungen für Evolventen-Interpolation - nur 840D sl.....	58
2.3.3	DEFAULT-Speicher aktivieren - nur 840D sl.....	61
2.3.4	PLC-Variable lesen und schreiben - nur 840D sl.....	61
2.3.5	Zugriffsschutz über Kennwort und Schlüsselschalter.....	65
2.3.5.1	Kennwort.....	66
2.3.5.2	Schlüsselschalter-Stellungen (DB10, DBX56.4 bis 7).....	68
2.3.5.3	Parametrierbare Schutzstufen.....	69
2.3.6	Umschalten von Motor- / Antriebsdatensätzen.....	70
2.3.6.1	Allgemeine Informationen.....	70
2.3.6.2	Formatierungsschnittstellen.....	70
2.3.6.3	Anforderungsschnittstelle.....	71
2.3.6.4	Anzeigeschnittstelle.....	72
2.3.6.5	Beispiel.....	72
2.3.6.6	Übersicht der Schnittstellen.....	73
2.3.6.7	Randbedingungen.....	74
2.4	Beispiele.....	75
2.4.1	Parametersatzumschaltung.....	75
2.5	Datenlisten.....	77
2.5.1	Maschinendaten.....	77
2.5.1.1	Anzeige-Maschinendaten.....	77
2.5.1.2	NC-spezifischen Maschinendaten.....	77
2.5.1.3	Kanal-spezifische Maschinendaten.....	77
2.5.1.4	Achs-/Spindel-spezifische Maschinendaten.....	78
2.5.2	Systemvariablen.....	79

2.5.3	Signale.....	79
2.5.3.1	Signale an NC.....	79
2.5.3.2	Signale von NC.....	79
2.5.3.3	Signale an Bedientafelfront.....	80
2.5.3.4	Signale von Bedientafelfront.....	80
2.5.3.5	Signale an Kanal.....	81
2.5.3.6	Signale von Kanal.....	81
2.5.3.7	Signale an Achse/Spindel.....	81
2.5.3.8	Signale von Achse/Spindel.....	81
3	A3: Achsüberwachungen.....	83
3.1	Konturüberwachung.....	83
3.1.1	Konturfehler.....	83
3.1.2	Schleppabstandsüberwachung.....	84
3.2	Positionier-, Stillstands- und Klemmungsüberwachung.....	86
3.2.1	Zusammenhang zwischen Positionier-, Stillstands- und Klemmungsüberwachung.....	86
3.2.2	Positionierüberwachung.....	86
3.2.3	Stillstandsüberwachung.....	88
3.2.4	Parametersatzabhängige Genauhalt- und Stillstandstoleranz.....	89
3.2.5	Klemmungsüberwachung.....	89
3.2.5.1	Funktion.....	89
3.2.5.2	Maschinendaten.....	90
3.2.5.3	NC/PLC-Nahtstellensignale.....	90
3.2.5.4	Fehlerreaktionen.....	91
3.2.5.5	Klemmungsfunktion "Automatisches Anhalten zum Lösen der Klemmung".....	91
3.2.5.6	Klemmungsfunktion "Zeitoptimiertes Lösen der Klemmung".....	93
3.2.5.7	Klemmungsfunktion "Automatisches Anhalten zum Setzen der Klemmung".....	94
3.2.5.8	Randbedingungen.....	96
3.3	Drehzahlsollwertüberwachung.....	99
3.4	Istgeschwindigkeitsüberwachung.....	100
3.5	Messsystem-Überwachung.....	101
3.5.1	Gebergrenzfrequenzüberwachung.....	103
3.5.2	Plausibilitätskontrolle bei Absolutwertgebern.....	104
3.5.3	Anwenderspezifische Fehlerreaktionen.....	106
3.6	Endschalter-Überwachung.....	109
3.6.1	Hardware-Endschalter.....	109
3.6.2	Software-Endschalter.....	110
3.7	Arbeitsfeldbegrenzungs-Überwachung.....	112
3.7.1	Allgemein.....	112
3.7.2	Arbeitsfeldbegrenzung im BKS.....	114
3.7.3	Arbeitsfeldbegrenzung im WKS/ENS.....	116
3.7.4	Beispiel: Arbeitsfeldbegrenzung im WKS/ENS.....	120
3.8	Parken einer Maschinenachse.....	122
3.9	Parken des passiven Lagemesssystems.....	124
3.9.1	Funktion.....	124
3.9.2	Randbedingungen.....	127
3.9.3	Beispiel: Vorsatzkopfwechsel bei einem direkten Lagemesssystem.....	128
3.9.4	Beispiel: Vorsatzkopfwechsel bei zwei direkten Lagemesssystemen.....	133

3.9.5	Beispiel: Messsystemumschaltung bei bereichsweise fehlenden Gebern.....	137
3.10	Randbedingungen.....	139
3.11	Datenlisten.....	140
3.11.1	Maschinendaten.....	140
3.11.1.1	NC-spezifische Maschinendaten.....	140
3.11.1.2	Kanal-spezifische Maschinendaten.....	140
3.11.1.3	Achs-/Spindel-spezifische Maschinendaten.....	141
3.11.2	Settingdaten.....	142
3.11.2.1	Achs-/Spindel-spezifische Settindaten.....	142
3.11.3	Signale.....	142
3.11.3.1	Signale an Achse/Spindel.....	142
3.11.3.2	Signale von Achse/Spindel.....	142
4	A5: Schutzbereiche.....	143
4.1	Funktion.....	143
4.2	Inbetriebnahme.....	149
4.2.1	Maschinendaten.....	149
4.3	Programmierung.....	150
4.3.1	Schutzbereiche definieren (CPROTDEF, NPROTDEF).....	150
4.3.2	Schutzbereiche aktivieren/deaktivieren (CPROT, NPROT).....	154
4.3.3	Überprüfung auf Schutzbereichsverletzung, Arbeitsfeldbegrenzung und Software-Endschalter (CALCPOSI).....	157
4.4	Besonderheiten.....	167
4.4.1	Zeitweise Freigabe von Schutzbereichen.....	167
4.4.2	Verhalten in den Betriebsarten AUTOMATIK und MDA.....	168
4.4.3	Verhalten in der Betriebsart JOG.....	169
4.5	Randbedingungen.....	172
4.6	Beispiel.....	173
4.6.1	Schutzbereiche an einer Drehmaschine.....	173
4.6.2	Schutzbereichsdefinition im Teileprogramm.....	174
4.6.3	Schutzbereichsdefinition mit Systemvariablen.....	175
4.6.4	Aktivierung der Schutzbereiche.....	182
4.7	Datenlisten.....	182
4.7.1	Maschinendaten.....	182
4.7.1.1	NC-spezifische Maschinendaten.....	182
4.7.1.2	Kanal-spezifische Maschinendaten.....	183
4.7.2	Signale.....	183
4.7.2.1	Signale an Kanal.....	183
4.7.2.2	Signale von Kanal.....	183
5	B1: Bahnsteuerbetrieb, Genauhalt, LookAhead.....	185
5.1	Kurzbeschreibung.....	185
5.2	Genauhaltbetrieb.....	188
5.3	Bahnsteuerbetrieb.....	192
5.3.1	Allgemeine Funktionalität.....	192
5.3.2	Geschwindigkeitsabsenkung gemäß Überlastfaktor.....	193
5.3.3	Überschleifen.....	195

5.3.3.1	Überschleifen nach Wegkriterium (G641).....	198
5.3.3.2	Überschleifen unter Einhaltung definierter Toleranzen (G642/G643).....	200
5.3.3.3	Überschleifen mit maximal möglicher Achsdynamik (G644).....	203
5.3.3.4	Überschleifen tangentialer Satzübergänge (G645).....	206
5.3.3.5	Überschleifen und Repositionieren (REPOS).....	207
5.3.4	LookAhead.....	208
5.3.4.1	Standardfunktionalität.....	208
5.3.4.2	Freiformflächenmodus: Erweiterungsfunktion.....	214
5.4	Dynamikanpassungen.....	218
5.4.1	Glättung der Bahngeschwindigkeit.....	218
5.4.2	Anpassung der Bahndynamik.....	222
5.4.3	Ermittlung der Dynamikgrenzwerte.....	226
5.4.4	Zusammenwirken der Funktionen "Glättung der Bahngeschwindigkeit" und "Anpassung der Bahndynamik".....	226
5.4.5	Dynamikmodus für Bahninterpolation.....	230
5.4.6	Freiformflächenmodus: Grundfunktionen.....	232
5.5	Kompressor-Funktionen.....	236
5.5.1	Kompression von Linear-, Kreis- und Eilgangsätzen.....	236
5.5.1.1	Funktion.....	236
5.5.1.2	Inbetriebnahme.....	238
5.5.1.3	Programmierung.....	247
5.5.1.4	Randbedingungen.....	248
5.5.2	Kompression kurzer Spline-Sätze.....	248
5.6	Kontur-/Orientierungstoleranz.....	250
5.6.1	Inbetriebnahme.....	250
5.6.1.1	Parametrierung.....	250
5.6.2	Programmierung.....	251
5.6.2.1	Kontur-/Orientierungstoleranz programmieren (CTOL, OTOL, ATOL).....	251
5.6.2.2	Kontur-/Orientierungstoleranz programmieren (CTOL, OTOL, ATOL): Weitere Informationen.....	254
5.7	Toleranzfaktor für G0-Bewegungen.....	256
5.7.1	Funktion.....	256
5.7.2	Inbetriebnahme.....	256
5.7.2.1	Parametrierung.....	256
5.7.3	Programmierung.....	257
5.7.3.1	Toleranzfaktor für G0-Bewegungen programmieren (STOLF).....	257
5.8	RESET-Verhalten.....	258
5.9	Randbedingungen.....	259
5.9.1	Satzwechsel und Positionierachsen.....	259
5.9.2	Satzwechselverzögerung.....	259
5.10	Datenlisten.....	260
5.10.1	Maschinendaten.....	260
5.10.1.1	Allgemeine Maschinendaten.....	260
5.10.1.2	Kanal-spezifische Maschinendaten.....	260
5.10.1.3	Achs-/Spindel-spezifische Maschinendaten.....	261
5.10.2	Settingdaten.....	262
5.10.2.1	Kanal-spezifische Settingdaten.....	262
5.10.3	Signale.....	262
5.10.3.1	Signale von Kanal.....	262

5.10.3.2	Signale von Achse/Spindel.....	263
6	B2: Beschleunigung.....	265
6.1	Kurzbeschreibung.....	265
6.1.1	Allgemeine Informationen.....	265
6.1.2	Merkmale.....	265
6.2	Funktionen.....	267
6.2.1	Beschleunigung ohne Ruckbegrenzung (BRISK/BRISKA) (kanal-/achsspezifisch).....	267
6.2.1.1	Allgemeine Informationen.....	267
6.2.1.2	Parametrierung.....	268
6.2.1.3	Programmierung.....	269
6.2.2	Konstantfahrzeit (kanalspezifisch).....	270
6.2.2.1	Allgemeine Informationen.....	270
6.2.2.2	Parametrierung.....	271
6.2.3	Beschleunigungsanpassung (ACC) (achsspezifisch).....	272
6.2.3.1	Allgemeine Informationen.....	272
6.2.3.2	Programmierung.....	272
6.2.4	Beschleunigungsreserve (kanalspezifisch).....	273
6.2.4.1	Allgemeine Informationen.....	273
6.2.4.2	Parametrierung.....	273
6.2.5	Begrenzung der Bahnbeschleunigung (kanalspezifisch).....	274
6.2.5.1	Allgemeine Informationen.....	274
6.2.5.2	Parametrierung.....	274
6.2.5.3	Programmierung.....	274
6.2.6	Bahnbeschleunigung für Echtzeitereignisse (kanalspezifisch).....	275
6.2.6.1	Allgemeine Informationen.....	275
6.2.6.2	Programmierung.....	276
6.2.7	Beschleunigung bei programmiertem Eilgang (G00) (achsspezifisch).....	277
6.2.7.1	Allgemeine Informationen.....	277
6.2.7.2	Parametrierung.....	277
6.2.8	Beschleunigung bei aktiver Ruckbegrenzung (SOFT/SOFTA) (achsspezifisch).....	278
6.2.8.1	Allgemeine Informationen.....	278
6.2.8.2	Parametrierung.....	278
6.2.9	Beschleunigungsüberhöhung bei nicht tangentialen Satzübergängen (achsspezifisch).	279
6.2.9.1	Allgemeine Informationen.....	279
6.2.9.2	Parametrierung.....	279
6.2.10	Beschleunigungsreserve für die Radialbeschleunigung (kanalspezifisch).....	279
6.2.10.1	Allgemeine Informationen.....	279
6.2.10.2	Parametrierung.....	281
6.2.11	Ruckbegrenzung bei Bahninterpolation (SOFT) (kanalspezifisch).....	281
6.2.11.1	Allgemeine Informationen.....	281
6.2.11.2	Parametrierung.....	283
6.2.11.3	Programmierung.....	283
6.2.12	Ruckbegrenzung bei Einzelachsinterpolation (SOFTA) (achsspezifisch).....	284
6.2.12.1	Parametrierung.....	284
6.2.12.2	Programmierung.....	285
6.2.13	Begrenzung des Bahnruks (kanalspezifisch).....	285
6.2.13.1	Allgemeine Informationen.....	285
6.2.13.2	Parametrierung.....	286
6.2.13.3	Programmierung.....	286
6.2.14	Bahnruks für Echtzeitereignisse (kanalspezifisch).....	287
6.2.14.1	Allgemeine Informationen.....	287

6.2.14.2	Programmierung.....	288
6.2.15	Ruck bei programmiertem Eilgang (G00) (achsspezifisch).....	289
6.2.15.1	Allgemeine Informationen.....	289
6.2.15.2	Parametrierung.....	289
6.2.16	Rucküberhöhung bei nicht krümmungsstetigen Satzübergängen (achsspezifisch).....	290
6.2.16.1	Allgemeine Informationen.....	290
6.2.16.2	Parametrierung.....	290
6.2.17	Geschwindigkeitsabhängige Ruckadaption (achsspezifisch).....	290
6.2.18	Ruckfilter (achsspezifisch).....	292
6.2.18.1	Allgemeine Informationen.....	292
6.2.18.2	Parametrierung.....	295
6.2.19	Geknickte Beschleunigungskennlinie.....	296
6.2.19.1	Anpassung an die Motorkennlinie.....	296
6.2.19.2	Auswirkungen auf die Bahnbeschleunigung.....	298
6.2.19.3	Ersatzkennlinie.....	298
6.2.19.4	Parametrierung.....	301
6.2.19.5	Programmierung.....	302
6.2.19.6	Randbedingungen.....	303
6.2.20	Beschleunigung und Ruck bei JOG-Bewegungen.....	304
6.2.20.1	Parametrierung.....	304
6.2.20.2	Randbedingungen.....	306
6.3	Beispiele.....	307
6.3.1	Beschleunigung.....	307
6.3.1.1	Bahngeschwindigkeitsverlauf.....	307
6.3.2	Ruck.....	308
6.3.2.1	Bahngeschwindigkeitsverlauf.....	308
6.3.3	Beschleunigung und Ruck.....	310
6.3.4	Geknickte Beschleunigungskennlinie.....	311
6.3.4.1	Aktivierung.....	311
6.4	Datenlisten.....	312
6.4.1	Maschinendaten.....	312
6.4.1.1	NC-spezifische Maschinendaten.....	312
6.4.1.2	Kanal-spezifische Maschinendaten.....	312
6.4.1.3	Achs-/Spindel-spezifische Maschinendaten.....	313
6.4.2	Settingdaten.....	314
6.4.2.1	Kanal-spezifische Settingdaten.....	314
6.4.3	Systemvariablen.....	314
7	F1: Fahren auf Festanschlag.....	315
7.1	Kurzbeschreibung.....	315
7.2	Ausführliche Beschreibung.....	316
7.2.1	Programmierung.....	316
7.2.2	Funktionsablauf.....	318
7.2.2.1	Anwahl.....	318
7.2.2.2	Festanschlag wird erreicht.....	319
7.2.2.3	Festanschlag wird nicht erreicht.....	321
7.2.2.4	Abwahl.....	322
7.2.3	Verhalten bei Satzsuchlauf.....	324
7.2.4	Verhalten bei Reset und Funktionsabbruch.....	327
7.2.5	Verhalten bezüglich anderer Funktionen.....	328
7.2.6	Settingdaten.....	329

7.2.7	Systemvariablen.....	330
7.2.8	Alarmer.....	332
7.2.9	Fahren mit begrenztem Moment/Kraft FOC.....	333
7.3	Beispiele.....	336
7.4	Datenlisten.....	337
7.4.1	Maschinendaten.....	337
7.4.1.1	Achs-/Spindel-spezifische Maschinendaten.....	337
7.4.2	Settingdaten.....	338
7.4.2.1	Achs-/Spindel-spezifische Settingdaten.....	338
7.4.3	Signale.....	338
7.4.3.1	Signale an Achse/Spindel.....	338
7.4.3.2	Signale von Achse/Spindel.....	339
8	G2: Geschwindigkeiten, Soll-/Istwertsysteme, Regelung.....	341
8.1	Kurzbeschreibung.....	341
8.2	Geschwindigkeiten, Verfahrbereiche, Genauigkeiten.....	341
8.2.1	Geschwindigkeiten.....	341
8.2.2	Verfahrbereiche.....	343
8.2.3	Positioniergenauigkeit der Steuerung	344
8.2.4	Eingabe-/Anzeigeinheit, Rechenfeinheit.....	344
8.2.5	Normierung physikalischer Größen der Maschinen- und Settingdaten.....	346
8.3	Maßsystem inch/metrisch.....	349
8.3.1	Funktion.....	349
8.3.1.1	Parametriertes und Programmirtes Maßsystem.....	349
8.3.1.2	Erweiterte Maßsystem-Funktionalität.....	350
8.3.1.3	Maßsystem-Umschaltung über die Bedienoberfläche.....	351
8.3.2	Inbetriebnahme.....	353
8.3.3	Programmierung.....	357
8.3.3.1	Maßsystem umschalten (G70/G71/G700/G710).....	357
8.4	Soll-/Istwertsystem.....	360
8.4.1	Allgemeine Informationen.....	360
8.4.2	Sollwert- und Geberzuordnung.....	363
8.4.3	Anpassungen der Motor/Last-Verhältnisse.....	366
8.4.4	Drehzahlsollwertausgabe.....	369
8.4.5	Maschinendaten des Istwertsystems.....	370
8.4.6	Istwertauflösung.....	371
8.4.6.1	Maschinendaten der Istwertauflösung.....	371
8.4.6.2	Beispiel: Linearachse mit Linearmaßstab.....	374
8.4.6.3	Beispiel: Linearachse mit rotatorischem Geber am Motor.....	375
8.4.6.4	Beispiel: Linearachse mit rotatorischem Geber an der Maschine.....	376
8.4.6.5	Beispiel: Rundachse mit rotatorischem Geber am Motor.....	377
8.4.6.6	Beispiel: Rundachse mit rotatorischem Geber an der Maschine.....	378
8.4.6.7	Beispiel: Vorsatz-Getriebe mit Geber am Werkzeug.....	379
8.5	Regelung.....	380
8.5.1	Allgemeine Informationen.....	380
8.5.2	Parametersätze des Lagereglers.....	383
8.6	Optimierung der Regelung.....	384
8.6.1	Lageregler Lagesollwertfilter: Symmetrierfilter.....	384
8.6.2	Lageregler Lagesollwertfilter: Ruckfilter.....	388

8.6.3	Lageregler Lagesollwertfilter: Phasenfilter.....	390
8.6.4	Lageregler: Lagedifferenz-Aufschaltung.....	392
8.6.5	Lageregelung mit PI-Regler.....	393
8.7	Datenlisten.....	395
8.7.1	Maschinendaten.....	395
8.7.1.1	Anzeige-Maschinendaten.....	395
8.7.1.2	NC-spezifische Maschinendaten.....	396
8.7.1.3	Kanal-spezifische Maschinendaten.....	396
8.7.1.4	Achs-/Spindel-spezifische Maschinendaten.....	396
8.7.2	Signale.....	398
8.7.2.1	Signale von NC.....	398
8.7.2.2	Signale an Achse/Spindel.....	398
9	H2: Hilfsfunktionsausgaben an PLC.....	399
9.1	Kurzbeschreibung.....	399
9.1.1	Funktion.....	399
9.1.2	Definition einer Hilfsfunktion.....	400
9.1.3	Übersicht der Hilfsfunktionen.....	400
9.2	Vordefinierte Hilfsfunktionen.....	406
9.2.1	Übersicht: vordefinierten Hilfsfunktionen.....	407
9.2.2	Übersicht: Ausgabeverhalten.....	417
9.2.3	Parametrierung.....	421
9.2.3.1	Gruppenzuordnung.....	421
9.2.3.2	Typ, Adresserweiterung und Wert.....	421
9.2.3.3	Ausgabeverhalten.....	422
9.3	Anwenderdefinierte Hilfsfunktionen.....	426
9.3.1	Parametrierung.....	428
9.3.1.1	Maximale Anzahl anwenderdefinierter Hilfsfunktionen.....	428
9.3.1.2	Gruppenzuordnung.....	428
9.3.1.3	Typ, Adresserweiterung und Wert.....	429
9.3.1.4	Ausgabeverhalten.....	430
9.4	Assoziierte Hilfsfunktionen.....	430
9.5	Typ-spezifisches Ausgabeverhalten.....	432
9.6	Prioritäten des parametrierten Ausgabeverhaltens.....	433
9.7	Programmierung einer Hilfsfunktion.....	434
9.8	Programmierbare Ausgabedauer.....	436
9.9	Hilfsfunktionsausgabe an die PLC.....	437
9.10	Hilfsfunktionen ohne Satzwechselverzögerung.....	438
9.11	M-Funktion mit implizitem Vorlaufstopp.....	439
9.12	Verhalten bei Überspeichern.....	440
9.13	Verhalten bei Satzsuchlauf.....	441
9.13.1	Hilfsfunktionsausgabe bei Satzsuchlauf Typ 1,2 und 4.....	441
9.13.2	Zuordnung einer Hilfsfunktion zu mehreren Gruppen.....	442
9.13.3	Zeitstempel der aktiven M-Hilfsfunktion.....	444
9.13.4	Ermittlung der Ausgabe-Reihenfolge.....	444
9.13.5	Ausgabeunterdrückung von Spindel-spezifischen Hilfsfunktionen.....	446

9.13.6	Hilfsfunktionsausgabe bei Satzsuchlauf Typ 5 (SERUPRO).....	449
9.13.7	SERUPRO-Ende-ASUP.....	454
9.14	Implizit ausgegebene Hilfsfunktionen.....	460
9.15	Informationsmöglichkeiten.....	461
9.15.1	Gruppenspezifische modale M-Hilfsfunktionsanzeige.....	461
9.15.2	Abfrage von Systemvariablen.....	462
9.16	Randbedingungen.....	464
9.16.1	Allgemeine Randbedingungen.....	464
9.16.2	Ausgabeverhalten.....	465
9.17	Beispiele.....	466
9.17.1	Erweiterung von vordefinierten Hilfsfunktionen.....	466
9.17.2	Definition von Hilfsfunktionen.....	468
9.18	Datenlisten.....	472
9.18.1	Maschinendaten.....	472
9.18.1.1	NC-spezifische Maschinendaten.....	472
9.18.1.2	Kanal-spezifische Maschinendaten.....	472
9.18.2	Signale.....	473
9.18.2.1	Signale an Kanal.....	473
9.18.2.2	Signale von Kanal.....	473
9.18.2.3	Signale an Achse/Spindel.....	475
9.18.2.4	Signale von Achse/Spindel.....	476
10	K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten.....	477
10.1	Kurzbeschreibung.....	477
10.2	Betriebsartengruppe (BAG).....	480
10.2.1	BAG-Stop.....	482
10.2.2	BAG-Reset.....	482
10.3	Betriebsarten und Betriebsartenwechsel.....	483
10.3.1	Überwachungen und Verriegelungen der einzelnen Betriebsarten.....	488
10.3.2	Betriebsartenwechsel.....	489
10.4	Kanal.....	490
10.4.1	Startsperr, global und kanalspezifisch.....	493
10.5	Programmtest.....	494
10.5.1	Programmbearbeitung ohne Sollwertausgaben.....	494
10.5.2	Programmbearbeitung im Einzelsatzbetrieb.....	496
10.5.3	Programmbearbeitung mit Probelaufvorschub.....	498
10.5.4	Teileprogrammsätze ausblenden.....	501
10.6	Werkstücksimulation.....	502
10.7	Satzsuchlauf Typ 1, 2 und 4.....	503
10.7.1	Funktionsbeschreibung.....	504
10.7.2	Satzsuchlauf im Zusammenhang mit weiteren NCK-Funktionen.....	507
10.7.2.1	ASUP nach und bei Satzsuchlauf.....	507
10.7.2.2	PLC-Aktionen nach Satzsuchlauf.....	508
10.7.2.3	Spindelfunktionen nach Satzsuchlauf.....	508
10.7.2.4	Lesen von Systemvariablen bei Satzsuchlauf.....	509
10.7.3	Automatischer Start eines ASUP nach Satzsuchlauf.....	510
10.7.4	Kaskadierter Satzsuchlauf.....	512

10.7.5	Beispiele zum Satzsuchlauf mit Berechnung.....	513
10.7.6	Randbedingungen.....	517
10.7.6.1	Kompressor-Funktionen (COMPON, COMPCURV, COMPCAD).....	517
10.8	Satzsuchlauf Typ 5 (SERUPRO).....	517
10.8.1	Funktionsbeschreibung.....	517
10.8.2	Wiederanfahren an die Kontur (REPOS).....	521
10.8.2.1	Wiederanfahren an die Kontur mit gesteuertem REPOS.....	530
10.8.3	Suchlaufvorgang beschleunigen.....	531
10.8.4	SERUPRO-ASUP.....	533
10.8.5	Self-Acting SERUPRO.....	536
10.8.6	Programmabschnitt für Wiederaufsetzen sperren.....	537
10.8.7	Verhalten bei Power On, Betriebsartenwechsel und RESET.....	541
10.8.8	Randbedingungen.....	541
10.8.8.1	STOPRE im Zielsatz.....	541
10.8.8.2	SPOS im Zielsatz.....	542
10.8.8.3	Fahren auf Festanschlag (FXS).....	542
10.8.8.4	Fahren mit begrenztem Moment/Kraft (FOC).....	543
10.8.8.5	Synchronspindel.....	544
10.8.8.6	Kopplungen und Master-Slave.....	544
10.8.8.7	Achsfunktionen.....	547
10.8.8.8	Getriebestufenwechsel.....	549
10.8.8.9	Überlagerte Bewegung.....	549
10.8.8.10	NC/PLC-Nahtstellensignale.....	550
10.8.8.11	Flexibilisierung der Grundeinstellung.....	550
10.8.8.12	Kompressor-Funktionen (COMPON, COMPCURV, COMPCAD).....	551
10.8.9	Systemvariable.....	551
10.9	Programmbetrieb.....	551
10.9.1	Grundstellungen.....	552
10.9.1.1	Maschinendaten.....	553
10.9.1.2	Programmierung.....	553
10.9.2	Anwahl und Start eines NC-Programms.....	557
10.9.3	Programmunterbrechung.....	559
10.9.4	Kanal-Reset.....	560
10.9.5	Programmzustand.....	562
10.9.6	Kanalzustand.....	563
10.9.7	Reaktionen auf Bedienungs- und Programmaktionen.....	564
10.9.8	Zeitdiagramm-Beispiel für einen Programmablauf.....	565
10.9.9	Programmsprünge.....	565
10.9.9.1	Rücksprung auf Programmanfang (GOTOS).....	565
10.9.10	Programmteilwiederholungen.....	568
10.9.10.1	Programmierung.....	568
10.9.11	Ereignisgesteuerter Programmaufruf (PROG_EVENT).....	573
10.9.11.1	Funktion.....	573
10.9.11.2	Parametrierung.....	577
10.9.11.3	Programmierung.....	581
10.9.11.4	Randbedingungen.....	582
10.9.11.5	Beispiele.....	583
10.9.12	Beeinflussung von Stopp-Ereignissen durch Stop-Delay-Bereiche.....	584
10.9.12.1	Funktion.....	584
10.9.12.2	Parametrierung.....	588
10.9.12.3	Programmierung.....	588

10.9.12.4	Randbedingungen.....	590
10.10	Asynchrone Unterprogramme (ASUPs).....	591
10.10.1	Funktion.....	591
10.10.1.1	Ablauf eines ASUPs im Programmbetrieb.....	594
10.10.1.2	ASUP mit REPOSA.....	595
10.10.1.3	NC-Verhalten.....	596
10.10.2	Inbetriebnahme: Maschinendaten.....	597
10.10.2.1	NC-spez.: BAG-spezifische NC/PLC-Nahtstellensignale und Betriebsartenumschaltung.....	597
10.10.2.2	NC-spez.: ASUP Startfreigabe.....	598
10.10.2.3	NC-spez.: Wirksamkeit der parametrisierten Startfreigaben.....	599
10.10.2.4	Kanalspez.: Startfreigabe trotz nicht referenzierter Achsen.....	599
10.10.2.5	Kanalspez.: Startfreigabe trotz Einlesesperre.....	599
10.10.2.6	Kanalspez.: Kontinuierliche Abarbeitung trotz Einzelsatz.....	600
10.10.2.7	Kanalspez.: Aktualisierung der Anzeige.....	600
10.10.3	Programmierung: Systemvariablen.....	601
10.10.3.1	REPOS-Möglichkeit (\$P_REPINF).....	601
10.10.3.2	Aktivierungsereignis (\$AC_ASUP).....	602
10.10.4	Programmierung (SETINT, PRIO).....	602
10.10.5	Randbedingungen.....	604
10.10.6	Beispiele.....	605
10.11	Anwenderspezifisches ASUP für RET und REPOS.....	606
10.11.1	Funktion.....	606
10.11.2	Parametrierung.....	606
10.11.3	Programmierung.....	607
10.12	ASUP-Start bei Anwenderalarmen durchführen.....	608
10.12.1	Funktion.....	608
10.12.2	Aktivierung.....	609
10.12.3	Beispiele.....	610
10.12.3.1	Anwender-ASUP aus Reset - Beispiel 1.....	610
10.12.3.2	Anwender-ASUP aus Reset - Beispiel 2.....	611
10.12.3.3	Anwender-ASUP mit M0.....	612
10.12.3.4	Anwender-ASUP mit Stopp.....	613
10.12.3.5	Anwender-ASUP aus gestoppt.....	614
10.13	Einzelsatz.....	615
10.13.1	Parametrierung.....	616
10.13.2	Programmierung.....	618
10.13.2.1	Einzelsatzbearbeitung aus/einschalten (SBLOF, SBLON).....	618
10.13.2.2	Randbedingungen.....	619
10.13.2.3	Beispiele.....	619
10.13.3	BAG-spezifischer Einzelsatz Typ A / B.....	622
10.13.4	Randbedingungen.....	623
10.13.4.1	Einzelsatztyp SBL2 und satzbezogene Synchronaktionen.....	623
10.13.4.2	Programmierter Halt (M0), Einzelsatz und Einzelsatztypumschaltung.....	623
10.14	Programmbeeinflussung.....	624
10.14.1	Funktionenwahl über Bedienoberfläche oder PLC-Anwenderprogramm.....	624
10.14.2	Aktivierung von Ausblendeblenden.....	625
10.14.3	Größenanpassung des Interpolationspuffers.....	626
10.14.4	Darstellungsweise der Programmanzeige über eine zusätzliche Basis-Satzanzeige.....	628
10.14.5	Basis-Satzanzeige bei ShopMill/ShopTurn.....	629
10.14.6	Aufbau für einen DIN-Satz.....	631

10.14.7	Abarbeiten von Extern.....	634
10.14.8	Abarbeiten von externen Unterprogrammen (EXTCALL).....	636
10.15	Abarbeiten vom externen Speicher (EES) (Option).....	639
10.15.1	Funktion.....	639
10.15.2	Inbetriebnahme.....	642
10.15.2.1	Projektierung der Laufwerke.....	642
10.15.2.2	Globaler Teileprogrammspeicher (GDIR).....	644
10.15.2.3	Einstellungen für Dateihandling im Teileprogramm bei EES.....	646
10.15.2.4	Speicherkonfiguration.....	647
10.15.3	Randbedingungen.....	648
10.16	Process DataShare - Ausgabe auf ein externes Gerät/Datei.....	648
10.16.1	Funktion.....	648
10.16.2	Inbetriebnahme.....	650
10.16.3	Programmierung.....	653
10.16.4	Randbedingungen.....	657
10.17	Systemeinstellungen für Hochlauf, RESET/Teileprogrammende und Teileprogramm-Start.....	658
10.17.1	Werkzeugrückzug nach POWER ON mit Orientierungstransformation.....	662
10.18	Ersetzung von Funktionen durch Unterprogramme.....	665
10.18.1	Übersicht.....	665
10.18.2	Ersetzung von M-, T/TCA- und D/DL-Funktionen.....	666
10.18.2.1	Ersetzung von M-Funktionen.....	666
10.18.2.2	Ersetzung von T/TCA- und D/DL-Funktionen.....	668
10.18.2.3	Systemvariable.....	671
10.18.2.4	Beispiel: Ersetzung einer M-Funktion.....	672
10.18.2.5	Beispiel: Ersetzung einer T- und D-Funktion.....	675
10.18.2.6	Verhalten im Konfliktfall.....	676
10.18.3	Ersetzung von Spindel-Funktionen.....	677
10.18.3.1	Allgemeine Informationen.....	677
10.18.3.2	Ersetzung von M40 - M45 (Getriebestufenwechsel).....	679
10.18.3.3	Ersetzung von SPOS, SPOSA, M19 (Spindelpositionieren).....	679
10.18.3.4	Systemvariable.....	680
10.18.3.5	Beispiel: Getriebestufenwechsel.....	681
10.18.3.6	Beispiel: Spindelpositionieren.....	683
10.18.4	Eigenschaften der Unterprogramme.....	686
10.18.5	Randbedingungen.....	688
10.19	Umbenennung/Sperrung von NC-Befehlen.....	688
10.20	Programmlaufzeit / Werkstückzähler.....	691
10.20.1	Programmlaufzeit.....	691
10.20.1.1	Funktion.....	691
10.20.1.2	Inbetriebnahme.....	694
10.20.1.3	Randbedingungen.....	696
10.20.1.4	Beispiele.....	696
10.20.2	Werkstückzähler.....	697
10.20.2.1	Funktion.....	697
10.20.2.2	Inbetriebnahme.....	698
10.20.2.3	Randbedingungen.....	700
10.20.2.4	Beispiele.....	700
10.21	Datenlisten.....	702

10.21.1	Funktion.....	702
10.21.2	Maschinendaten.....	703
10.21.2.1	Allgemeine Maschinendaten.....	703
10.21.2.2	Kanal-spezifische Maschinendaten.....	704
10.21.2.3	Achs-/Spindel-spezifische Maschinendaten.....	707
10.21.3	Settingdaten.....	708
10.21.3.1	Kanal-spezifische Settingdaten.....	708
10.21.4	Signale.....	708
10.21.4.1	Signale an NC.....	708
10.21.4.2	Signale an BAG.....	708
10.21.4.3	Signale von BAG.....	709
10.21.4.4	Signale an Kanal.....	709
10.21.4.5	Signale von Kanal.....	710
10.21.4.6	Signale an Achse/Spindel.....	711
10.21.4.7	Signale von Achse/Spindel.....	711
11	K2: Achsen, Koordinatensysteme, Frames.....	713
11.1	Kurzbeschreibung.....	713
11.1.1	Achsen.....	713
11.1.2	Koordinatensysteme.....	715
11.1.3	Frames.....	716
11.2	Achsen.....	720
11.2.1	Übersicht.....	720
11.2.2	Maschinenachsen.....	721
11.2.3	Kanalachsen.....	722
11.2.4	Geometrieachsen.....	722
11.2.5	Zusatzachsen.....	722
11.2.6	Bahnachsen.....	723
11.2.7	Positionierachsen.....	723
11.2.8	Hauptlaufachsen.....	724
11.2.9	Synchronachsen.....	726
11.2.10	Achskonfiguration.....	728
11.2.11	Link-Achsen.....	730
11.3	Nullpunkte und Referenzpunkte.....	731
11.3.1	Bezugspunkte im Arbeitsraum.....	731
11.3.2	Lage der Koordinatensysteme und Referenzpunkte.....	733
11.4	Koordinatensysteme.....	734
11.4.1	Übersicht.....	734
11.4.2	Maschinenkoordinatensystem (MKS).....	737
11.4.2.1	Istwertsetzen mit Verlust des Referenzierstatus (PRESETON).....	738
11.4.2.2	Istwertsetzen ohne Verlust des Referenzierstatus (PRESETONS).....	743
11.4.3	Basiskoordinatensystem (BKS).....	748
11.4.4	Additive Korrekturen.....	750
11.4.4.1	Externe Nullpunktverschiebungen.....	750
11.4.4.2	DRF-Verschiebung.....	752
11.4.4.3	Überlagerte Bewegungen.....	752
11.4.4.4	Reset-Verhalten.....	753
11.4.5	Basis-Nullpunktsystem (BNS).....	754
11.4.6	Einstellbares Nullpunktsystem (ENS).....	756
11.4.7	Werkstückkoordinatensystem (WKS).....	757

11.5	Frames.....	758
11.5.1	Frame-Arten.....	758
11.5.2	Frame-Komponenten.....	759
11.5.2.1	Translation.....	759
11.5.2.2	Feinverschiebung.....	760
11.5.2.3	Drehung: Übersicht (nur Geometrieachsen).....	761
11.5.2.4	Drehung mit Euler-Winkeln: ZY'X"-Konvention (RPY-Winkel).....	762
11.5.2.5	Drehung mit Euler-Winkeln: ZX'Z"-Konvention.....	767
11.5.2.6	Drehung in beliebiger Ebene.....	768
11.5.2.7	Skalierung.....	769
11.5.2.8	Spiegelung.....	770
11.5.2.9	Verkettungsoperator.....	770
11.5.2.10	Programmierbare Achsname.....	770
11.5.2.11	Koordinatentransformation.....	772
11.5.3	Datenhaltungs-Frames und aktive Frames.....	772
11.5.3.1	Übersicht.....	772
11.5.3.2	Aktivierung von Datenhaltungsframes.....	774
11.5.3.3	NCU-lobale und kanalspezifische Frames.....	775
11.5.4	Framekette und Koordinatensysteme.....	776
11.5.4.1	Übersicht.....	776
11.5.4.2	Relative Koordinatensysteme.....	777
11.5.4.3	Wählbares ENS.....	779
11.5.4.4	Manuelle Verfahren von Geometrieachsen wahlweise im WKS oder ENS (\$AC_JOG_COORD).....	780
11.5.4.5	Unterdrückung von Frames.....	781
11.5.5	Frames der Framekette.....	782
11.5.5.1	Übersicht.....	782
11.5.5.2	Einstellbare Frames \$P_UIFR[<n>].....	783
11.5.5.3	Schleifframes \$P_GFR[<n>].....	785
11.5.5.4	Kanal-spezifische Basisframes[<n>].....	789
11.5.5.5	NCU-globale Basisframes \$P_NCBFR[<n>].....	790
11.5.5.6	Aktiver Gesamt-Basisframe \$P_ACTBFRAME.....	791
11.5.5.7	Programmierbarer Frame \$P_PFRAME.....	793
11.5.5.8	Kanalspezifische Systemframes.....	795
11.5.6	Implizite Frame-Änderungen.....	797
11.5.6.1	Umschalten von Geometrieachsen.....	797
11.5.6.2	An- und Abwahl von Transformationen: Allgemein.....	800
11.5.6.3	An- und Abwahl von Transformationen: TRANSMIT.....	801
11.5.6.4	An- und Abwahl von Transformationen: TRACYL.....	807
11.5.6.5	An- und Abwahl von Transformationen: TRAANG.....	812
11.5.6.6	Adaptionen von aktiven Frames.....	818
11.5.6.7	Mapped Frames.....	819
11.5.7	Vordefinierte Frame-Funktionen.....	823
11.5.7.1	Inverses Frame.....	823
11.5.7.2	Additives Frame in der Framekette.....	825
11.5.8	Funktionen.....	827
11.5.8.1	Setzen von Nullpunkten, Werkstück- und Werkzeugvermessung.....	827
11.5.8.2	Achsspezifische Externe Nullpunktverschiebung.....	827
11.5.8.3	Werkzeugträger.....	828
11.5.9	Unterprogramme mit Attribut SAVE.....	838
11.5.10	Datensicherung.....	839
11.5.11	Positionen in den Koordinatensystemen.....	840

11.5.12	Steuerungsverhalten	840
11.5.12.1	POWER ON.....	840
11.5.12.2	Betriebsartenwechsel.....	841
11.5.12.3	Kanal-Reset / Teileprogrammende.....	841
11.5.12.4	Teileprogrammstart.....	845
11.5.12.5	Satzsuchlauf.....	845
11.5.12.6	REPOS.....	846
11.6	Werkstücknahes Istwertsystem.....	846
11.6.1	Übersicht.....	846
11.6.2	Benutzung des werkstücknahen Istwertsystems.....	846
11.6.3	Besondere Reaktionen.....	849
11.7	Randbedingungen.....	851
11.8	Beispiele.....	851
11.8.1	Achsen.....	851
11.8.2	Koordinatensysteme.....	853
11.8.3	Frames.....	855
11.9	Datenlisten.....	857
11.9.1	Maschinendaten.....	857
11.9.1.1	Anzeige-Maschinendaten.....	857
11.9.1.2	NC-spezifischen Maschinendaten.....	858
11.9.1.3	Kanal-spezifische Maschinendaten.....	858
11.9.1.4	Achs-/Spindel-spezifische Maschinendaten.....	859
11.9.2	Settingdaten.....	859
11.9.2.1	Kanal-spezifische Settingdaten.....	859
11.9.3	Systemvariablen.....	860
11.9.4	Signale.....	861
11.9.4.1	Signale von Kanal.....	861
11.9.4.2	Signale an Achse/Spindel.....	861
11.9.4.3	Signale von Achse/Spindel.....	861
12	N2: Not-Halt.....	863
12.1	Kurzbeschreibung.....	863
12.2	Normen.....	863
12.3	Not-Halt-Stellteile.....	864
12.4	Not-Halt-Ablauf.....	865
12.5	Not-Halt-Quittierung.....	866
12.6	Datenlisten.....	868
12.6.1	Maschinendaten.....	868
12.6.1.1	Achs-/Spindel-spezifische Maschinendaten.....	868
12.6.2	Signale.....	868
12.6.2.1	Signale an NC.....	868
12.6.2.2	Signale von NC.....	868
12.6.2.3	Signale an BAG.....	869
13	P1: Planachsen.....	871
13.1	Funktion.....	871
13.2	Parametrierung.....	874

13.3	Programmierung.....	877
13.4	Randbedingungen.....	878
13.5	Beispiele.....	879
13.6	Datenlisten.....	880
13.6.1	Maschinendaten.....	880
13.6.1.1	Kanal-spezifische Maschinendaten.....	880
13.6.1.2	Achs-/Spindel-spezifische Maschinendaten.....	881
14	P3: PLC-Grundprogramm für SINUMERIK 840D sl.....	883
14.1	Kurzbeschreibung.....	883
14.2	Eckdaten der PLC-CPU.....	885
14.3	PLC-Betriebssystemversion.....	886
14.4	PLC-Betriebsartenschalter.....	886
14.5	Ressourcen (Timer, Zähler, FC, FB, DB, Peripherie) reservieren.....	887
14.6	Inbetriebnahme Hardware-Konfiguration der PLC-CPU.....	887
14.7	Inbetriebnahme PLC-Programm.....	887
14.7.1	Installation des Grundprogramms.....	887
14.7.2	Anwendung des Grundprogramms.....	888
14.7.3	Versionskennzeichnungen.....	889
14.7.4	Maschinenprogramm.....	890
14.7.5	Datensicherung.....	890
14.7.6	PLC-Serieninbetriebnahme, PLC Archive.....	891
14.7.7	Software-Hochrüstung.....	893
14.7.8	Peripheriebaugruppen (FM-, CP-Baugruppen).....	894
14.7.9	Fehler-Beseitigung.....	895
14.8	Ankopplung der PLC-CPU.....	896
14.8.1	Allgemeine Informationen.....	896
14.8.2	Eigenschaften der PLC-CPU.....	896
14.8.3	Nahtstelle mit integrierter PLC.....	896
14.8.4	Diagnosepuffer der PLC.....	899
14.9	Struktur der Nahtstelle.....	899
14.9.1	Nahtstelle PLC/NCK.....	899
14.9.2	Nahtstelle PLC/HMI.....	905
14.9.3	Nahtstelle PLC/MCP/BHG.....	910
14.10	Struktur und Funktionen des Grundprogramms.....	912
14.10.1	Anlauf und Synchronisation NCK-PLC.....	914
14.10.2	Zyklischer Betrieb (OB1).....	914
14.10.3	Zeitalarm Bearbeitung (OB35).....	916
14.10.4	Prozessalarm Bearbeitung (OB40)	917
14.10.5	Diagnosealarm, Baugruppenausfall Bearbeitung (OB82, OB86).....	917
14.10.6	Verhalten bei NCK-Ausfall.....	918
14.10.7	Funktionen des Grundprogramms mit Aufruf vom Anwenderprogramm.....	919
14.10.8	Symbolische Programmierung des Anwenderprogramms mit Nahtstellen-DB.....	922
14.10.9	M-Dekodierung nach Liste.....	924
14.10.10	PLC-Maschinendaten.....	928
14.10.11	Projektierung von Maschinensteuertafel, Bedienhandgerät, Direkttasten.....	932

14.10.12	Umschaltung von Maschinensteuertafel, Bedienhandgerät.....	941
14.11	SPL für Safety Integrated.....	943
14.12	Belegungsübersicht.....	943
14.12.1	Belegung: NCK/PLC-Nahtstelle.....	943
14.12.2	Belegung: FB/FC.....	943
14.12.3	Belegung: DB.....	944
14.12.4	Belegung: Timer.....	945
14.13	PLC-Funktionen für HMI (DB19).....	945
14.13.1	Kanalanwahl.....	945
14.13.2	Programmanwahl.....	946
14.13.3	Tastensperre aktivieren.....	949
14.13.4	Bedienbereichsnummern.....	950
14.13.5	Bildnummern.....	950
14.13.5.1	Bildnummern: JOG, manuelle Maschine.....	951
14.13.5.2	Bildnummern: Referenzpunktfahren.....	956
14.13.5.3	Bildnummern: MDA.....	956
14.13.5.4	Bildnummern: AUTOMATIK.....	956
14.13.5.5	Bildnummern: Bedienbereich Parameter.....	957
14.13.5.6	Bildnummern: Bedienbereich Programm.....	958
14.13.5.7	Bildnummern: Bedienbereich Prog., Programm-Manager.....	958
14.13.5.8	Bildnummern: Bedienbereich Diagnose.....	959
14.13.6	HMI-Monitor.....	959
14.14	PLC-Funktionen für Antriebskomponenten am integrierten PROFIBUS.....	960
14.14.1	Überblick.....	960
14.14.2	Inbetriebnahme.....	961
14.14.3	Beispiel.....	961
14.15	Speicherbedarf des PLC-Grundprogramms.....	962
14.16	Rahmenbedingungen und NC-VAR-Selector.....	966
14.16.1	Rahmenbedingungen.....	966
14.16.1.1	Programmier- und Parametrierwerkzeuge.....	966
14.16.1.2	Notwendige SIMATIC-Dokumentation.....	967
14.16.1.3	Relevante SINUMERIK-Dokumente.....	968
14.16.2	NC-VAR-Selector.....	968
14.16.2.1	Übersicht.....	968
14.16.2.2	Funktionsbeschreibung.....	971
14.16.2.3	Inbetriebnahme, Installation.....	980
14.17	Bausteinbeschreibungen.....	980
14.17.1	FB1: RUN_UP - Grundprogramm, Anlaufteil.....	980
14.17.2	FB2: GET - NC-Variable lesen.....	989
14.17.3	FB3: PUT - NC-Variable schreiben.....	997
14.17.4	FB4: PI_SERV - PI-Dienst anfordern.....	1005
14.17.4.1	Liste verfügbarer PI-Dienste.....	1009
14.17.4.2	PI-Dienst: ASUP	1010
14.17.4.3	PI-Dienst: CANCEL.....	1011
14.17.4.4	PI-Dienst: CONFIG.....	1011
14.17.4.5	PI-Dienst: DIGION.....	1012
14.17.4.6	PI-Dienst: DIGIOF.....	1012
14.17.4.7	PI-Dienst: FINDBL.....	1012
14.17.4.8	PI-Dienst: LOGIN.....	1013

14.17.4.9	PI-Dienst: LOGOUT.....	1013
14.17.4.10	PI-Dienst: NCRES.....	1013
14.17.4.11	PI-Dienst: SELECT.....	1014
14.17.4.12	PI-Dienst: SETUDT.....	1014
14.17.4.13	PI-Dienst: SETUFR.....	1015
14.17.4.14	PI-Dienst: RETRAC.....	1015
14.17.4.15	PI-Dienst: CRCEDN.....	1016
14.17.4.16	PI-Dienst: CREAM.....	1017
14.17.4.17	PI-Dienst: CREATO.....	1017
14.17.4.18	PI-Dienst: DELECE.....	1017
14.17.4.19	PI-Dienst: DELETO.....	1018
14.17.4.20	PI-Dienst: MMCSEM.....	1018
14.17.4.21	PI-Dienst: TMCRT0.....	1019
14.17.4.22	PI-Dienst: TMFDPL.....	1021
14.17.4.23	PI-Dienst: TMFPBP.....	1022
14.17.4.24	PI-Dienst: TMGETT.....	1023
14.17.4.25	PI-Dienst: TMMVTL.....	1024
14.17.4.26	PI-Dienst: TMPOSM.....	1025
14.17.4.27	PI-Dienst: TMPCIT.....	1026
14.17.4.28	PI-Dienst: TMRASS.....	1027
14.17.4.29	PI-Dienst: TRESMO.....	1027
14.17.4.30	PI-Dienst: TSEARC.....	1028
14.17.4.31	PI-Dienst: TMCRTM.....	1031
14.17.4.32	PI-Dienst: TMDLMT.....	1032
14.17.4.33	PI-Dienst: POSMT.....	1032
14.17.4.34	PI-Dienst: FDPLMT.....	1033
14.17.5	FB5: GETGUD - GUD-Variable lesen.....	1034
14.17.6	FB7: PI_SERV2 - PI-Dienst anfordern.....	1043
14.17.7	FB9: MzuN - Bedieneinheitenumschaltung.....	1044
14.17.8	FB10: Sicherheits-Relais (SI-Relais).....	1048
14.17.9	FB11: Bremsentest.....	1051
14.17.10	FB29: Diagnose Signalrekorder und Datentrigger.....	1057
14.17.11	FC2: GP_HP - Grundprogramm, zyklischer Teil.....	1060
14.17.12	FC3: GP_PRAL - Grundprogramm, alarmgesteuerter Teil.....	1062
14.17.13	FC5: GP_DIAG - Grundprogramm, Diagnosealarm und Baugruppenausfall.....	1064
14.17.14	FC6: TM_TRANS2 - Transfer-Baustein für Werkzeugverwaltung und Multitool.....	1066
14.17.15	FC7: TM_REV - Transfer-Baustein für Werkzeugwechsel mit Revolver.....	1067
14.17.16	FC8: TM_TRANS - Transfer-Baustein für Werkzeugverwaltung.....	1071
14.17.17	FC9: ASUP - Start von asynchronen Unterprogrammen.....	1078
14.17.18	FC10: AL_MSG - Fehler- und Betriebsmeldungen.....	1080
14.17.19	FC12: AUXFU - Aufrufschnittstelle für Anwender bei Hilfsfunktionen.....	1083
14.17.20	FC13: BHGDisp - Displaysteuerung für Bedienhandgerät.....	1084
14.17.21	FC17: YDelta - Stern-Dreieck-Umschaltung.....	1088
14.17.22	FC18: SpinCtrl - Spindelsteuerung.....	1092
14.17.23	FC19: MCP_IFM - Übertragung der MCP-Signale an die Nahtstelle.....	1102
14.17.24	FC21: Transfer - Datenaustausch NC/PLC.....	1108
14.17.24.1	Funktion.....	1108
14.17.24.2	Deklaration der Funktion.....	1109
14.17.24.3	Erläuterungen der Formalparameter.....	1109
14.17.24.4	Funktion 1, 2: Signale Synchronaktionen an / von Kanal.....	1110
14.17.24.5	Funktion 3, 4: Schneller Datenaustausch PLC-NC.....	1111
14.17.24.6	Funktion 5: Steuersignale an Kanal aktualisieren.....	1115

14.17.24.7	Funktion 6: Steuersignale an Achsen aktualisieren.....	1115
14.17.24.8	Funktion 7: Steuersignale an Achsen aktualisieren.....	1116
14.17.25	FC22: TM_DIR - Richtungsauswahl für Werkzeugverwaltung.....	1116
14.17.26	FC24: MCP_IFM2 - Übertragung der MCP-Signale an die Nahtstelle.....	1118
14.17.27	FC25: MCP_IFT - Übertragung der MCP-/BT-Signale an die Nahtstelle.....	1122
14.17.28	FC26: HPU_MCP - Übertragung der HT 8-Signale an die Nahtstelle.....	1125
14.17.28.1	Übersicht der NC/PLC-Nahtstellensignale von HT 8.....	1129
14.17.28.2	Übersicht der NC/PLC-Nahtstellensignale an HT 8.....	1131
14.17.29	FC19, FC24, FC25, FC26 - Beschreibung Quellcode.....	1131
14.17.30	FC1005: AG_SEND - Übergibt Daten an Ethernet-CP.....	1133
14.17.31	FC1006: AG_RECV - Empfängt Daten von Ethernet-CP.....	1134
14.18	Signal-/Datenbeschreibungen.....	1136
14.18.1	Nahtstellensignale NCK/PLC, HMI/PLC, MSTT/PLC.....	1136
14.18.2	Dekodierte M-Signale.....	1136
14.18.3	G-Befehle.....	1137
14.18.4	Meldesignale im DB2.....	1138
14.19	Hinweise zur Programmierung in STEP 7.....	1138
14.19.1	Daten kopieren.....	1139
14.19.2	ANY und POINTER.....	1139
14.19.2.1	Verwendung von POINTER und ANY im FC.....	1139
14.19.2.2	Verwendung von POINTER und ANY im FB.....	1141
14.19.2.3	Variable POINTER bzw. ANY für Übergabe an FC oder FB.....	1142
14.19.3	Multi-Instanz-DB.....	1144
14.19.4	Strings.....	1145
14.19.5	Ermittlung von Offsetadressen auf Datenbaustein-Strukturen.....	1146
14.19.6	Aufrufe von FBs.....	1147
14.20	Datenlisten.....	1148
14.20.1	Maschinendaten.....	1148
14.20.1.1	Anzeige-Maschinendaten.....	1148
14.20.1.2	NC-spezifische Maschinendaten.....	1148
14.20.1.3	Kanal-spezifische Maschinendaten.....	1149
14.20.2	Signale.....	1149
14.20.2.1	Signale von Bedientafel.....	1149
15	P4: PLC für SINUMERIK 828D.....	1151
15.1	Überblick.....	1151
15.1.1	PLC-Firmware.....	1151
15.1.2	PLC-Anwendernahtstelle.....	1151
15.1.2.1	Zyklisch ausgetauschte Daten.....	1153
15.1.2.2	Alarmer und Meldungen.....	1153
15.1.2.3	Remanente Daten.....	1154
15.1.2.4	Nichtremanente Daten.....	1154
15.1.2.5	PLC-Maschinendaten.....	1154
15.1.3	PLC-Eckdaten.....	1154
15.1.4	PLC Peripherie, schnelle Onboard-Ein-/Ausgänge.....	1155
15.1.5	PLC-Toolbox.....	1155
15.1.5.1	Stern/Dreieck-Umschaltung.....	1155
15.2	Schnelle Onboard-Ein- und Ausgänge.....	1155
15.3	Ladder Viewer und Ladder add-on.....	1157
15.3.1	Überblick.....	1157

15.3.2	Parametrierung.....	1157
15.4	PLC Programming Tool.....	1158
15.5	Datenschnittstelle.....	1158
15.5.1	Nahtstelle PLC-NC.....	1159
15.5.1.1	Betriebsartensignale.....	1159
15.5.1.2	NC-Kanal-Signale.....	1160
15.5.1.3	Achs- und Spindelsignale.....	1161
15.5.1.4	Allgemeine NC-Signale.....	1162
15.5.1.5	Schneller Datenaustausch PLC-NC.....	1162
15.5.2	Nahtstelle PLC-HMI.....	1163
15.5.2.1	Bedienbereichsnummern.....	1163
15.5.2.2	Bildnummern.....	1164
15.6	Funktionsschnittstelle.....	1172
15.6.1	NC-Variable lesen/schreiben.....	1172
15.6.1.1	Anwenderschnittstelle.....	1172
15.6.1.2	Auftragsspezifizierung.....	1173
15.6.1.3	Auftragsmanagement: Auftrag starten.....	1174
15.6.1.4	Auftragsmanagement: Warten auf Auftragsende.....	1175
15.6.1.5	Auftragsmanagement: Auftragsabschluss.....	1175
15.6.1.6	Auftragsmanagement: Ablaufdiagramm.....	1176
15.6.1.7	Auftragsauswertung.....	1177
15.6.1.8	Bedienbare Variablen.....	1179
15.6.1.9	Spezifizierung von ausgewählten NC-Variablen.....	1185
15.6.2	Programminstanz-Dienste (PI-Dienste).....	1187
15.6.2.1	Auftragsspezifizierung.....	1187
15.6.2.2	Auftragsrückmeldung.....	1187
15.6.2.3	PI-Dienst ASUP.....	1188
15.6.2.4	PI-Dienst LOGOUT.....	1190
15.6.2.5	PI-Dienst DATA_SAVE.....	1190
15.6.2.6	PI-Dienst TMMVTL.....	1191
15.6.2.7	PI-Dienste: Takt diagramm.....	1192
15.6.3	PLC-Anwenderalarme.....	1192
15.6.3.1	Anwenderschnittstelle.....	1192
15.6.3.2	Aktivierungsschnittstelle der Anwenderalarme.....	1193
15.6.3.3	Variablenschnittstelle der Anwenderalarme.....	1194
15.6.3.4	Anwenderalarme projektieren.....	1195
15.6.3.5	Aktive Alarmreaktionen und Löschkriterien auslesen.....	1197
15.6.3.6	Quittierungsschnittstelle der Anwenderalarme.....	1198
15.6.3.7	Schnittstelle zum HMI.....	1198
15.6.4	PLC-Achssteuerung.....	1198
15.6.4.1	Allgemeine Informationen.....	1198
15.6.4.2	Anwenderschnittstelle: Vorbereiten der NC-Achse als PLC-Achse.....	1200
15.6.4.3	Anwenderschnittstelle: Funktionalität.....	1201
15.6.4.4	Spindel positionieren.....	1202
15.6.4.5	Spindel drehen.....	1204
15.6.4.6	Spindel pendeln.....	1206
15.6.4.7	Teilungsachse.....	1207
15.6.4.8	Positionierachse metrisch.....	1209
15.6.4.9	Positionierachse Inch.....	1210
15.6.4.10	Positionierachse metrisch mit Handradüberlagerung.....	1212
15.6.4.11	Positionierachse Inch mit Handradüberlagerung.....	1213

15.6.4.12	Spindel drehen mit automatischer Getriebestufenanwahl.....	1215
15.6.4.13	Spindel drehen mit konstanter Schnittgeschwindigkeit [m/min].....	1216
15.6.4.14	Spindel drehen mit konstanter Schnittgeschwindigkeit [feet/min].....	1217
15.6.4.15	Fehlermeldungen.....	1219
15.6.5	ASUP starten.....	1221
15.6.5.1	Auftragsstart.....	1221
15.6.5.2	Auftragsergebnis.....	1222
15.6.5.3	Signalfluss.....	1223
15.6.6	Kanalwahl am HMI.....	1223
15.7	CNC-Sperrfunktion (Option).....	1225
15.7.1	Funktion.....	1225
15.7.2	Voraussetzungen.....	1226
15.7.3	Einschränkungen.....	1226
15.7.4	Schutz vor Manipulation.....	1227
15.7.5	Ersteinrichtung der CNC-Sperrfunktion.....	1227
15.7.6	Verlängerung der CNC-Sperrfunktion.....	1231
15.7.7	Deaktivierung der CNC-Sperrfunktion.....	1233
15.7.8	Austausch einer defekten Steuerungshardware (PPU).....	1235
15.7.9	Austausch einer defekten CF-Card.....	1237
15.7.10	OEM-PIN vergessen.....	1239
15.7.11	Sonstige Informationen.....	1240
16	R1: Referenzieren.....	1243
16.1	Kurzbeschreibung.....	1243
16.2	Achsspezifisches Referenzieren.....	1244
16.3	Kanalspezifisches Referenzieren.....	1246
16.4	Referenzpunktfahren aus dem Teileprogramm (G74).....	1248
16.5	Referenzieren bei inkrementellen Messsystemen.....	1249
16.5.1	Hardware-Signale.....	1249
16.5.2	Nullmarkenauswahl.....	1250
16.5.3	Zeitlicher Ablauf.....	1252
16.5.4	Phase 1: Fahren auf den Referenznocken.....	1253
16.5.5	Phase 2: Synchronisation mit der Nullmarke.....	1255
16.5.6	Phase 3: Fahren auf den Referenzpunkt.....	1260
16.6	Referenzieren bei abstandscodierten Referenzmarken.....	1262
16.6.1	Allgemeiner Überblick.....	1262
16.6.2	Grundlegende Parametrierung.....	1263
16.6.3	Zeitlicher Ablauf.....	1265
16.6.4	Phase 1: Überfahren der Referenzmarken mit Synchronisation.....	1265
16.6.5	Phase 2: Fahren auf Zielpunkt.....	1267
16.7	Referenzieren über Istwertabgleich.....	1269
16.7.1	Istwertabgleich auf das referenzierende Messsystem.....	1269
16.7.2	Istwertabgleich bei Messsystemen mit abstandscodierten Referenzmarken.....	1270
16.8	Referenzieren im Nachführbetrieb.....	1271
16.9	Referenzieren bei Absolutwertgebern.....	1274
16.9.1	Informationen zur Justage.....	1274
16.9.2	Justage durch Eingabe der Referenzpunktverschiebung.....	1276
16.9.3	Justage durch Eingabe des Referenzpunktwertes.....	1278

16.9.4	Automatische Justage mittels Messtaster.....	1280
16.9.5	Justage mittels BERO.....	1281
16.9.6	Referenzpunktfahren bei Absolutwertgebern.....	1282
16.9.7	Referenzpunktfahren bei rotatorischen Absolutwertgebern mit Ersatznullmarke.....	1283
16.9.8	Einschalten des Messsystems.....	1285
16.9.9	Nicht unterstützte Referenziervarianten.....	1286
16.10	Automatisches Wiederherstellen des Maschinenbezugs.....	1287
16.10.1	Automatisches Referenzieren.....	1288
16.10.2	Restaurieren der Istposition.....	1289
16.11	Randbedingungen.....	1291
16.11.1	Große Verfahrbereiche.....	1291
16.12	Datenlisten.....	1292
16.12.1	Maschinendaten.....	1292
16.12.1.1	NC-spezifische Maschinendaten.....	1292
16.12.1.2	Kanal-spezifische Maschinendaten.....	1292
16.12.1.3	Achs-/Spindel-spezifische Maschinendaten.....	1292
16.12.2	Signale.....	1293
16.12.2.1	Signale an BAG.....	1293
16.12.2.2	Signale von BAG.....	1293
16.12.2.3	Signale an Kanal.....	1294
16.12.2.4	Signale von Kanal.....	1294
16.12.2.5	Signale an Achse/Spindel.....	1294
16.12.2.6	Signale von Achse/Spindel.....	1294
17	S1: Spindeln.....	1295
17.1	Kurzbeschreibung.....	1295
17.2	Betriebsarten.....	1296
17.2.1	Übersicht.....	1296
17.2.2	Betriebsartenwechsel.....	1297
17.2.3	Steuerbetrieb.....	1298
17.2.4	Pendelbetrieb.....	1301
17.2.5	Positionierbetrieb.....	1301
17.2.5.1	Allgemeine Funktionalität.....	1301
17.2.5.2	Positionieren aus der Drehung.....	1308
17.2.5.3	Positionieren aus dem Stillstand.....	1313
17.2.5.4	Signal "Spindel in Position" für Werkzeugwechsel.....	1316
17.2.6	Achsbetrieb.....	1317
17.2.6.1	Allgemeine Funktionalität.....	1317
17.2.6.2	Impliziter Übergang in den Achsbetrieb.....	1320
17.2.7	Spindel-Grundstellung.....	1323
17.2.8	Gewindebohren ohne Ausgleichsfutter.....	1324
17.2.8.1	Funktion.....	1324
17.2.8.2	Programmierung.....	1325
17.2.8.3	Beispiel: Gewindebohren mit G331 / G332.....	1326
17.2.8.4	Beispiel: Programmierte Bohrdrehzahl in der aktuellen Getriebestufe ausgeben.....	1326
17.2.8.5	Beispiel: Anwendung des zweiten Getriebestufen-Datensatzes.....	1327
17.2.8.6	Beispiel: Keine Drehzahlprogrammierung, Überwachung der Getriebestufe.....	1327
17.2.8.7	Beispiel: Getriebestufenwechsel nicht möglich, Überwachung der Getriebestufe.....	1328
17.2.8.8	Beispiel: Programmierung ohne SPOS.....	1328
17.2.8.9	Sonderfall: Drehrichtungsumkehr über NC/PLC-Nahtstellensignal im NC-Programm.....	1329

17.2.9	Gewindebohren mit Ausgleichsfutter.....	1330
17.2.9.1	Funktion.....	1330
17.2.9.2	Programmierung.....	1331
17.3	Referenzieren / Synchronisieren.....	1332
17.4	Projektierbare Getriebearpassungen.....	1337
17.4.1	Getriebestufen für Spindeln und Getriebestufenwechsel.....	1337
17.4.2	Spindelgetriebestufe 0.....	1348
17.4.3	Ermittlung der Spindelgetriebestufe.....	1350
17.4.4	Parametersatzanwahl beim Getriebestufenwechsel.....	1351
17.4.5	Vorsatz-Getriebe.....	1354
17.4.6	Nicht quittierter Getriebestufenwechsel.....	1355
17.4.7	Getriebestufenwechsel mit Pendelbetrieb.....	1356
17.4.8	Getriebestufenwechsel auf Festposition.....	1362
17.4.9	Projektierbare Getriebestufe bei M70.....	1368
17.4.10	Unterdrückung des Getriebestufenwechsels bei DryRun, Programmtest und SERUPRO..	1371
17.5	Weitere projektierbare Anpassungen der Spindelfunktionalität.....	1373
17.6	Umschaltbare Spindeln.....	1374
17.7	Programmierung.....	1378
17.7.1	Programmierung aus dem Teileprogramm.....	1378
17.7.2	Programmierung über Synchronaktionen.....	1382
17.7.3	Programmierung von Spindelsteuerungen über PLC mit FC18 - nur 840D sl.....	1382
17.7.4	Programmierung über NC/PLC-Nahtstellensignale.....	1383
17.7.4.1	Funktion.....	1383
17.7.4.2	Inbetriebnahme: Maschinendaten.....	1384
17.7.4.3	Inbetriebnahme: NC/PLC-Nahtstellensignale.....	1385
17.7.4.4	Drehzahlvorgabe (SD43200).....	1386
17.7.4.5	Vorgabe der konstanten Schnittgeschwindigkeit (SD43202).....	1386
17.7.4.6	Vorgabe des Spindeldrehzahltyps für die Masterspindel (SD43206).....	1387
17.7.5	Programmierung von extern (PLC, HMI).....	1388
17.8	Spindelüberwachungen.....	1389
17.8.1	Zulässige Drehzahlbereiche.....	1389
17.8.2	Achse/Spindel steht.....	1390
17.8.3	Spindel im Sollbereich.....	1390
17.8.4	Minimale/Maximale Drehzahl der Getriebestufe.....	1392
17.8.5	Diagnose von Spindeldrehzahlbegrenzungen.....	1393
17.8.6	Maximale Drehzahl der Spindel.....	1394
17.8.7	Maximale Gebergrenzfrequenz.....	1396
17.8.8	Zielpunktüberwachung.....	1397
17.8.9	M40: Automatische Getriebestufenauswahl bei Drehzahlen außerhalb projektierter Schaltschwellen.....	1399
17.9	Spindel mit SMI 24 (Weiss-Spindel).....	1401
17.9.1	Allgemeine Informationen.....	1401
17.9.2	Sensordaten.....	1401
17.9.3	Spannzustand.....	1403
17.9.4	Weitere Antriebsparameter.....	1405
17.10	Randbedingungen.....	1405
17.10.1	Ändern von Regelparametern.....	1405
17.11	Beispiele.....	1406

17.11.1	Automatische Getriebestufenanwahl (M40).....	1406
17.12	Datenlisten.....	1407
17.12.1	Maschinendaten.....	1407
17.12.1.1	NC-spezifische Maschinendaten.....	1407
17.12.1.2	Kanal-spezifische Maschinendaten.....	1407
17.12.1.3	Achs-/Spindel-spezifische Maschinendaten.....	1407
17.12.2	Settingdaten.....	1409
17.12.2.1	Kanal-spezifische Settingdaten.....	1409
17.12.2.2	Achs-/Spindel-spezifische Settingdaten.....	1409
17.12.3	Signale.....	1410
17.12.3.1	Signale an Achse/Spindel.....	1410
17.12.3.2	Signale von Achse/Spindel.....	1411
18	V1: Vorschübe.....	1413
18.1	Kurzbeschreibung.....	1413
18.2	Bahnvorschub F.....	1414
18.2.1	Vorschubart G93, G94, G95.....	1416
18.2.2	Vorschubart G96, G961, G962, G97, G971.....	1419
18.2.3	Vorschub bei Gewindschneiden (G33, G34, G35, G335, G336)	1423
18.2.3.1	Vorschub bei G33.....	1423
18.2.3.2	Linear progressive/degressive Gewindesteigungsänderung bei G34 und G35	1424
18.2.3.3	Beschleunigungsverhalten der Achse bei G33, G34 und G35	1426
18.2.3.4	Programmierter Einlauf- und Auslaufweg bei G33, G34 und G35 (DITS, DITE).....	1428
18.2.3.5	Schnellrückzug während Gewindeschneiden.....	1430
18.2.3.6	Ballige Gewinde (G335, G336).....	1434
18.2.4	Vorschub bei Gewindebohren ohne Ausgleichsfutter (G331, G332).....	1439
18.2.5	Vorschub bei Gewindebohren mit Ausgleichsfutter (G63).....	1441
18.2.6	FGROUP und FREF.....	1441
18.3	Vorschub für Positionierachsen (FA).....	1444
18.4	Vorschubbeeinflussung.....	1445
18.4.1	Vorschubsperrung und Vorschub/Spindel Halt.....	1445
18.4.2	Vorschub-Override über Maschinensteuertafel.....	1447
18.4.3	Programmierbarer Vorschub-Override.....	1451
18.4.4	Probelaufvorschub.....	1452
18.4.5	Mehrere Vorschubwerte in einem Satz.....	1454
18.4.6	Feste Vorschubwerte	1459
18.4.7	Programmierbare Vorschubverläufe.....	1461
18.4.8	Vorschub für Fase/Rundung FRC, FRCM.....	1462
18.4.9	Satzweiser Vorschub FB.....	1464
18.4.10	Beeinflussung der Einzelachsdynamik	1465
18.5	Randbedingungen.....	1470
18.6	Datenlisten.....	1471
18.6.1	Maschinendaten.....	1471
18.6.1.1	NC-spezifische Maschinendaten.....	1471
18.6.1.2	Kanal-spezifische Maschinendaten.....	1472
18.6.1.3	Achs-/Spindel-spezifische Maschinendaten.....	1472
18.6.2	Settingdaten.....	1472
18.6.2.1	Kanal-spezifische Settingdaten.....	1472
18.6.2.2	Achs-/Spindel-spezifische Settingdaten.....	1473

18.6.3	Signale.....	1473
18.6.3.1	Signale an Kanal.....	1473
18.6.3.2	Signale von Kanal.....	1473
18.6.3.3	Signale an Achse/Spindel.....	1474
18.6.3.4	Signale von Achse/Spindel.....	1474
19	W1: Werkzeugkorrektur.....	1475
19.1	Kurzbeschreibung.....	1475
19.2	Werkzeug.....	1477
19.2.1	Allgemeine Informationen.....	1477
19.2.2	Korrekturspeicherstruktur.....	1480
19.2.3	Verrechnung der WZ-Korrektur.....	1482
19.2.4	Adresserweiterung für die NC-Adressen T und M.....	1482
19.2.5	Freie D-Nummernvergabe.....	1484
19.2.6	Korrektursatz bei Fehler im Werkzeugwechsel.....	1491
19.2.7	Definition der Wirkung der Werkzeugparameter.....	1493
19.3	Flache D-Nummern-Struktur.....	1494
19.3.1	Allgemeine Informationen.....	1494
19.3.2	Neuanlegen einer D-Nummer (Korrektursatz).....	1495
19.3.3	D-Nummer programmieren.....	1496
19.3.4	T-Nummer programmieren.....	1498
19.3.5	M6 programmieren.....	1498
19.3.6	Programmtest.....	1499
19.3.7	Werkzeugverwaltung oder "Flache D-Nummern-Struktur".....	1499
19.4	Werkzeugschneide.....	1500
19.4.1	Allgemeine Informationen.....	1500
19.4.2	WZ-Parameter 1: Werkzeugtyp.....	1502
19.4.3	WZ-Parameter 2: Schneidenlage.....	1505
19.4.4	WZ-Parameter 3 - 5: Geometrie - Werkzeuglängen.....	1507
19.4.5	WZ-Parameter 6 - 11: Geometrie - Werkzeugform.....	1508
19.4.6	WZ-Parameter 12 - 14: Verschleiß - Werkzeuglängen.....	1510
19.4.7	WZ-Parameter 15 - 20: Verschleiß - Werkzeugform.....	1511
19.4.8	WZ-Parameter 21 - 23: Basismaß / Adaptermaß.....	1511
19.4.9	WZ-Parameter 24: Freischneidwinkel.....	1513
19.4.10	Werkzeuge mit relevanter Schneidenlage.....	1515
19.5	2D-Werkzeugradiuskorrektur (2D-WRK).....	1515
19.5.1	Allgemeine Informationen.....	1515
19.5.2	Anwahl der WRK (G41/G42).....	1516
19.5.3	An- und Abfahrverhalten (NORM/KONT/KONTC/KONTT).....	1517
19.5.4	Weiches An- und Abfahren.....	1522
19.5.4.1	Funktion.....	1522
19.5.4.2	Parameter.....	1522
19.5.4.3	Geschwindigkeiten.....	1530
19.5.4.4	Systemvariablen.....	1532
19.5.4.5	Randbedingungen.....	1532
19.5.4.6	Beispiele.....	1533
19.5.5	Abwahl der WRK (G40).....	1536
19.5.6	Korrektur an Außenecken.....	1536
19.5.7	Korrektur an Innenecken.....	1540
19.5.8	Kollisionsüberwachung und Flaschenhalserkennung.....	1542

19.5.9	Schlitzformerkennung (Option) - nur 840D sl.....	1544
19.5.10	Sätze mit veränderlichem Korrekturwert.....	1547
19.5.11	Alarmverhalten.....	1548
19.5.12	Schnittpunktverfahren für Polynome.....	1549
19.5.13	G461/G462: Erweiterung An- / Abfahrstrategie.....	1550
19.6	Werkzeugradiuskorrektur konstant halten.....	1553
19.7	Orientierbare Werkzeugträger.....	1557
19.7.1	Allgemeine Informationen.....	1557
19.7.2	Kinematische Zusammenhänge und Maschinenaufbau.....	1564
19.7.3	Schrägbearbeitung mit 3 + 2 Achsen.....	1571
19.7.4	Maschine mit drehbarem Werkstück.....	1572
19.7.5	Vorgehen bei Benutzung der orientierbaren Werkzeugträger.....	1576
19.7.6	Programmierung.....	1580
19.7.7	Randbedingungen und Steuerungsverhalten für Orientierungen.....	1582
19.7.8	Beispiele.....	1585
19.7.8.1	Beispiel: Orientierbare Werkzeugträger.....	1585
19.7.8.2	Beispiel Orientierbare Werkzeugträger mit drehbarem Tisch.....	1585
19.7.8.3	Korrekturen einsatzort- und werkstückspezifisch einrechnen.....	1588
19.8	Schneidendaten-Modifikation bei drehbaren Werkzeugen.....	1591
19.8.1	Funktion.....	1591
19.8.2	Ermittlung des Drehwinkels.....	1591
19.8.3	Schneidenlage, Schnitttrichtung und Winkel bei Drehwerkzeugen.....	1592
19.8.4	Modifikationen bei der Drehung von Drehwerkzeugen.....	1594
19.8.5	Schneidenlage bei Fräs- und Bohrwerkzeugen.....	1596
19.8.6	Modifikationen bei der Drehung von Fräs- und Bohrwerkzeugen.....	1597
19.8.7	Parametrierung.....	1598
19.8.8	Programmierung.....	1599
19.8.9	Beispiel.....	1603
19.9	Inkrementell programmierte Korrekturwerte.....	1605
19.9.1	G91 Erweiterung.....	1605
19.9.2	Verfahren in Richtung der Werkzeugorientierung (MOVT).....	1606
19.10	Werkzeuggrundorientierung.....	1608
19.10.1	Funktion.....	1608
19.10.2	Parametrierung.....	1610
19.10.3	Beispiel.....	1611
19.11	Werkzeugkorrektur-Sonderbehandlungen.....	1612
19.11.1	Relevante Settingdaten.....	1612
19.11.2	Werkzeuglängen spiegeln (SD42900 \$SC_MIRROR_TOOL_LENGTH).....	1613
19.11.3	Verschleißlängen spiegeln (SD42920 \$SC_WEAR_SIGN_CUTPOS).....	1614
19.11.4	Werkzeuglängen und Ebenenwechsel (SD42940 \$SC_TOOL_LENGTH_CONST).....	1616
19.11.5	Werkzeugtyp (SD42950 \$SC_TOOL_LENGTH_TYPE).....	1617
19.11.6	Werkzeuglängen im WKS unter Berücksichtigung der Orientierung.....	1618
19.11.7	Werkzeuglängenoffsets in Werkzeugrichtung.....	1618
19.12	Summen- und Einrichtekorrekturen.....	1623
19.12.1	Allgemeine Informationen.....	1623
19.12.2	Funktionale Beschreibung.....	1624
19.12.3	Aktivierung.....	1627
19.12.4	Beispiele.....	1633
19.12.5	Erweiterungen der Werkzeuglängenbestimmung.....	1634

19.12.5.1	Korrekturen einsatzort- und werkstückspezifisch einrechnen.....	1634
19.12.5.2	Funktionalität der einzelnen Verschleißwerte.....	1638
19.13	Mit Werkzeugumgebungen arbeiten.....	1642
19.13.1	Werkzeugumgebung speichern (TOOLENV).....	1642
19.13.2	Werkzeugumgebung löschen (DELTOOLENV).....	1645
19.13.3	T-, D- und DL-Nummer lesen (GETTENV).....	1646
19.13.4	Informationen zu gespeicherten Werkzeugumgebungen lesen (\$P_TOOLENVN, (\$P_TOOLENV).....	1647
19.13.5	Werkzeuglängen bzw. -längenkomponenten lesen (GETTCOR).....	1647
19.13.6	Werkzeugkomponenten ändern (SETTCOR).....	1654
19.14	Zuordnung der Werkzeuglängen L1, L2, L3 zu den Koordinatenachsen lesen (LENTOAX).....	1667
19.15	Randbedingungen.....	1670
19.15.1	Flache D-Nummern-Struktur.....	1670
19.15.2	SD42935 Erweiterungen	1670
19.15.3	Ankratzen.....	1670
19.16	Datenlisten.....	1671
19.16.1	Maschinendaten.....	1671
19.16.1.1	NC-spezifische Maschinendaten.....	1671
19.16.1.2	Kanal-spezifische Maschinendaten.....	1671
19.16.1.3	Achs-/Spindel-spezifische Maschinendaten.....	1672
19.16.2	Settingdaten.....	1673
19.16.2.1	Kanal-spezifische Settingdaten.....	1673
19.16.3	Signale.....	1673
19.16.3.1	Signale von Kanal.....	1673
20	Z1: NC/PLC-Nahtstellensignale.....	1675
A	Anhang.....	1677
A.1	Liste der Abkürzungen.....	1677
A.2	Dokumentationsübersicht.....	1686
	Glossar.....	1687
	Index.....	1709

Grundlegende Sicherheitshinweise

1.1 Allgemeine Sicherheitshinweise



WARNUNG

Lebensgefahr bei Nichtbeachtung von Sicherheitshinweisen und Restrisiken

Bei Nichtbeachtung der Sicherheitshinweise und Restrisiken in der zugehörigen Hardware-Dokumentation können Unfälle mit schweren Verletzungen oder Tod auftreten.

- Halten Sie die Sicherheitshinweise der Hardware-Dokumentation ein.
- Berücksichtigen Sie bei der Risikobeurteilung die Restrisiken.



WARNUNG

Lebensgefahr durch Fehlfunktionen der Maschine infolge fehlerhafter oder veränderter Parametrierung

Durch fehlerhafte oder veränderte Parametrierung können Fehlfunktionen an Maschinen auftreten, die zu Körperverletzungen oder Tod führen können.

- Schützen Sie die Parametrierungen vor unbefugtem Zugriff.
- Beherrschen Sie mögliche Fehlfunktionen durch geeignete Maßnahmen (z. B. NOT-HALT oder NOT-AUS).

1.2 Industrial Security

Hinweis

Industrial Security

Siemens bietet Produkte und Lösungen mit Industrial-Security-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen nur einen Bestandteil eines solchen Konzepts.

Der Kunde ist dafür verantwortlich, unbefugten Zugriff auf seine Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und entsprechende Schutzmaßnahmen (z. B. Nutzung von Firewalls und Netzwerksegmentierung) ergriffen wurden.

Zusätzlich sollten die Empfehlungen von Siemens zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Industrial Security finden Sie unter:

Industrial Security (<http://www.siemens.com/industrialsecurity>).

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Aktualisierungen durchzuführen, sobald die entsprechenden Updates zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Security RSS Feed unter:

Industrial Security (<http://www.siemens.com/industrialsecurity>).



WARNUNG

Lebensgefahr durch unsichere Betriebszustände wegen Manipulation der Software

Manipulationen der Software, z. B. Viren, Trojaner, Malware oder Würmer, können unsichere Betriebszustände in Ihrer Anlage verursachen, die zu Tod, schwerer Körperverletzung und zu Sachschäden führen können.

- Halten Sie die Software aktuell.
- Integrieren Sie die Automatisierungs- und Antriebskomponenten in ein ganzheitliches Industrial Security-Konzept der Anlage oder Maschine nach dem aktuellen Stand der Technik.
- Berücksichtigen Sie bei Ihrem ganzheitlichen Industrial Security-Konzept alle eingesetzten Produkte.
- Schützen Sie die Dateien in Wechselspeichermedien vor Schadsoftware durch entsprechende Schutzmaßnahmen, z. B. Virens Scanner.

A2: Diverse NC/PLC-Nahtstellensignale und Funktionen

2

2.1 Kurzbeschreibung

Inhalt

Die Nahtstelle PLC/NC wird einerseits durch eine Datenschnittstelle und andererseits durch eine Funktionsschnittstelle gebildet. In der Datenschnittstelle sind Status- und Steuersignale, Hilfs- und G-Befehl enthalten, während über die Funktionsschnittstelle Aufträge von der PLC an den NC übergeben werden.

In der vorliegenden Beschreibung wird die Funktionalität von Nahtstellensignalen beschrieben, die von allgemeiner Bedeutung sind und die in den funktionsspezifischen Beschreibungen nicht beschrieben sind:

- Asynchrone Ereignisse
- Zustandssignale
- PLC-Variable (lesen und schreiben)

2.2 NC/PLC-Nahtstellensignale

2.2.1 Allgemeine Informationen

NC/PLC-Nahtstelle

Die NC/PLC-Nahtstelle besteht aus den Teilen:

- Datenschnittstelle
- Funktionsschnittstelle

Datenschnittstelle

Die Datenschnittstelle dient zur Koordination der Komponenten:

- PLC-Anwenderprogramm
- NC
- HMI (Bedienkomponente)
- MTT (Maschinensteuertafel)

Der Datenaustausch wird durch das PLC-Grundprogramm organisiert.

Zyklischer Signalaustausch

Folgende Nahtstellensignale werden vom PLC-Grundprogramm zyklisch, d. h. im Taktraster des OB1, übertragen:

- NC- und Bedientafelfront-spezifische Signale
- BAG-spezifische Signale
- Kanal-spezifische Signale
- Achs-/Spindel-spezifische Signale

NC- und Bedientafelfront-spezifische Signale (DB10)

PLC an NC:

- Signale für die Beeinflussung der CNC-Ein- und -Ausgänge
- Schlüsselschalter-Signale (und Kennwort)

NC an PLC:

- Istwerte der CNC-Eingänge
- Sollwerte der CNC-Ausgänge
- Bereitschaftssignale (Ready-Signale) von NC und HMI
- Statussignale der NC (Alarmsignale)

Kanalspezifische Signale (DB21, ...)

PLC an NC:

- Steuersignal "Restweg löschen"

NC an PLC:

- Statussignale der NC (NC-Alarm steht an)

Achs-/Spindel-spezifische Signale (DB31, ...)

PLC an NC:

- Steuersignale an Achse/Spindel (z. B. Nachführbetrieb, Reglerfreigabe, ...)
- Steuersignale an Antrieb (Byte 20, 21)

NC an PLC:

- Statussignale von Achse/Spindel (z. B. Lageregler aktiv, Stromregler aktiv, ...)
- Statussignale vom Antrieb (Byte 93, 94)

Funktionsschnittstelle

Die Funktionsschnittstelle wird durch Funktionsbausteine (FB) und Funktionen (FC) gebildet. Über die Funktionsschnittstelle werden Funktionsanforderungen z. B. zum Verfahren von Achsen, von der PLC an die NC übergeben.

Literatur

- Beschreibung des PLC-Grundprogramms:
→ Funktionshandbuch Grundfunktionen; PLC-Grundprogramm (P3)
- Beschreibung des ereignisgesteuerten Signalaustausches (Hilfs- und G-Befehle):
→ Funktionshandbuch Grundfunktionen; Hilfsfunktionsausgabe an PLC (H2)
- Übersicht aller Nahtstellensignale, Funktionsbausteine und Datenbausteine:
→ Listenhandbuch 2

2.2.2 Bereitschaftssignale an PLC

DB10 DBX104.7 (NC-CPU ready)

Die NC-CPU ist betriebsbereit und meldet sich zyklisch bei der PLC.

DB10 DBX108.3 (HMI - CPU an BTSS ready)

SINUMERIK Operate ist betriebsbereit und meldet sich zyklisch bei der NC.

DB10 DBX108.5 (Antriebe im zyklischen Betrieb)

Voraussetzung: Bei **allen** Maschinenachsen der NC befinden sich die zugehörigen Antriebe im zyklischen Betrieb, d. h. sie tauschen zyklisch mit der NC PROFIdrive-Telegramme aus.

DB10 DBX108.6 (Antrieb ready)

Voraussetzung: Für **alle** Maschinenachsen der NC steht bei den zugehörigen Antrieben und auch den Fremdantrieben über PROFIBUS die Betriebsbereitschaft an:

DB31, ... DBX93.5 == 1 (Drive Ready)

DB10 DBX108.7 (NC ready)

Die NC ist betriebsbereit.

2.2.3 Statussignale an PLC

DB10 DBX103.0 (Ferndiagnose aktiv)

Die HMI-Komponente meldet an die PLC, dass die Ferndiagnose (Option!) aktiv ist, d. h. die Bedienung der Steuerung erfolgt über einen externen PC.

DB10 DBX109.6 (Luft-Temperatur-Alarm)

Die Umgebungstemperatur- oder die Lüfterüberwachung hat angesprochen.

DB10 DBX109.7 (NC-Batterie-Alarm)

Die Batteriespannung ist unter den Grenzwert abgesunken. Die Steuerung kann weiterhin betrieben werden. Ein Ausschalten der Steuerung oder Ausfall der Versorgungsspannung führt zum Datenverlust.

DB10 DBX109.0 (NC-Alarm steht an)

Die NC meldet, dass mindestens ein NC-Alarm ansteht. An der kanalspezifischen Nahtstelle kann abgefragt werden, welche Kanäle betroffen sind und ob ein Bearbeitungsstillstand ausgelöst wurde.

DB21, ... DBX36.6 (Kanalspezifischer NC-Alarm steht an)

Die NC meldet an die PLC, dass für den betroffenen Kanal mindestens ein NC-Alarm ansteht (siehe auch DB21, ... DBX36.7)

DB21, ... DBX36.7 (NC-Alarm mit Bearbeitungsstillstand steht an)

Die NC meldet an die PLC, dass für den betroffenen Kanal mindestens ein NC-Alarm ansteht, der die aktuelle Teileprogrammbearbeitung unter- bzw. abgebrochen hat.

2.2.4 Signale an/von Bedientafelfront

DB19 DBX0.0 (Bildschirm hell)

Die Dunkelschaltung des Bildschirms wird außer Kraft gesetzt.

DB19 DBX0.1 (Bildschirm dunkel)

Der Bildschirm der Bedientafel wird dunkel gesteuert.

Bei aktiver Dunkelsteuerung durch das Nahtstellensignal gilt:

- Eine Hellsteuerung über die Tastatur (siehe unten) ist damit nicht mehr möglich.
- Bereits der erste Tastendruck an der Bedientafelfront löst eine Bedienhandlung aus.

Hinweis

Um unbeabsichtigte Bedienhandlungen bei Dunkelsteuerung des Bildschirms über das Nahtstellensignal auszuschließen, wird empfohlen **gleichzeitig** die Tastatur zu sperren:

DB19 DBX0.1 = 1 **UND** DB19 DBX0.2 = 1 (Tastensperre)

Dunkelsteuerung über die Tastatur bzw. automatischer Bildschirmschoner

Wird in der parametrierten Zeit (Default = 3 Minuten):

MD9006 \$MM_DISPLAY_SWITCH_OFF_INTERVAL (Zeit für Bildschirmdunkelschaltung) keine Taste an der Bedientafelfront betätigt, wird der Bildschirm automatisch dunkel gesteuert.

Die Hellsteuerung erfolgt mit dem ersten Tastendruck nach Dunkelsteuerung. Dieser Tastendruck löst keine Bedienhandlung aus.

Parametrierung:

- DB19 DBX0.1 = 0
- MD9006 \$MM_DISPLAY_SWITCH_OFF_INTERVAL > 0

DB19 DBX0.2 (Tastensperre)

Alle Eingaben über die angeschlossene Tastatur werden gesperrt.

DB19 DBX 0.3 / 0.4 (Cancel-Alarme löschen / Recall-Alarme löschen)

Anforderung zum Löschen aller aktuell anstehenden Alarme mit Löschkriterium Cancel bzw. Recall. Das Löschen der Alarme wird über folgende Nahtstellensignale quittiert:

- DB19 DBX20.3 (Cancel-Alarm gelöscht)
- DB19 DBX20.4 (Recall-Alarm gelöscht)

DB19 DBX0.7 (Istwerte in WKS, 0=MKS)

Umschalten der Istwertanzeige zwischen Maschinen- und Werkstückkoordinatensystem:

- DB19 DBX0.7 = 0: Maschinenkoordinatensystem (MKS)
- DB19 DBX0.7 = 1: Werkstückkoordinatensystem (WKS)

DB19 DBB13 (Steuerung des Datei-Transfers)

Auftragsbyte zur Steuerung des Datei-Transfers. Die Aufträge beziehen sich auf die Anwender-Steuerdatei in den Nahtstellensignalen:

- DB19 DBB16 (Teile-Programmhandling: Nummer der Steuerdatei für Anwender-Dateinamen)
- DB19 DBB17 (Teile-Programmhandling: Index der zu übertragenden File aus der Anwender-Liste)

DB19 DBB16 (Steuerung des Datei-Transfers)

Steuerbyte des Datei-Transfers zur Beschreibung des Index zur Steuerdatei (Jobliste). Diese Datei wird behandelt entsprechend des Auftrags in DB19 DBB13

DB19 DBB17 (Teileprogrammhandling: Index des zu übertragenden Files aus der Anwenderliste)

Steuerbyte des Datei-Transfers zur Angabe in welcher Zeile der Anwender-Steuerdatei die zu übertragene Steuerdatei steht.

DB19 DBB26 (Teileprogrammhandling: Status)

Zustandsbyte für den aktuellen Zustand der Datenübertragung bei "Anwahl", "Laden" bzw. "Entladen" oder ob die Datenübertragung fehlerhaft war.

DB19 DBB27 (Error Programmhandling)

Ausgabebyte für die Fehlerwerte der Datenübertragung.

2.2.5 Signale an Kanal

DB21, ... DBX6.2 (Restweg löschen)

Mit der steigenden Flanke des Nahtstellensignals wird im entsprechenden Kanal der NC auf der programmierten Bahn mit der aktuell wirksame Bahnbeschleunigung angehalten. Anschließend wird der noch nicht verfahrens Bahnrestweg gelöscht und der Satzwechsel zum nächsten Teileprogrammsatz freigegeben.

2.2.6 Signale an Achse/Spindel

DB31, ... DBX1.0 (Antriebtest Fahrfreigabe)

ACHTUNG

Vorgaben für den Antriebtest

Es liegt in der alleinigen Verantwortung des Maschinenherstellers / Inbetriebnehmers durch geeignete Maßnahmen / Prüfungen sicherzustellen, dass die Maschinenachse im Rahmen des Antriebtests ohne Gefahr für Mensch und Maschine verfahren werden kann.

Werden Maschinenachsen von speziellen Testfunktionen wie z. B. "Funktionsgenerator" verfahren, wird für die Verfahrbewegung eine explizite Antriebtest-spezifische Freigabe angefordert:

DB31, ... DBX61.0 = 1 (Antriebtest Fahrenanforderung)

Die Verfahrbewegung wird ausgeführt, sobald die Fahrfreigabe erfolgt ist:

DB31, ... DBX1.0 == 1 (Antriebtest Fahrfreigabe)

DB31, ... DBX1.3 (Achsen- / Spindelsperre)**Stehende Achse**

Wird das Nahtstellensignal bei einer stehenden Achse gesetzt, werden ab da alle Fahranforderungen ignoriert.

Die Fahranforderungen bleibt erhalten. Wird die Achsensperre bei anstehender Verfahranforderung aufgehoben DB31, ... DBX1.3 = 0 wird die Verfahrbewegung ausgeführt.

Fahrende Achse

Wird das Nahtstellensignal bei einer fahrender Achse gesetzt, wird die Achse anhand ihrer aktuell wirksamen Bremskennlinie angehalten. Wird das Nahtstellensignal wieder zurückgesetzt während die Fahranforderung noch ansteht, wird die Verfahrbewegung fortgesetzt.

Spindel

- Steuerbetrieb: Es wird sofort Drehzahlsollwert Null ausgegeben
- Positionierbetrieb: siehe oben "Stehende Achse" / "Fahrende Achse"

DB31, ... DBX1.4 (Nachführbetrieb)

Das Nahtstellensignal ist nur zusammen mit dem Nahtstellensignal DB31, ... DBX2.1 (Reglerfreigabe) wirksam

DB31, ... DBX2.1	DB31, ... DBX1.4	Betriebsart
1	unwirksam	In Regelung
0	1	Nachführen
0	0	Halten

Nachführen

Im Nachführbetrieb wird die Sollposition der Maschinenachse kontinuierlich auf die Istposition gesetzt (Sollposition = Istposition).

Rückmeldung: DB31, ... DBX61.3 == 1 (Nachführen aktiv)

Während des Nachführbetriebs sind Klemmungs- oder Stillstandsüberwachung der Achse **inaktiv**.

Hinweis

Wird das Nahtstellensignal DB31, ... DBX2.1 (Reglerfreigabe) während des Nachführbetriebs gesetzt, erfolgt bei aktivem NC-Programm ein Wiederanfahren der zuletzt programmierten Position mit **REPOSA** (Anfahren auf einer Geraden mit allen Achsen). In allen anderen Fällen beginnen alle weiteren Verfahrbewegungen an der aktuellen Istposition.

Halten

In der Betriebsart "Halten" wird die Sollposition der Maschinenachse nicht auf die Istposition nachgeführt. Wird die Maschinenachse von der Sollposition weg bewegt, ergibt sich eine immer größer werdende Differenz zwischen Soll- und Istposition (Schleppabstand). Der

2.2 NC/PLC-Nahtstellensignale

Schleppabstand wird beim Setzen der Reglerfreigabe ohne Einhaltung der axialen Beschleunigungskennlinie schlagartig abgebaut (Drehzahlsprung).

Rückmeldung: DB31, ... DBX61.3 == 0 (Nachführen aktiv)

Während des "Haltens" sind Klemmungs- oder Stillstandsüberwachung **aktiv**.

Hinweis

Der Schleppabstand wird beim Setzen der Reglerfreigabe ohne Einhaltung der axialen Beschleunigungskennlinie schlagartig abgebaut (Drehzahlsprung).

Anwendungsbeispiel

Positionierverhalten der Maschinenachse Y nach einem Klemmungsvorgang beim Setzen der "Reglerfreigabe". Die Maschinenachse wurde dabei durch den Klemmungsvorgang aus der Istposition Y_1 auf die Klemmungsposition Y_k gedrückt.

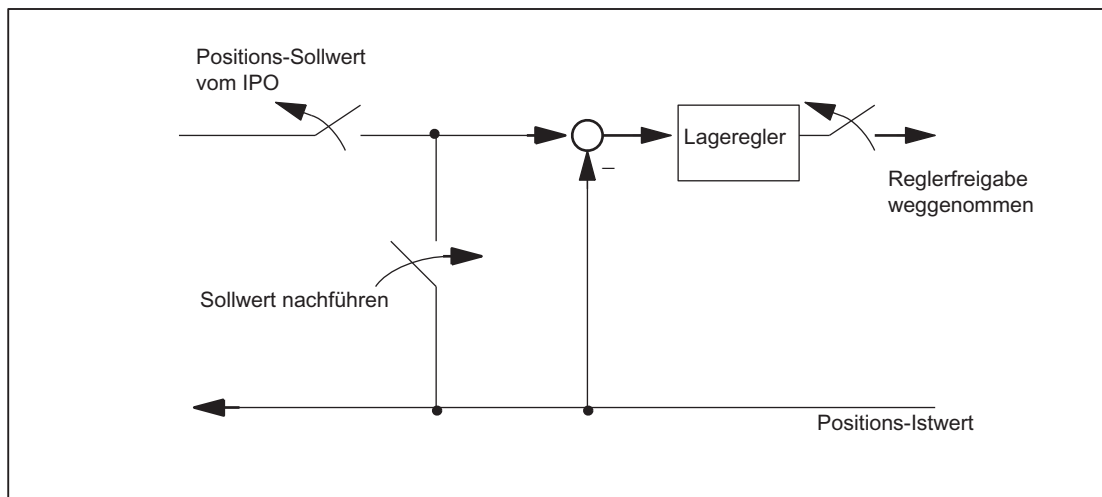


Bild 2-1 Wirkung von Reglerfreigabe und Nachführbetrieb

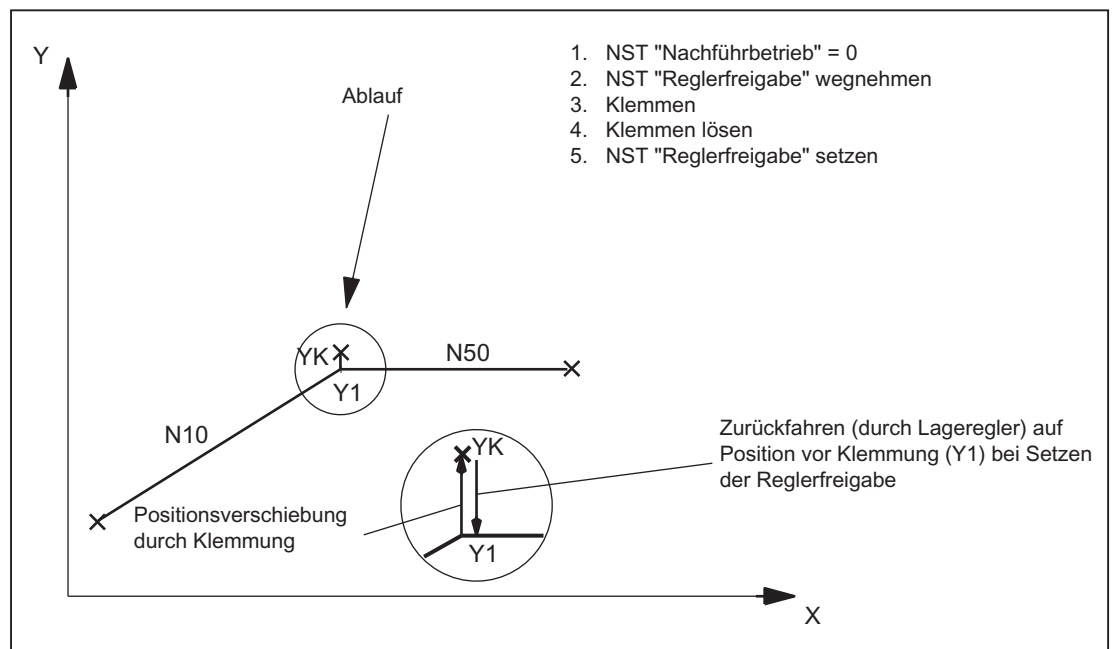


Bild 2-2 Bahnverlauf bei Klemmungsvorgang und "Halten"

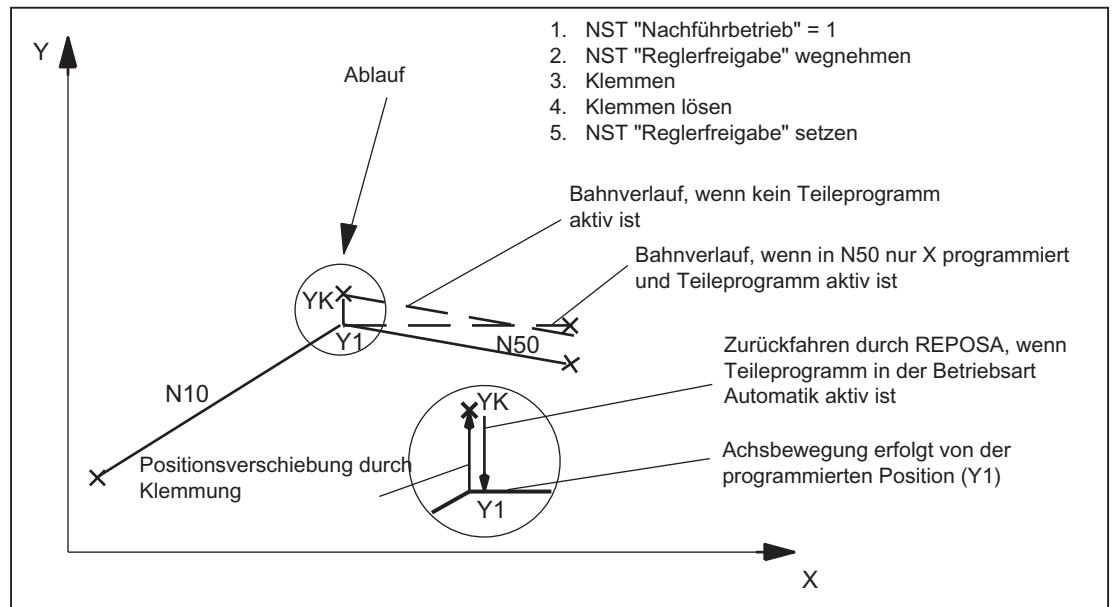


Bild 2-3 Bahnverlauf bei Klemmungsvorgang und "Nachführen"

Antrieben mit analoger Sollwertschnittstelle

Bei einem Antrieb mit analoger Sollwertschnittstelle besteht die Möglichkeit, die Maschinenachse mit einem externen Sollwert zu verfahren. Wird für die Maschinenachse "Nachführbetrieb" gesetzt, wird die Istposition weiterhin erfasst. Nach Aufheben des Nachführbetriebs ist dann kein Referenzieren erforderlich.

2.2 NC/PLC-Nahtstellensignale

Dabei ist folgendes Vorgehen zu empfehlen:

1. Nachführbetrieb aktivieren:
DB31, ... DBX2.1 = 0 (Reglerfreigabe)
DB31, ... DBX1.4 = 1 (Nachführbetrieb) (im gleichen oder vorhergehenden OB1-Zyklus)
→ Achse/Spindel befindet sich im Nachführbetrieb
2. Externe Reglerfreigabe und externen Drehzahlsollwert aufschalten
→ Achse/Spindel fährt mit externem Sollwert
→ NC erfasst weiterhin die Istposition und führt die Sollposition der Istposition nach
3. Externe Reglerfreigabe und externen Drehzahlsollwert wegnehmen
→ Achse/Spindel steht
4. Nachführbetrieb aufheben
DB31, ... DBX2.1 = 1 (Reglerfreigabe)
DB31, ... DBX1.4 = 0 (Nachführbetrieb)
→ NC synchronisiert auf die aktuelle Istposition. Die nächste Verfahrbewegung beginnt von dieser Position.

Hinweis

"Nachführbetrieb" kann gesetzt bleiben, da es nur in Verbindung mit "Reglerfreigabe" wirkt.

Nachführbetrieb aufheben

Nach Aufhebung des Nachführbetriebes ist keine erneutes Referenzpunktfahren der Maschinenachse erforderlich, falls die maximal zulässige Gebergrenzfrequenz des aktiven Messsystems während des Nachführbetriebes nicht überschritten wurde. Wird die Gebergrenzfrequenz überschritten, wird dies von der Steuerung erkannt:

- DB31, ... DBX60.4 / 60.5 = 0 (Referenziert / Synchronisiert 1 / 2)
- Alarm: "21610 Encoderfrequenz überschritten"

Hinweis

Wird der "Nachführbetrieb" für eine Maschinenachse ausgeschaltet, die Bestandteil einer aktiven Transformation (z.B. TRANSMIT) ist, kann es zu Verfahrbewegungen im Rahmen von Rückpositionierbewegungen (REPOS) von anderen, an der Transformation beteiligten Maschinenachsen kommen.

Überwachungen

Befindet sich eine Maschinenachse im Nachführbetrieb, sind folgende Überwachungen nicht wirksam:

- Stillstandsüberwachung
- Klemmungsüberwachung
- Positionierüberwachung

Auswirkungen auf andere Nahtstellensignale:

- DB31, ... DBX60.7 = 0 (Position erreicht mit Genauhalt fein)
- DB31, ... DBX60.6 = 0 (Position erreicht mit Genauhalt grob)

- Indirektes Motormesssystem
- Direktes Messsystem an der Last

1: LM1
2: LM2
3: Parkende Achse

Funktionalität der Nahstellensignale "Lagemesssystem 1 / 2" in Zusammenhang mit der "Reglerfreigabe":

DB31, ... DBX1.5	DB31, ... DBX1.6	DB31, ... DBX2.1	Funktion
1	beliebig	1	Lagemesssystem 1 aktiv
0	1	1	Lagemesssystem 2 aktiv
0	0	0	"Parken" aktiv
0	0	1	Spindel ohne Lagemesssystem (drehzahlge- regelt)
1 → 0	0 → 1	1	Umschaltung: Lagemesssystem 1 → 2
0 → 1	1 → 0	1	Umschaltung: Lagemesssystem 2 → 1

DB31, ... DBX2.1 == 0

Aktivierungsarten

Die regelungstechnische Reglerfreigabe einer Maschinenachse wird beeinflusst durch:

- NC/PLC-Nahtstellensignal:
 - DB31, ... DBX2.1 (Reglerfreigabe)
 - DB31, ... DBX21.7 (Impulsfreigabe)
 - DB31, ... DBX93.5 (Drive Ready)
 - DB10, DBX56.1 (Not-Halt)
- NC-intern
Alarme, die als Alarmreaktion die Reglerfreigabe der Maschinenachsen wegnehmen. Die Alarmreaktionen sind beschrieben in:
Literatur:
Diagnosehandbuch

Wegnahme der Reglerfreigabe bei stehender Maschinenachse:

- Der Lageregelkreis der Maschinenachse wird geöffnet
- DB31, ... DBX61.5 == 0 (Lageregler aktiv)

Wegnahme der Reglerfreigabe bei fahrender Maschinenachse:

Ist eine Maschinenachse Teil einer interpolatorischen Bahnbewegung oder Kopplung und wird für diese die Reglerfreigabe weggenommen, werden alle betroffenen Achsen mit Schnellstopp (Drehzahlsollwert = 0) stillgesetzt und ein Alarm angezeigt:

Alarm: "21612 Reglerfreigabe zurückgesetzt während der Bewegung"

- Die Maschinenachse wird, unter Berücksichtigung der parametrierten Zeitdauer der Bremsrampe, bei Fehlerzuständen mit Schnellstopp (Drehzahlsollwert = 0) abgebremst:
MD36610 \$MA_AX_EMERGENCY_STOP_TIME (max. Zeitdauer der Bremsrampe bei Fehlern)
Ein Alarm wird angezeigt:
Alarm: "21612 Reglerfreigabe zurückgesetzt während der Bewegung"

Hinweis

Die Reglerfreigabe wird spätestens nach Ablauf der Abschaltverzögerungszeit weggenommen:

MD36610 \$MA_AX_EMERGENCY_STOP_TIME

- Der Lageregelkreis der Maschinenachse wird geöffnet. Rückmeldung über Nahtstellensignal:
DB31, ... DBX61.5 == 0 (Lageregler aktiv).
Die Zeit für die parametrierte Abschaltverzögerung der Reglerfreigabe wird gestartet durch das Maschinendatum:
MD36620 \$MA_SERVO_DISABLE_DELAY_TIME (Abschaltverzögerung Reglerfreigabe)
- Sobald die Istgeschwindigkeit den Stillstandsbereich erreicht, wird die Reglerfreigabe des Antriebs weggenommen. Rückmeldung über Nahtstellensignal:
DB31, ... DBX61.6 == 0 (Drehzahlregler aktiv)

- Der Lageistwert der Maschinenachse wird von der Steuerung weiterhin erfasst.
- Am Ende des Bremsvorganges wird die Maschinenachse unabhängig vom entsprechenden NC/PLC-Nahtstellensignal in den Nachführbetrieb geschaltet. Stillstands- und Klemmungsüberwachung sind dabei unwirksam. Siehe oben die Beschreibung zum Nahtstellensignal:
DB31, ... DBX1.4 (Nachführbetrieb).

Istwert synchronisieren (Referenzpunktfahren)

Nach dem Setzen der Reglerfreigabe ist keine erneutes Synchronisieren der Istposition der Maschinenachse (Referenzpunktfahren) erforderlich, falls während der Zeit in der die Maschinenachse nicht in Lageregelung war, die maximal zulässige Grenzfrequenz des Messsystems nicht überschritten wurde.

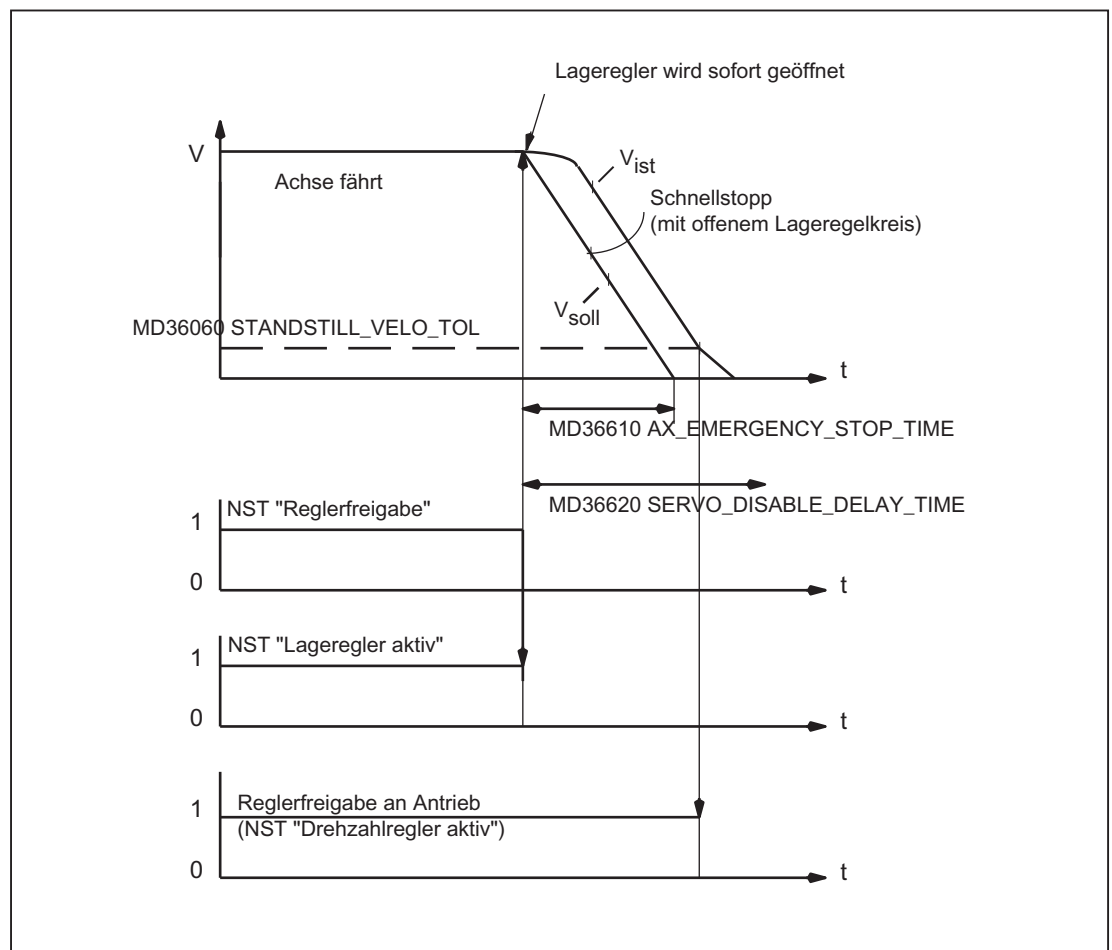


Bild 2-5 Wegnahme der Reglerfreigabe bei fahrender Maschinenachse

DB31, ... DBX2.2 (Restweg/Spindel-Reset (achs-/spindelspezifisch))

"Restweg löschen" wirkt in den Betriebsarten: AUTOMATIK bzw. MDA nur im Zusammenhang mit Positionierachsen. Die Positionierachse wird dabei über die aktuelle Bremskennlinie bis zum Stillstand abgebremst. Der noch nicht abgefahrte Restweg der Achse wird gelöscht.

Spindel-Reset

Eine ausführliche Beschreibung des Spindel-Reset siehe Kapitel "S1: Spindeln (Seite 1295)".

DB31, ... DBX9.0 / 9.1 / 9.2 (Regler-Parametersatz)

Anforderung zur Aktivierung des vorgegebenen Regler-Parametersatzes.

Regler-Parametersatz	DBX9.2	DBX9.1	DBX9.0
1	0	0	0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1

Die Parametersatzumschaltung muss freigegeben sein (nicht erforderlich bei Spindeln) über das Maschinendatum:

MD35590 \$MA_PARAMSET_CHANGE_ENABLE = 1 oder 2

Ausführliche Informationen zur Parametersatzumschaltung siehe Kapitel "Parametersatzanwahl beim Getriebestufenwechsel (Seite 1351)".

Parametersatzumschaltung bei fahrender Maschinenachse

Das Verhalten bei Parametersatzumschaltung ist abhängig von der dabei stattfindenden Änderung des Regelkreis-Verstärkungsfaktor K_V :

MD32200 \$MA_POSCTRL_GAIN (KV-Faktor)

- "Gleiche K_V -Faktoren" oder "Lageregelung nicht aktiv":
Die NC reagiert sofort auf die Parametersatzumschaltung. Der Parametersatz wird auch während der Bewegung gewechselt.
- "Ungleiche K_V -Faktoren" und "Lageregelung aktiv":
Um eine möglichst ruckfreie Umschaltung durchführen zu können, wird mit der Parametersatzumschaltung so lange gewartet, bis die Achse "steht", d. h. die parametrisierte Stillstandsgeschwindigkeit erreicht bzw. unterschritten wurde:
DB31, ... DBX61.4 = 1 (Achse/Spindel steht)
MD36060 \$MA_STANDSTILL_VELO_TOL (Schwellgeschwindigkeit/Drehzahl "Achse/Spindel steht")

Parametersatzumschaltung aus dem Teileprogramm

Zur Parametersatzumschaltung aus dem Teileprogramm sind vom Anwender (Maschinenhersteller) entsprechende Anwender-spezifische Hilfsfunktionen zu definieren und

im PLC-Anwenderprogramm auszuwerten. Das PLC-Anwenderprogramm setzt daraufhin die Anforderung zur Umschaltung auf den entsprechenden Parametersatz.

Ausführliche Informationen zur Hilfsfunktionsausgabe siehe Kapitel "H2: Hilfsfunktionsausgaben an PLC (Seite 399)".

DB31, ... DBX9.3 (Parametersatzvorgabe von NC gesperrt)

Bei gesetztem Nahtstellensignal wird eine Anforderung zur Regler-Parametersatzumschaltung (DB31, ... DBX9.0 / 9.1 / 9.2) ignoriert.

2.2.7 Signale von Achse/Spindel

DB31, ... DBX61.0 (Antriebtest Fahrenanforderung)

Werden Maschinenachsen von speziellen Testfunktionen wie z. B. "Funktionsgenerator" verfahren, wird für die Verfahrbewegung eine explizite Antriebtest-spezifische Freigabe angefordert:

DB31, ... DBX61.0 == 1 (Antriebtest Fahrenanforderung)

Die Verfahrbewegung wird ausgeführt, sobald die Fahrfreigabe erfolgt ist:

DB31, ... DBX1.0 == 1 (Antriebtest Fahrfreigabe)

DB31, ... DBX61.3 (Nachführen aktiv)

Die Maschinenachse befindet sich im Nachführbetrieb.

DB31, ... DBX61.4 (Achse/Spindel steht ($n < n_{\min}$))

"Achse/Spindel steht" wird von der NC gesetzt, wenn:

- Keine neuen Sollwerte mehr ausgegeben werden **UND**
- Die Istgeschwindigkeit der Maschinenachse kleiner der parametrisierten Stillstandsgeschwindigkeit ist:
MD36060 \$MA_STANDSTILL_VELO_TOL (Schwellgeschwindigkeit Achse steht)

DB31, ... DBX61.5 (Lageregler aktiv)

Der Lageregelkreis der Maschinenachse ist geschlossen und die Lageregelung ist aktiv.

DB31, ... DBX61.6 (Drehzahlregler aktiv)

Der Drehzahlregelkreis der Maschinenachse ist geschlossen und die Drehzahlregelung ist aktiv.

DB31, ... DBX61.7 (Stromregler aktiv)

Der Stromregelkreis der Maschinenachse ist geschlossen und die Stromregelung ist aktiv.

DB31, ... DBX69.0 / 69.1 / 69.2 (Parametersatz Servo)

Aktiver Parametersatz. Codierung entsprechend:

DB31, ... DBX9.0 / 9.1 / 9.2 (Reglerparametersatz Anwahl)

DB31, ... DBX76.0 (Schmierimpuls)

Nach Power On/Reset der Steuerung ist der Signalzustand 0 (FALSE).

Der "Schmierimpuls" wird **invertiert** (Flankenwechsel), sobald die Maschinenachse die parametrisierte Verfahrstrecke für Schmierung zurückgelegt hat:

MD33050 \$MA_LUBRICATION_DIST (Verfahrstrecke für Schmierung von PLC)

2.2.8 Signale an Achse/Spindel (Digitale Antriebe)

DB31, ... DBX21.0 - 4 (Anforderung zum Umschaltung von Motor- und/oder Antriebsdatensatz)

Von der PLC aus wird die Anforderung an den Antrieb gestellt, auf den neuen Motor- und/oder Antriebsdatensatz umzuschalten.

Die Schnittstelle ist flexibel parametrierbar über: DB31,DBX130.0 - 4 (siehe Kapitel "Umschalten von Motor- / Antriebsdatensätzen (Seite 70)")

Die Anzeige des aktiven Motor- und/oder Antriebsdatensatzes erfolgt über: DB31, ... DBX93.0 - 4

Hinweis

Hauptspindelantrieb

Bei einem Hauptspindelantrieb sind nur die Motordatensätze 1 und 2 zum Umschalten zwischen Stern- und Dreieckbetrieb gültig:

- Motordatensatz 1: Sternbetrieb
 - Motordatensatz 2: Dreieckbetrieb
-

DB31, ... DBX21.5 (Motor-Anwahl erfolgt)

Damit meldet das PLC-Anwenderprogramm an den Antrieb den Abschluss der Motor-Anwahl. Anschließend werden vom Antrieb die Impulse freigegeben.

DB31, ... DBX21.6 (Integratorsperre n-Regler)

Das PLC-Anwenderprogramm sperrt beim Antrieb den Integrator des Drehzahlreglers. Der Drehzahlregler wird dabei von PI- auf P-Regler umgeschaltet.

Bei Aktivierung der Integratorsperre des Drehzahlreglers können je nach Anwendungsfall Ausgleichsvorgänge auftreten (z.B. wenn der Integrator zuvor stationär eine Last hält).

Rückmeldung über: DB31, ... DBX93.6 = 1 (Integrator n-Regler gesperrt).

DB31, ... DBX21.7 (Anforderung Impulsfreigabe)

Die Impulsfreigabe für das Antriebsmodul wird nur angefordert, wenn alle Freigabesignale (Hardware und Software) anstehen:

- Steuersatzfreigabe
- Regler- und Impulsfreigabe
- Impulsfreigabe (sicherer Betriebshalt)
- Gespeicherter Hardwareeingang
- Sollwertfreigabe
- "Zustand fahrbereit"
 - Kein Antriebsalarm (ZK1-Fehler)
 - Zwischenkreis zugeschaltet
 - Hochlauf abgeschlossen

Rückmeldung über: DB31, ... DBX93.7 (Impulse freigegeben)

2.2.9 Signale von Achse/Spindel (Digitale Antriebe)

DB31, ... DBX92.1 (Hochlaufgebersperre aktiv)

Vom Antrieb wird an die PLC zurückgemeldet, dass der Hochlaufgeber-Schnellstopp aktiv ist. Damit wird der Antrieb ohne Hochlaufgeberrampe (mit Drehzahlsollwert 0) stillgesetzt.

DB31, ... DBX93.0 - 4 (Anzeige aktiver Motor- und Antriebsdatensatz)

Vom Antrieb wird der aktive Motor- und Antriebsdatensatz an die PLC zurückgemeldet.

Die Schnittstelle ist flexibel parametrierbar über: DB31, ... DBX130.0 - 4 (siehe Kapitel "Umschalten von Motor- / Antriebsdatensätzen (Seite 70)")

Die Anforderung zum Umschalten des Motor- und/oder Antriebsdatensatzes erfolgt über: DB31, ... DBX21.0 - 4

DB31, ... DBX93.5 (DRIVE ready)

Rückmeldung, dass der Antrieb betriebsbereit ist. Damit sind die Voraussetzungen vom Antrieb für ein Verfahren der Achse/Spindel gegeben.

DB31, ... DBX93.6 (Integrator n-Regler gesperrt)

Der Integrator des Drehzahlreglers ist gesperrt. Der Drehzahlregler wurde somit von PI- auf P-Regler umgeschaltet.

DB31, ... DBX93.7 (Impulse freigegeben)

Die Impulsfreigabe für das Antriebsmodul ist vorhanden. Damit kann die Achse/Spindel verfahren werden.

DB31, ... DBX94.0 (Temperaturvorwarnung Motor)

Die Temperatur im Motor ist größer als die eingestellte Motortemperatur-Warnschwelle (Antriebsparameter p0604).

Siehe auch unten Hinweis zu "DB31, ... DBX94.1 (Temperaturvorwarnung Kühlkörper)".

DB31, ... DBX94.1 (Temperaturvorwarnung Kühlkörper)

Die Temperatur des Kühlkörpers im Leistungsteil ist außerhalb des zulässigen Bereichs. Bleibt die zu hohe Temperatur bestehen, schaltet der Antrieb nach ca. 20 s ab.

Hinweis

Temperaturvorwarnung DB31, ... DBX94.0 und DBX94.1

Die Nahtstellensignale werden von folgenden Signalen des zykl. Antriebstelegramms abgeleitet:

- Fall 1: Temperaturwarnung im Meldungswort
 - DB31, ... DBX94.0 = MELDW, Bit 6 (Keine Warnung Übertemperatur Motor)
 - DB31, ... DBX94.1 = MELDW, Bit 7 (Keine Warnung thermische Überlast Leistungsteil)
- Fall 2: Warnung der Warnungsklasse B (nur im Interface Mode "SIMODRIVE 611u", p2038=1)
DB31, ... DBX94.0 == 1 und DBX94.1 == 1, wenn gilt:
zykl. Antriebstelegramm, Zustandswort 1 (ZSW1), Bit 11 / 12 == 2 (Warnungsklasse B)

Die Nahtstellensignale werden aus der Warnung der Warnklasse B abgeleitet, wenn keine spezifische Information aus dem Meldewort vorliegt.

Es wird ein Alarm angezeigt: Alarmnummer = 200.000 + Warnwert (r2124)

Für eine ausführliche Beschreibung zur Einstellung der Motortemperaturüberwachung siehe:

Literatur:

- S120 Inbetriebnahmehandbuch, Kapitel "Inbetriebnahme" > "Temperatursensoren bei SINAMICS-Komponenten"
 - S120 Funktionshandbuch, Kapitel "Überwachungs- und Schutzfunktionen"
 - S120 Listenhandbuch
 - MELDW, Bit 6 \triangleq BO: r2135.14 \rightarrow Funktionsplan: 2548, 8016
 - MELDW, Bit 7 \triangleq BO: r2135.15 \rightarrow Funktionsplan: 2548, 2452, 2456, 8016
-

DB31, ... DBX94.2 (Hochlaufvorgang beendet)

Der Drehzahlwert liegt nach einer Änderung des Drehzahlsollwerts wieder innerhalb des parametrisierten Toleranzbandes. Der Hochlaufvorgang ist damit abgeschlossen.

Nachfolgende Drehzahlschwankungen, auch außerhalb des Toleranzbandes z. B. infolge Belastungsänderungen, haben auf das Nahtstellensignal keinen Einfluss.

DB31, ... DBX94.3 ($|M_d| < M_{dx}$)

Der Betrag des aktuellen Moments $|M_d|$ ist kleiner als das parametrisierte Schwellenmoment M_{dx} (Drehmomentschwellwert 2, p2194).

Das Schwellenmoment wird in Prozent [%], bezogen auf die aktuelle drehzahlabhängige Momentenbegrenzung, eingestellt.

DB31, ... DBX94.4 ($|n_{ist}| < n_{min}$)

Der Drehzahlwert n_{ist} ist kleiner als n_{min} (Drehzahlschwellwert 3, p2161).

DB31, ... DBX94.5 ($|n_{ist}| < n_x$)

Der Drehzahlwert n_{ist} ist kleiner als n_x (Drehzahlschwellwert 2, p2155).

DB31, ... DBX94.6 ($n_{ist} = n_{soll}$)

Der Drehzahlwert liegt innerhalb des Toleranzbandes (p2163) um den Drehzahlsollwert.

DB31, ... DBX95.7 (Warnung der Warnungsklasse C steht an)

Der Antrieb meldet, dass eine Warnung der Warnungsklasse C ansteht.

2.3 Funktionen

2.3.1 Verhalten der Bildaktualisierung bei Überlastung - nur 840D sl

Es gibt Teileprogramme, bei denen der Hauptlauf (HL) warten muss, bis der Vorlauf (VL) neue Sätze zur Verfügung stellt.

Vorlauf und Anzeige-Aktualisierung konkurrieren um die NC-Rechenzeit.

Über das folgende Maschinendatum kann eingestellt werden, wie sich die NC verhalten soll, wenn der Vorlauf zu langsam ist:

MD10131 \$MN_SUPPRESS_SCREEN_REFRESH

Wert	Bedeutung
0	Wenn der VL eines Kanals zu langsam ist, wird die Aktualisierung der Anzeige in allen Kanälen unterdrückt.
1	Wenn der VL eines Kanals zu langsam ist, wird die Aktualisierung der Anzeige nur in den zeitkritischen Kanälen unterdrückt, um Rechenzeit für den Vorlauf zu gewinnen.
2	Die Aktualisierung der Anzeige wird grundsätzlich nicht unterdrückt.

2.3.2 Einstellungen für Evolventen-Interpolation - nur 840D sl

Einführung

Die Evolvente des Kreises ist eine Kurve, die vom Endpunkt eines fest gespannten, von einem Kreis abgewickelten Fadens beschrieben wird. Die Evolventen-Interpolation ermöglicht Bahnkurven entlang einer Evolvente.

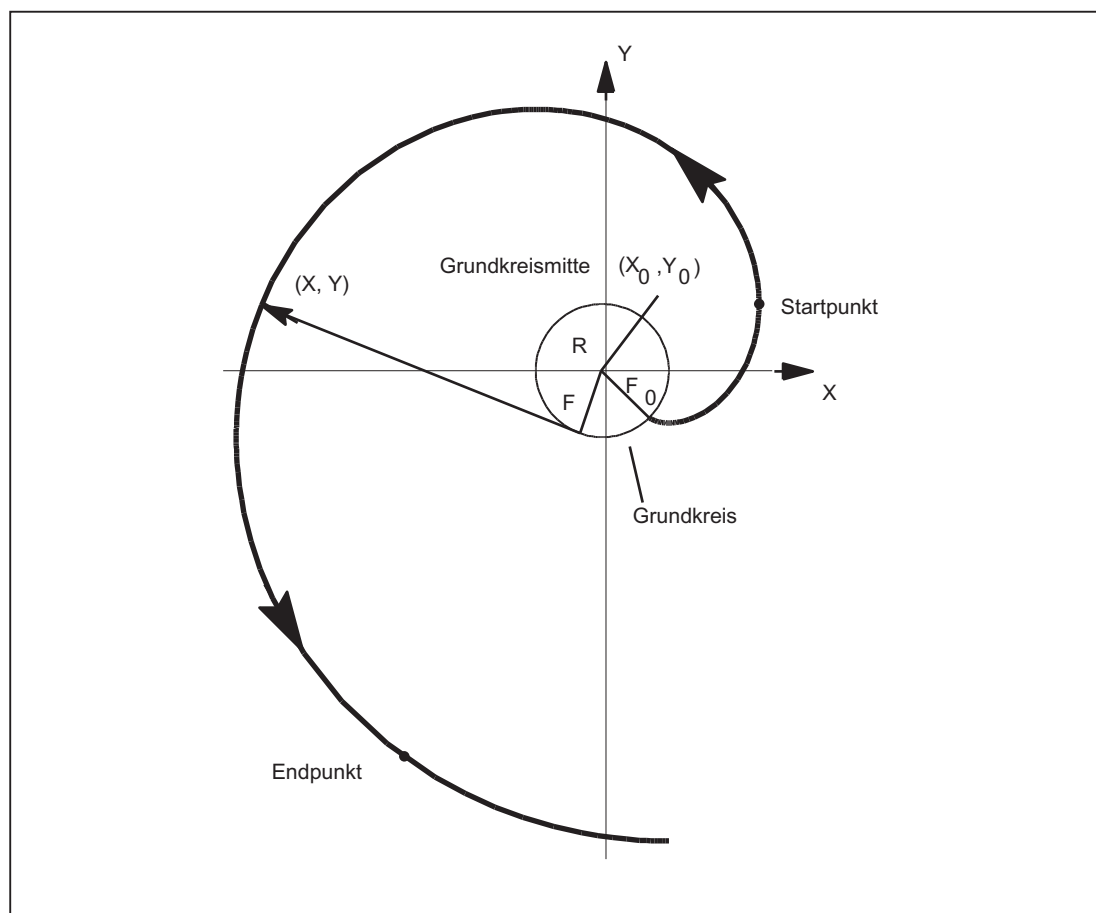


Bild 2-6 Evolvente (vom Grundkreis weg)

Programmierung

Die Programmierung der Evolventen-Interpolation ist allgemein beschrieben in:

Literatur:

Programmierhandbuch Grundlagen

Für zwei Fälle der Evolventen-Interpolation haben Maschinendaten zusätzlich zu den programmierten Parametern eine Bedeutung, die ggf. auch durch den Maschinenhersteller/Endbenutzer eingestellt werden müssen.

Genauigkeit

Falls der programmierte Endpunkt nicht exakt auf der durch den Startpunkt festgelegten Evolventen liegt, wird zwischen den beiden Evolventen, die durch den Startpunkt bzw. den Endpunkt definiert sind, interpoliert (siehe nachfolgende Abbildung).

Die maximale Abweichung des Endpunkts wird festgelegt durch das Maschinendatum:

MD21015 \$MC_INVOLUTE_RADIUS_DELTA(Endpunktüberwachung bei Evolvente)

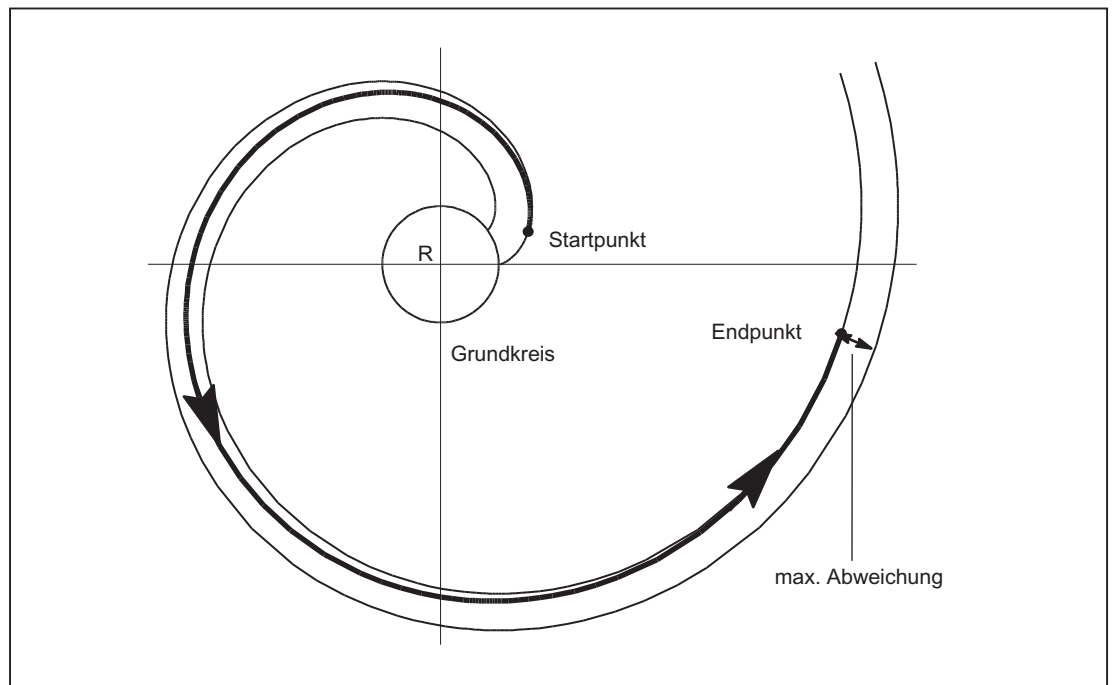


Bild 2-7 MD21015 legt die max. zulässige Abweichung fest

Grenzwinkel

Wird mit AR eine zum Grundkreis führende Evolvente mit einem Drehwinkel programmiert, der größer als der maximal mögliche Wert ist, wird ein Alarm ausgegeben und die Programmverarbeitung gestoppt.

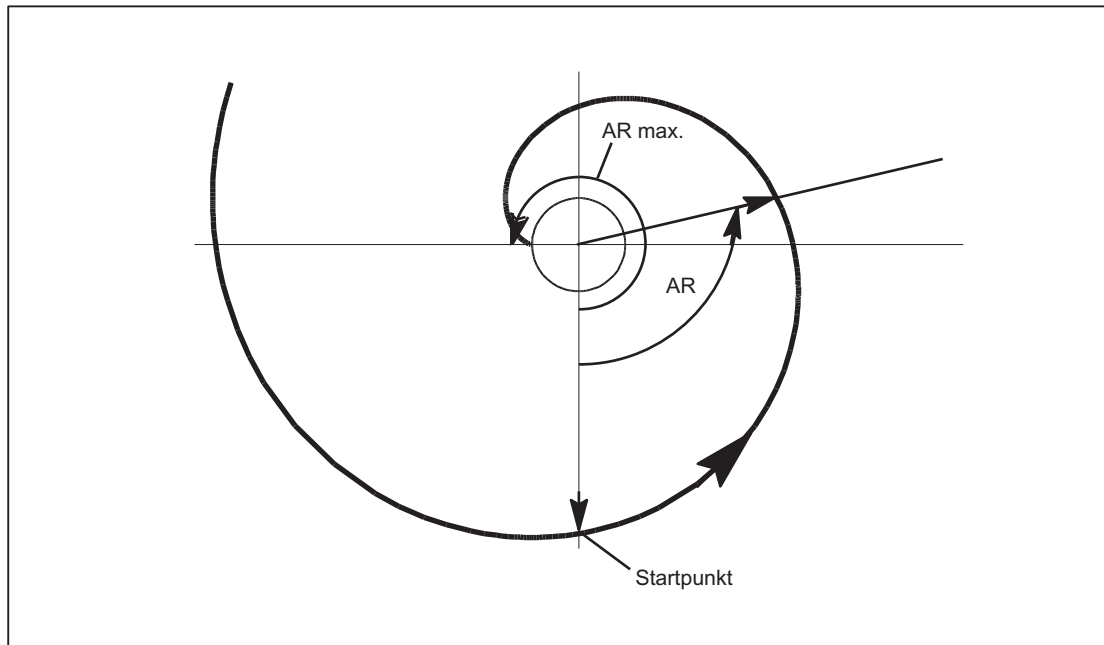


Bild 2-8 Begrenzter Drehwinkel zum Grundkreis hin

Die Anzeige des Alarms kann über folgende Parametrierung unterdrückt werden:

MD21016 \$MC_INVOLUTE_AUTO_ANGLE_LIMIT = TRUE (automatische Winkelbegrenzung bei Evolventen-Interpolation)

Dann wird der programmierte Drehwinkel gegebenenfalls automatisch begrenzt und die interpolierte Bahn endet an dem Punkt an dem die Evolvente den Grundkreis trifft. Dies erlaubt die einfache Programmierung einer Evolvente, die von einem Punkt außerhalb des Grundkreises startet und direkt auf dem Grundkreis endet.

Werkzeugradiuskorrektur

Für Evolventen ist nur die 2-1/2D-Werkzeugradiuskorrektur zulässig. Bei aktiver 3D-Werkzeugradiuskorrektur (sowohl Umfangs- als auch Stirnfräsen) wird bei Programmierung einer Evolvente die Bearbeitung mit Alarm 10782 abgebrochen.

Bei 2-1/2-D-Werkzeugradiuskorrektur muss die Ebene der Evolvente in der Korrekturebene liegen. Andernfalls wird der Alarm 10781 erzeugt. Ein zusätzlicher Helixanteil zu einer in der Korrekturebene liegenden Evolvente ist jedoch zulässig.

Dynamik

Evolventen, die auf dem Grundkreis beginnen oder enden, haben an dieser Stelle eine unendliche Krümmung. Damit die Geschwindigkeit in diesem Punkt bei aktiver Werkzeugradiuskorrektur hinreichend beschränkt wird, ohne diese an anderen Stellen zu stark zu begrenzen, muss die Funktion "Geschwindigkeitsbegrenzungsprofile" aktiviert werden:

MD28530 \$MC_MM_PATH_VELO_SEGMENTS > 1 (Anzahl Speicherelemente zur Begrenzung der Bahngeschwindigkeit)

Empfohlen wird der Wert 5. Diese Einstellung ist nicht notwendig, wenn nur Evolventenstücke verwendet werden, deren Krümmungsradien sich über einen relativ kleinen Bereich ändern.

2.3.3 DEFAULT-Speicher aktivieren - nur 840D sl

GUD Startwerte

Mit den Sprachbefehlen DEF... / REDEF... können globalen Anwendervariablen (GUD) Defaultwerte zugewiesen werden. Damit diese Defaultwerte zum parametrierten Initialisierungszeitpunkt, z. B. durch das Attribut `INIPO` nach Power On, verfügbar sind, müssen diese im System dauerhaft gespeichert werden. Der dafür erforderliche Speicherplatz muss über folgendes Maschinendatum freigegeben werden:

MD18150 \$MM_GUD_VALUES_MEM (remanenter Speicherplatz für GUD-Werte)

Literatur:

- Funktionshandbuch Erweiterungsfunktionen; S7: "Speicherkonfiguration"
- Programmierhandbuch Arbeitsvorbereitung

2.3.4 PLC-Variable lesen und schreiben - nur 840D sl

Schneller Datenkanal

Für den schnellen Austausch von Informationen zwischen PLC und NC wird in dem Koppelspeicher dieser Baugruppen (DPR) ein Speicherbereich reserviert. In diesem Speicherbereich können beliebige PLC-Variablen (E/A, DB, DW, Merker) ausgetauscht werden.

Auf diesen Speicher wird von der PLC mit 'FunctionCalls' (FC), von der NC mit Systemvariablen zugegriffen.

Organisation des Speicherbereichs

Für die Organisation (Struktur) des Speicherbereiches ist der Anwenderprogrammierer (NC und PLC) selbst verantwortlich.

Dabei kann jede beliebige Speicherstelle angesprochen werden, jedoch muss dabei die Grenze entsprechend dem Datenformat gewählt werden (ein DWORD also auf einer 4-Byte-Grenze, ein WORD auf einer 2-Byte-Grenze ...).

Der Zugriff in den Speicherbereich erfolgt über den Datentyp und den Positionsoffset innerhalb des Speicherbereichs.

Zugriff von NC

Für den schnellen Zugriff aus einem Teileprogramm oder Synchronaktion auf PLC-Variable, werden in der NC Systemvariablen zur Verfügung gestellt. Das Schreiben / Lesen der Daten durch die NC erfolgt direkt. Der Datentyp ergibt sich aus dem Bezeichner der Systemvariablen. Die Position innerhalb des Speicherbereichs wird als Index in Byte angegeben.

Systemvariable	Datentyp	Wertebereich
\$A_DBB(<Index>)	Byte (8 Bit)	$0 \leq x \leq 255$
\$A_DBW(<Index>)	Wort (16 Bit)	$-32768 \leq x \leq 32767$
\$A_DBD(<Index>)	Doppelwort (32 Bit)	$-2147483648 \leq x \leq 2147483647$
\$A_DBR(<Index>)	Fließkomma (32 Bit)	$\pm(1,5 \cdot 10^{-45} \leq x \leq 3,4 \cdot 10^{38})$

Zugriff von PLC

Der Zugriff von der PLC erfolgt mittels "FunctionCall" (FC). Im FC werden die Daten sofort und nicht erst bei Zyklusbeginn der PLC, im DPR gelesen bzw. geschrieben. Datentyp und Position im Speicherbereich werden dem FC als Parameter übergeben.

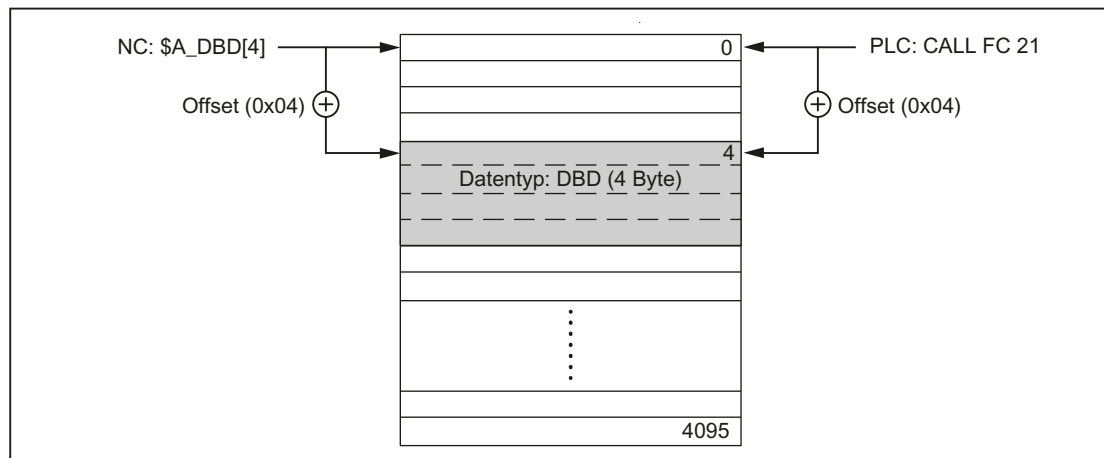


Bild 2-9 Koppelspeicher DPR für die NC/PLC-Kommunikation

Randbedingungen

- Die Strukturierung des DPR-Speicherbereichs liegt ausschließlich in der Verantwortung der Anwenders. Es werden keine Überprüfungen auf übereinstimmende Projektierung vorgenommen.
- In Ein- und Ausgaberrichtung stehen in Summe 4096 Bytes zur Verfügung.
- Einzelbitoperationen werden nicht unterstützt und müssen vom Anwender auf Byte-Operationen zurückgeführt werden.

- Da die Manipulation der Inhalte der Variablen im Koppelspeicher unmittelbar erfolgt, ist durch den Anwenderprogrammierer darauf zu achten, dass bei mehrfacher Auswertung einer Variablen oder bei der Verknüpfung von Variablen sich die Werte zwischenzeitlich geändert haben (ggf. Werte in lokaler Variable oder R-Parameter zwischenspeichern oder eine Semaphore einrichten).
- Für die Koordination der Zugriffe von verschiedenen Kanälen auf den Koppelspeicher ist der Anwenderprogrammierer verantwortlich.
- Es wird nur für die Datenzugriffe bis 16 Bit (Byte und Word) Datenkonsistenz zugesichert. Für konsistente Übertragung von 32 Bit-Variablen (Double und Real) ist ausschließlich der Anwender verantwortlich. Auf der PLC-Seite steht dazu ein einfacher Semaphor-Mechanismus zur Verfügung.
- Die Daten werden von der PLC im 'Little Endian'-Format im DPR abgelegt.
- Mit \$A_DBR transferierte Werte unterliegen einer Datenwandlung und damit einem Genauigkeitsverlust. Das Datenformat für Fließkommazahlen ist in der NC DOUBLE (64Bit), in der PLC jedoch nur FLOAT (32Bit). Die Ablage im DPR ist FLOAT. Die Wandlung erfolgt jeweils vor/nach der Ablage im DPR.
Wird z.B. von der NC auf eine Variable im DPR geschrieben und wieder zurückgegeben, wird die Wandlung zweimal durchlaufen. Eine Differenz des geschriebenen/gelesenen Wertes ist aufgrund der Datenablage in den beiden Formaten nicht zu verhindern.

Beispiel

Umgehung des Problems durch Vergleich auf "EPSILON" (kleine Abweichung)

Programmcode	
N10	DEF REAL DBR
N12	DEF REAL EPSILON = 0.00001
N20	\$A_DBR[0]=145.145
N30	G4 F2
N40	STOPRE
N50	DBR=\$A_DBR[0]
N60	IF (ABS(DBR/145.145-1.0) < EPSILON) GOTOF ENDE
N70	MSG ("Fehler")
N80	M0
N90	ENDE:
N99	M30

Aktivierung

Die maximale Anzahl der gleichzeitig schreibbaren Ausgangsvariablen ist einstellbar über: MD28150 \$MC_MM_NUM_VDIVAR_ELEMENTS (Anzahl Elemente für das schreiben von PLC-Variablen)

Beispiel

Es soll eine Variable vom Typ WORD von der PLC an die NC übergeben werden.

Der Positions-Offset innerhalb des NC-Eingangs (PLC-Ausgangsbereich) soll das 4. Byte sein. Der Positions-Offset muss ein ganzzahliges Vielfaches der Datenbreite sein.

Schreiben von der PLC:

Programmcode	Kommentar
<pre> . . . CALL FC21 (Enable :=M10.0, Funct :=B#16#4, S7Var :=P#M 104.0 WORD1, IVAR1 :=04, IVAR2 :=-1, Error :=M10.1, ErrCode :=MW12); . . .) </pre>	<pre> ; wenn TRUE, dann FC21 aktiv </pre>

Lesen im Teileprogramm

Programmcode	Kommentar
<pre> . . . PLCDATA = \$A_DBW[4]; . . . </pre>	<pre> ; Lesen eines Wortes </pre>

Verhalten bei POWER ON, Satzsuchlauf

Bei "POWER ON" wird der Koppelspeicher DPR initialisiert.

Bei "Satzsuchlauf" wird die Ausgabe der PLC-Variablen aufgesammelt und mit dem Anfahr Satz in den Koppelspeicher DPR ausgegeben (analog dem Schreiben von analogen und digitalen Ausgängen).

Andere Zustandsübergänge haben hier keine Auswirkung.

Literatur

Eine ausführliche Beschreibung zum Datenaustausch seitens der PLC mit FC 21 findet sich in:

SINUMERIK 840D sl: Kapitel "Funktion (Seite 1108)"

2.3.5 Zugriffsschutz über Kennwort und Schlüsselschalter

Zugriffsrechte

Der Zugriff auf Funktionen, Programme und Daten und ist benutzerorientiert über 8 hierarchische Schutzstufen geschützt. Diese sind unterteilt in:

- Kennwort-Stufen für Siemens, Maschinenhersteller und Endanwender
- Schlüsselschalter-Stellungen für Endanwender

Mehrstufiges Sicherheitskonzept

Über die Kennwort-Stufen und Schlüsselschalter-Stellungen steht ein mehrstufiges Sicherheitskonzept zur Regelung der Zugriffsrechte zur Verfügung.

Schutzstufe	Art	Benutzer	Zugriff auf (Beispiele)
0	Kennwort	Siemens	alle Funktionen, Programme und Daten
1	Kennwort	Maschinenhersteller: Entwicklung	definierte Funktionen, Programme und Daten; z.B.: Optionen eingeben
2	Kennwort	Maschinenhersteller: Inbetriebnehmer	definierte Funktionen, Programme und Daten; z.B.: Großteil der Maschinendaten
3	Kennwort	Endanwender: Service	zugeordnete Funktionen, Programme und Daten
4	Schlüsselschalter-Stellung 3	Endanwender: Programmierer Einrichter	weniger als Schutzstufe 0 bis 3; festgelegt vom Maschinenhersteller oder Endanwender
5	Schlüsselschalter-Stellung 2	Endanwender: Qualifizierter Bediener, der nicht programmiert	weniger als Schutzstufe 0 bis 3; festgelegt vom Endanwender
6	Schlüsselschalter-Stellung 1	Endanwender: Ausgebildeter Bediener, der nicht programmiert	Beispiel: nur Programmanwahl, Werkzeugverschleißeingabe und Eingabe von Nullpunktverschiebungen
7	Schlüsselschalter-Stellung 0	Endanwender: Angelernter Bediener	Beispiel: keine Eingaben und Programmanwahl möglich, nur Maschinensteuertafel bedienbar

Zugriffsmerkmale

- Die Schutzstufe 0 besitzt die höchsten Zugriffsrechte, die Schutzstufe 7 hat die geringsten Zugriffsrechte.
- Wird einer Schutzstufe ein Zugriffsrecht erteilt, so schließt die höhere Schutzstufe automatisch dieses Zugriffsrecht mit ein.
- Umgekehrt kann ein Zugriffsrecht für eine bestimmte Schutzstufe nur aus einer höheren Schutzstufe heraus geändert werden.
- Die Zugriffsrechte für die Schutzstufen 0 bis 3 werden von Siemens standardmäßig vorgegeben (Default).
- Die Zugriffsberechtigung wird durch Abfrage der aktuellen Schlüsselschalterstellung und durch Vergleich der eingegebenen Kennwörter gesetzt. Dabei überschreibt ein eingegebenes Kennwort die Zugriffsrechte der Schlüsselschalterstellung.
- In jeder Schutzstufe können Optionen gesichert werden. Aber nur in Schutzstufe 0 und 1 können Optionsdaten eingegeben werden.
- Die Zugriffsrechte für die Schutzstufen 4 bis 7 sind Vorschlagswerte und können vom Maschinenhersteller oder Endanwender geändert werden.

2.3.5.1 Kennwort

Kennwort setzen

Das Kennwort einer Schutzstufe (0 - 3) wird über die HMI-Bedienoberfläche eingegeben:

Bedienbereich "Inbetriebnahme" > "Kennwort" > "Kennwort setzen"

Kennwort löschen

Die Zugriffsberechtigung durch ein gesetztes Kennwort bleibt so lange wirksam, bis sie explizit durch Löschen des Kennworts zurückgenommen wird:

Bedienbereich "Inbetriebnahme" > "Kennwort" > "Kennwort löschen"

Kennwort ändern

Das Kennwort einer Schutzstufe (0 - 3) wird über die HMI-Bedienoberfläche geändert:

Bedienbereich "Inbetriebnahme" > "Kennwort" > "Kennwort ändern"

Hinweis

Warmstart

Eine Zugriffsberechtigung bzw. der Status des Kennworts (gesetzt / gelöscht) wird durch Warmstart nicht beeinflusst!

Neues Kennwort

Ein Kennwort kann maximal aus acht Zeichen bestehen. Bei der Wahl des Kennworts wird empfohlen, sich auf den Zeichenvorrat der Bedientafelfront zu beschränken.

Bei einem Kennwort mit weniger als 8 Zeichen werden die restlichen Zeichen als Leerzeichen (Blank) interpretiert.

Bei der Vergabe eines neuen Kennworts müssen die Regeln zur Vergabe von sicheren Passwörtern beachtet werden.

Hinweis

Sichere Passwörter vergeben

Beachten Sie bei der Vergabe von neuen Passwörtern die folgenden Regeln:

- Beachten Sie bei der Vergabe von neuen Passwörtern, dass Sie niemals leicht zu erratende Passwörter vergeben, z. B. einfache Wörter, leicht zu erratende Tastenfolgen auf der Tastatur, etc.
- Passwörter müssen immer eine Kombination aus Groß- und Kleinbuchstaben enthalten sowie Zahlen und Sonderzeichen. Passwörter müssen aus mindestens 8 Zeichen bestehen. PINS müssen aus einer willkürlichen Abfolge von Ziffern bestehen.
- Sie müssen – wo immer es möglich ist und wo es von den IT-Systemen und der Software unterstützt wird – stets die höchste Komplexität von Zeichen für die Wahl eines Passworts ansetzen.

Weitere Regeln zur Vergabe von sicheren Passwörtern finden Sie beim Bundesamt für Sicherheit in der Informationstechnik (BSI). (<https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/content/m/m02/m02011.html>)

Zur Unterstützung im Umgang mit Passwörtern können Sie ein Programm zur Passwortverwaltung nutzen. Mit dessen Hilfe können Sie Passwörter und Geheimzahlen verschlüsselt speichern, verwalten sowie sichere Passwörter erzeugen.

Standardeinstellungen

Standardmäßig sind für die Schutzstufen 1 - 3 folgende Kennwörter wirksam:

- Schutzstufe 1: SUNRISE
- Schutzstufe 2: EVENING
- Schutzstufe 3: CUSTOMER

ACHTUNG

Kaltstart bewirkt ein Rücksetzen der Kennwörter auf die Standardeinstellungen
--

Nach einem Kaltstart (Hochlauf der NC mit Laden der Standard-Maschinendaten) sind die Kennwörter der Schutzstufen 1 - 3 wieder auf die Standardeinstellungen zurückgesetzt. Aus Sicherheitsgründen wird daher nach einem Kaltstart dringend empfohlen, die Standardeinstellungen der Kennwörter zu ändern.
--

Literatur

- Inbetriebnahmehandbuch "SINUMERIK 840D sl Basesoftware und Bedien-Software"; Kapitel "SINUMERIK Operate (IM9)" > "Allgemeine Einstellungen" > "Zugriffsstufen"
- Inbetriebnahmehandbuch "Inbetriebnahme CNC: NC, PLC, Antrieb"; Kapitel "Voraussetzungen für die Inbetriebnahme" > "Einschalten und Hochlauf" und "Zugriffsstufen"

2.3.5.2 Schlüsselschalter-Stellungen (DB10, DBX56.4 bis 7)

Schlüsselschalter

Der Schlüsselschalter verfügt über vier Schalterstellungen, denen die Schutzstufen 4 bis 7 zugeordnet sind. Zum Schlüsselschalter gehören verschiedenfarbige Schlüssel, die in unterschiedlichen Schalterstellungen abgezogen werden können.





Schalterstellung	Abzugsstellung	DB10, DBB56	Schutzstufe
Stellung 0 	-	Bit 4	7
Stellung 1 	0 oder 1 schwarzer Schlüssel	Bit 5	6
Stellung 2 	0 oder 1 oder 2 grüner Schlüssel	Bit 6	5
Stellung 3 	0 oder 1 oder 2 oder 3 roter Schlüssel	Bit 7	4

Bild 2-10 Schalterstellungen 0 bis 3

Schalterstellungen

Schalterstellung 0 hat die geringsten Zugriffsrechte. Schalterstellung 3 hat die höchsten Zugriffsrechte:

DB10, DBX56.4 / .5 / .6 / .7 (Schalterstellungen 0 / 1 / 2 / 3)

Den Schalterstellungen können maschinenspezifische Freigaben für den Zugriff auf Programme, Daten und Funktionen zugeordnet werden. Ausführliche Informationen dazu finden sich in:

Literatur

- Inbetriebnahmehandbuch CNC: NC, PLC, Antriebe; Grundlagen, Kapitel: Prinzipielles zu den Schutzstufen
- Inbetriebnahmehandbuch SINUMERIK Operate (IM9); Allgemeine Einstellungen, Kapitel: Zugriffsstufen

Vorgaben durch das PLC-Anwenderprogramm

Die Schalterstellungen des Schlüsselschalters werden durch PLC-Grundprogramm in die NC/PLC-Nahtstelle übertragen. Die entsprechenden Nahtstellensignale können durch das PLC-Anwenderprogramm verändert werden. Dabei sollte aus Sicht der NC immer nur eine Schalterstellung aktiv, d.h. das entsprechende Nahtstellensignal auf den Wert 1 gesetzt sein. Sind aus Sicht der NC gleichzeitig mehrere Schalterstellungen aktiv, wird NC-intern die Schalterstellung 3, d.h. die Schlüsselschalterstellung mit den höchsten Zugriffsrechten aktiv.

2.3.5.3 Parametrierbare Schutzstufen**Parametrierbare Schutzstufe**

Für verschiedene Funktionen und Datenbereiche kann die Schutzstufe frei parametriert werden. Die Einstellung der Schutzstufe erfolgt über Bedientafel-Maschinendaten mit folgender Bezeichnungssystematik: \$MM_USER_CLASS_<Funktion_Datenbereich>

Beispiele:

\$MM_USER_CLASS_READ_TOA	Werkzeugkorrekturen lesen
\$MM_USER_CLASS_WRITE_TOA	Werkzeugkorrekturen schreiben
\$MM_USER_CLASS_READ_PROGRAM	Teileprogramme lesen
\$MM_USER_CLASS_WRITE_PROGRAM	Teileprogramme schreiben/editieren

Standardwerte

Bei Auslieferung oder nach einer Standard-Inbetriebnahme ist der Standardwert für die Schutzstufe bis auf wenige Ausnahmen auf 7, d.h. niedrigste Schutzstufe eingestellt.

2.3.6 Umschalten von Motor- / Antriebsdatensätzen

2.3.6.1 Allgemeine Informationen

Motor- und Antriebsdatensätzen

Zur optimalen Anpassung an die jeweilige Bearbeitungssituation oder aufgrund unterschiedlicher Maschinenkonfigurationen kann es erforderlich sein, dass in einem Antrieb für Motoren, Antriebsparameter und Geber jeweils mehrere unterschiedliche Datensätze vorliegen. Die Erstellung der grundlegenden Datensätze der Antriebsobjekte erfolgt während der Inbetriebnahme mit Hilfe des "Antriebsassistenten".

Hinweis

Literatur

Inbetriebnahmehandbuch: CNC: NC, PLC- Antrieb, Kapitel "Inbetriebnahme NC-gesteuerte Antriebe"

Die nachfolgende Duplizierung und Verwaltung der Datensätze erfolgt über die Bedienoberfläche:

SINUMERIK Operate: Bedienbereich "Inbetriebnahme" > "Antriebssystem" > "Antriebe" > "Datensätze"

Die Aktivierung des für eine Maschinenachse in der konkreten Bearbeitungssituation erforderlichen Motor- (MDS) bzw. Antriebsdatensatzes (DDS), muss vom PLC-Anwenderprogramm über die im Folgenden beschriebenen Schnittstellen vorgenommen werden.

Axiale NC/PLC-Nahtstelle

Die Schnittstellen in der axialen NC/PLC-Nahtstelle zur Umschaltung der Motor- und Antriebsdatensätze ist in drei Bereiche unterteilt:

- Formatierungsschnittstelle (Seite 70)
- Anforderungsschnittstelle (Seite 71)
- Anzeigeschnittstelle (Seite 72)

2.3.6.2 Formatierungsschnittstellen

Formatierung

Über die Formatierungsschnittstelle wird eingestellt, welche welche Bits der Anforderungs- und Anzeigeschnittstelle zur Adressierung der Motordatensätze (MDS) und welche zur Adressierung der Antriebsdatensätze (DDS) verwendet werden:

DB31, ... DBX130.0 - 4, mit Bit x = <Wert>

<Wert>	Bedeutung
0	Bit-Stelle für Motordatensatz (MDS) bzw. ungültige Bit-Stelle
1	Bit-Stelle für Antriebsdatensatz (DDS)

Motor- und Antriebsdatensätzen im Antrieb

Die Formatierung ist abhängig von der Anzahl der im Antrieb vorliegenden Motor- (MDS) und Antriebsdatensätzen (DDS). Die jeweilige Anzahl kann über folgende Antriebsparameter ermittelt werden:

- p0130 (Anzahl Motordatensätze)
- p0180 (Anzahl Antriebsdatensätze)

Gültigkeit der Schnittstellen

Sobald im Hochlauf der Steuerung alle erforderlichen Informationen vom Antrieb übermittelt und von der NC ausgewertet wurden, werden die Anforderungs (Seite 71)- und Anzeigeschnittstelle (Seite 72) als gültig angezeigt:

DB31, ... DBX130.7 == 1 (Anforderungs- und Anzeigeschnittstelle gültig)

Werden vom Antrieb keine oder inkompatible Informationen übertragen, werden die Anforderungs- und Anzeige-Schnittstellen als ungültig angezeigt.

Hinweis

Bei ungültigen Anforderungs- und Anzeigeschnittstellen liegt es in der alleinigen Verantwortung des Anwenders / Maschinenherstellers über die ungültigen Schnittstellen dennoch eine Datensatzumschaltung vorzunehmen.

Siehe auch

Beispiel (Seite 72)

Übersicht der Schnittstellen (Seite 73)

2.3.6.3 Anforderungsschnittstelle

Das Umschalten auf einen neuen Motor- (MDS) und/oder Antriebsdatensatz (DDS) wird angefordert über die Schnittstelle :

DB31, ... DBX21.0 - .4 = <MDS / DDS-Index>

Wertebereich

Die Adressierung eines Motor- oder Antriebsdatensatzes n, mit n = 1, 2, 3, ..., erfolgt anhand seines Index i, mit i = n - 1 = 0, 1, 2, ...

- Motordatensätze: MDS[0, 1, 2, ... 15]
- Antriebsdatensätze: DDS[0, 1, 2, ... 31]

Schnittstellenformatierung

Die Formatierung der Anforderungsschnittstelle, d. h. welche Bits zur Adressierung der Motordatensätze (MDS) und welche zur Adressierung der Antriebsdatensätze (DDS) verwendet werden, wird über die Formatierungsschnittstelle (Seite 70) eingestellt.

Motor- und Antriebsdatensätzen im Antrieb

Die Anzahl der im Antrieb vorliegenden Motor- (MDS) und Antriebsdatensätzen (DDS) kann über folgende Antriebsparameter ermittelt werden:

- p0130 (Anzahl Motordatensätze)
- p0180 (Anzahl Antriebsdatensätze)

Motordatensatz (MDS) bei Hauptspindelantrieben

Bei Hauptspindelantrieben gilt folgender Zusammenhang:

- MDS[0] → Stern-Betrieb
- MDS[1] → Dreieck-Betrieb

2.3.6.4 Anzeigeschnittstelle

Der aktive Motor- (MDS) und Antriebsdatensatzes (DDS) wird angezeigt über die Schnittstelle:

DB31, ... DBX93.0 - .4 == <MDS / DDS-Index>

Wertebereich und Formatierung sind identisch zur Anforderungsschnittstelle (Seite 71).

2.3.6.5 Beispiel

Im Antrieb sind zwei Motordatensätze (MDS) und zwei Antriebsdatensätze (DDS) pro Motordatensatz vorhanden. Dies entspricht "Nr.": 9 der im Bild 2-11 Prinzip der Motor- / Antriebsdatensatz-Umschaltung (Seite 74) dargestellten möglichen Datensatzkombinationen.

Format

Bit-Stellen für Antriebsdatensatz-Umschaltung (DDS):

- DB31, ... DBX130.0 == 1

Bit-Stellen für Motordatensatz -Umschaltung (MDS):

- DB31, ... DBX130.1 == 0

Ungültige Bit-Stellen:

- DB31, ... DBX130.2 == 0
- DB31, ... DBX130.3 == 0
- DB31, ... DBX130.4 == 0

Schnittstellen der Antriebsdatensätze (DDS)

Relevante Bit-Stellen der Anforderungs- und Anzeige-Schnittstellen:

- DB31, ... DBX21.0 / DBX93.0
 - DB31, ... DBX21.0 / DBX93.0 == 0 \Rightarrow 1. Antriebsdatensatz DDS[0]
 - DB31, ... DBX21.0 / DBX93.0 == 1 \Rightarrow 2. Antriebsdatensatz DDS[1]

Schnittstellen der Motordatensätze (MDS)

Relevante Bit-Stellen der Anforderungs- und Anzeige-Schnittstellen:

- DB31, ... DBX21.1 / DBX93.1
 - DB31, ... DBX21.1 / DBX93.1 == 0 \Rightarrow 1. Motordatensatz MDS[0]
 - DB31, ... DBX21.1 / DBX93.1 == 1 \Rightarrow 2. Motordatensatz MDS[1]

Ungültige Bit-Stellen (MDS / DDS)

Ungültige Bit-Stellen der Anforderungs- und Anzeige-Schnittstellen:

- DB31, ... DBX21.1 / DBX93.2 == 0
- DB31, ... DBX21.1 / DBX93.3 == 0
- DB31, ... DBX21.1 / DBX93.4 == 0

Siehe auch

Übersicht der Schnittstellen (Seite 73)

2.3.6.6 Übersicht der Schnittstellen

Tabelle 2-1 Konfigurierbare MDS / DDS Kombinationen

Anzahl MDS (Motoren)	Anzahl von DDS (Antriebe) pro MDS
1	1 ... 32
2	1, 2, 4, 8, 16
3	1, 2, 4, 8
4	1, 2, 4, 8
5	1, 2, 4
6	1, 2, 4
7	1, 2, 4
8	1, 2, 4
9	1, 2
10	1, 2
11	1, 2
12	1, 2
13	1, 2
14	1, 2

2.3 Funktionen

Anzahl MDS (Motoren)	Anzahl von DDS (Antriebe) pro MDS
15	1, 2
16	1, 2

Nr.	DDS pro MDS			DB31, ... DBX21.x / 93.x					DB31, ... DBX130.x				
	MDS												
				4	3	2	1	0	4	3	2	1	0
1	1	1							0	0	0	0	0
2	2	1							0	0	0	0	0
3	3	1							0	0	0	0	0
4	4	1							0	0	0	0	0
5	8	1							0	0	0	0	0
6	16	1							0	0	0	0	0
7	32	1							0	0	0	0	0
8	1	2							0	0	0	0	1
9	2	2							0	0	0	0	1
10	3	2							0	0	0	0	1
11	4	2							0	0	0	0	1
12	8	2							0	0	0	0	1
13	16	2							0	0	0	0	1
14	1	4							0	0	0	1	1
15	2	4							0	0	0	1	1
16	3	4							0	0	0	1	1
17	4	4							0	0	0	1	1
18	8	4							0	0	0	1	1
19	1	8							0	0	1	1	1
20	2	8							0	0	1	1	1
21	3	8							0	0	1	1	1
22	4	8							0	0	1	1	1
23	1	16							0	1	1	1	1
24	2	16							0	1	1	1	1
25	1	32							1	1	1	1	1

← Nicht unterstützte Kombination

Relevante Bit-Stelle bezüglich:

- Motor Datensatz (MDS)
- Antriebs Datensatz (DDS)
- Ungültige Bit-Stelle

MDS	Anzahl Motordatensätze
DDS pro MDS	Anzahl Antriebsdatensätze pro Motordatensatz
DB31, ... DBX21.x	Anforderungsschnittstelle
DB31, ... DBX93.x	Anzeigeschnittstelle
DB31, ... DBX130.x	Formatierungsschnittstelle

Bild 2-11 Prinzip der Motor- / Antriebsdatensatz-Umschaltung

2.3.6.7 Randbedingungen

Variable Anzahl Antriebsdatensätze für den "letzten" Motordatensatz

Der "letzte" Motordatensatz ist der Motordatensatz mit der höchsten Nummer bzw. Index.

Im Allgemeinen gilt, dass im Antrieb für jeden Motordatensatz die gleiche Anzahl von Antriebsdatensätzen (Anzahl "DDS pro MDS") angelegt wird. Nur für den "letzten" Motordatensatz kann davon abweichend, eine beliebige Anzahl von Antriebsdatensätzen a parametrisiert werden kann:

$$1 \leq a \leq (\text{Anzahl "DDS pro MDS"})$$

Beispiel

Es sollen 4 Motordatensätze (MDS) und 8 Antriebsdatensätze (DDS) pro Motordatensatz (DDS pro MDS) parametrisiert werden. Dies entspricht "Nr.": 22 der im Bild 2-11 Prinzip der

Motor- / Antriebsdatensatz-Umschaltung (Seite 74) dargestellten möglichen Datensatzkombinationen:

- Motordatensätze: MDS[0], MDS[1], ... MDS[3] ("letzter" Motordatensatz)
- Antriebsdatensätzen pro Motordatensatz: DDS[0] ... DDS[7]

Die Anzahl Antriebsdatensätze für die einzelnen Motordatensätze ist somit:

Motordatensatz (MDS)	Anzahl Antriebsdatensätze (DDS) pro Motordatensatz (MDS)
MDS[0] ... MDS[2]	8
MDS[3]	1 - 8

Umschaltpunkt: Antriebsparametersatz

Prinzipiell ist das Umschalten der Antriebsparametersätze zu jedem beliebigen Zeitpunkt möglich. Während eine Achse verfährt, können insbesondere bei der Umschaltung von Drehzahlreglerparameter und Motordrehzahlnormierung Momentensprünge auftreten. Es wird daher empfohlen, eine Umschaltung des Antriebsparametersatzes nur bei stationären Zuständen, insbesondere bei Achsstillstand, vorzunehmen.

Siehe auch

Übersicht der Schnittstellen (Seite 73)

2.4 Beispiele

2.4.1 Parametersatzumschaltung

Parametersatzumschaltung

Über eine Parametersatzumschaltung wird für die Maschinenachse X1 der Verstärkungsfaktor der Lageregelung (K_V -Faktor) von $K_V = 4.0$ auf $K_V = 0.5$ umgeschaltet.

Voraussetzungen

Die Parametersatzumschaltung muss freigegeben sein durch das Maschinendatum:

MD35590 \$MA_PARAMSET_CHANGE_ENABLE [AX1] = 1 oder 2 (Parametersatzwechsel möglich)

Angewählt ist der 1. Parametersatz der Maschinenachse X1, entsprechend den Maschinendaten mit dem Index "0". NC/PLC-Nahtstelle:

DB31, ... DBX9.0 - DBX9.2 = 0 (Regler Parametersatz)

Parametersatz-abhängige Maschinendaten

Die Parametersatz-abhängigen Maschinendaten sind wie folgt eingestellt:

Maschinendatum	Bemerkung
MD32200 \$MA_POSCTRL_GAIN [0, AX1] = 4.0	K _V -Einstellung für Parametersatz 1
MD32200 \$MA_POSCTRL_GAIN [1, AX1] = 2.0	K _V -Einstellung für Parametersatz 2
MD32200 \$MA_POSCTRL_GAIN [2, AX1] = 1.0	K _V -Einstellung für Parametersatz 3
MD32200 \$MA_POSCTRL_GAIN [3, AX1] = 0.5	K _V -Einstellung für Parametersatz 4
MD32200 \$MA_POSCTRL_GAIN [4, AX1] = 0.25	K _V -Einstellung für Parametersatz 5
MD32200 \$MA_POSCTRL_GAIN [5, AX1] = 0.125	K _V -Einstellung für Parametersatz 6
MD31050 \$MA_DRIVE_AX_RATIO_DENOM [0, AX1] = 3	Nenner Lastgetriebe für Parametersatz 1
MD31050 \$MA_DRIVE_AX_RATIO_DENOM [1, AX1] = 3	Nenner Lastgetriebe für Parametersatz 2
MD31050 \$MA_DRIVE_AX_RATIO_DENOM [2, AX1] = 3	Nenner Lastgetriebe für Parametersatz 3
MD31050 \$MA_DRIVE_AX_RATIO_DENOM [3, AX1] = 3	Nenner Lastgetriebe für Parametersatz 4
MD31050 \$MA_DRIVE_AX_RATIO_DENOM [4, AX1] = 3	Nenner Lastgetriebe für Parametersatz 5
MD31050 \$MA_DRIVE_AX_RATIO_DENOM [5, AX1] = 3	Nenner Lastgetriebe für Parametersatz 6
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA [0, AX1] = 5	Zähler Lastgetriebe für Parametersatz 1
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA [1, AX1] = 5	Zähler Lastgetriebe für Parametersatz 2
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA [2, AX1] = 5	Zähler Lastgetriebe für Parametersatz 3
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA [3, AX1] = 5	Zähler Lastgetriebe für Parametersatz 4
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA [4, AX1] = 5	Zähler Lastgetriebe für Parametersatz 5
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA [5, AX1] = 5	Zähler Lastgetriebe für Parametersatz 6
MD35130 \$MA_AX_VELO_LIMIT [0...5, AX1]	Einstellung für jeden Parametersatz*)
MD32800 \$MA_EQUIV_CURRCTRL_TIME [0..5, AX1]	Einstellung für jeden Parametersatz*)
MD32810 \$MA_EQUIV_SPEEDCTRL_TIME [0..5, AX1]	Einstellung für jeden Parametersatz*)
MD32910 \$MA_DYN_MATCH_TIME [0...5, AX1]	Einstellung für jeden Parametersatz*)
*) Die entsprechende Zeile ist für jeden Parametersatz nach den Syntaxregeln gesondert anzugeben.	

Umschaltung

Zur Umschaltung des Verstärkungsfaktor der Lageregelung wird vom PLC-Anwenderprogramm für die Maschinenachse X1 der 4. Parametersatz angewählt.

- Anforderung durch PLC-Anwenderprogramm:
DB31, ... DBX9.0 – DBX9.2 = 3 (Parametersatz Servo)
 - Für die Maschinenachse AX1 wird die Umschaltung auf den 4. Parametersatz angefordert.
 - Nach einer Verzögerungszeit wird der Parametersatz umgeschaltet.
 - Aktiv ist jetzt der Parametersatz 4, entsprechend den Maschinendaten mit dem Index "3"
- Rückmeldung durch NC:
DB31, ... DBX69.0 – DBX69.2 = 3 (Parametersatz Servo)
 - Die Umschaltung des Parametersatzes wird von der NC bestätigt / quittiert.

2.5 Datenlisten

2.5.1 Maschinendaten

2.5.1.1 Anzeige-Maschinendaten

Nummer	Bezeichner: \$MM_	Beschreibung
SINUMERIK Operate		
9000	LCD_CONTRAST	Kontrast
9001	DISPLAY_TYPE	Monitortyp
9004	DISPLAY_RESOLUTION	Anzeigefeinheit
9006	DISPLAY_SWITCH_OFF_INTERVAL	Zeit für Bildschirmdunkelschaltung

2.5.1.2 NC-spezifischen Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10350	FASTIO_DIG_NUM_INPUTS	Anzahl der aktiven digitalen NC-Eingangsbytes
10360	FASTIO_DIG_NUM_OUTPUTS	Anzahl der aktiven digitalen NC-Ausgangsbytes
10361	FASTIO_DIG_SHORT_CIRCUIT	Kurzschluss digitaler ein- und Ausgänge
11120	LUD_EXTENDED_SCOPE	Programmglobale Variablen aktivieren (PUD)
11270	DEFAULT_VALUES_MEM_MSK	Aktiv. Funktion: DEFAULT-Werte von GUD speichern.
18150	MM_GUD_VALUES_MEM	Speicherplatz für GUD reservieren

2.5.1.3 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
21015	INVOLUTE_RADIUS_DELTA	NC-Startsperre ohne Referenzpunkt
21016	INVOLUTE_AUTO_ANGLE_LIMIT	Automatische Winkelbegrenzung bei Evolventen-Interpolation
27800	TECHNOLOGY_MODE	Technologie im Kanal
28150	MM_NUM_VDIVAR_ELEMENTS	Anzahl Schreibelemente für PLC-Variable
28530	MM_PATH_VELO_SEGMENTS	Anzahl von Speicherelementen zur Begrenzung der Bahngeschwindigkeit im Satz

2.5.1.4 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30350	SIMU_AX_VDI_OUTPUT	Ausgabe der Achssignale bei Simulationsachsen
33050	LUBRICATION_DIST	Schmierimpulsdistanz
35590	PARAMSET_CHANGE_ENABLE	Parametersatzvorgabe durch PLC möglich
36060	STANDSTILL_VELO_TOL	Maximale Geschwindigkeit/Drehzahl bei Achse/Spindel steht
36610	AX_EMERGENCY_STOP_TIME	Zeitdauer der Bremsrampe bei Fehlerzuständen
36620	SERVO_DISABLE_DELAY_TIME	Abschaltverzögerung Reglerfreigabe

2.5.2 Systemvariablen

Bezeichner	Beschreibung
\$P_FUMB	freier Teileprogrammspeicher (F ree U ser M emory B uffer)
\$A_DBB[n]	Datum auf der PLC (Daten vom Type BYTE)
\$A_DBW[n]	Datum auf der PLC (Daten vom Type WORD)
\$A_DBD[n]	Datum auf der PLC (Daten vom Type DWORD)
\$A_DBR[n]	Datum auf der PLC (Daten vom Type REAL)

2.5.3 Signale

2.5.3.1 Signale an NC

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Schlüsselschalter-Stellung 0 bis 3	DB10.DBX56.4 - 7	DB2600.DBX0.4 - 7

2.5.3.2 Signale von NC

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Ferndiagnose aktiv (HMI-Alarm steht an)	DB10.DBX103.0	-
AT-Box ready	DB10.DBX103.5	-
HMI Temperaturgrenze	DB10.DBX103.6	-
HMI Batteriealarm	DB10.DBX103.7	-
NC--Ready	DB10.DBX104.7	-
Bedientafel 2: "ready"	DB10.DBX108.1	-
Bedientafel an MPI: "ready"	DB10.DBX108.2	-
Bedientafel an BTSS: "ready"	DB10.DBX108.3	DB2700.DBX2.3
Antriebe im zyklischen Betrieb	DB10.DBX108.5	DB2700.DBX2.5
Antriebe-Ready	DB10.DBX108.6	DB2700.DBX2.6
NC-Ready	DB10.DBX108.7	DB2700.DBX2.7
NC-Alarm steht an	DB10.DBX109.0	DB2700.DBX3.0
Kühlk.-Temperatur-Alarm NCU	DB10.DBX109.5	-
Luft-Temperatur-Alarm	DB10.DBX109.6	DB2700.DBX3.6
NC-Batterie-Alarm	DB10.DBX109.7	-

2.5.3.3 Signale an Bedientafelfront

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Bildschirm hell	DB19.DBX0.0	-
Bildschirm dunkel	DB19.DBX0.1	DB1900.DBX5000.1 ¹⁾
Tastensperre	DB19.DBX0.2	DB1900.DBX5000.2
Cancel-Alarme löschen (nur HMI Advanced)	DB19.DBX0.3	-
Recall-Alarme löschen (nur HMI Advanced)	DB19.DBX0.4	-
Istwert im WKS (1) / MKS (0)	DB19.DBX0.7	DB1900.DBX5000.7
Teileprogramm entladen	DB19.DBX13.5	-
Teileprogramm laden	DB19.DBX13.6	-
Teileprogramm Anwahl	DB19.DBX13.7	DB1700.DBX1000.7
Filesystem aktiv (0) / passiv (1)	DB19.DBX14.7	-
Teileprogrammhandling: Nummer der Steuerdaten für Anwenderdateinamen	DB19.DBB16 ²⁾	DB1700.DBB1001 ²⁾
Teileprogrammhandling: Index des zu übertragenden Files aus der Anwenderliste	DB19.DBB17	DB1700.DBB1002
BA-Wechselsperre	DB19.DBX44.0	-

¹⁾ Bei SINUMERIK 828D erfolgt die Hell-/Dunkelsteuerung des Bildschirms nur über DB1900.DBX5000.1:

DB1900.DBX5000.1=0: Bildschirm hell

DB1900.DBX5000.1=1: Bildschirm dunkel

DB1900.DBX5000.0 hat für die Hell-/Dunkelsteuerung keine Bedeutung.

²⁾ Bit 7: immer 1

2.5.3.4 Signale von Bedientafelfront

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Bildschirm ist dunkel	DB19.DBX20.1	-
Bedienoberfläche: Die Funktion "Simulation" ist aktiv	DB19.DBX20.6	DB1900.DBX0.6
MKS/WKS umschalten	DB19.DBX20.7	DB1900.DBX0.7
Teileprogramm-Handling-Status: Error	DB19.DBX26.0	DB1700.DBX2000.2
Teileprogramm-Handling-Status: O.K.	DB19.DBX26.1	DB1700.DBX2000.1
Teileprogramm-Handling-Status: Aktiv	DB19.DBX26.3	DB1700.DBX2000.3
Teileprogramm-Handling-Status: Entladen	DB19.DBX26.5	DB1700.DBX2000.5
Teileprogramm-Handling-Status: Laden	DB19.DBX26.6	DB1700.DBX2000.6
Teileprogramm-Handling-Status: Anwahl	DB19.DBX26.7	DB1700.DBX2000.7
FC9: Start "Messen in JOG"	DB19.DBX42.0	-
FC9 Out: Active	DB19.DBX45.0	-
FC9 Out: Done	DB19.DBX45.1	-
FC9 Out: Error	DB19.DBX45.2	-
FC9 Out: StartErr	DB19.DBX45.3	-

2.5.3.5 Signale an Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Restweg löschen (kanalspezifisch)	DB21,DBX6.2	DB320x.DBX6.2

2.5.3.6 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Kanalspezifischer NC-Alarm steht an	DB21,DBX36.6	DB330x.DBX4.6
NC-Alarm mit Bearbeitungsstillstand steht an	DB21,DBX36.7	DB330x.DBX4.7
Überspeichern aktiv	DB21,DBX318.7	-

2.5.3.7 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Achsen-/Spindelsperre	DB31,DBX1.3	DB380x.DBX1.3
Nachführbetrieb	DB31,DBX1.4	DB380x.DBX1.4
Lagemesssystem 1	DB31,DBX1.5	DB380x.DBX1.5
Lagemesssystem 2	DB31,DBX1.6	DB380x.DBX1.6
Reglerfreigabe	DB31,DBX2.1	DB380x.DBX2.1
Restweg löschen (achsspezifisch)/Spindel-Reset	DB31,DBX2.2	DB380x.DBX2.2
Motor- / Antriebsdatensatz: Anforderungsschnittstelle	DB31,DBX21.0 - 4	DB380x.DBX4001.0 - 4
Motoranwahl erfolgt	DB31,DBX21.5	DB380x.DBX4001.5
Integratorsperre n-Regler	DB31,DBX21.6	DB380x.DBX4001.6
Impulsfreigabe	DB31,DBX21.7	DB380x.DBX4001.7

2.5.3.8 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Referenziert / synchronisiert, Geber 1 / 2	DB31,DBX60.4 / 5	DB390x.DBX0.4 / 5
Fahrbefehl minus / plus	DB31,DBX64.6 / 7	DB390x.DBX4.6 / 7
Nachführen aktiv	DB31,DBX61.3	DB390x.DBX1.3
Achse/Spindel steht ($n < n_{\min}$)	DB31,DBX61.4	DB390x.DBX1.4
Lageregler aktiv	DB31,DBX61.5	DB390x.DBX1.5
Drehzahlregler aktiv	DB31,DBX61.6	DB390x.DBX1.6
Stromregler aktiv	DB31,DBX61.7	DB390x.DBX1.7
Schmierimpuls	DB31,DBX76.0	DB390x.DBX1002.0
Motor- / Antriebsdatensatz: Anzeigeschnittstelle	DB31,DBX93.0 - 4	DB390x.DBX4001.0 - 4
Drive Ready	DB31,DBX93.5	DB390x.DBX4001.5
Integrator n-Regler gesperrt	DB31,DBX93.6	DB390x.DBX4001.6

2.5 Datenlisten

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Impulse freigegeben	DB31,DBX93.7	DB390x.DBX4001.7
Motor-Temperatur-Vorwarnung	DB31,DBX94.0	DB390x.DBX4002.0
Kühlkörper-Temperatur-Vorwarnung	DB31,DBX94.1	DB390x.DBX4002.1
Hochlaufvorgang beendet	DB31,DBX94.2	DB390x.DBX4002.2
$ M_d < M_{dx}$	DB31,DBX94.3	DB390x.DBX4002.3
$ n_{ist} < n_{min}$	DB31,DBX94.4	DB390x.DBX4002.4
$ n_{ist} < n_x$	DB31,DBX94.5	DB390x.DBX4002.5
$n_{ist} = n_{soll}$	DB31,DBX94.6	DB390x.DBX4002.6
Motor- / Antriebsdatensatz: Formatierungsschnittstelle	DB31,DBX130.0 - 4	DB390x.DBX4008.0 - 4
Motor- / Antriebsdatensatz: Formatierung ist gültig	DB31,DBX130.7	DB390x.DBX4008.7

A3: Achsüberwachungen

3.1 Konturüberwachung

3.1.1 Konturfehler

Konturfehler entstehen durch Signalverzerrungen im Lageregelkreis.

Man unterscheidet lineare und nichtlineare Signalverzerrungen.

Lineare Signalverzerrungen

Lineare Signalverzerrungen entstehen durch:

- Nicht optimal eingestellte Drehzahl- bzw. Lageregler
- Ungleiche K_V -Faktoren der an der Bahnerzeugung beteiligten Vorschubachsen
Bei gleichem K_V -Faktor zweier linear interpolierender Achsen folgt der Istpunkt dem Sollpunkt auf gleicher Bahn, jedoch zeitlich verzögert. Bei ungleichem K_V -Faktor entsteht ein Parallelversatz zwischen Soll- und Istbahn.
- Ungleiche Dynamik der Vorschubantriebe
Eine ungleiche Antriebsdynamik führt insbesondere an Konturänderungen zu Bahnabweichungen. Kreise werden durch eine ungleiche Dynamik der beiden Vorschubantriebe zu Ellipsen verzerrt.

Nichtlineare Signalverzerrungen

Nichtlineare Signalverzerrungen entstehen durch:

- Wirksamwerden der Strombegrenzung innerhalb des Bearbeitungsbereichs
- Wirksamwerden der Begrenzung des Drehzahlsollwerts
- Umkehrspanne innerhalb und/oder außerhalb des Lageregelkreises
Beim Durchfahren einer Kreisbahn treten vor allem Konturfehler durch die Umkehrspanne und durch Reibung auf.
Beim Fahren von Geraden entsteht ein Konturfehler durch eine Umkehrspanne außerhalb des Lageregelkreises z. B. durch eine kippende Frässpindel. Sie führt zu einem Parallelversatz zwischen Ist- und Sollbahn. Dieser ist umso größer, je flacher die Steigung der Geraden verläuft.
- Nichtlineares Reibungsverhalten der Schlittenführungen

3.1.2 Schleppabstandsüberwachung

Funktion

Regelungstechnisch entsteht beim Verfahren einer Maschinenachse immer ein gewisser Schleppabstand, d. h. eine Differenz zwischen Soll- und Istposition.

Der sich einstellende Schleppabstand ist abhängig von:

- Lageregelkreisverstärkung
MD32200 \$MA_POSCTRL_GAIN (KV-Faktor)
- Maximale Beschleunigung
MD32300 \$MA_MAX_AX_ACCEL (Maximale Achsbeschleunigung)
- Maximale Geschwindigkeit
MD32000 \$MA_MAX_AX_VELO (Maximale Achsgeschwindigkeit)
- Bei aktiver Vorsteuerung:
Genauigkeit des Streckenmodells und den Parametern:
MD32610 \$MA_VELO_FFW_WEIGHT (Faktor für Geschwindigkeits-Vorsteuerung)
MD32800 \$MA_EQUIV_CURRCTRL_TIME (Ersatzzeitkonstante Stromregelkreis für Vorsteuerung)
MD32810 \$MA_EQUIV_SPEEDCTRL_TIME (Ersatzzeitkonstante Drehzahlregelkreis für Vorsteuerung)

In der Beschleunigungsphase wird der Schleppabstand beim Verfahren einer Maschinenachse zunächst immer größer. Nach einer von der Parametrierung des Lageregelkreises abhängigen Zeit bleibt er dann im Idealfall konstant. Aufgrund äußerer Einflüsse kommt es während eines Bearbeitungsvorgangs immer zu mehr oder weniger großen Schwankungen des Schleppabstandes. Damit diese Schwankungen des Schleppabstandes keinen Alarm auslösen, ist für die Schleppabstandsüberwachung ein Toleranzband vorzugeben, innerhalb dessen sich der Schleppabstand bewegen darf:

MD36400 \$MA_CONTOUR_TOL (Toleranzband Konturüberwachung)

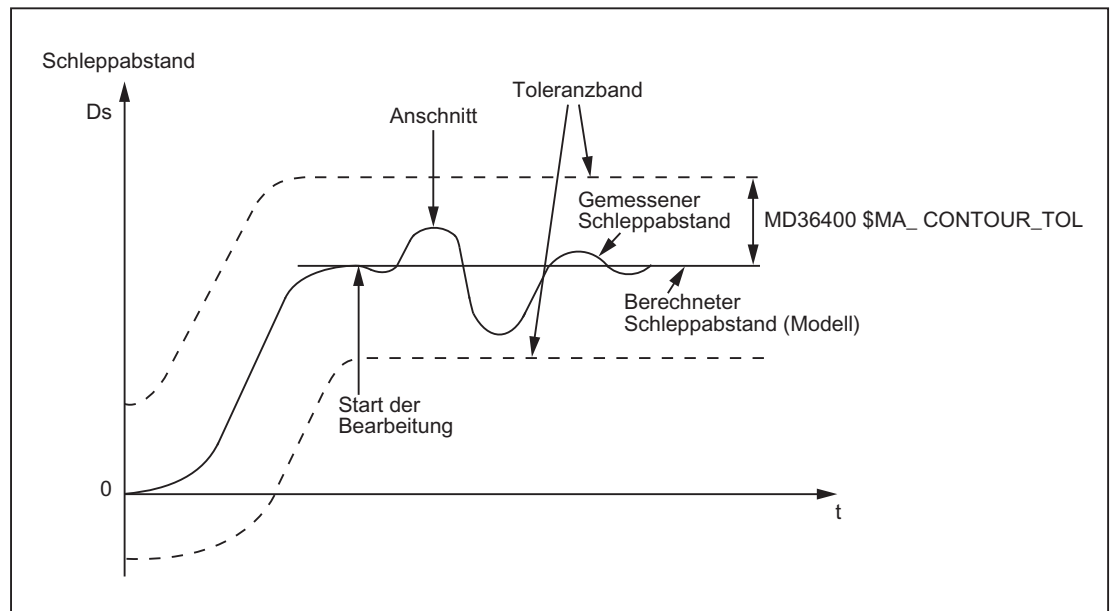


Bild 3-1 Schleppabstandsüberwachung

Wirksamkeit

Die Schleppabstandsüberwachung ist nur wirksam bei aktiver Lageregelung und folgenden Achstypen:

- Linearachsen mit und ohne Vorsteuerung
- Rundachsen mit und ohne Vorsteuerung
- Lagegeregelten Spindeln

Fehlerfall

Bei Überschreiten der parametrisierten Toleranzgrenze wird folgender Alarm angezeigt:

25050 "Achse <Achsname> Konturüberwachung"

Die betroffene Achse / Spindel wird im Nachführbetrieb über die parametrisierte Bremsrampe stillgesetzt:

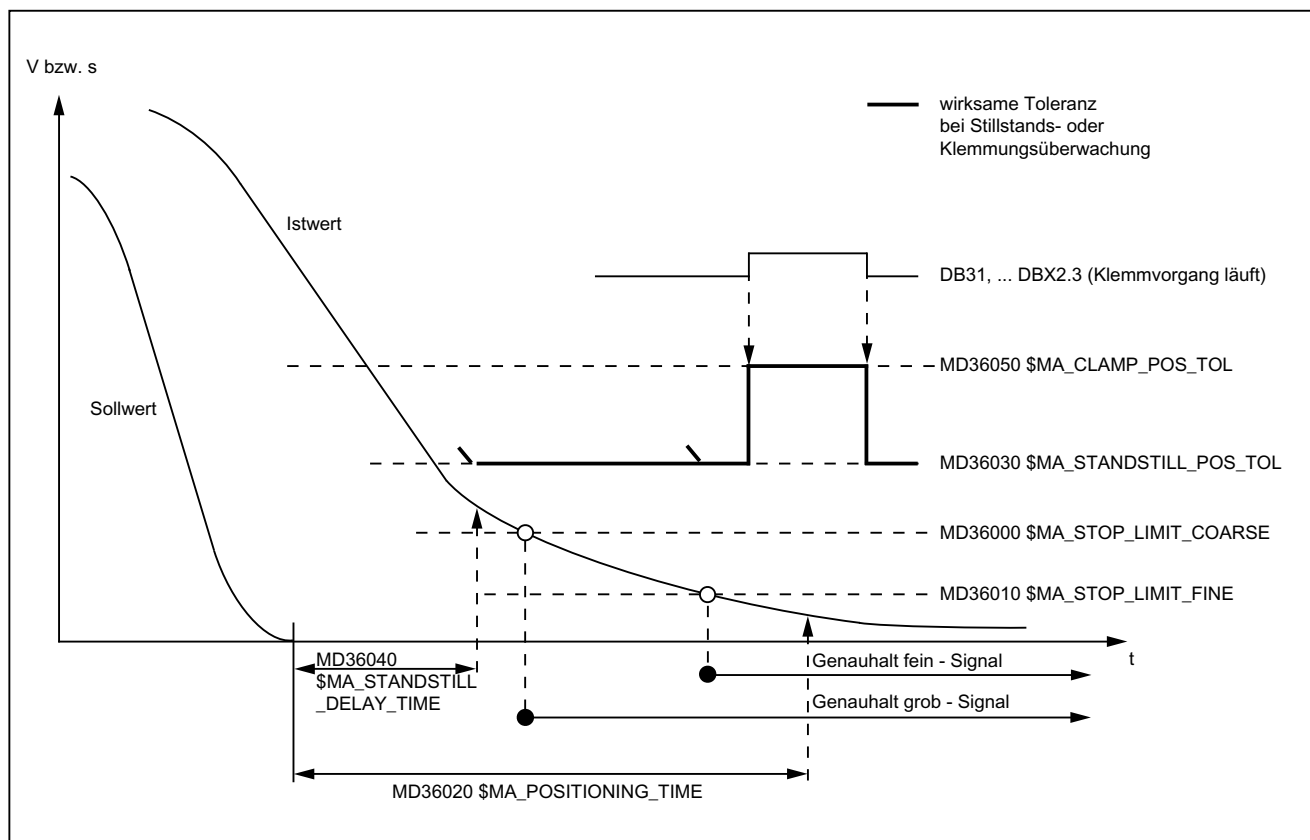
MD36610 \$MA_AX_EMERGENCY_STOP_TIME
(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

3.2 Positionier-, Stillstands- und Klemmungsüberwachung

3.2.1 Zusammenhang zwischen Positionier-, Stillstands- und Klemmungsüberwachung

Übersicht

Die folgende Übersicht zeigt den Zusammenhang zwischen Positionier-, Stillstands- und Klemmungsüberwachung:



3.2.2 Positionierüberwachung

Funktion

Zum Abschluss eines Positioniervorgangs:

- Sollgeschwindigkeit = 0 **UND**
- DB31, ... DBX64.6/7 (Fahrbefehl minus/plus) = 0

überwacht die Positionierüberwachung, dass der Schleppabstand jeder beteiligten Maschinenachse innerhalb der Verzögerungszeit kleiner der Genauhalttoleranz fein wird:

MD36010 \$MA_STOP_LIMIT_FINE (Genauhalt fein)

MD36020 \$MA_POSITIONING_TIME (Verzögerungszeit Genauhalt fein)

Nach dem Erreichen von "Genauhalt fein" wird die Positionierüberwachung abgeschaltet.

Hinweis

Je kleiner die Genauhalttoleranz fein gewählt wird, desto länger dauert der Positioniervorgang und damit die Zeit bis zum Satzwechsel.

Regeln zur MD-Einstellung

MD36010 \$MA_STOP_LIMIT_FINE	MD36020 \$MA_POSITIONING_TIME
groß	kann relativ kurz gewählt werden
klein	muss relativ lang gewählt werden

MD32200 \$MA_POSCTRL_GAIN (KV-Faktor)	MD36020 \$MA_POSITIONING_TIME
klein	muss relativ lang gewählt werden
groß	kann relativ kurz gewählt werden

Wirksamkeit

Die Positionierüberwachung ist nur wirksam bei aktiver Lageregelung und folgenden Achstypen:

- Linearachsen
- Rundachsen
- Lagegeregelten Spindeln

Fehlerfall

Bei Überschreiten der parametrisierten Positionierüberwachungszeit wird folgender Alarm angezeigt:

25080 "Achse <Achsname> Positionierüberwachung"

Die betroffene Achse wird im Nachführbetrieb über die parametrisierte Bremsrampe stillgesetzt:

MD36610 \$MA_AX_EMERGENCY_STOP_TIME
(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

3.2.3 Stillstandsüberwachung

Funktion

Zum Abschluss eines Positioniervorgangs:

- Sollgeschwindigkeit = 0 **UND**
- DB31, ... DBX64.6/7 (Fahrbefehl minus/plus) = 0

überwacht die Stillstandsüberwachung, dass der Schleppabstand jeder beteiligten Maschinenachse innerhalb der Verzögerungszeit kleiner der Stillstandstoleranz wird:

MD36040 \$MA_STANDSTILL_DELAY_TIME (Verzögerungszeit Stillstandsüberwachung)

MD36030 \$MA_STANDSTILL_POS_TOL (Stillstandstoleranz)

Nach Erreichen des erforderlichen Genauhalt-Zustandes ist der Positioniervorgang abgeschlossen:

DB31, ... DBX60.6/7 (Position erreicht mit Genauhalt grob/fein) = 1

Die Positionierüberwachung wird abgeschaltet und von der Stillstandsüberwachung abgelöst.

Die Stillstandsüberwachung überwacht die Einhaltung der Stillstandstoleranz. Solange keine neue Fohranforderung ansteht, darf die Maschinenachse die Stillstandstoleranz nicht verlassen.

Wirksamkeit

Die Stillstandsüberwachung ist nur wirksam bei aktiver Lageregelung und folgenden Achstypen:

- Linearachsen
- Rundachsen
- Lagegeregelten Spindeln

Fehlerfall

Bei Überschreiten der Verzögerungszeit und / oder der Stillstandstoleranz wird folgender Alarm angezeigt:

25040 "Achse <Achsnamc> Stillstandsüberwachung"

Die betroffene Achse wird im Nachföhrbetrieb über die parametrisierte Bremsrampe stillgesetzt:

MD36610 \$MA_AX_EMERGENCY_STOP_TIME

(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

3.2.4 Parametersatzabhängige Genauhalt- und Stillstandstoleranz

Zur Anpassung an unterschiedliche Bearbeitungssituationen und / oder Achsdynamiken, z. B.:

- Betriebszustand A: Hohe Genauigkeit, lange Bearbeitungszeit
- Betriebszustand B: Geringere Genauigkeit, kürzere Bearbeitungszeit
- Änderung der Massenverhältnisse nach Getriebeumschaltung

können die Positionstoleranzen:

- MD36000 \$MA_STOP_LIMIT_COARSE (Genauhalt grob)
- MD36010 \$MA_STOP_LIMIT_FINE (Genauhalt fein)
- MD36030 \$MA_STANDSTILL_POS_TOL (Stillstandstoleranz)

parametersatzabhängig mit einem gemeinsamen Faktor bewertet werden:

MD36012 \$MA_STOP_LIMIT_FACTOR (Faktor Genauhalt grob / fein und Stillstand)

Da der Faktor für alle drei Positionstoleranzen gemeinsam gilt, bleibt das Verhältnis der Werte untereinander konstant.

3.2.5 Klemmungsüberwachung

3.2.5.1 Funktion

Bei Maschinenachsen, die nach Abschluss eines Positioniervorgangs mechanisch geklemmt werden, kann es aufgrund des Klemmvorgangs zu einer Verschiebung der Achse aus der Sollposition kommen. Durch das Setzen des NC/PLC-Nahtstellensignals DB31, ... DBX2.3 (Klemmvorgang läuft) wird für die Dauer des Klemmvorgangs statt der Stillstandstoleranz (MD36030 \$MA_STANDSTILL_POS_TOL) die Klemmungstoleranz (MD36050 \$MA_CLAMP_POS_TOL) überwacht. Wird die Klemmungstoleranz überschritten, wird der Alarm 26000 "Klemmungsüberwachung" angezeigt.

Alarmverzögerungszeit

Soll eine zeitlich begrenzte Überschreitung der Klemmungstoleranz zugelassen werden, muss über das Maschinendatum MD36051 \$MA_CLAMP_POS_TOL_TIME eine Alarmverzögerungszeit vorgegeben werden. Die Alarmausgabe erfolgt dann bei einer Überschreitung der Klemmungstoleranz erst nach Ablauf der parametrisierten Zeit. Wird vor Ablauf der Zeit die Klemmungstoleranz wieder unterschritten, erfolgt keine Alarmausgabe. Mit der nächsten Überschreitung der Klemmungstoleranz wird die Zeit neu gestartet.

Damit vor Ablauf der Alarmverzögerungszeit auf die Überschreitung der Klemmungstoleranz reagiert werden kann, wird das achsspezifische NC/PLC-Nahtstellensignal DB31, ... DBX102.3 (Klemmungstoleranz überschritten) gesetzt. Bei Unterschreitung der Klemmungstoleranz wird das Signal wieder zurückgesetzt.

3.2.5.2 Maschinendaten

Klemmungstoleranz

In das Maschinendatum wird die gegenüber der Stillstandstoleranz größere Klemmungstoleranz eingetragen:

MD36050 \$MA_CLAMP_POS_TOL[<Achse>] = <Klemmungstoleranz>

Alarmverzögerungszeit

Soll eine zeitlich begrenzte Überschreitung der Klemmungstoleranz toleriert werden, muss in das Maschinendatum die maximal zulässige Alarmverzögerungszeit eingetragen werden.

MD36051 \$MA_CLAMP_POS_TOL_TIME[<Achse>] = <Alarmverzögerungszeit>

Bei einer Überschreitung der Klemmungstoleranz wird der Alarm 26000 "Klemmungsüberwachung" erst nach Ablauf der Alarmverzögerungszeit angezeigt.

Wird die Klemmungstoleranz vor Ablauf der Alarmverzögerungszeit wieder unterschritten, erfolgt keine Alarmausgabe.

Mit der nächsten Überschreitung der Klemmungstoleranz wird die Alarmverzögerungszeit neu gestartet.

Spezielle Klemmungsfunktionen

Über das Maschinendatum werden die speziellen Klemmungsfunktionen, die das Lösen und Setzen der Klemmung im Teileprogrammablauf automatisieren, bitweise aktiviert:

MD36052 \$MA_STOP_ON_CLAMPING[<Achse>], <Bit> = <Wert>

<Bit>	<Wert>	Bedeutung
0		Automatisches Anhalten zum Lösen der Klemmung
	0	Nicht aktiv
	1	Aktiv
1		Optimiertes Lösen der Klemmung
	0	Nicht aktiv
	1	Aktiv, Voraussetzung: Bit 0 == 1
2		Automatisches Anhalten zum Setzen der Klemmung
	0	Nicht aktiv
	1	Aktiv

3.2.5.3 NC/PLC-Nahtstellensignale

Aktivierung der Klemmungsüberwachung

Die Aktivierung der Klemmungsüberwachung erfolgt durch Setzen des NC/PLC-Nahtstellensignals:

DB31, ... DBX2.3 = 1 (Klemmvorgang läuft)

Überschreitung der Klemmungstoleranz

Das Überschreiten der Klemmungstoleranz wird durch das NC/PLC-Nahtstellensignal angezeigt:

DB31, ... DBX102.3 == 1 (Klemmungstoleranz überschritten)

Das Signal wird **gesetzt**, wenn innerhalb der Alarmverzögerungszeit die Klemmungstoleranz überschritten wird.

Das Signal wird **zurückgesetzt**, wenn innerhalb der Alarmverzögerungszeit die Klemmungstoleranz unterschritten wird oder für die Achse Nachführbetrieb aktiviert wird.

3.2.5.4 Fehlerreaktionen

Fehlerreaktionen bei Überschreitung der Klemmungstoleranz:

- Alarm 26000 "Klemmungsüberwachung" wird angezeigt
- Die Achse wird mit der parametrierten Maximalbeschleunigung stillgesetzt:
MD32300 \$MA_MAX_AX_ACCEL
Dabei wird die maximale Zeitdauer der Bremsrampe bei Fehlerzuständen überwacht:
MD36610 \$MA_AX_EMERGENCY_STOP_TIME
- Für die Achse wird Nachführbetrieb aktiviert:
DB31, ... DBX61.3 == 1
- Das Signal "Klemmungstoleranz überschritten" wird zurückgesetzt:
DB31, ... DBX102.3 == 0

3.2.5.5 Klemmungsfunktion "Automatisches Anhalten zum Lösen der Klemmung"

Durch die Klemmungsfunktion "Automatisches Anhalten zum Lösen der Klemmung" wird bei Bahnsteuerbetrieb vor dem Verfahrssatz der Klemmungsachse ein NC-interner Stopp eingefügt.

Der Stopp wird nicht wirksam bzw. der Bahnsteuerbetrieb nicht unterbrochen, wenn vor dem Satzwechsel die Reglerfreigabe (DB31, ... DBX2.1) der Klemmungsachse gesetzt ist.

Ist vor dem Satzwechsel die Reglerfreigabe der Klemmungsachse **nicht** gesetzt, wird der Stopp wirksam.

Parametrierung

MD36052 \$MA_STOP_ON_CLAMPING[<Klemmungsachse>] = 'H01'

Voraussetzungen / Annahmen

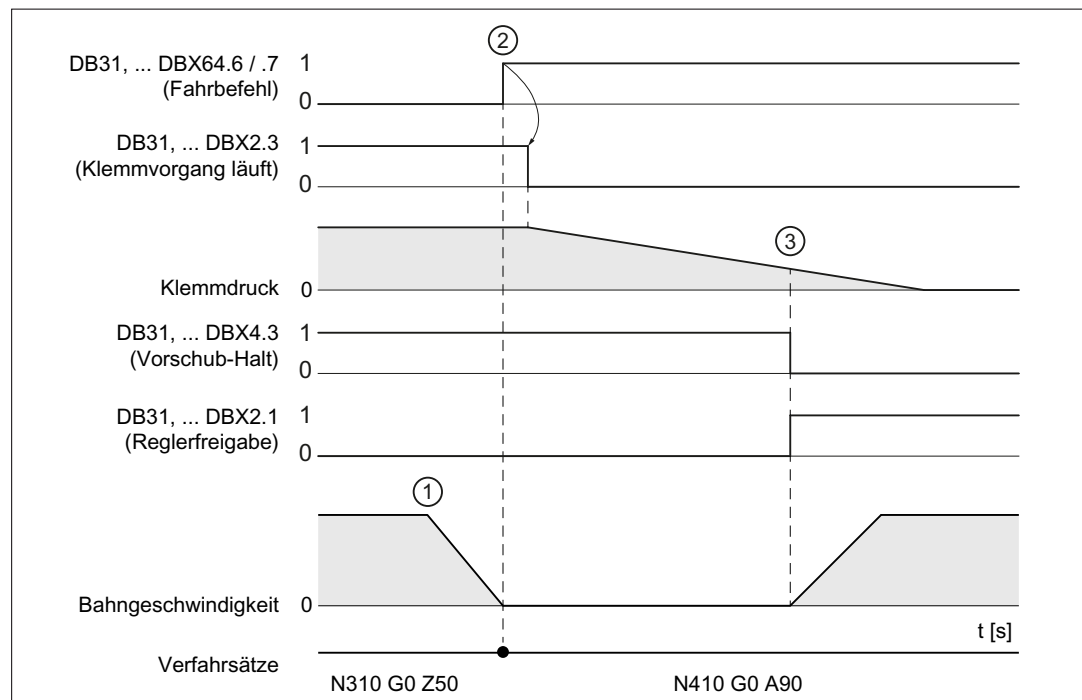
- Wenn für die Klemmungsachse ein **Fahrbefehl** (DB31, ... DBX64.6 / .7) ansteht, wird die Klemmung durch das PLC-Anwenderprogramm **gelöst**.
- Zwischen Reglerfreigabe (DB31, ... DBX2.1) und Klemmung der Klemmungsachse muss folgender Zusammenhang bestehen:

Reglerfreigabe	⇒	Klemmungsachse
nicht gesetzt	⇒	geklemmt
gesetzt	⇒	nicht geklemmt

Beispiel

Programmcode	Kommentar
N100 G0 X0 Y0 Z0 A0 G90 G54 F500	; Grundstellung anfahren
N101 G641 ADIS=.1 ADISPOS=5	; Bahnsteuerbetrieb einschalten
N210 G1 X10	; Verfahrsatz
N220 G1 X5 Y20	; "
N310 G0 Z50	; Positioniersatz
N410 G0 A90	; " (Klemmungsachse)
N510 G0 X100	; "
N520 G0 Z2	; "
N610 G1 Z-4	; Verfahrsatz
N620 G1 X0 Y-20	; "

Schematischer Verlauf der NC/PLC-Nahtstellensignale und Zustände für die Sätze N310 und N410:



- ① NC: Aufgrund des automatisch eingefügten Stopps wird zum Satzende von N310 angehalten.
- ② NC → PLC: Nach dem Satzwechsel wird der Fahrbehl für die Klemmungsachse gesetzt
PLC: Aufgrund des Fahrbefehls wird die Klemmung gelöst.
- ③ PLC → NC: Der Klemmdruck ist ausreichend abgebaut. Die Klemmungsachse wird zum Verfahrfahren freigegeben.

3.2.5.6 Klemmungsfunktion "Zeitoptimiertes Lösen der Klemmung"

Durch die Klemmungsfunktion "Zeitoptimiertes Lösen der Klemmung" wird im Zusammenhang mit der Klemmungsfunktion "Automatisches Anhalten zum Lösen der Klemmung" bei Bahnsteuerbetrieb das Lösen der Klemmung NC-intern durch vorausschauendes Setzen des Fahrbefehls für die Klemmungsachse angefordert. Der Fahrbefehl wird nur gesetzt, wenn bis zum Verfahren der Klemmungsachse ausschließlich positioniert (G0-Sätze) und nicht bearbeitet (G1-Sätze) wird.

Um den Bezug zum Verfahrssatz der Klemmungsachse zu erhalten, wird der Fahrbefehl maximal zwei Eilgangsätze (G0), einschließlich eventuell intern erzeugter Zwischensätze, vor dem Verfahrssatz gesetzt, .

Aktivierung

MD36052 \$MA_STOP_ON_CLAMPING[<Klemmungsachse>] = 'H03'

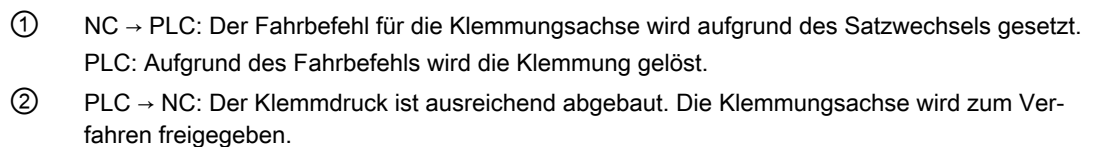
Voraussetzungen / Annahmen

- Wenn für die Klemmungsachse ein **Fahrbefehl** (DB31, ... DBX64.6 / .7) ansteht, wird die Klemmung durch das PLC-Anwenderprogramm **gelöst**.
- Während andere Achsen mit Eilgang (G0) verfahren, muss die Klemmungsachse **nicht** geklemmt sein.

Beispiel

Programmcode	Kommentar
N100 G0 X0 Y0 Z0 A0 G90 G54 F500	; Grundstellung anfahren
N101 G641 ADIS=.1 ADISPOS=5	; Bahnsteuerbetrieb einschalten
N210 G1 X10	; Bearbeitungssatz
N220 G1 X5 Y20	; "
N310 G0 Z50	; Positioniersatz
N410 G0 A90	; " (Klemmungsachse)
N510 G0 X100	; "
N520 G0 Z2	; "
N610 G1 Z-4	; Bearbeitungssatz
N620 G1 X0 Y-20	; "

Schematischer Verlauf der NC/PLC-Nahtstellensignale und Zustände für die Sätze N220 bis N410:



Der Klemmungsvorgang nimmt eine gewisse Zeit in Anspruch. Im Bahnsteuerbetrieb müsste dazu ein expliziter Stopp der Verfahrbewegung durch Programmierung von z.B. G09, G60 oder einer Hilfsfunktionsausgabe vorgesehen werden, damit die Klemmung sicher wirkt, bevor mit der Bearbeitung begonnen wird.

Aktivierung

Grundfunktionen
Funktionshandbuch, 05/2017, A5E40870716

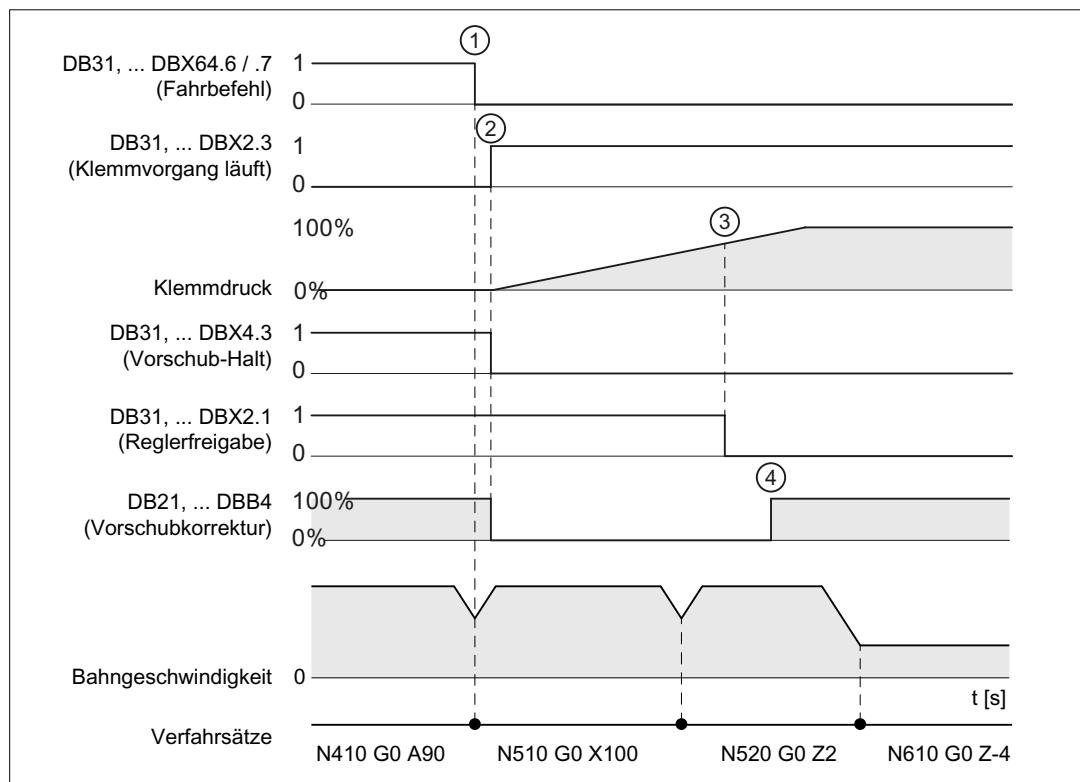
Voraussetzungen / Annahmen

- Wenn für die Klemmungsachse **kein** Fahrbefehl (DB31, ... DBX64.6 / .7) ansteht, wird die Klemmung durch das PLC-Anwenderprogramm **geschlossen**, .
- Während andere Achsen mit Eilgang (G0) verfahren, muss die Klemmungsachse **nicht** geklemmt sein.
- Wenn die kanalspez. Vorschub-Override **ungleich 0%** ist (DB21, ... DBB4 \neq 0%), ist die Klemmungsachse **geklemt**.

Beispiel

Programmcode	Kommentar
N100 G0 X0 Y0 Z0 A0 G90 G54 F500	; Grundstellung anfahren
N101 G641 ADIS=.1 ADISPOS=5	; Bahnsteuerbetrieb einschalten
N210 G1 X10	; Bearbeitungssatz
N220 G1 X5 Y20	; "
N310 G0 Z50	; Positioniersatz
N410 G0 A90	; " (Klemmungsachse)
N510 G0 X100	; "
N520 G0 Z2	; "
N610 G1 Z-4	; Bearbeitungssatz
N620 G1 X0 Y-20	; "

Schematischer Verlauf der NC/PLC-Nahtstellensignale und Zustände für die Sätze N410 bis N610:



3.2.5.8

Randbedingungen

Unterbrochener Bahnsteuerbetrieb

Wird während der oben beschriebenen Klemmungsfunktionen der Bahnsteuerbetrieb und damit auch die Funktion "LookAhead" durch Sätze ohne Verfahrbewegung (z. B. Ausgabe einer M-Funktion M82 / M83) unterbrochen, verhalten sich die Funktionen wie folgt:

- **Klemmungsfunktion: "Zeitoptimiertes Lösen der Achsklemmung"**
(MD36052 \$MA_STOP_ON_CLAMPING[<Achse>] = 'B011')
Die Funktion wirkt sich nicht mehr aus, da das vorausschauende Setzen des Fahrbefehls nur für Sätze mit aktivem Bahnsteuerbetrieb durchgeführt wird. Die Ausgabe der M-Funktion M82 im Satz N320 des unten stehenden Beispielprogramms bewirkt einen Stopp der Verfahrbewegung und unterbricht somit den Bahnsteuerbetrieb.
Das von der Funktion ansonsten bewirkte "vorausschauende" Anhalten auf N410 ist nicht erforderlich, da durch N320 sowieso angehalten wird.
- **Klemmungsfunktion: "Automatisches Anhalten zum Setzen der Klemmung"**
(MD36052 \$MA_STOP_ON_CLAMPING[<Achse>] = 'B100')
Die Funktion erzeugt unabhängig von M83 einen Stopp, der abhängig von "Vorschub-Override 0%" durchgeführt wird. Das Anhalten vor dem ersten Bearbeitungssatz ist somit gegeben.

Hinweis

Klemmungsfunktionen ohne Klemmung verwenden

Folgende Klemmungsfunktionen können auch unabhängig vom Klemmen der Achse verwendet werden:

- "Automatisches Anhalten zum Lösen der Klemmung":
MD36052 \$MA_STOP_ON_CLAMPING[<Achse>] = 'B001'
Verhalten: Es wird im aktuellen Satz auf der Bahn gestoppt, wenn die Reglerfreigabe (DB31, ... DBX2.1) für die parametrisierte <Achse> **nicht** gesetzt ist, sie aber in einem der nachfolgenden Sätze verfahren wird.
- "Automatisches Anhalten zum Setzen der Klemmung":
MD36052 \$MA_STOP_ON_CLAMPING[<Achse>] = 'B100'
Verhalten: Es wird im aktuellen Satz auf der Bahn gestoppt, wenn am Übergang von Eilgang- (G0) zu Verfahrätzen (G1) die kanalspezifische Vorschub-Override (DB21, ... DBB4) == 0% ist.

In beiden Fällen wird sichergestellt, dass die Bahnbewegung im Bahnsteuerbetrieb bereits vor dem Beginn des betreffenden Satzes gestoppt ist und nicht erst innerhalb des Satzes angehalten wird.

Tabelle 3-1 Beispielprogramm: Unterbrochener Bahnsteuerbetrieb

Programmcode	Kommentar
N100 G0 X0 Y0 Z0 A0 G90 G54 F500	; Grundstellung anfahren
N101 G641 ADIS=.1 ADISPOS=5	; Bahnsteuerbetrieb einschalten
N210 G1 X10	; Verfahrssatz
N220 G1 X5 Y20	; "
N310 G0 Z50	; Eilgangssatz
N320 M82	; Unterbrechung des Bahnsteuerbetriebs
N410 G0 A90	; Eilgangssatz

Programmcode	Kommentar
N420 M83	; Unterbrechung des Bahnsteuerbetriebs
N510 G0 X100	; Eilgangsatz
N520 G0 Z2	; "
N610 G1 Z-4	; Verfahrssatz
N620 G1 X0 Y-20	; "

Satzwechselkriterium: Klemmungstoleranz

Nach Aktivierung der Klemmungsüberwachung (DB31, ... DBX2.3) wirkt als Satzwechselkriterium bei Verfahrssätzen, bei denen am Satzende angehalten wird, für die Klemmungsachse nicht mehr die Genauhalt- sondern die Klemmungstoleranz:

MD36050 \$MA_CLAMP_POS_TOL (Klemmungstoleranz bei Nahtstellensignal "Klemmung aktiv")

Verhalten beim Lösen der Klemmung

Ist die Klemmungsachse durch den Klemmungsvorgang aus der Sollposition bewegt worden, wird sie nach dem Lösen der Klemmung und dem Setzen der Reglerfreigabe (DB31, ... DBX2.1) von der NC wieder auf die Sollposition verfahren. Die Rückpositionierung erfolgt abhängig davon, ob für die Achse während des Klemmungsvorgangs "Nachführbetrieb" aktiv war:

- DB31, ... DBX1.4 == 0 (Nachführbetrieb nicht aktiv) ⇒ Sprungförmig durch den Lageregler
- DB31, ... DBX1.4 == 1 (Nachführbetrieb aktiv) ⇒ Interpolatorisches Verfahren

Hinweis

Als Kriterium zur Aktivierung des Nachführbetriebs (DB31, ... DBX1.4) können vom PLC-Anwenderprogramm folgende Daten ausgewertet werden:

- DB31, ... DBX60.6/.7 (Position erreicht mit Genauhalt grob/fein)
- Istposition der Klemmungsachse

Nachführbetrieb

Die Klemmungsüberwachung ist im Nachführbetrieb nicht aktiv

DB31, ... DBX1.4 == 1 (Nachführbetrieb).

3.3 Drehzahlsollwertüberwachung

Funktion

Der Drehzahlsollwert setzt sich zusammen aus:

- Drehzahlsollwert des Lagereglers
- Drehzahlsollwertanteil der Vorsteuerung (nur bei aktiver Vorsteuerung)
- Driftkompensation (nur bei Antrieben mit analoger Sollwertschnittstelle)

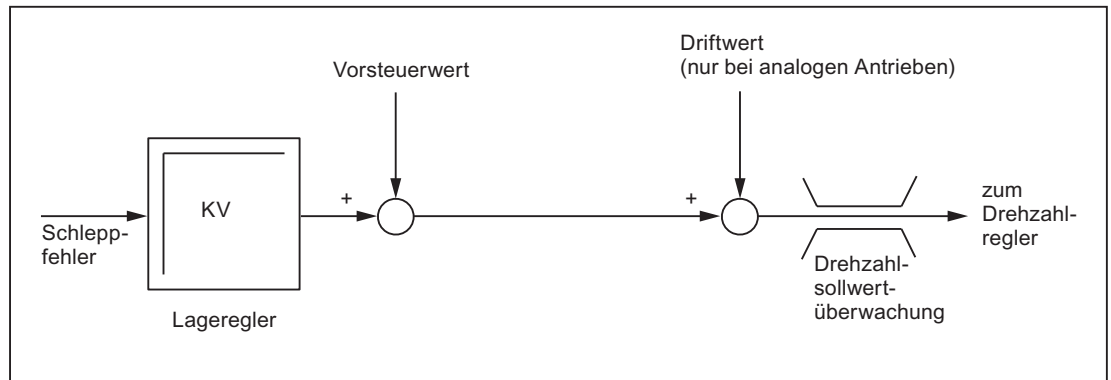


Bild 3-2 Drehzahlsollwertberechnung

Die Drehzahlsollwertüberwachung stellt durch Begrenzung der Stell- bzw. Ausgangsgröße (10 V bei analoger Sollwertschnittstelle oder Nenndrehzahl bei digitalen Antrieben) sicher, dass die physikalischen Begrenzungen der Antriebe nicht überschritten werden:

MD36210 \$MA_CTRLOUT_LIMIT (Maximaler Drehzahlsollwert)

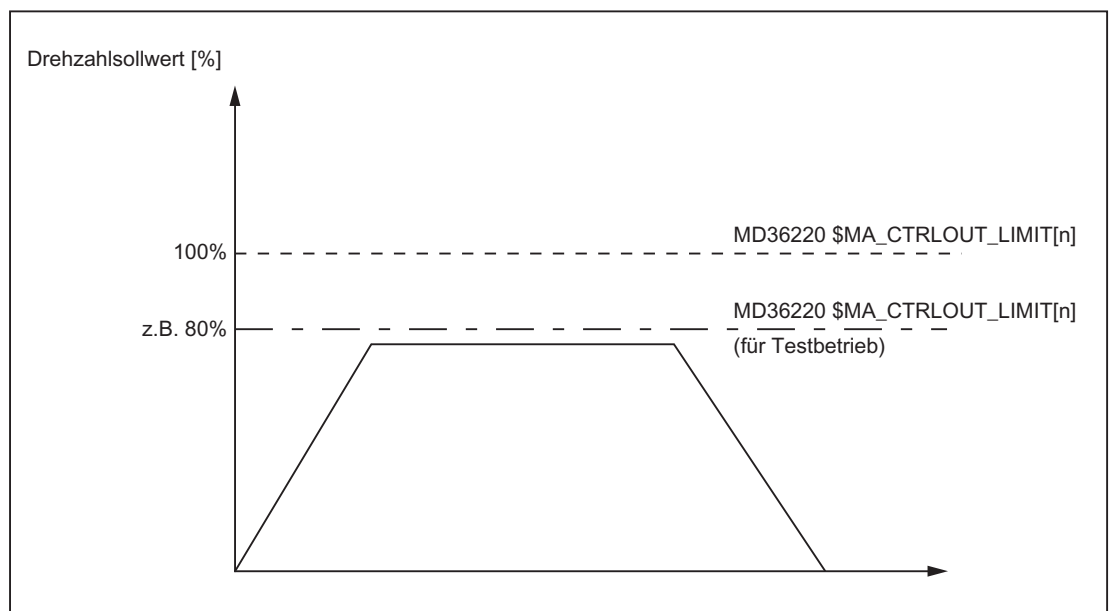


Bild 3-3 Drehzahlsollwertbegrenzung

Verzögerung Drehzahlsollwertüberwachung

Damit es nicht in jedem Fall einer Drehzahlbegrenzung zu einer Fehlerreaktion kommt, kann eine Verzögerungszeit parametrierbar werden:

MD36220 \$MA_CTRL_OUT_LIMIT_TIME (Verzögerung Drehzahlsollwertüberwachung)

Erst wenn eine Drehzahlbegrenzung länger als die eingestellte Zeit erforderlich wird, erfolgt die entsprechende Fehlerreaktion.

Wirksamkeit

Die Drehzahlsollwertüberwachung ist nur für lagegeregelte Achsen aktiv und kann nicht ausgeschaltet werden.

Fehlerfall

Bei Überschreiten der parametrierbaren Verzögerungszeit wird folgender Alarm angezeigt:

25060 "Achse <Achsenname> Drehzahlsollwertbegrenzung"

Die betroffene Achse wird im Nachführbetrieb über die parametrierbare Bremsrampe stillgesetzt:

MD36610 \$MA_AX_EMERGENCY_STOP_TIME

(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

Hinweis

Mit dem Erreichen der Drehzahlsollwertüberwachung wird der Lageregelkreis der Achse durch die Begrenzung nichtlinear. Hieraus resultieren Konturfehler, sofern die Achse an der Konturerzeugung beteiligt ist.

3.4 Istgeschwindigkeitsüberwachung

Funktion

Die Istgeschwindigkeitsüberwachung überwacht, dass die aktuelle Istgeschwindigkeit einer Achse/Spindel den parametrierbaren Schwellwert nicht überschreitet:

MD36200 \$MA_AX_VELO_LIMIT (Schwellwert Geschwindigkeitsüberwachung)

Der Schwellwert sollte 10-15% über der parametrierbaren Maximalgeschwindigkeit liegen:

- Bei Achsen:
MD32000 \$MA_MAX_AX_VELO (Maximale Achsgeschwindigkeit)
- Bei Spindeln:
MD35110 \$MA_GEAR_STEP_MAX_VELO_LIMIT[n] (Maximaldrehzahl der Getriebestufe)

Mit dieser Einstellung tritt eine Überschreitung des Schwellwerts der Geschwindigkeitsüberwachung normalerweise nicht auf (Ausnahme: Antriebsfehler).

Aktivierung

Die Istgeschwindigkeitsüberwachung wird aktiv, sobald das aktive Messsystem gültige Istwerte (Gebergrenzfrequenz nicht überschritten) liefert.

Wirksamkeit

Die Istgeschwindigkeitsüberwachung ist nur wirksam bei aktiver Lageregelung und folgenden Achstypen:

- Linearachsen
- Rundachsen
- Gesteuerten und lagegeregelten Spindeln

Fehlerfall

Bei Überschreitung des Schwellwerts wird folgender Alarm angezeigt:

25030 "Achse <Achsname> Istgeschwindigkeit Alarmgrenze"

Die betroffene Achse wird im Nachführbetrieb über die parametrisierte Bremsrampe stillgesetzt:

MD36610 \$MA_AX_EMERGENCY_STOP_TIME

(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

3.5 Messsystem-Überwachung

Die NC hat keinen direkten Zugriff auf die Messsystem-Hardware, die Messsystem-Überwachungen werden deshalb hauptsächlich von der Antriebs-Software durchgeführt.

Überwachungsfunktionen im Antrieb

- Überwachung von Hardware-Fehlern (z. B. Messsystem-Ausfall, Leitungsbruch)
- Nullmarkenüberwachung

Literatur:

Antriebsfunktionen SINAMICS S120

Im Antrieb durchgeführte Messsystem-Überwachungen werden auf NC-Alarme (Alarm 25000 und folgende) oder NC-Reaktionen (z. B. Abbruch von Referenzieren oder fliegendem Messen) abgebildet. Das genaue Verhalten der NC ist abhängig von der Einstellung im Maschinendatum:

MD36310 \$MA_ENC_ZERO_MONITORING

Wert	Bedeutung	
= 0	Überwachung von HW-Fehlern:	<p>EIN</p> <p>Bei detektierten Hardware-Fehlern im aktiven Messsystem wird der POWER ON-Alarm 25000 angezeigt:</p> <p>"Achse <Achsnamen> Hardwarefehler aktiver Geber"</p> <p>Die betroffene Achse wird im Nachfuhrbetrieb über die parametrisierte Bremsrampe stillgesetzt:</p> <p>MD36610 \$MA_AX_EMERGENCY_STOP_TIME (Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)</p> <p>Bei detektierten Hardware-Fehlern im passiven Messsystem wird der Alarm 25001 angezeigt:</p> <p>"Achse <Achsnamen> Hardwarefehler passiver Geber"</p> <p>Es erfolgt keine weitere Alarmreaktion.</p>
	Nullmarkenüberwachung:	<p>AUS</p> <p>Die Alarmer 25020 und 25021 (s. u.) werden unterdrückt.</p>
= 100	Keine Nullmarkenüberwachung sowie Ausblenden sämtlicher Geberüberwachungen (d. h. neben Alarm 25020 (25021) werden auch die Alarmer 25000 (25001) und 25010 (25011) unterdrückt.	
> 0 aber < 100	Überwachung von HW-Fehlern:	EIN (s. o.)
	Nullmarkenüberwachung:	<p>EIN</p> <p>Bei Ansprechen der Nullmarkenüberwachung im aktiven Messsystem wird der Alarm 25020 angezeigt:</p> <p>"Achse <Achsnamen> Nullmarkenüberwachung aktiver Geber"</p> <p>Die betroffene Achse wird im Nachfuhrbetrieb über die parametrisierte Bremsrampe stillgesetzt:</p> <p>MD36610 \$MA_AX_EMERGENCY_STOP_TIME (Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)</p> <p>Bei Ansprechen der Nullmarkenüberwachung im passiven Messsystem wird der Alarm 25021 angezeigt:</p> <p>"Achse <Achsnamen> Nullmarkenüberwachung passiver Geber"</p> <p>Es erfolgt keine weitere Alarmreaktion.</p>
> 100	Überwachung von HW-Fehlern:	<p>EIN mit abgeschwächter Fehlermeldung:</p> <p>Der POWER ON-Alarm 25000 wird durch den Reset-Alarm 25010 ersetzt, der Reset-Alarm 25001 durch den Cancel-Alarm 25011.</p>
	Nullmarkenüberwachung:	EIN (s. o.)

Details zu den Alarmen siehe:

Literatur:
Diagnosehandbuch

Hinweis

Bei Hardware-Fehlern wird der Referenzierstatus der Maschinenachse zurückgesetzt:
DB31, ... DBX60.4/5 (Referenziert/Synchronisiert 1/2) = 0

Überwachungsfunktionen im NC

- Gebergrenzfrequenzüberwachung
- Plausibilitätskontrolle bei Absolutwertgebern

3.5.1 Gebergrenzfrequenzüberwachung

Funktion

Die NC-seitige Gebergrenzfrequenzüberwachung basiert auf den Projektier- und Telegramm-Informationen des Antriebs. Sie überwacht, dass die Geberfrequenz die projektierte Gebergrenzfrequenz nicht überschreitet:

MD36300 \$MA_ENC_FREQ_LIMIT (Gebergrenzfrequenz)

Die Gebergrenzfrequenzüberwachung bezieht sich immer auf das in der NC/PLC-Nahtstelle angewählte, aktive Messsystem:

DB31, ... DBX1.5 / 1.6 (Lagemesssystem 1/2)

Wirksamkeit

Die Gebergrenzfrequenzüberwachung ist wirksam bei:

- Linearachsen
- Rundachsen
- Gesteuerten und lagegeregelten Spindeln

Fehlerfall

Bei Überschreiten der Gebergrenzfrequenz erfolgt:

- Meldung an die PLC:
DB31, ... DBX60.2 bzw. 60.3 = 1 (Gebergrenzfrequenz überschritten 1 bzw. 2)
- Spindeln
Spindeln werden nicht stillgesetzt, sondern drehen drehzahl geregelt weiter.
Wird die Spindeldrehzahl soweit reduziert, dass die Geberfrequenz die Gebergrenzfrequenz wieder unterschreitet, wird das Istwertsystem der Spindel automatisch neu synchronisiert.
- Achsen
Es wird folgender Alarm angezeigt:
21610 "Kanal <Kanalnummer> Achse <Achsenname> Geber <Gebernummer> -Frequenz überschritten"
Die betroffene Achse wird im Nachführbetrieb über die parametrisierte Bremsrampe stillgesetzt:
MD36610 \$MA_AX_EMERGENCY_STOP_TIME
(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

Hinweis

Nach Überschreitung der Gebergrenzfrequenz muss eine lagegeregelter Maschinenachse neu referenziert werden (siehe Kapitel "R1: Referenzieren (Seite 1243)").

3.5.2 Plausibilitätskontrolle bei Absolutwertgebern

Funktion

Bei Absolutwertgebern (MD30240 \$MA_ENC_TYPE = 4) werden vom Messsystem gelieferte Absolutwerte zur Plausibilitätskontrolle des Istwerts genutzt.

Dazu vergleicht die NC den softwaretechnisch zyklisch im Lageregler-Takt anhand der Inkrementalinformationen des Gebers mitgeführten Positionswert mit einem unmittelbar aus den Absolut- und Inkrementalinformationen des Gebers neu gebildeten Positionswert und überwacht, dass die ermittelte Positionsdifferenz die zulässige Abweichung nicht überschreitet:

MD36310 \$MA_ENC_ZERO_MONITORING (zulässige Abweichung in 1/2 Grobstrichen zwischen der absoluten und der inkrementellen Geberspur)

Hinweis

Die Plausibilitätskontrolle bei Absolutwertgebern deckt vor allem Abweichungen aufgrund von Verschmutzungen der Absolutspur bzw. Störungen der Absolutwert-Übertragung auf. Kleine Verfälschungen der Inkrementalspur (Störimpulse, Pulsfehler) werden dagegen nicht erkannt, die Plausibilitätskontrolle reagiert in solchen Fällen erst bei Abweichungen im Millimeterbereich. Sie ist deshalb als Zusatzüberwachung gedacht, die vornehmlich Störungen der Absolutlage diagnostizieren hilft.

Hinweis

Rotatorische Absolutwertgeber

Soll die Plausibilitätskontrolle bei einem rotatorischen Absolutwertgeber eingesetzt werden, muss bei der Einstellung des Modulo-Bereichs (MD34220 \$MA_ENC_ABS_TURNS_MODULO) der SINAMICS-Parameter p0979 berücksichtigt werden.

Hinweis

Hochrüstung der NC-Software

Ist bei Absolutwertgebern die Plausibilitätskontrolle aktiv (MD36310 > 0), müssen bei einer Hochrüstung der NC-Software die bisherigen Einstellwerte von MD36310 überprüft und ggf. erhöht werden.

Nullmarken-Diagnose

Bei Absolutwertgebern muss bei der Inbetriebnahme die zulässige Abweichung für die Plausibilitätskontrolle ermittelt werden. Dies kann erfolgen über das Maschinendatum:

MD36312 \$MA_ENC_ABS_ZEROMON_WARNING (Nullmarkenüberwachung Warnschwelle)

Wert	Bedeutung
0	keine Nullmarken-Diagnose
> 0	zulässige Abweichung in 1/2 Grobstrichen zwischen der absoluten und der inkrementellen Geberspur

Vorgehensweise bei der Inbetriebnahme:

1. Nullmarkenüberwachung deaktivieren:
MD36310 \$MA_ENC_ZERO_MONITORING = 0
2. Nullmarken-Diagnose aktivieren:
MD36312 \$MA_ENC_ABS_ZEROMON_WARNING = 1
3. Achse bewegen und Systemvariable \$VA_ENC_ZERO_MON_ERR_CNT (Anzahl der erkannten Grenzwert-Überschreitungen) beobachten.
4. Falls \$VA_ENC_ZERO_MON_ERR_CNT ≠ 0:
MD36312-Wert erhöhen und Schritt 3 wiederholen.
5. Falls \$VA_ENC_ZERO_MON_ERR_CNT = 0 (über einen längeren Zeitraum!):
Der richtige Wert für MD36310 ist gefunden! Wert aus MD36312 in MD36310 übernehmen und anschließend MD36312 auf "0" setzen.

Hinweis

Abhängig von der Steifigkeit der Maschine (optimal sind möglichst kleine Last-Massen/ Trägheitsmomente) und der Regler-Einstellungen wird das Regelspiel unterschiedlich stark "pendeln". Dies muss durch maschinenspezifische Grenzwert-Eingabe in MD36310 berücksichtigt werden.

Fehlerfall

Alarm 25020

Bei Ansprechen der Plausibilitätskontrolle im **aktiven** Messsystem wird Alarm 25020 angezeigt:

"Achse <Achsname> Nullmarkenueberwachung aktiver Geber"

Die betroffene Achse wird im Nachföhrbetrieb über die parametrisierte Bremsrampe stillgesetzt:

MD36610 \$MA_AX_EMERGENCY_STOP_TIME

(Maximale Zeitdauer der Bremsrampe bei Fehlerzuständen)

Alarm 25021

Bei Ansprechen der Plausibilitätskontrolle im **passiven** Messsystem wird Alarm 25021 angezeigt:

"Achse <Achsname> Nullmarkenueberwachung passiver Geber"

Es erfolgt keine weitere Alarmreaktion.

Hinweis

Im Fehlerfall geht die Justage des Absolutwertgebers verloren und die Achse ist nicht mehr referenziert. Eine Neujustage des Absolutwertgebers muss durchgeführt werden (siehe Kapitel "Referenzieren bei Absolutwertgebern (Seite 1274)").

Hinweis

Fehler in der Inkrementalspur, die nicht durch die Amplitudenüberwachung erfasst werden können, können zu Positionsabweichungen im Millimeter-Bereich föhren. Die Abweichung ist dabei abhängig von der Gitterteilung/Strichzahl und der Verfahrgeschwindigkeit der Achse bei Auftreten des Fehlers.

Eine vollständige Positionsüberwachung ist nur durch Redundanz, d. h. durch Vergleich mit einem unabhängigen zweiten Messsystem möglich.

3.5.3 Anwenderspezifische Fehlerreaktionen

Anwenderspezifische Nullmarkenüberwachung

Das standardmäßige Alarm- und Reaktionsverhalten der Nullmarkenüberwachung kann bei absoluten Messsystemen (MD30240 \$MA_ENC_TYPE = 4) mit Hilfe von Systemvariablen anwenderspezifisch angepasst werden. Damit hat der Anwender die Möglichkeit, per Synchronaktion oder OEM-Applikation eine eigene Überwachung zu realisieren und alle in diesem Applikations-Zusammenhang vorhandenen Reaktionsmöglichkeiten zu nutzen, z. B.:

- Alarm absetzen
- Zyklen nutzen (z. B. Werkzeugwechsel-Position anfahren)
- ...

Beispiel:

Der Anwender kann das Alarm- und Reaktionsverhalten so anpassen, dass bei einem teuren Werkstück, bei dem das Stillsetzen der Achse infolge eines Alarms zur Beschädigung führen würde, die Bearbeitung erst beendet wird, bevor mit entsprechenden Synchronaktions-Befehlen die Bearbeitungsgüte des Werkstücks bewertet wird.

Wirksamkeit

Die anwenderspezifische Überwachung kann parallel oder alternativ zur standardmäßigen Nullmarkenüberwachung wirksam sein, abhängig von der Einstellung im Maschinendatum:

MD36310 \$MA_ENC_ZERO_MONITORING

Wert	Bedeutung
0	Soll ausschließlich eine anwenderspezifische Überwachung realisiert werden, muss die standardmäßige Nullmarkenüberwachung deaktiviert werden: MD36310 = 0 und MD36312 = 0
> 0	Anwenderspezifische Überwachung und Standard-Nullmarkenüberwachung wirken parallel.
100	Alle Geberüberwachungen sind ausgeschaltet.

Wenn beide Überwachungen wirksam sind (MD36310 > 0), kann eine **kaskadierte Überwachung** realisiert werden.

Beispiel:

Bei einem Wert unterhalb des im MD36310 angegebenen Schwellenwerts setzt die anwenderspezifische Überwachung eine Vorwarnung ab, erst bei Überschreiten des Schwellenwerts erkennt dann die Standard-Nullmarkenüberwachung auf Störung und schaltet ab.

Systemvariablen

Zur Realisierung der anwenderspezifischen Fehlerreaktionen stehen folgende Systemvariablen zur Verfügung:

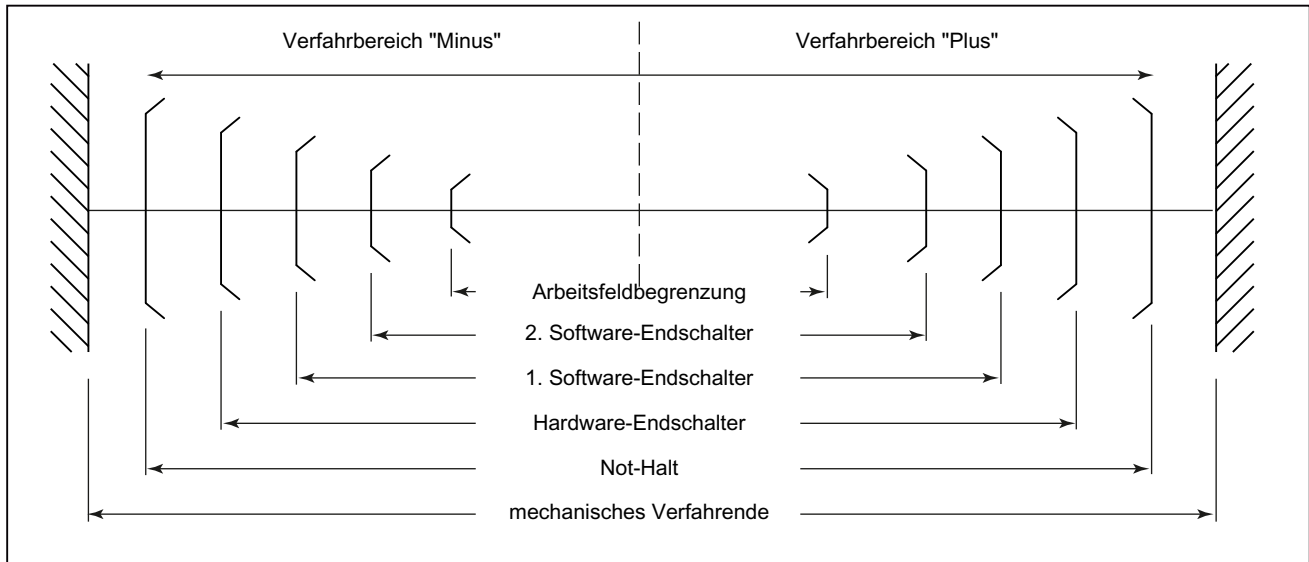
Systemvariable	Bedeutung
\$VA_ENC_ZERO_MON_ERR_CNT[<n>,<Achse>]	<p>Anzahl der erkannten Grenzwert-Überschreitungen</p> <p>Enthält die aktuelle Anzahl der erkannten Grenzwert-Überschreitungen beim Vergleich zwischen absoluter und inkrementeller Geberspur</p> <p>Der Wert wird auf 0 zurückgesetzt bei:</p> <ul style="list-style-type: none"> • POWER ON • An- bzw. Abwahl von Parken <p>Reset bewirkt kein Zurücksetzen.</p>
\$VA_ABSOLUTE_ENC_DELTA_INIT[<n>,<Achse>]	<p>Anfangs-Differenz bei Absolutgeber</p> <p>Enthält den Anfangs-Differenzwert zwischen der letzten im statischen NC-Speicher gepufferten Absolut-Position und der aktuellen Absolut-Position.</p> <p>Format des Differenzwerts: Zahl der internen Inkremente (siehe MD10200 \$MN_INT_INCR_PER_MM bzw. MD10210 \$MN_INT_INCR_PER_DEG)</p> <p>Der Wert wird aktualisiert bei:</p> <ul style="list-style-type: none"> • POWER ON • Warmstart • Abwahl von Parken • Rückkehr unter die Gebergrenzfrequenz <p>Es erfolgt kein Rücksetzen bei Reset.</p>

<n>: Gebernummer

<Achse>: Achsname

3.6 Endschalter-Überwachung

Übersicht der Endbegrenzungen und möglichen Endschalter-Überwachungen:



3.6.1 Hardware-Endschalter

Funktion

Ein Hardware-Endschalter wird normalerweise am Verfahrbereichsende einer Maschinenachse angebracht. Er dient zum Schutz vor einem versehentlichen Überfahren des maximalen Verfahrbereichs der Maschinenachse, während die Maschinenachse noch nicht referenziert ist.

Wird der Hardware-Endschalter ausgelöst, setzt das vom Maschinenhersteller erstellte PLC-Anwenderprogramm das entsprechende Nahtstellensignal:

DB31, ... DBX12.0/1 = 1 (Hardware-Endschalter minus/plus)

Parametrierung

Das Bremsverhalten der Maschinenachse bei Erreichen des Hardware-Endschalters ist parametrierbar über das Maschinendatum:

MD36600 \$MA_BRAKE_MODE_CHOICE (Bremsverhalten bei Hardware-Endschalter)

Wert	Bedeutung
0	Bremsen mit der parametrierten axialen Beschleunigung
1	Schnellstopp (Sollgeschwindigkeit = 0)

Wirksamkeit

Die Hardware-Endschalterüberwachung ist nach dem Hochlauf der Steuerung in allen Betriebsarten aktiv.

Auswirkung

Bei Erreichen des Hardware-Endschalters erfolgt:

- Alarm 21614 "Kanal <Kanalnummer> Achse <Achsname> Hardware-Endschalter <Richtung>"
- Abbremsen der Maschinenachse entsprechend dem parametrisierten Bremsverhalten.
- Steht die Achse/Spindel in einem interpolatorischen Zusammenhang mit anderen Achsen/Spindeln, so werden auch diese gemäß ihrem parametrisierten Bremsverhalten abgebremst.
- Die Verfahrtasten der betroffenen Maschinenachse werden richtungsabhängig gesperrt.

3.6.2 Software-Endschalter

Funktion

Software-Endschalter dienen zur Begrenzung des Verfahrbereiches einer Maschinenachse. Pro Maschinenachse stehen pro Verfahrrichtung zwei (1. und 2.) Software-Endschalter zur Verfügung:

MD36100 POS_LIMIT_MINUS (1. Software-Endschalter minus)

MD36110 POS_LIMIT_PLUS (1. Software-Endschalter plus)

MD36120 POS_LIMIT_MINUS2 (2. Software-Endschalter minus)

MD36130 POS_LIMIT_PLUS2 (2. Software-Endschalter plus)

Standardmäßig ist der 1. Software-Endschalter aktiv. Durch das PLC-Anwenderprogramm kann richtungsspezifisch der 2. Software-Endschalter aktiviert werden:

DB31, ... DBX12.2 / 12.3 (2. Software-Endschalter minus / plus)

Wirksamkeit

Die Software-Endschalter sind wirksam:

- Sofort nach dem erfolgreichem Referenzieren der Maschinenachse.
- In allen Betriebsarten.

Randbedingungen

- Die Software-Endschalter beziehen sich auf das Maschinenkoordinatensystem.
- Die Software-Endschalter müssen innerhalb des Bereichs der Hardware-Endschalter liegen.

- Die Maschinenachse kann auf die Position des aktiven Software-Endschalters gefahren werden.
- **PRESET**
Nach Verwendung der Funktion **PRESET** erfolgt keine Software-Endschalterüberwachung mehr. Die Maschinenachse muss erst erneut referenziert werden.
- **Endlos drehende Rundachsen**
Bei endlos drehenden Rundachsen erfolgt keine Software-Endschalterüberwachung:
MD30310 \$MA_ROT_IS_MODULO == 1 (Modulo-Wandlung für Rundachse und Spindel)
Ausnahme: Aufbau-Rundachsen

Auswirkungen

Automatik-Betriebsarten (AUTOMATIK, MDA)

- Ohne Transformation, ohne überlagerte Bewegung, unveränderter Software-Endschalter:
Ein Teileprogrammsatz, dessen programmierte Verfahrbewegung zu einem Überfahren des Software-Endschalters führen würde, wird nicht begonnen.
- Mit Transformation:
Abhängig von der Transformationsart ergeben sich unterschiedliche Reaktionen:
 - Verhalten wie oben.
oder
 - Der Teileprogrammsatz, dessen programmierte Verfahrbewegung zu einem Überfahren des Software-Endschalters führen würde, wird begonnen. Die betroffene Maschinenachse kommt auf dem aktiven Software-Endschalter zum Stehen. Die anderen an der Verfahrbewegung beteiligten Maschinenachsen werden abgebremst. Die programmierte Kontur wird dabei verlassen.
- Mit überlagerter Bewegung
Der Teileprogrammsatz, dessen programmierte Verfahrbewegung zu einem Überfahren des Software-Endschalters führen würde, wird begonnen. Maschinenachsen, die mit überlagerter Bewegung verfahren werden oder worden sind, kommen auf dem jeweiligen aktiven Software-Endschalter zum Stehen. Die anderen an der Verfahrbewegung beteiligten Maschinenachsen werden abgebremst. Die programmierte Kontur wird dabei verlassen.

Manuelle Betriebsarten

- JOG ohne Transformation
Die Maschinenachse kommt auf der Software-Endschalterposition zum Stehen.
- JOG mit Transformation
Die Maschinenachse kommt auf der Software-Endschalterposition zum Stehen. Andere an der Verfahrbewegung beteiligte Maschinenachsen werden abgebremst. Die vorgegebene Bahn wird dabei verlassen.

Allgemein

- Umschalten des Software-Endschalters (1. ↔ 2. Software-Endschalter)
Liegt die Istposition der Maschinenachse nach dem Umschalten hinter dem Software-Endschalter, wird die Maschinenachse mit der maximal zulässigen Beschleunigung angehalten.
- Überfahren des Software-Endschalters in der Betriebsart JOG
Ist die Position eines Software-Endschalters erreicht und soll per erneuter Betätigung der Verfahrtaste weiter in dieser Richtung verfahren werden, wird ein Alarm angezeigt und die Achse nicht weiter verfahren:
Alarm 10621 "Kanal <Kanalnummer> Achse <Achsenname> steht auf Software-Endschalter <Richtung>"

3.7 Arbeitsfeldbegrenzungs-Überwachung

3.7.1 Allgemein

Funktion

Mit der Funktion "Arbeitsfeldbegrenzung" kann der Verfahrbereich von Geometrie- und Zusatzachsen eines Kanals auf einen zulässigen Arbeitsbereich eingegrenzt werden. Die Funktion überwacht das Einhalten der Arbeitsfeldgrenzen sowohl in der Betriebsart AUTOMATIK als auch in der Betriebsart JOG.

Folgende Varianten stehen zur Verfügung:

- Arbeitsfeldbegrenzung im Basiskoordinatensystem (BKS)
Die Angabe der Verfahrbereichsgrenzen erfolgt bezogen auf das Basiskoordinatensystem.
- Arbeitsfeldbegrenzung im Werkstück-Koordinatensystem (WKS) oder im einstellbaren Nullpunktsystem (ENS)
Die Angabe der Verfahrbereichsgrenzen erfolgt bezogen auf das Werkstück-Koordinatensystem oder das Einstellbare Nullpunktsystem.

Die beiden Überwachungsarten sind unabhängig voneinander. Sind beide gleichzeitig aktiv, wirkt richtungsabhängig diejenige Verfahrbereichsgrenze, welche den Verfahrbereich der Achse am weitesten einschränkt.

Bezugspunkt am Werkzeug

Die Berücksichtigung der Werkzeugdaten (Werkzeuglänge und Werkzeugradius) und damit der Bezugspunkt am Werkzeug bei der Überwachung der Arbeitsfeldbegrenzung ist abhängig vom Status der Transformation im Kanal:

- **Transformation nicht aktiv**

Ohne Transformation wird bei Verfahrbewegungen mit einem aktiven Werkzeug die Position der Werkzeugspitze P überwacht, d. h. bei der Überwachung wird die Werkzeuglänge automatisch berücksichtigt.

Die Berücksichtigung des Werkzeugradius muss separat aktiviert werden:

MD21020 \$MC_WORKAREA_WITH_TOOL_RADIUS (Berücksichtigung des Werkzeugradius bei Arbeitsfeldbegrenzung)

- **Transformation aktiv**

Bei bestimmten Transformationen kann die Überwachung der Arbeitsfeldbegrenzung vom Verhalten ohne Transformation abweichen:

- Die Werkzeuglänge ist Bestandteil der Transformation (\$MC_TRAFO_INCLUDES_TOOL_X = TRUE):
In diesem Fall wird die Werkzeuglänge nicht berücksichtigt, d. h. die Überwachung bezieht sich auf den Werkzeugträgerbezugspunkt.
- Transformation mit Orientierungsänderung:
Bei Transformationen mit Orientierungsänderung bezieht sich die Überwachung immer auf den Werkzeugmittelpunkt. MD21020 hat keinen Einfluss.

Hinweis

Das Maschinendatum \$MC_TRAFO_INCLUDES_TOOL... wird nur bei bestimmten Transformationen ausgewertet. Bedingung für eine mögliche Auswertung ist, dass die Orientierung des Werkzeugs in Bezug auf das Basiskoordinatensystem durch die Transformation nicht verändert werden kann. Bei den Standardtransformationen ist diese Bedingung nur für die Transformationsart "Schräge Achse" erfüllt.

Verhalten

Automatikbetriebsarten

- Mit / ohne Transformation
Der Teileprogrammsatz, dessen programmierte Verfahrbewegung zu einem Überfahren der Arbeitsfeldbegrenzung führen würde, wird nicht ausgeführt.
- Mit überlagerter Bewegung
Die Achse, die durch eine überlagerte Bewegung die Arbeitsfeldbegrenzung verletzen würde, wird mit maximaler Bremsbeschleunigung ohne Ruckbegrenzung (**BRISK**) abgebremst und kommt auf der Position der Arbeitsfeldbegrenzung zum Stehen. Andere an der Bewegung beteiligte Achsen werden entsprechend des aktuellen Beschleunigungsverhaltens (z.B. **SOFT**) abgebremst. Durch unterschiedliche Bremsbeschleunigungen kann dabei der Bahnzusammenhang verloren gehen (Konturverletzung).

Manuelle Betriebsarten

- JOG mit / ohne Transformation
Die Achse kommt auf der Position der Arbeitsfeldbegrenzung zum Stehen.

Einschaltverhalten

Bewegt sich eine Achse beim Einschalten der Arbeitsfeldbegrenzung außerhalb des zulässigen Arbeitsfeldes, wird sie sofort mit maximaler Beschleunigung angehalten.

Überfahren der Arbeitsfeldbegrenzung in der Betriebsart JOG

In der Betriebsart JOG wird eine Achse von der Steuerung maximal bis zu ihrer Arbeitsfeldgrenze verfahren. Nach einer erneuten Betätigung der Verfahrtaste wird ein Alarm angezeigt und die Achse nicht weiter verfahren.

Geometrieachstausch

Über das folgende Maschinendatum ist einstellbar, ob beim Geometrieachstausch die aktive Arbeitsfeldbegrenzung erhalten bleibt oder deaktiviert wird:

MD10604 \$MN_WALIM_GEOAX_CHANGE_MODE = <Wert>

<Wert>	Bedeutung
0	Die Arbeitsfeldbegrenzung wird beim Geometrieachstausch deaktiviert.
1	Die Arbeitsfeldbegrenzung bleibt beim Geometrieachstausch aktiviert.

3.7.2 Arbeitsfeldbegrenzung im BKS**Anwendung**

Durch die "Arbeitsfeldbegrenzung im BKS" wird der Arbeitsraum einer Werkzeugmaschine so begrenzt, dass die umliegenden Anlageteile (z. B. Werkzeugrevolver, Messstationen) vor Beschädigung geschützt sind.

Arbeitsfeldgrenzen

Die unteren und oberen Arbeitsfeldgrenzen der einzelnen Achsen werden über Settingdaten eingestellt oder über Teileprogrammanweisungen programmiert:

Arbeitsfeldbegrenzung über Settingdaten

Die Einstellungen erfolgen über die sofort wirksamen achsspezifischen Settingdaten:

SD43420 \$SA_WORKAREA_LIMIT_PLUS (Arbeitsfeldbegrenzung plus)

SD43430 \$SA_WORKAREA_LIMIT_MINUS (Arbeitsfeldbegrenzung minus)

Programmierte Arbeitsfeldbegrenzung

Die Programmierung erfolgt über die G-Befehle:

G25 X...Y...Z... untere Arbeitsfeldbegrenzung

G26 X...Y...Z... obere Arbeitsfeldbegrenzung

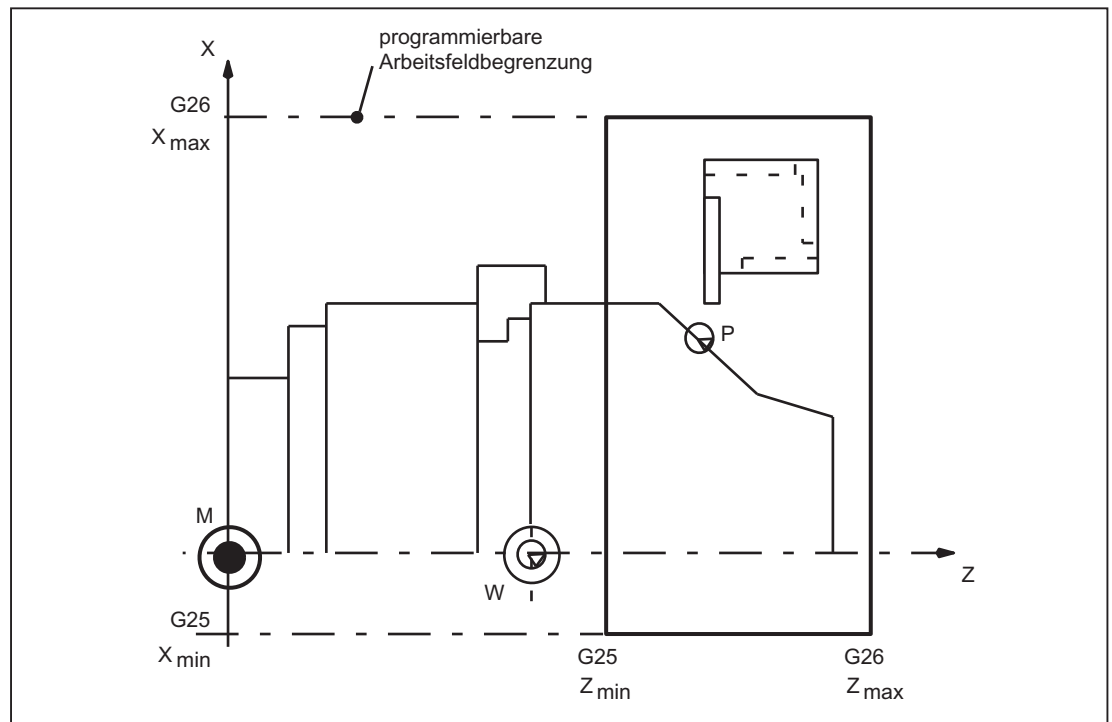


Bild 3-4 Programmierter Arbeitsfeldbegrenzung

Die programmierte Arbeitsfeldbegrenzung hat Vorrang und überschreibt die in SD43420 und SD43430 eingetragenen Werte.

Aktivierung/Deaktivierung

Arbeitsfeldbegrenzung über Settingdaten

Die Aktivierung bzw. Deaktivierung der Arbeitsfeldbegrenzung für einzelne Achsen erfolgt richtungsspezifisch über die sofort wirksamen Settingdaten:

SD43400 \$SA_WORKAREA_PLUS_ENABLE (Arbeitsfeldbegrenzung in positiver Richtung aktiv)

SD43410 \$SA_WORKAREA_MINUS_ENABLE (Arbeitsfeldbegrenzung in negativer Richtung aktiv)

Wert	Bedeutung
0	Die Arbeitsfeldbegrenzung in positiver bzw. negativer Richtung ist ausgeschaltet .
1	Die Arbeitsfeldbegrenzung in positiver bzw. negativer Richtung ist aktiv .

Programmierte Arbeitsfeldbegrenzung

Die Aktivierung bzw. Deaktivierung der gesamten "Arbeitsfeldbegrenzung im BKS" erfolgt durch die Teileprogrammanweisungen:

WALIMON	Arbeitsfeldbegrenzung EIN
bzw.	
WALIMOF	Arbeitsfeldbegrenzung AUS

Ändern der Arbeitsfeldbegrenzung

Arbeitsfeldbegrenzung über Settingdaten

HMI-Bedienoberfläche: Bedienbereich "Parameter"

- Automatikbetriebsarten:
 - Änderungen: nur im RESET-Zustand möglich
 - Wirksamkeit: sofort
- Manuelle Betriebsarten:
 - Änderungen: immer möglich
 - Wirksamkeit: mit Start der nächsten Verfahrbewegung

Programmierte Arbeitsfeldbegrenzung

Die Arbeitsfeldbegrenzung kann im Teileprogramm über G25 bzw. G26 <Achsenname> <Wert> geändert werden. Die Änderung ist sofort wirksam.

Der neue Wert der Arbeitsfeldbegrenzung bleibt auch nach NC-RESET und POWER ON erhalten, wenn für SD43420 und SD43430 die Sicherung in die remanente Datenhaltung des NC aktiviert wurde:

MD10710 \$MN_PROG_SD_RESET_SAVE_TAB[0] = 43420

MD10710 \$MN_PROG_SD_RESET_SAVE_TAB[1] = 43430

Löschstellung

Die Löschstellung der Arbeitsfeldbegrenzung (WALIMON oder WALIMOF) ist einstellbar über:

MD20150 \$MC_GCODE_RESET_VALUES (Löschstellung der G-Gruppen)

3.7.3 Arbeitsfeldbegrenzung im WKS/ENS

Anwendung

Die "Arbeitsfeldbegrenzung im WKS/ENS" ermöglicht eine flexible werkstückspezifische Begrenzung des Verfahrbereichs der Kanalachsen im Werkstückkoordinatensystem (WKS) oder Einstellbaren Nullpunktsystem (ENS). Sie ist hauptsächlich für den Einsatz im Bereich konventioneller Drehmaschinen gedacht.

Voraussetzung

Die Kanalachsen müssen referenziert sein.

Arbeitsfeldbegrenzungsgruppe

Damit beim Umschalten von Achszuordnungen, z. B. beim Ein/Ausschalten von Transformationen oder des aktiven Frames, die achsspezifischen Arbeitsfeldgrenzen nicht für alle Kanalachsen neu geschrieben werden müssen, stehen Arbeitsfeldbegrenzungsgruppen zur Verfügung.

Eine Arbeitsfeldbegrenzungsgruppe umfasst folgende Daten:

- Arbeitsfeldgrenzen für alle Kanalachsen
- Bezugssystem der Arbeitsfeldbegrenzung

Die Anzahl der Arbeitsfeldbegrenzungsgruppen wird kanalspezifisch eingestellt im Maschinendatum:

MD28600 \$MC_MM_NUM_WORKAREA_CS_GROUPS

Pro Kanal sind maximal 10 Arbeitsfeldbegrenzungsgruppen möglich.

Arbeitsfeldgrenzen einstellen

Die Arbeitsfeldgrenzen innerhalb eines Kanals werden für jede Kanalachse über folgende Systemvariablen eingestellt:

- \$P_WORKAREA_CS_LIMIT_PLUS[<WALimNo>, <Ax>]
- \$P_WORKAREA_CS_LIMIT_MINUS[<WALimNo>, <Ax>]

mit: <WALimNo> = Arbeitsfeldbegrenzungsgruppe

Wertebereich: 0 (Gruppe 1) ... 9 (Gruppe 10)

<Ax> = Kanalachsname

Arbeitsfeldgrenzen freigeben

Die Arbeitsfeldgrenzen innerhalb eines Kanals werden für jede Kanalachse über folgende Systemvariablen freigegeben:

- \$P_WORKAREA_CS_PLUS_ENABLE[<WALimNo>, <Ax>]
- \$P_WORKAREA_CS_MINUS_ENABLE[<WALimNo>, <Ax>]

mit: <WALimNo> = Arbeitsfeldbegrenzungsgruppe

Wertebereich: 0 (Gruppe 1) ... 9 (Gruppe 10)

<Ax> = Kanalachsname

Durch die richtungsspezifische Freigabe ist es möglich, den Arbeitsbereich für eine Achse nur in einer Richtung zu begrenzen.

Durch die Freigabe erfolgt keine Aktivierung.

Bezugssystem auswählen

Das Bezugssystem für eine Arbeitsfeldbegrenzungsgruppe innerhalb eines Kanals wird über folgende Systemvariable eingestellt:

\$P_WORKAREA_CS_COORD_SYSTEM[<WALimNo>] = <Wert>

mit: <WALimNo> = Arbeitsfeldbegrenzungsgruppe

Wertebereich: 0 (Gruppe 1) ... 9 (Gruppe 10)

<Wert>	Bedeutung
1	Bezugssystem ist das WKS.
3	Bezugssystem ist das ENS.

Arbeitsfeldgrenzen einschalten

Das Einschalten der Arbeitsfeldbegrenzungen einer Arbeitsfeldbegrenzungsgruppe erfolgt im Teileprogramm über den G-Befehl `WALCS<n>`

mit: <n> = Nummer der Arbeitsfeldbegrenzungsgruppe

Wertebereich: 1 ... 10

Arbeitsfeldgrenzen ausschalten

Das Ausschalten der im Kanal aktiven Arbeitsfeldbegrenzungen einer Arbeitsfeldbegrenzungsgruppe erfolgt im Teileprogramm über den G-Befehl `WALCS0`.

Arbeitsfeldgrenzen ändern

Über die oben genannten Systemvariablen können die Arbeitsfeldgrenzen jederzeit geändert werden. Änderungen werden mit dem nächsten Einschalten der Arbeitsfeldbegrenzungsgruppe (`WALCSn`) wirksam.

Datenablage

Die Systemvariablen der Arbeitsfeldbegrenzungen werden remanent im statischen Speicher der NC abgelegt.

Hinweis

Bei Linearachsen wird bei der Ablage der Begrenzungswerte die Grundeinstellung für das Maßsystem (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) berücksichtigt.

Datensicherung

Die Systemvariablen der Arbeitsfeldbegrenzungen können in eigenen Dateien gesichert werden:

- `_N_CHx_WAL`
Zur Sicherung der Systemvariablenwerte für den Kanal x.
- `_N_COMPLETE_WAL`
Zur Sicherung der Systemvariablenwerte für alle Kanäle.

Hinweis

Die Systemvariablen der Arbeitsfeldbegrenzungen sind Bestandteil der Datei "`_N_INITIAL_INI`".

Verhalten in Betriebsart JOG

Ausgangssituation:

- In der Betriebsart JOG verfahren gleichzeitig **mehrere** Geometrieachsen (z. B. durch mehrere Handräder).
- Zwischen dem Basiskoordinatensystem (BKS) und dem Bezugskoordinatensystem der Arbeitsfeldbegrenzung (WKS oder ENS) ist ein **drehender** Frame aktiv.

Verhalten bei Ansprechen einer Arbeitsfeldbegrenzung:

- Die Verfahrbewegungen der nicht betroffenen Geometrieachsen werden fortgesetzt.
- Die betroffene Geometrieachse wird auf der Arbeitsfeldgrenze gestoppt.

Löschstellung einstellen

Die Arbeitsfeldbegrenzungsgruppe, die bei Hochlauf, Reset bzw. Teileprogrammende und Teileprogrammstart wirksam werden soll, wird kanalspezifisch voreingestellt über das Maschinendatum:

MD20150 \$MC_GCODE_RESET_VALUE[59] = <n>

mit: <n> = Nummer der Arbeitsfeldbegrenzungsgruppe

Wertebereich: 1 ... 10

Ob die voreingestellte Arbeitsfeldbegrenzungsgruppe bei Hochlauf und Reset bzw. Teileprogrammende tatsächlich zur Anwendung kommt, ist von folgender Einstellung abhängig:

MD20152 \$MC_GCODE_RESET_MODE[59] = <Wert>

<Wert>	Bedeutung
0	Die Arbeitsfeldgruppe gemäß MD20150 wird wirksam (Grundeinstellung).
1	Die letzte aktive Arbeitsfeldgruppe bleibt aktiv.

3.7.4 Beispiel: Arbeitsfeldbegrenzung im WKS/ENS

Annahme

Kanalachsen

Im Kanal sind vier Achsen definiert:

- Linearachsen: X, Y, Z
- Rundachse: A (nicht modulo)

Anforderungen

Kanalachsen

Im Kanal sind vier Achsen definiert:

- Linearachsen: X, Y, Z
- Rundachse: A (nicht modulo)

Arbeitsfeldbegrenzungsgruppen

Im Kanal sollen drei Arbeitsfeldbegrenzungsgruppen zur Verfügung stehen:

MD28600 \$MC_MM_NUM_WORKAREA_CS_GROUP = 3

Von diesen drei Arbeitsfeldbegrenzungsgruppen werden im Folgenden zwei Gruppen definiert.

Koordinatensysteme

- Arbeitsfeldbegrenzungsgruppe 1: Arbeitsfeldbegrenzung im Einstellbaren Nullpunktsystem (**ENS**).
- Arbeitsfeldbegrenzungsgruppe 2: Arbeitsfeldbegrenzung im Werkstückkoordinatensystem (**WKS**).

Arbeitsfeldbegrenzungsgruppe 1

- X-Achse in Plus-Richtung: 10 mm
- X-Achse in Minus-Richtung: keine Begrenzung
- Y-Achse in Plus-Richtung: keine Begrenzung
- Y-Achse in Minus-Richtung: 25 mm
- Z-Achse in Plus-Richtung: keine Begrenzung
- Z-Achse in Minus-Richtung: keine Begrenzung
- A-Achse in Plus-Richtung: 10 Grad
- A-Achse in Minus-Richtung: -40 Grad

Definition über Systemvariablen im NC-Programm**Programmcode**

```

; Arbeitsfeldbegrenzungsgruppe 1
$P_WORKAREA_CS_COORD_SYSTEM[1] = 3          ; Arbeitsfeldbegrenzung im ENS
$P_WORKAREA_CS_PLUS_ENABLE[1,X] = TRUE
$P_WORKAREA_CS_LIMIT_PLUS[1,X] = 10
$P_WORKAREA_CS_MINUS_ENABLE[1,X] = FALSE
$P_WORKAREA_CS_PLUS_ENABLE[1,Y] = FALSE
$P_WORKAREA_CS_MINUS_ENABLE[1,Y] = TRUE
$P_WORKAREA_CS_LIMIT_MINUS[1,Y] = 25
$P_WORKAREA_CS_PLUS_ENABLE[1,Z] = FALSE
$P_WORKAREA_CS_MINUS_ENABLE[1,Z] = FALSE
$P_WORKAREA_CS_PLUS_ENABLE[1,A] = TRUE
$P_WORKAREA_CS_LIMIT_PLUS[1,A] = 10
$P_WORKAREA_CS_MINUS_ENABLE[1,A] = TRUE
$P_WORKAREA_CS_LIMIT_MINUS[1,A] = -40

```

Arbeitsfeldbegrenzungsgruppe 2

- X-Achse in Plus-Richtung: 10 mm
- X-Achse in Minus-Richtung: keine Begrenzung
- Y-Achse in Plus-Richtung: 34 mm
- Y-Achse in Minus-Richtung: -25 mm
- Z-Achse in Plus-Richtung: keine Begrenzung
- Z-Achse in Minus-Richtung: -600 mm
- A-Achse in Plus-Richtung: keine Begrenzung
- A-Achse in Minus-Richtung: keine Begrenzung

Definition über Systemvariablen im NC-Programm**Programmcode**

```

; Arbeitsfeldbegrenzungsgruppe 2
$P_WORKAREA_CS_COORD_SYSTEM[2] = 1          ; Arbeitsfeldbegrenzung im WKS
$P_WORKAREA_CS_PLUS_ENABLE[2,X] = TRUE
$P_WORKAREA_CS_LIMIT_PLUS[2,X] = 10
$P_WORKAREA_CS_MINUS_ENABLE[2,X] = FALSE
$P_WORKAREA_CS_PLUS_ENABLE[2,Y] = TRUE
$P_WORKAREA_CS_LIMIT_PLUS[2,Y] = 34
$P_WORKAREA_CS_MINUS_ENABLE[2,Y] = TRUE
$P_WORKAREA_CS_LIMIT_MINUS[2,Y] = -25
$P_WORKAREA_CS_PLUS_ENABLE[2,Z] = FALSE
$P_WORKAREA_CS_MINUS_ENABLE[2,Z] = TRUE
$P_WORKAREA_CS_LIMIT_PLUS[2,Z] = -600

```

Programmcode

```
$P_WORKAREA_CS_PLUS_ENABLE[2,A] = FALSE  
$P_WORKAREA_CS_MINUS_ENABLE[2,A] = FALSE
```

Aktivierung

Die Aktivierung der Arbeitsfeldbegrenzungsgruppen erfolgt im NC-Programm mit dem Befehl `WALCS<x>`, mit x: Nummer der Arbeitsfeldbegrenzungsgruppe

3.8 Parken einer Maschinenachse

Wird eine Maschinenachse in den Zustand "Parken" versetzt, werden bezüglich dieser Achse keine Geberistwerte mehr erfasst, sowie alle in den vorherigen Kapiteln beschriebenen Überwachungen (Messsystem-, Stillstands-, Klemmungsüberwachung etc.) ausgeschaltet.

Aktivierung / Deaktivierung

Parken aktivieren

Die Funktion "Parken" wird für eine Maschinenachse durch **Rücksetzen** der achsspezifischen NC/PLC-Nahtstellensignale für die Lagemesssysteme und der Reglerfreigabe **aktiviert**:

- DB31, ... DBX1.5 = 0 (Lagemesssystem 1)
- DB31, ... DBX1.6 = 0 (Lagemesssystem 2)
- DB31, ... DBX2.1 = 0 (Reglerfreigabe)

Der Geberstatus der Lagemesssysteme der Achse wird daraufhin auf "Nicht referenziert" gesetzt:

- DB31, ... DBX60.4 = 0 (Referenziert/Synchronisiert, Lagemesssystem 1)
- DB31, ... DBX60.5 = 0 (Referenziert/Synchronisiert, Lagemesssystem 2)

Folgende weitere NC/PLC-Nahtstellensignale werden ebenfalls zurückgesetzt:

- DB31, ... DBX61.5 = 0 (Lageregler aktiv)
- DB31, ... DBX61.6 = 0 (Drehzahlregler aktiv)
- DB31, ... DBX61.7 = 0 (Stromregler aktiv)
- DB31, ... DBX93.7 = 0 (Impulse freigegeben)
- DB31, ... DBX102.5 = 0 (Lagemesssystem 1 eingeschaltet)
- DB31, ... DBX102.6 = 0 (Lagemesssystem 2 eingeschaltet)

Parken deaktivieren

Die Funktion "Parken" wird für eine Maschinenachse durch **Setzen** der achsspezifischen NC/PLC-Nahtstellensignale für das zu aktivierende Lagemesssystem und der Reglerfreigabe **deaktiviert**:

- DB31, ... DBX1.5 = 1 (Lagemesssystem 1)
bzw.
DB31, ... DBX1.6 = 1 (Lagemesssystem 2) = 1
- DB31, ... DBX2.1 = 1 (Reglerfreigabe) = 1

Die Lageregelung wird für die Maschinenachse an der aktuellen Position wieder aktiv.

Der Geberstatus der Lagemesssysteme ist abhängig vom Messsystemtyp:


- Inkrementelles Lagemesssystem \Rightarrow Zustand "Nicht referenziert"
 - DB31, ... DBX60.4 = 0 (Referenziert/Synchronisiert, Lagemesssystem 1)
 - DB31, ... DBX60.5 = 0 (Referenziert/Synchronisiert, Lagemesssystem 2)
- Absolutes Lagemesssystem \Rightarrow Zustand "Referenziert/Synchronisiert"
 - DB31, ... DBX60.4 = 1 (Referenziert/Synchronisiert, Lagemesssystem 1)
 - DB31, ... DBX60.5 = 1 (Referenziert/Synchronisiert, Lagemesssystem 2)

Folgende weitere NC/PLC-Nahtstellensignale werden ebenfalls wieder gesetzt:

- DB31, ... DBX61.5 = 1 (Lageregler aktiv)
- DB31, ... DBX61.6 = 1 (Drehzahlregler aktiv)
- DB31, ... DBX61.7 = 1 (Stromregler aktiv)
- DB31, ... DBX93.7 = 1 (Impulse freigegeben)
- DB31, ... DBX102.5 = 1 (Lagemesssystem 1 eingeschaltet)
- DB31, ... DBX102.6 = 1 (Lagemesssystem 2 eingeschaltet)

Inkrementelle Lagemesssysteme

Nach dem Ausschalten des Zustands "Parken" müssen inkrementelle Lagemesssysteme zum Erreichen des Geberstatus "Referenziert" neu referenziert werden.

 WARNUNG	<p>Fehlsynchronisation des Lagemesssystems durch Versatz der Maschinenachsen-Istposition</p> <p>Wurden während des "Parkens" Veränderungen am Lagemesssystem vorgenommen, die eine Änderung an den parametrisierten Maschinendaten erforderlich machen, z. B. Anbau eines anderen Gebers, muss das Lagemesssystem vollständig neu vermessen und referenziert werden. Siehe Kapitel "R1: Referenzieren (Seite 1243)".</p>
--	---

Maschinenachse ohne Lagemesssystem

Bei einer Maschinenachse ohne Lagemesssystem (drehzahlgeregelte Spindel) wird ein dem "Parken" entsprechender Zustand durch Wegnahme der Reglerfreigabe aktiviert:

- DB31, ... DBX2.1 = 0 (Reglerfreigabe)

3.9 Parken des passiven Lagemesssystems

3.9.1 Funktion

Im Unterschied zur Funktion "Parken einer Maschinenachse (Seite 122)", bei der alle Lagemesssysteme einer Maschinenachse ausgeschaltet werden, hat der Anwender mit der Funktion "Parken des passiven Lagemesssystems" die Möglichkeit, nur das passive Lagemesssystem einer Maschinenachse zu "parken" (d. h. die Geberauswertung und -überwachung im Antrieb und in der Steuerung auszuschalten), während das aktive Lagemesssystem in Betrieb bleiben kann.

Hinweis

Erläuterungen bezüglich aktives/passives Messsystem siehe "Soll-/Istwertsystem (Seite 360)".

Anwendung

Die Funktion "Parken des passiven Lagemesssystems" kann z. B. in folgenden Fällen angewendet werden:

- Wechsel von Vorsatzköpfen mit und ohne eingebautem Geber
Über die Funktion "Parken des passiven Lagemesssystems" ist es möglich, Vorsatzköpfe mit und ohne eingebautem Geber für unterschiedliche Bearbeitungsaufgaben abwechselnd auf der Hauptspindel zu montieren, ohne dass die fehlenden Gebersignale Antriebs- und Steuerungsfehler verursachen.
Siehe auch:
 - Beispiel: Vorsatzkopfwechsel bei einem direkten Lagemesssystem (Seite 128)
 - Beispiel: Vorsatzkopfwechsel bei zwei direkten Lagemesssystemen (Seite 133)
- Verwendung von linearen Lagemesssystemen, die nicht im gesamten Verfahrbereich einer Maschinenachse verfügbar sind
Über die Funktion "Parken des passiven Lagemesssystems" ist es möglich, den Bereich außerhalb des linearen Lagemesssystems zu durchfahren, ohne dass die fehlenden Gebersignale Antriebs- und Steuerungsfehler verursachen.
Siehe auch:
 - "Beispiel: Messsystemumschaltung bei bereichsweise fehlenden Gebern (Seite 137)".

Aktivierung / Deaktivierung

Aktivierung

Das passive Lagemesssystem einer Maschinenachse wird unter folgenden Bedingungen geparkt:

- Für das Messsystem ist die Funktion "Parken des passiven Lagemesssystems" eingeschaltet:
MD31046 \$MA_ENC_PASSIVE_PARKING[<n>] = 1
mit <n> = 0 (Lagemesssystem 1) bzw. 1 (Lagemesssystem 2)

Hinweis

MD31046 ist **nicht wirksam**:

- bei Achsen mit weniger als zwei Gebern:
MD30200 \$MA_NUM_ENCS < 2
 - bei simulierten Gebern:
MD30240 \$MA_ENC_TYPE = 0
-

Hinweis

Für Lagemesssysteme, die als Motormesssysteme verwendet werden, sollte die Funktion "Parken des passiven Lagemesssystems" ausgeschaltet sein (MD31046 = 0)!

und

- Vom Anwender wird das folgende NC/PLC-Nahtstellensignal auf "0" gesetzt:
DB31, ... DBX1.5 (Lagemesssystem 1) = 0
bzw.
DB31, ... DBX1.6 (Lagemesssystem 2) = 0

Die Steuerung setzt daraufhin das Statussignal für den Einschaltzustand des Lagemesssystems auf "0":

DB31, ... DBX102.5 (Lagemesssystem 1 eingeschaltet) == 0

bzw.

DB31, ... DBX102.6 (Lagemesssystem 2 eingeschaltet) == 0

Das Lagemesssystem wird jetzt nicht mehr überwacht und aktualisiert.

Deaktivierung

"Parken" wird ausgeschaltet, wenn das Lagemesssystem vom Anwender aktiv geschaltet wird:

DB31, ... DBX1.5 (Lagemesssystem 1) = 1

bzw.

DB31, ... DBX1.6 (Lagemesssystem 2) = 1

Die Steuerung setzt daraufhin das Statussignal für den Einschaltzustand des Lagemesssystems wieder auf "1":

DB31, ... DBX102.5 (Lagemesssystem 1 eingeschaltet) == 1

bzw.

DB31, ... DBX102.6 (Lagemesssystem 2 eingeschaltet) == 1

Hinweis

Die Umschaltung auf ein geparktes Lagemesssystem dauert länger als auf ein nicht geparktes Lagemesssystem. Aufgrund der Zeitdauer sollte die Umschaltung im Achsstillstand vorgenommen werden.

Position des Lagemesssystems

Absolute Lagemesssysteme

Bei absoluten Lagemesssystemen entspricht die Position nach dem Ausschalten von "Parken" der aktuellen Absolutposition des Gebers.

Das Lagemesssystem ist referenziert:

DB31, ... DBX60.4 (Referenziert/Synchronisiert, Lagemesssystem 1) == 1

bzw.

DB31, ... DBX60.5 (Referenziert/Synchronisiert, Lagemesssystem 2) == 1

Inkrementelle Lagemesssysteme

Bei inkrementellen Lagemesssystemen entspricht die Position nach dem Ausschalten von "Parken" grundsätzlich der letzten Ausschaltposition des Lagemesssystems.

Die Umschaltung auf das geparkte Lagemesssystem erfolgt nur, wenn die parametrisierte zulässige Abweichung zwischen den Istwerten der beiden Lagemesssysteme (siehe MD36500 \$MA_ENC_CHANGE_TOL) nicht überschritten wird. Andernfalls muss der Anwender die Funktion "Parken einer Maschinenachse" anwenden, bei der keine derartige Überprüfung stattfindet.

Das Lagemesssystem ist **nicht** referenziert:

DB31, ... DBX60.4 (Referenziert/Synchronisiert, Lagemesssystem 1) == 0

bzw.

DB31, ... DBX60.5 (Referenziert/Synchronisiert, Lagemesssystem 2) == 0

Inkrementelle Lagemesssysteme mit Positionsübernahme

Alternativ besteht bei inkrementellen Lagemesssystemen mit aktiver Funktion "Parken des passiven Lagemesssystems" (MD31046 \$MA_ENC_PASSIVE_PARKING[<n>] = 1) die Möglichkeit, nach dem Ausschalten von "Parken" die Position und ggf. auch den "Referenziert"-Status vom zuvor aktiven Lagemesssystem zu übernehmen.

Diese Funktion ist für jedes Lagemesssystem einer Maschinenachse einschaltbar über das Maschinendatum:

MD34210 \$MA_ENC_REFP_STATE[<n>]

mit <n> = 0 (Lagemesssystem 1) bzw. 1 (Lagemesssystem 2)

Wert	Bedeutung
1	Nur die Position wird vom zuvor aktiven Lagemesssystem übernommen. Das Lagemesssystem ist nicht referenziert: DB31, ... DBX60.4 (Referenziert/Synchronisiert, Lagemesssystem 1) == 0 bzw. DB31, ... DBX60.5 (Referenziert/Synchronisiert, Lagemesssystem 2) == 0
2	Position und "Referenziert"-Status werden vom zuvor aktiven Lagemesssystem übernommen. Das Lagemesssystem ist referenziert: DB31, ... DBX60.4 (Referenziert/Synchronisiert, Lagemesssystem 1) == 1 bzw. DB31, ... DBX60.5 (Referenziert/Synchronisiert, Lagemesssystem 2) == 1

Hinweis

Nur bei inkrementellen Lagemesssystemen werden in Abhängigkeit von MD34210 \$MA_ENC_REFP_STATE[<n>] und nur bei aktiver Funktion "Parken des passiven Lagemesssystems" (MD31046 \$MA_ENC_PASSIVE_PARKING[<n>] = 1) Position und "Referenziert"-Status vom zuvor aktiven Lagemesssystem übernommen.

Die übernommene Position hat die Genauigkeit des zuvor aktiven Lagemesssystems. Sollte diese Genauigkeit nicht ausreichen, ist das Lagemesssystem neu zu referenzieren.



WARNUNG

Fehlsynchronisation des Lagemesssystems durch Versatz der Maschinenachsen-Istposition

Wurden während des "Parkens" Veränderungen am Lagemesssystem vorgenommen, die eine Änderung an den parametrisierten Maschinendaten erforderlich machen, z. B. Anbau eines anderen Gebers, muss das Lagemesssystem vollständig neu vermessen und referenziert werden. Siehe Kapitel "R1: Referenzieren (Seite 1243)".

3.9.2 Randbedingungen

Wechselwirkung mit "Dual-Position-Feedback"

Die Anwendung der Funktion "Parken des passiven Lagemesssystems" kann **nicht** in Verbindung mit der Funktion "Dual-Position-Feedback" (MD32960 \$MA_POSCTRL_DUAL_FEEDBACK_TIME > 0) genutzt werden.

Wechselwirkung mit "Lagedifferenz-Aufschaltung"

Die Anwendung der Funktion "Parken des passiven Lagemesssystems" kann **nicht** in Verbindung mit der Funktion "Lagedifferenz-Aufschaltung" (MD32950 \$MA_POSCTRL_DAMPING > 0) genutzt werden.

Wechselwirkung mit APC (Option bei SINUMERIK 840D sl)

Die Funktion "Parken des passiven Lagemesssystems" kann **nicht** in Verbindung mit der Antriebsfunktion "Advanced Positioning Control (APC)" genutzt werden.

Wechselwirkung mit Geber-Safety-Schutzkonzept

In Verbindung mit der Funktion "Parken des passiven Lagemesssystems" ist **nur** das 1-Geber-Safety-Schutzkonzept einsetzbar.

Wechselwirkung beim Ziehen und Stecken am DRIVE-CLiQ

Werden statt der Geberleitung die DRIVE-CLiQ-Leitungen z. B. zwischen SMC und Motor Module gezogen und gesteckt, so können solche Geber nur über die Funktion "Parken einer Maschinenachse (Seite 122)" fehlerfrei entparkt werden.

3.9.3 Beispiel: Vorsatzkopfwechsel bei einem direkten Lagemesssystem

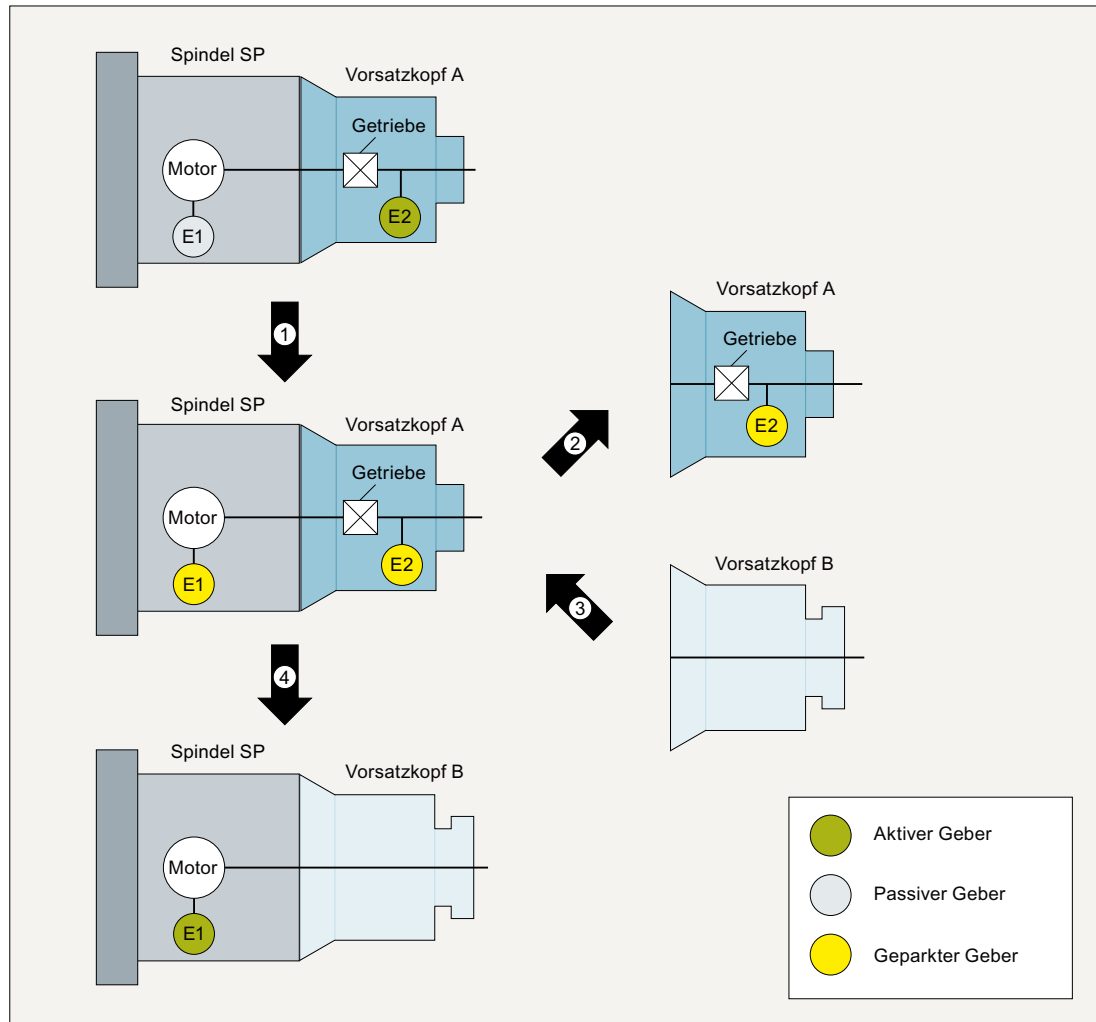
Ausgangssituation

- Der Vorsatzkopf "A" hat einen Geber E2.
- Der Vorsatzkopf "B" hat keinen Geber.
- Die Spindel "SP" hat einen Geber E1.
- In MD13060 \$MN_DRIVE_TELEGRAM_TYPE (Standard-Telegramm-Typ für PROFIdrive) ist einer der beiden folgenden Telegrammtypen projektiert:
 - Telegramm 116 (Motorgeber + ein externer Geber)
 - oder
 - Telegramm 136 (Motorgeber + ein externer Geber, zzgl. Momentenvorsteuerung)
- In der Spindel "SP" sind folgende Lagemesssysteme projektiert:
 - Motorgeber E1 als Lagemesssystem 1
 - Direkter Geber E2 als Lagemesssystem 2
- Aktuell ist Vorsatzkopf "A" mit Geber E2 auf der Spindel montiert.
- Lagemesssystem 2 ist das aktive Messsystem:
DB31, ... DBX1.6 = 1
Lagemesssystem 1 ist passiv.
- Die Funktion "Parken des passiven Lagemesssystems" ist:
 - nicht aktiv für Lagemesssystem 1:
MD31046 \$MA_ENC_PASSIVE_PARKING [0] = 0
 - aktiv für Lagemesssystem 2:
MD31046 \$MA_ENC_PASSIVE_PARKING[1] = 1

Ziel

Der Anwender möchte vom Vorsatzkopf "A" auf Vorsatzkopf "B" wechseln.

Durchführung



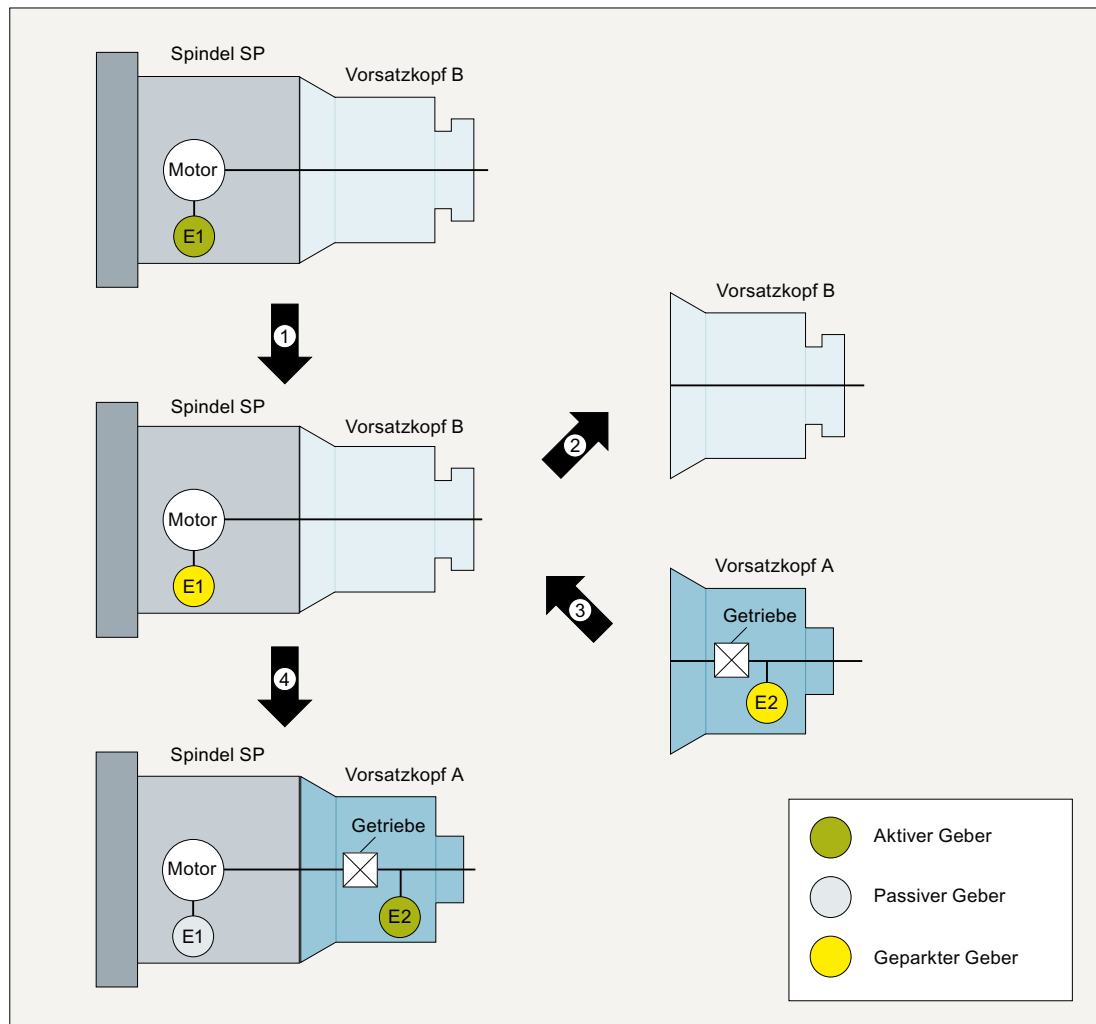
- ① Vor dem Wechsel eines Vorsatzkopfs muss der Anwender über die Funktion "Parken einer Maschinenachse (Seite 122)" alle Lagemesssysteme der Maschinenachse ausschalten:
 DB31, ... DBX1.5 (Lagemesssystem 1) = 0
 DB31, ... DBX1.6 (Lagemesssystem 2) = 0
 Die Steuerung setzt daraufhin die Statussignale für die Lagemesssysteme zurück:
 DB31, ... DBX102.5 (Lagemesssystem 1 eingeschaltet) == 0
 DB31, ... DBX102.6 (Lagemesssystem 2 eingeschaltet) == 0
- ② Der Anwender wartet die Statussignale ab und entfernt erst dann den Vorsatzkopf "A" von der Spindel. Dabei wird auch die Geberleitung zwischen Vorsatzkopf "A" und Kupplung elektrisch getrennt. Das Fehlen von Geber E2 erzeugt keine NC- oder Antriebsfehler.

- ③ Jetzt wird der Vorsatzkopf "B" auf der Spindel montiert.
- ④ Der Anwender schaltet nur das Lagemesssystem 1 ein:
DB31, ... DBX1.5 (Lagemesssystem 1) = 1
Die Steuerung setzt daraufhin das Statussignal:
DB31, ... DBX102.5 (Lagemesssystem 1 eingeschaltet) == 1
Da für Lagemesssystem 2 die Funktion "Parken des passiven Lagemesssystems" aktiv ist, wird das Lagemesssystem 2 nicht passiv, sondern bleibt im Zustand "Parken".

Ziel

Als Nächstes möchte der Anwender wieder den Vorsatzkopf "A" einwechseln.

Durchführung



- ① Der Anwender schaltet über die Funktion "Parken einer Maschinenachse" Lagemesssystem 1 aus:
 $DB31, \dots DBX1.5 \text{ (Lagemesssystem 1)} = 0$
 Die Steuerung setzt daraufhin das Statussignal für das Lagemesssystem zurück:
 $DB31, \dots DBX102.5 \text{ (Lagemesssystem 1 eingeschaltet)} == 0$
- ② Der Anwender wartet das Statussignal ab und entfernt erst dann den Vorsatzkopf "B" von der Spindel.
- ③ Jetzt wird der Vorsatzkopf "A" auf der Spindel montiert.
- ④ Der Anwender schaltet Lagemesssystem 2 aktiv:
 $DB31, \dots DBX1.6 \text{ (Lagemesssystem 2)} = 1$
 Damit wird gleichzeitig auch Lagemesssystem 1 eingeschaltet, da für Lagemesssystem 1 (Motormesssystem!) die Funktion "Parken des passiven Lagemesssystems" nicht aktiv ist. Lagemesssystem 1 wird zum passiven Lagemesssystem.
 Die Steuerung setzt die Statussignale für die Lagemesssysteme:
 $DB31, \dots DBX102.5 \text{ (Lagemesssystem 1 eingeschaltet)} == 1$
 $DB31, \dots DBX102.6 \text{ (Lagemesssystem 2 eingeschaltet)} == 1$

3.9.4 Beispiel: Vorsatzkopfwechsel bei zwei direkten Lagemesssystemen

Ausgangssituation

- Der Vorsatzkopf "A" hat einen Geber E3.
- Der Vorsatzkopf "B" hat keinen Geber.
- Die Spindel "SP" hat zwei Geber E1 und E2.
- In MD13060 \$MN_DRIVE_TELEGRAM_TYPE (Standard-Telegramm-Typ für PROFIdrive) ist einer der beiden folgenden Telegrammtypen projektiert:
 - Telegramm 118 (zwei externe Geber)
oder
 - Telegramm 138 (zwei externe Geber, zzgl. Momentenvorsteuerung)

Literatur:

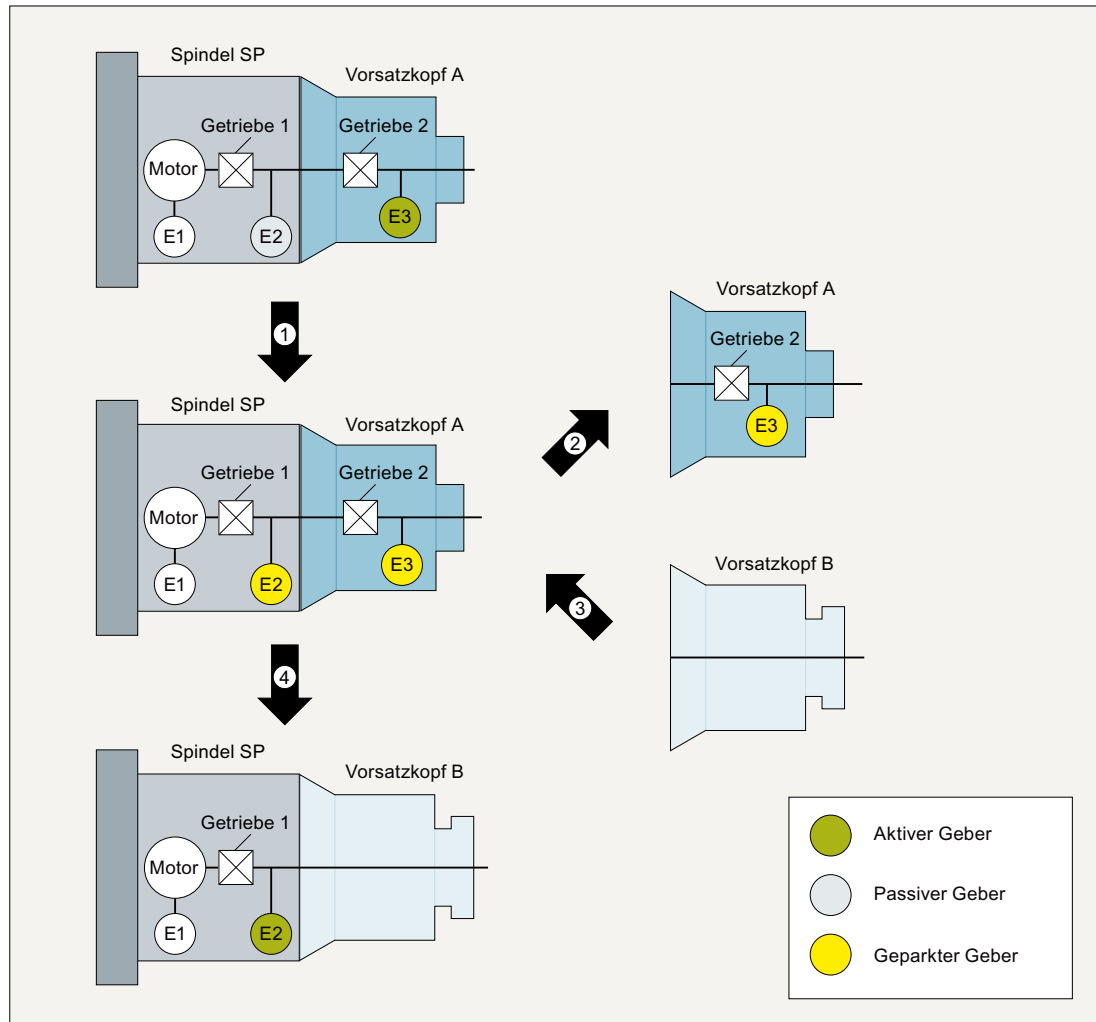
Informationen bezüglich Geberzuordnung siehe:
Inbetriebnahmehandbuch Inbetriebnahme CNC: NC, PLC, Antrieb;
Kapitel: "Kommunikation zwischen NC und Antrieb" > "Antriebe: Achse zuordnen"

- In der Spindel "SP" sind folgende Lagemesssysteme projektiert:
 - Direkter Geber E2 als Lagemesssystem 1
 - Direkter Geber E3 als Lagemesssystem 2
- Aktuell ist Vorsatzkopf "A" mit Geber E3 auf der Spindel montiert.
- Lagemesssystem 2 ist das aktive Messsystem:
DB31, ... DBX1.6 = 1
Lagemesssystem 1 ist passiv.
- Die Funktion "Parken des passiven Lagemesssystems" ist:
 - nicht aktiv für Lagemesssystem 1:
MD31046 \$MA_ENC_PASSIVE_PARKING[0] = 0
 - aktiv für Lagemesssystem 2:
MD31046 \$MA_ENC_PASSIVE_PARKING[1] = 1

Ziel

Der Anwender möchte vom Vorsatzkopf "A" auf Vorsatzkopf "B" wechseln.

Durchführung



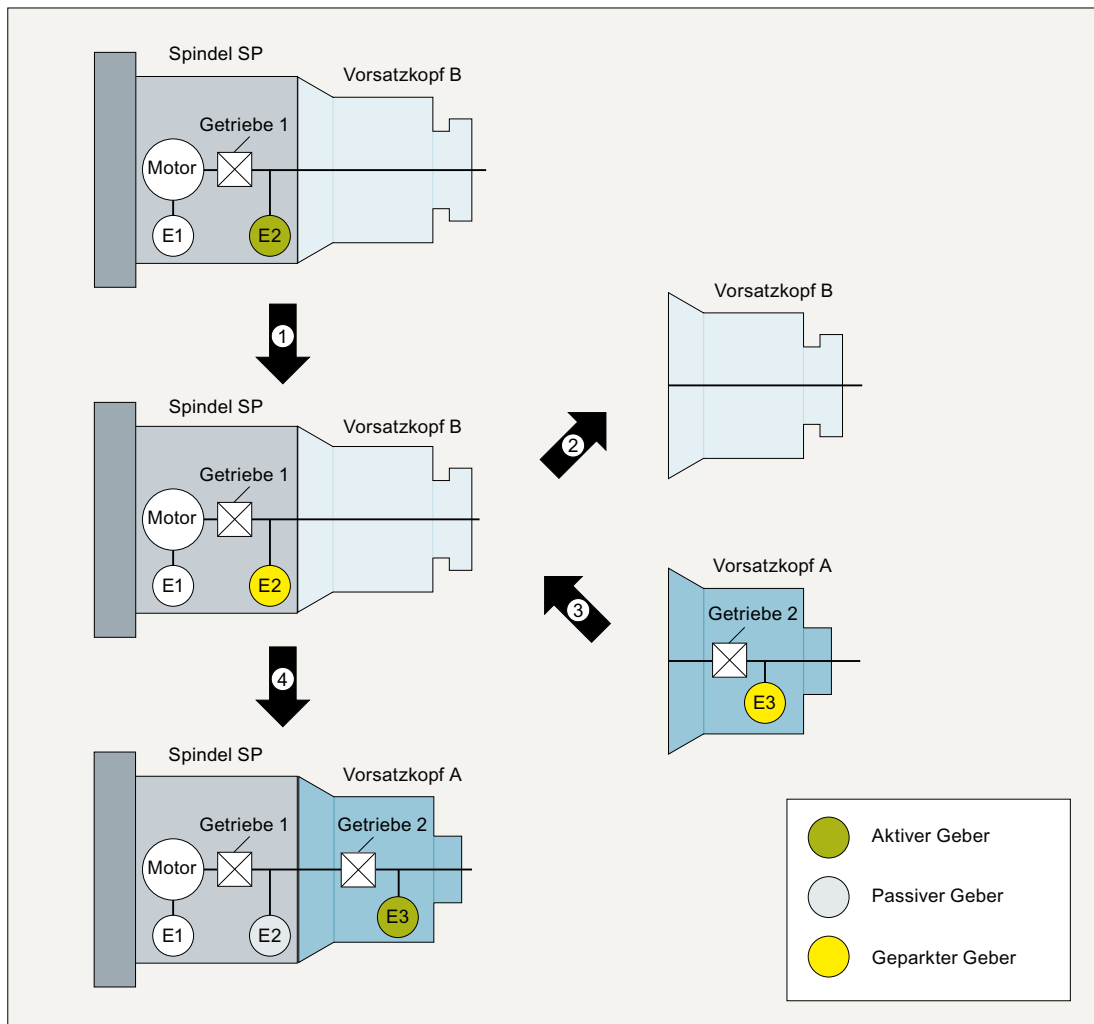
- ① Vor dem Wechsel eines Vorsatzkopfs muss der Anwender über die Funktion "Parken einer Maschinenachse (Seite 122)" alle Lagemesssysteme der Maschinenachse ausschalten:
 DB31, ... DBX1.5 (Lagemesssystem 1) = 0
 DB31, ... DBX1.6 (Lagemesssystem 2) = 0
 Die Steuerung setzt daraufhin die Statussignale für die Lagemesssysteme zurück:
 DB31, ... DBX102.5 (Lagemesssystem 1 eingeschaltet) == 0
 DB31, ... DBX102.6 (Lagemesssystem 2 eingeschaltet) == 0
- ② Der Anwender wartet die Statussignale ab und entfernt erst dann den Vorsatzkopf "A" von der Spindel. Dabei wird auch die Geberleitung zwischen Vorsatzkopf "A" und Kupplung elektrisch getrennt. Das Fehlen von Geber E3 erzeugt keine NC- oder Antriebsfehler.

- ③ Jetzt wird der Vorsatzkopf "B" auf der Spindel montiert.
- ④ Der Anwender schaltet nur das Lagemesssystem 1 ein:
DB31, ... DBX1.5 (Lagemesssystem 1) = 1
Die Steuerung setzt das Statussignal:
DB31, ... DBX102.5 (Lagemesssystem 1 eingeschaltet) == 1
Da für Lagemesssystem 2 die Funktion "Parken des passiven Lagemesssystems" aktiv ist, wird das Lagemesssystem 2 nicht passiv, sondern bleibt im Zustand "Parken".

Ziel

Als Nächstes möchte der Anwender wieder den Vorsatzkopf "A" einwechseln.

Durchführung



- ① Der Anwender schaltet über die Funktion "Parken einer Maschinenachse" Lagemesssystem 1 aus:
 $DB31, \dots DBX1.5 \text{ (Lagemesssystem 1)} = 0$
 Die Steuerung setzt daraufhin das Statussignal für das Lagemesssystem zurück:
 $DB31, \dots DBX102.5 \text{ (Lagemesssystem 1 eingeschaltet)} == 0$
- ② Der Anwender wartet das Statussignal ab und entfernt erst dann den Vorsatzkopf "B" von der Spindel.
- ③ Jetzt wird der Vorsatzkopf "A" auf der Spindel montiert.
- ④ Der Anwender schaltet Lagemesssystem 2 aktiv:
 $DB31, \dots DBX1.6 \text{ (Lagemesssystem 2)} = 1$
 Damit wird gleichzeitig auch Lagemesssystem 1 eingeschaltet, da für Lagemesssystem 1 die Funktion "Parken des passiven Lagemesssystems" nicht aktiv ist. Lagemesssystem 1 wird zum passiven Lagemesssystem.
 Die Steuerung setzt die Statussignale für die Lagemesssysteme:
 $DB31, \dots DBX102.5 \text{ (Lagemesssystem 1 eingeschaltet)} == 1$
 $DB31, \dots DBX102.6 \text{ (Lagemesssystem 2 eingeschaltet)} == 1$

3.9.5 Beispiel: Messsystemumschaltung bei bereichsweise fehlenden Gebern

Im folgenden Beispiel ist das direkte lineare Lagemesssystem nur in den Bearbeitungszonen vorhanden, während in den Bereichen außerhalb der Bearbeitungszonen nur das Motormesssystem zur Verfügung steht.

Ausgangssituation

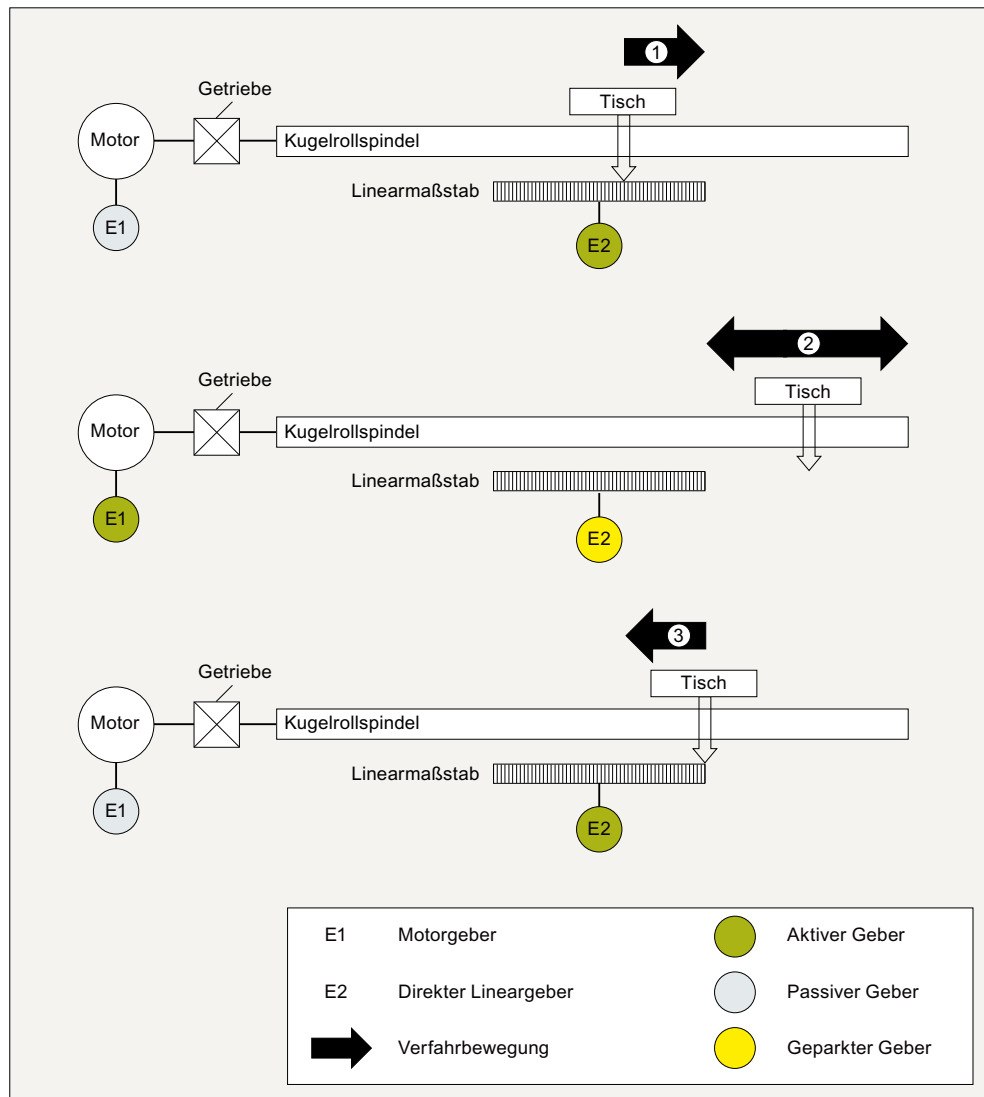
- Die Linearachse "X" hat zwei inkrementelle Geber:
 - Motorgeber E1
 - Direkter Lineargeber E2
- Der direkte Lineargeber E2 ist nur im Bearbeitungsbereich vorhanden.
- In MD13060 \$MN_DRIVE_TELEGRAM_TYPE (Standard-Telegramm-Typ für PROFIdrive) ist einer der beiden folgenden Telegrammtypen projektiert:
 - Telegramm 116 (Motorgeber + ein externer Geber)
oder
 - Telegramm 136 (Motorgeber + ein externer Geber, zzgl. Momentenvorsteuerung)
- Für die Maschinenachse sind folgende Lagemesssysteme projektiert:
 - Motorgeber E1 als Lagemesssystem 1
 - Direkter Lineargeber E2 als Lagemesssystem 2
- Beim Einschalten der Maschine schaltet der Anwender beide Lagemesssysteme ein:
 - DB31, ... DBX1.5 (Lagemesssystem 1) = 1
 - DB31, ... DBX1.6 (Lagemesssystem 2) = 1

Bei gleichzeitiger Aktivierung beider Lagemesssysteme wird von der Steuerung Lagemesssystem 1 als aktives Lagemesssystem ausgewählt.
- Zur Bearbeitung wird auf das Lagemesssystem 2 umgeschaltet:
 - DB31, ... DBX1.5 (Lagemesssystem 1) = 0
 - DB31, ... DBX1.6 (Lagemesssystem 2) = 1
- Die Funktion "Parken des passiven Lagemesssystems" ist:
 - nicht aktiv für Lagemesssystem 1:
MD31046 \$MA_ENC_PASSIVE_PARKING[0] = 0
 - aktiv für Lagemesssystem 2:
MD31046 \$MA_ENC_PASSIVE_PARKING[1] = 1
- Die Übernahme der Position und des "Referenziert"-Status für Lagemesssystem 2 ist eingeschaltet:
MD34210 \$MA_ENC_REFP_STATE[1] = 2

Ziel

Beim Durchfahren des Bereichs außerhalb des linearen Lagemesssystems E2 sollen die fehlenden Gebersignale keine Fehler im Antrieb und in der Steuerung verursachen.

Durchführung



- ③ Wenn der Tisch in den Bereich des linearen Lagemesssystems zurückkehrt, schaltet der Anwender im Stillstand vom Motormesssystem auf das lineare Lagemesssystem um:
- DB31, ... DBX1.5 (Lagemesssystem 1) = 0
- DB31, ... DBX1.6 (Lagemesssystem 2) = 1
- Die Steuerung setzt das Statussignal:
- DB31, ... DBX102.6 (Lagemesssystem 2 eingeschaltet) == 1
- Das Motormesssystem wird zum passiven Lagemesssystem.

Ergebnis

Beide Lagemesssysteme sind referenziert. Die Position des linearen Lagemesssystems entspricht der Position des Motormesssystems zum Umschaltzeitpunkt. Reicht die Positionsgenauigkeit nicht aus, ist das lineare Lagemesssystem neu zu referenzieren.

3.10 Randbedingungen

Weitere Maschinendaten

Neben den Maschinendaten der Achsüberwachung sind folgende weitere Maschinendaten einzustellen bzw. zu prüfen:

Alle Maschinenachsen

- MD31030 \$MA_LEADSCREW_PITCH (Steigung der Kugelrollspindel)
- MD31050 \$MA_DRIVE_AX_RATIO_DENOM (Nenner Lastgetriebe)
- MD31060 \$MA_DRIVE_AX_RATIO_NUMERA (Zähler Lastgetriebe)
- MD31070 \$MA_DRIVE_ENC_RATIO_DENOM (Nenner Messgetriebe)
- MD31080 \$MA_DRIVE_ENC_RATIO_NUMERA (Zähler Messgetriebe)
- MD32810 \$MA_EQUIV_SPEEDCTRL_TIME (Ersatzzeitkonstante Drehzahlregelkreis für Vorsteuerung)
- Geberauflösung: siehe Kapitel "G2: Geschwindigkeiten, Soll-/Istwertsysteme, Regelung (Seite 341)".

Maschinenachsen mit analoger Drehzahlsollwertschnittstelle

- MD32260 \$MA_RATED_VELO (Nenn-Motordrehzahl)
- MD32250 \$MA_RATED_OUTVAL (Nenn-Ausgangsspannung)

3.11 Datenlisten

3.11.1 Maschinendaten

3.11.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10604	WALIM_GEOAX_CHANGE_MODE	Arbeitsfeldbegrenzung beim Umschalten von Geometrieachsen
10710	PROG_SD_RESET_SAVE_TAB	Zu aktualisierende Settingdaten

3.11.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20150	GCODE_RESET_VALUES	Löschstellung der G-Gruppen
21020	WORKAREA_WITH_TOOL_RADIUS	Berücksichtigung des Werkzeugradius bei Arbeitsfeldbegrenzung
24130	TRAFO_INCLUDES_TOOL_1	Werkzeugbehandlung bei aktiver 1. Transformation
24230	TRAFO_INCLUDES_TOOL_2	Werkzeugbehandlung bei aktiver 2. Transformation
24330	TRAFO_INCLUDES_TOOL_3	Werkzeugbehandlung bei aktiver 3. Transformation
24426	TRAFO_INCLUDES_TOOL_4	Werkzeugbehandlung bei aktiver 4. Transformation
24436	TRAFO_INCLUDES_TOOL_5	Werkzeugbehandlung bei aktiver 5. Transformation
24446	TRAFO_INCLUDES_TOOL_6	Werkzeugbehandlung bei aktiver 6. Transformation
24456	TRAFO_INCLUDES_TOOL_7	Werkzeugbehandlung bei aktiver 7. Transformation
24466	TRAFO_INCLUDES_TOOL_8	Werkzeugbehandlung bei aktiver 8. Transformation
24476	TRAFO_INCLUDES_TOOL_9	Werkzeugbehandlung bei aktiver 9. Transformation
24486	TRAFO_INCLUDES_TOOL_10	Werkzeugbehandlung bei aktiver 10. Transformation
25106	TRAFO_INCLUDES_TOOL_11	Werkzeugbehandlung bei aktiver 11. Transformation
25116	TRAFO_INCLUDES_TOOL_12	Werkzeugbehandlung bei aktiver 12. Transformation
25126	TRAFO_INCLUDES_TOOL_13	Werkzeugbehandlung bei aktiver 13. Transformation
25136	TRAFO_INCLUDES_TOOL_14	Werkzeugbehandlung bei aktiver 14. Transformation
25146	TRAFO_INCLUDES_TOOL_15	Werkzeugbehandlung bei aktiver 15. Transformation
25156	TRAFO_INCLUDES_TOOL_16	Werkzeugbehandlung bei aktiver 16. Transformation
25166	TRAFO_INCLUDES_TOOL_17	Werkzeugbehandlung bei aktiver 17. Transformation
25176	TRAFO_INCLUDES_TOOL_18	Werkzeugbehandlung bei aktiver 18. Transformation
25186	TRAFO_INCLUDES_TOOL_19	Werkzeugbehandlung bei aktiver 19. Transformation
25196	TRAFO_INCLUDES_TOOL_20	Werkzeugbehandlung bei aktiver 20. Transformation
28600	MM_NUM_WORKAREA_CS_GROUPS	Anzahl an Koordinatensystem-spezifischen Arbeitsfeldbegrenzungen

3.11.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30200	NUM_ENCS	Anzahl der Geber
30240	ENC_TYPE	Geber-Typ der Istwerterfassung (Lageistwert)
30310	ROT_IS_MODULO	Modulo-Wandlung für Rundachse/Spindel
30800	WORK_AREA_CHECK_TYPE	Art der Prüfung der Arbeitsfeldgrenzen
31046	ENC_PASSIVE_PARKING	Parken des passiven Lagemesssystems
32200	POSCTRL_GAIN[n]	K _v -Faktor
32250	RATED_OUTVAL	Nennausgangsspannung
32260	RATED_VELO	Motorenndrehzahl
32300	MAX_AX_ACCEL	Maximale Achsbeschleunigung
32800	EQUIV_CURRCTRL_TIME	Ersatzzeitkonstante Stromregelkreis für Vorsteuerung
32810	EQUIV_SPEEDCTRL_TIME	Ersatzzeitkonstante Drehzahlregelkreis für Vorsteuerung
32910	DYN_MATCH_TIME[n]	Zeitkonstante der Dynamikanpassung
34210	ENC_REFP_STATE	Geberstatus
35160	SPIND_EXTERN_VELO_LIMIT	Spindeldrehzahlbegrenzung von PLCC
36000	STOP_LIMIT_COARSE	Genauhalt grob
36010	STOP_LIMIT_FINE	Genauhalt fein
36020	POSITIONING_TIME	Verzögerungszeit Genauhalt fein
36030	STANDSTILL_POS_TOL	Stillstandstoleranz
36040	STANDSTILL_DELAY_TIME	Verzögerungszeit Stillstandsüberwachung
36050	CLAMP_POS_TOL	Klemmungstoleranz bei NST "Klemmung aktiv"
36052	STOP_ON_CLAMPING	Sonderfunktionen bei geklemmter Achse
36060	STANDSTILL_VELO_TOL	Maximale Geschwindigkeit/Drehzahl "Achse/Spindel steht"
36100	POS_LIMIT_MINUS	1. Software-Endschalter minus
36110	POS_LIMIT_PLUS	1. Software-Endschalter plus
36120	POS_LIMIT_MINUS2	2. Software-Endschalter minus
36130	POS_LIMIT_PLUS2	2. Software-Endschalter plus
36200	AX_VELO_LIMIT	Schwellwert für Geschwindigkeitsüberwachung
36210	CTRLOUT_LIMIT	Maximaler Drehzahlsollwert
36220	CTRLOUT_LIMIT_TIME	Verzögerungszeit für Drehzahlsollwertüberwachung
36300	ENC_FREQ_LIMIT	Gebergrenzfrequenz
36302	ENC_FREQ_LIMIT_LOW	Gebergrenzfrequenz für Geber-Neusynchronisation
36310	ENC_ZERO_MONITORING	Nullmarkenüberwachung
36312	ENC_ABS_ZEROMON_WARNING	Nullmarkenüberwachung Warnschwelle
36400	CONTOUR_TOL	Toleranzband Konturüberwachung
36500	ENC_CHANGE_TOL	Maximale Toleranz bei Lageistwertumschaltung
36510	ENC_DIFF_TOL	Toleranz Messsystem-Gleichlauf
36600	BRAKE_MODE_CHOICE	Bremsverhalten bei Hardware-Endschalter
36610	AX_EMERGENCY_STOP_TIME	Maximale Zeitdauer der Bremsrampe bei Fehlern
36620	SERVO_DISABLE_DELAY_TIME	Abschaltverzögerung Reglerfreigabe

3.11.2 Settingdaten

3.11.2.1 Achs-/Spindel-spezifische Setindaten

Nummer	Bezeichner: \$SA_	Beschreibung
43400	WORKAREA_PLUS_ENABLE	Arbeitsfeldbegrenzung in positiver Richtung aktiv
43410	WORKAREA_MINUS_ENABLE	Arbeitsfeldbegrenzung in negativer Richtung aktiv
43420	WORKAREA_LIMIT_PLUS	Arbeitsfeldbegrenzung plus
43430	WORKAREA_LIMIT_MINUS	Arbeitsfeldbegrenzung minus

3.11.3 Signale

3.11.3.1 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Nachfuhrbetrieb	DB31,DBX1.4	DB380x.DBX1.4
Lagemesssystem 1/2	DB31,DBX1.5/6	DB380x.DBX1.5/6
Reglerfreigabe	DB31,DBX2.1	DB380x.DBX2.1
Klemmvorgang läuft	DB31,DBX2.3	DB380x.DBX2.3
Geschwindigkeits-/Spindeldrehzahlbegrenzung	DB31,DBX3.6	DB380x.DBX3.6
Vorschub-Halt / Spindel-Halt	DB31,DBX4.3	DB380x.DBX4.3
Hardware-Endschalter minus / Hardware-Endschalter plus	DB31,DBX12.0/1	DB380x.DBX1000.0/1
Software-Endschalter minus / 2. Software-Endschalter plus	DB31,DBX12.2/3	DB380x.DBX1000.2/3

3.11.3.2 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Gebergrenzfrequenz überschritten 1/2	DB31,DBX60.2/3	DB390x.DBX0.2
Referenziert/Synchronisiert 1/2	DB31,DBX60.4/5	DB390x.DBX0.4/5
Fahrbefehl minus/plus	DB31,DBX64.6/7	DB390x.DBX4.6/7
Lagemesssystem 1/2 eingeschaltet	DB31,DBX102.5/6	DB390x.DBX5006.5/6

A5: Schutzbereiche

4.1 Funktion

Schutzbereiche sind vom Anwender zu definierende statische oder bewegliche 2- bzw. 3-dimensionale Bereiche innerhalb einer Maschine zum Schutz von Maschinenelementen vor Kollisionen.

Folgende Elemente können geschützt werden:

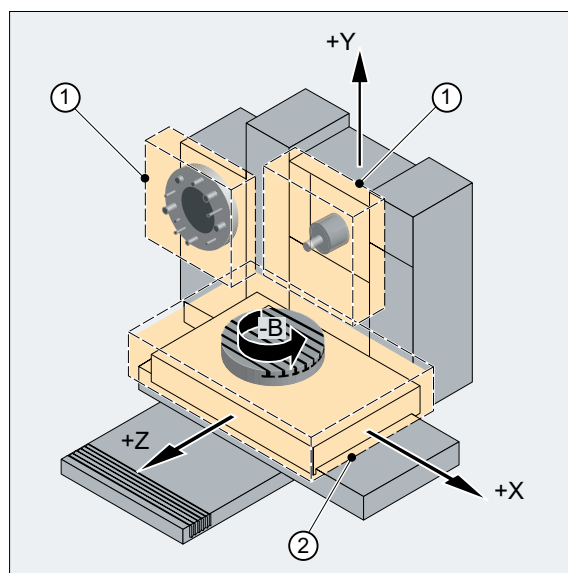
- Feststehende Maschinenelemente (z. B. Werkzeugmagazin, einschwenkbarer Messtaster)
- Bewegliche Maschinenelemente, die zum Werkzeug gehören (z. B. Werkzeug, Werkzeugträger)
- Bewegliche Maschinenelemente, die zum Werkstück gehören (z. B. Teile des Werkstücks, Aufspanntisch, Spannpratzen, Spindelfutter, Reitstock)

Damit der Schutz eines Maschinenelements gewährleistet ist, muss der Schutzbereich so definiert werden, dass er das zu schützende Element vollständig umschließt.

Die Überwachung der Schutzbereiche erfolgt kanalbezogen, d. h., es werden alle aktiven Schutzbereiche eines Kanals gegenseitig auf Kollision überwacht.

Beim automatischen Abarbeiten von Teileprogrammen in der Betriebsart AUTOMATIK bzw. MDA prüft die NC am Anfang jedes Teileprogrammsatzes, ob es beim Verfahren der programmierten Bahn zu Kollisionen von Schutzbereichen kommen würde.

Nach einer manuellen Deaktivierung eines aktiven Schutzbereichs kann in diesen gefahren werden. Nach Verlassen des Schutzbereichs wird der Schutzbereich automatisch wieder aktiv.



- ① Werkzeugbezogene Schutzbereiche
- ② Werkstückbezogener Schutzbereich

Bild 4-1 Beispiel für Schutzbereiche an einer Fräsmaschine

Definieren von Schutzbereichen

Ein Schutzbereich kann 2- oder 3-dimensional aus Polygonzügen mit maximal 10 Eckpunkten und Kreisbögen als Konturelemente definiert werden. Die Definition kann über Befehle im Teileprogramm (siehe "Schutzbereiche definieren (CROTDEF, NROTDEF) (Seite 150)") oder über Systemvariable erfolgen. Die Konturelemente liegen dabei alle in der mit G17, G18 oder G19 wählbaren Ebene.

3. Dimension

Die Ausdehnung des Schutzbereiches in der 3. Dimension kann zwischen $-\infty$ bis $+\infty$ begrenzt werden:

- $-\infty$ bis $+\infty$
- $-\infty$ bis oberer Grenzwert
- Unterer Grenzwert bis $+\infty$
- Unterer Grenzwert bis oberer Grenzwert

Koordinatensystem

Die Definition eines Schutzbereichs erfolgt bezogen auf die Geometrieachsen eines Kanals und daher im Basiskoordinatensystem (BKS).

Bezug

- Werkzeugbezogene Schutzbereiche
Koordinaten für **werkzeugbezogene** Schutzbereiche sind absolut, bezogen auf den **Werkzeugträgerbezugspunkt F**, anzugeben.
- Werkstückbezogene Schutzbereiche
Koordinaten für **werkstückbezogene** Schutzbereiche sind absolut, bezogen auf den Nullpunkt des **Basiskoordinatensystems (BKS)**, anzugeben.

Hinweis

Ist kein werkzeugbezogener Schutzbereich aktiv, wird die Werkzeugbahn gegen die Werkstück-bezogenen Schutzbereiche geprüft.

Ist kein werkstückbezogener Schutzbereich aktiv, findet keine Schutzbereichsüberwachung statt.

Orientierung

Die Orientierung der Schutzbereiche wird bestimmt durch die Festlegung der Ebene (Abszisse/ Ordinate), in der die Konturbeschreibung erfolgt und die senkrecht auf der Kontur des Schutzbereiches stehende Achse (Applikate).

Die Orientierung der Schutzbereiche muss für werkzeug- und werkstückbezogene Schutzbereiche gleich sein.

Maschinen- / Kanalspezifische Schutzbereiche

- Maschinenspezifische Schutzbereiche
Daten für maschinenspezifische Schutzbereiche sind einmal in der Steuerung definiert. Diese Schutzbereiche können von allen Kanälen aktiviert werden.
- Kanalspezifische Schutzbereiche
Daten für kanalspezifische Schutzbereiche sind in einem Kanal definiert. Diese Schutzbereiche können nur von diesem Kanal aktiviert werden.

Systemvariable

Bei der Definition der Schutzbereiche über Befehle im Teileprogramm werden die Schutzbereichsdaten in Systemvariablen abgelegt. Die Systemvariablen können auch direkt geschrieben werden, d. h. die Definition von Schutzbereichen kann auch direkt in den Systemvariablen vorgenommen werden. Für die Beschreibung der Kontur eines Schutzbereichs gelten dabei die gleichen Randbedingungen wie bei der Schutzbereichsdefinition über Befehle im Teileprogramm (siehe "Schutzbereiche definieren (CPROTDEF, NPROTDEF) (Seite 150)").

Die Schutzbereichsdefinitionen umfassen folgende Systemvariablen:

Systemvariable	Typ	Bedeutung
\$SN_PA_ACTIV_IMMED[<n>] \$SC_PA_ACTIV_IMMED[<n>]	BOOL	Aktivierungstyp Der Schutzbereich ist nach dem Hochlaufen der Steuerung und dem Referenzieren der Achsen sofort aktiv/nicht aktiv. FALSE nicht sofort aktiv TRUE sofort aktiv
\$SN_PA_T_W[<n>] \$SC_PA_T_W[<n>]	INT	Schutzbereichstyp 0 Werkstückbezogener Schutzbereich 1 reserviert 2 reserviert 3 Werkzeugbezogener Schutzbereich
\$SN_PA_ORI[<n>] \$SC_PA_ORI[<n>]	INT	Orientierung des Schutzbereichs, d. h. Polygonzug in der Ebene aus: 0 1. und 2. Geometrieachse 1 3. und 1. Geometrieachse 2 2. und 3. Geometrieachse
\$SN_PA_LIM_3DIM[<n>] \$SC_PA_LIM_3DIM[<n>]	INT	Art der Begrenzung in der 3. Dimension 0 keine Begrenzung 1 Begrenzung in Plus-Richtung 2 Begrenzung in Minus-Richtung 3 Begrenzung in Plus- und Minus-Richtung
\$SN_PA_PLUS_LIM[<n>] \$SC_PA_PLUS_LIM[<n>]	REAL	Wert der Begrenzung in Plus-Richtung in der 3. Dimension
\$SN_PA_MINUS_LIM[<n>] \$SC_PA_MINUS_LIM[<n>]	REAL	Wert der Begrenzung in Minus-Richtung in der 3. Dimension
\$SN_PA_CONT_NUM[<n>] \$SC_PA_CONT_NUM[<n>]	INT	Anzahl der gültigen Konturelemente

Systemvariable	Typ	Bedeutung
\$SN_PA_CONT_TYP[<n>, <i>] \$SC_PA_CONT_TYP[<n>, <i>]	INT	Konturtyp[<i>], Konturtyp (G1, G2, G3) des <i>-ten Konturelements
\$SN_PA_CONT_ABS[<n>, <i>] \$SC_PA_CONT_ABS[<n>, <i>]	REAL	Endpunkt der Kontur[<i>], Abszissenwert
\$SN_PA_CONT_ORD[<n>, <i>] \$SC_PA_CONT_ORD[<n>, <i>]	REAL	Endpunkt der Kontur[<i>], Ordinatenwert
\$SN_PA_CENT_ABS[<n>, <i>] \$SC_PA_CENT_ABS[<n>, <i>]	REAL	Mittelpunkt der Kreiskontur[<i>], absoluter Abszissenwert
\$SN_PA_CENT_ORD[<n>, <i>] \$SC_PA_CENT_ORD[<n>, <i>]	REAL	Mittelpunkt der Kreiskontur[<i>], absoluter Ordinatenwert
\$SN_... sind Systemvariable für NC- bzw. maschinenspezifische Schutzbereiche. \$SC_... sind Systemvariable für kanalspezifische Schutzbereiche. Der Index "<n>" entspricht der Nummer des Schutzbereichs: 0 = 1. Schutzbereich Der Index "<i>" entspricht der Nummer des Konturelements: 0 = 1. Konturelement Die Konturelemente sind in aufsteigender Reihenfolge zu definieren.		

Hinweis

Die Systemvariablen der Schutzbereichsdefinitionen werden bei REORG nicht restauriert!

Daten der Schutzbereichsdefinitionen**Datenablage**

Die Schutzbereichsdefinitionen werden in den folgenden Dateien abgelegt:

Datei	Bausteine
_N_NCK_PRO	Datenbaustein für maschinenspezifische Schutzbereiche
_N_CHAN1_PRO	Datenbaustein für kanalspezifische Schutzbereiche im Kanal 1
_N_CHAN2_PRO	Datenbaustein für kanalspezifische Schutzbereiche im Kanal 2

Datensicherung

Die Schutzbereichsdefinitionen werden in den folgenden Dateien gesichert:

Datei	Bausteine
_N_INITIAL_INI	Alle Datenbausteine der Schutzbereiche
_N_COMPLETE_PRO	Alle Datenbausteine der Schutzbereiche
_N_CHAN_PRO	Alle Datenbausteine der kanalspezifischen Schutzbereiche

Aktivieren, Voraktivieren und Deaktivieren von Schutzbereichen**Aktivierungsstatus**

Der Aktivierungsstatus eines Schutzbereichs kann folgende Werte annehmen:

- Aktiviert
- Voraktiviert

- Voraktiviert mit bedingtem Stopp
- Deaktiviert

Der Aktivierungsstatus ist immer kanalspezifisch, auch bei maschinenspezifischen Schutzbereichen!

Aktivieren, Voraktivieren und Deaktivieren im Teileprogramm

Der Aktivierungsstatus eines Schutzbereichs kann über Befehle im Teileprogramm jederzeit verändert werden (siehe "Schutzbereiche aktivieren/deaktivieren (CPROT, NPROT) (Seite 154)").

Hinweis

Ein Schutzbereich wird erst nach dem Referenzieren aller Geometrieachsen des Kanals, in dem er aktiviert wurde, berücksichtigt.

Schutzbereiche, die zu einem späteren Zeitpunkt vom PLC-Anwenderprogramm aus aktiviert werden sollen (siehe unten), müssen im Teileprogramm voraktiviert werden.

Voraktivierte Schutzbereiche werden über folgende NC/PLC-Nahtstellensignale angezeigt:

- DB21, ... DBX272.0 - 273.1 (Maschinenspezifischer Schutzbereich 1 - 10 voraktiviert) == 1
- DB21, ... DBX274.0 - 275.1 (Kanalspezifischer Schutzbereich 1 - 10 voraktiviert) == 1

Voraktivieren mit bedingtem Stopp

ACHTUNG

Schutzraumverletzung möglich

Wird ein mit bedingtem Stopp voraktivierter Schutzraum nicht rechtzeitig aktiviert, kann die NC aufgrund des bis zum Aktivierungszeitpunkt nicht berücksichtigten Bremswegs unter Umständen nicht mehr rechtzeitig vor dem Schutzraum anhalten.

Bei einem voraktivierten Schutzbereich mit bedingtem Stopp wird eine Verfahrbewegung **nicht** vor diesem angehalten, wenn die Verfahrbewegung in den Schutzraum hinein führt. Ein Stopp erfolgt nur, wenn der Schutzbereich aktiviert wurde. Dieses Verhalten soll eine vom Anwender steuerbare unterbrechungsfreie Bearbeitung für den Fall ermöglichen, dass der Schutzbereich nur zeitweise benötigt wird.

Aktivieren über NC/PLC-Nahtstellensignale

Nur Schutzbereiche, die über das Teileprogramm **voraktiviert** wurden, können im PLC-Anwenderprogramm über die NC/PLC-Nahtstellensignale aktiviert werden:

- DB21, ... DBX8.0 - 9.1 (Maschinenspezifischen Schutzbereich 1 - 10 aktivieren) = 1
- DB21, ... DBX10.0 - 11.1 (Kanalspezifischen Schutzbereich 1 - 10 aktivieren) = 1

Die Aktivierung voraktivierter Schutzbereiche muss vor der Verfahrbewegung der Geometrieachsen erfolgen! Erfolgt die Aktivierung während der Verfahrbewegung, werden diese Schutzbereiche für die aktuelle Verfahrbewegung nicht mehr berücksichtigt. Reaktion:

- Alarm 10704 "Schutzbereichsüberwachung ist nicht gewährleistet"
- DB21, ... DBX39.0 (Schutzbereichsüberwachung nicht gewährleistet) = 1

Hinweis

Die Aktivierung voraktivierter Schutzbereiche muss vor der Verfahrbewegung der Geometrieachsen erfolgen!

Deaktivieren über NC/PLC-Nahtstellensignale

Nur Schutzbereiche, die über ein Teileprogramm **voraktiviert** und über die NC/PLC-Nahtstellensignale aktiviert wurden, können über die NC/PLC-Nahtstellensignale wieder deaktiviert werden:

- DB21, ... DBX8.0 bis DBX9.1 (Maschinenspezifischen Schutzbereich 1 - 10 aktivieren) = 0
- DB21, ... DBX10.0 bis DBX11.1 (Kanalspezifischen Schutzbereich 1 - 10 aktivieren) = 0

Schutzbereiche, die über ein Teileprogramm direkt **aktiviert** wurden, können vom PLC-Anwenderprogramm aus **nicht** deaktiviert werden.

Automatisches Deaktivieren beim Transformationswechsel / Geometrieachstausch

In der Standardeinstellung werden aktive Schutzbereiche beim Wechsel einer Transformation oder beim Geometrieachstausch automatisch deaktiviert. Sollen aktive Schutzbereiche dagegen aktiv bleiben, muss das bitcodierte Maschinendatum MD10618 \$MN_PROTAREA_GEOAX_CHANGE_MODE entsprechend angepasst werden (siehe "Maschinendaten (Seite 149)").

Aktivierungsstatus in besonderen Systemzuständen

Satzsuchlauf mit Berechnung

Bei Satzsuchlauf mit Berechnung wird immer der zuletzt programmierte Aktivierungszustand eines Schutzbereichs berücksichtigt.

Programmtest

In den Betriebsarten AUTOMATIK und MDA werden aktivierte und voraktivierte Schutzbereiche auch im Zustand "Programmtest" überwacht.

NC-RESET und Programmende

Der Aktivierungsstatus eines Schutzbereichs bleibt über NC-RESET und Programmende hinaus erhalten.

Anzeige von Schutzbereichsverletzungen

Verletzungen von aktivierten Schutzbereichen oder mögliche Verletzungen von voraktivierten Schutzbereichen, falls diese aktiviert würden, werden über folgende NC/PLC-Nahtstellensignale angezeigt:

- DB21, ... DBX276.0 - 277.1 (Maschinenspezifischer Schutzbereich 1 - 10 verletzt) == 1
- DB21, ... DBX278.0 - 279.1 (Kanalspezifischer Schutzbereich 1 - 10 verletzt) == 1

Überprüfen auf Schutzbereichsverletzung

Ob ein vorgegebener Weg von den Geometrieachsen verfahren werden kann, ohne Schutzbereiche zu verletzen, kann mit der Funktion CALCPOSI überprüft werden (siehe "Überprüfung auf Schutzbereichsverletzung, Arbeitsfeldbegrenzung und Software-Endschalter (CALCPOSI) (Seite 157)").

4.2 Inbetriebnahme

4.2.1 Maschinendaten

Speicherbedarf

Der für Schutzbereiche benötigte Speicherbedarf wird über folgende Maschinendaten parametrisiert:

- Persistenten Speicher
 - MD18190 \$MN_MM_NUM_PROTECT_AREA_NCK (Anzahl der verfügbaren maschinenspezifischen Schutzbereiche)
 - MD28200 \$MC_MM_NUM_PROTECT_AREA_CHAN (Anzahl der verfügbaren kanalspezifischen Schutzbereiche)
- Dynamischer Speicher
 - MD28210 \$MC_MM_NUM_PROTECT_AREA_ACTIVE (Maximale Anzahl der gleichzeitig im Kanal aktivierbaren Schutzbereiche)
 - MD28212 \$MC_MM_NUM_PROTECT_AREA_CONTUR (Maximale Anzahl definierbarer Konturelemente im Kanal)

Verhalten beim Transformationswechsel / Geometrieachstausch

Mit dem folgenden Maschinendatum wird festgelegt, ob beim Wechsel einer Transformation oder beim Geometrieachstausch aktive Schutzbereiche erhalten bleiben oder deaktiviert werden sollen:

MD10618 \$MN_PROTAREA_GEOAX_CHANGE_MODE

Bit	Wert	Bedeutung
0	0	Die aktiven Schutzbereiche werden beim Transformationswechsel deaktiviert.
	1	Die aktiven Schutzbereiche bleiben beim Transformationswechsel aktiv.
1	0	Die aktiven Schutzbereiche werden beim Geometrieachstausch deaktiviert.
	1	Die aktiven Schutzbereiche bleiben beim Geometrieachstausch aktiv.

4.3 Programmierung

4.3.1 Schutzbereiche definieren (CROTDEF, NROTDEF)

Schutzbereiche, die Maschinenelemente vor Kollisionen schützen sollen, werden im Teileprogramm jeweils in Blöcken definiert. Diese enthalten folgende Elemente:

1. Festlegung der Arbeitsebene
Vor der eigentlichen Schutzbereichsdefinition muss die Arbeitsebene angewählt werden, auf die sich die Konturbeschreibung des Schutzbereichs beziehen soll.
2. Definitionsbeginn
Je nach NC-Befehl wird entweder ein Kanalspezifischer oder maschinenspezifischer Schutzbereich angelegt.
3. Konturbeschreibung des Schutzbereichs
Die Kontur eines Schutzbereichs wird mittels Verfahrbewegungen beschrieben. Diese werden nicht ausgeführt und haben keine Verbindung zu vorhergehenden oder nachfolgenden Geometriebeschreibungen. Sie definieren ausschließlich den Schutzbereich.
4. Definitionsende

Syntax

```
DEF INT <Var_Name>
G17/G18/G19
CROTDEF/NROTDEF(<n>,<t>,<AppLim>,<AppPlus>,<AppMinus>)
G0/G1/... X/Y/Z...
...
EXECUTE(<Var_Name>)
```

Bedeutung

DEF INT <Var_Name>:	Definition einer lokalen Hilfsvariable vom Datentyp INTEGER
<Var_Name>:	Name der Hilfsvariable

G17/G18/G19:	Arbeitsebene Hinweis: Die Arbeitsebene darf vor dem Definitionsende nicht geändert werden. Eine Programmierung der Applikate zwischen Definitionsbeginn und -ende ist nicht zulässig.	
CROTDEF():	Vordefinierte Prozedur zur Definition eines kanalspezifischen Schutzbereichs	
NROTDEF():	Vordefinierte Prozedur zur Definition eines maschinenspezifischen Schutzbereichs	
<n>:	Nummer des definierten Schutzbereichs	
	Datentyp:	INT
<t>:	Typ des Schutzbereichs	
	Datentyp:	BOOL
	Wert:	TRUE Werkzeugbezogener Schutzbereich
		FALSE Werkstückbezogener Schutzbereich
<AppLim>:	Art der Begrenzung in der 3. Dimension	
	Datentyp:	INT
	Wert:	0 keine Begrenzung
		1 Begrenzung in Plus-Richtung
		2 Begrenzung in Minus-Richtung
		3 Begrenzung in Plus- und Minus-Richtung
<AppPlus>:	Wert der Begrenzung in Plus-Richtung der 3. Dimension	
	Datentyp:	REAL
<AppMinus>:	Wert der Begrenzung in Minus-Richtung der 3. Dimension	
	Datentyp:	REAL

G0/G1/... X/Y/Z... :	<p>Die Kontur eines Schutzbereichs wird mit maximal 11 Verfahr- bewegungen in der angewählten Arbeitsebene beschrieben. Dabei ist die erste Verfahrbewegung die Bewegung an die Kon- tur. Der letzte Punkt der Konturbeschreibung muss immer mit dem ersten Punkt der Konturbeschreibung zusammenfallen.</p> <p>Als Schutzbereich gilt der Bereich links von der Kontur:</p> <ul style="list-style-type: none"> • Innenschutzbereich Die Kontur für einen Innenschutzbereich ist gegen den Uhrzeigersinn zu beschreiben. • Außenschutzbereich (nur zulässig für werkstückbezogene Schutzbereiche!) Die Kontur für einen Außenschutzbereich ist im Uhrzeigersinn zu beschreiben. <p>Folgende Konturelemente sind zulässig:</p> <ul style="list-style-type: none"> • G0, G1 für gerade Konturelemente • G2 für Kreisabschnitte im Uhrzeigersinn Nur zulässig bei werkstückbezogenen Schutzbereichen! Nicht zulässig bei werkzeugbezogenen Schutzbereichen, da sie nur konvex sein dürfen. • G3 für Kreisabschnitte gegen den Uhrzeigersinn <p>Hinweis: Ein Schutzbereich kann nicht durch einen Vollkreis beschrieben werden. Ein Vollkreis muss in zwei Teilkreise aufgeteilt werden.</p> <p>Hinweis: Die Folge G2 → G3 bzw. G3 → G2 ist nicht zulässig! Zwischen die beiden Kreissätze muss ein kurzer G1-Satz eingefügt wer- den.</p>
EXECUTE (<Var_Name>):	<p>Vordefinierte Prozedur, die das Ende der Definition markiert Mit EXECUTE wird zur normalen Programmbearbeitung zurück- geschaltet.</p>

Beispiel

Siehe Beispiel unter "Schutzbereiche aktivieren/deaktivieren (CPROT, NPROT) (Seite 154)".

Weitere Informationen

Maschinenspezifische Schutzbereiche

Ein maschinenspezifischer Schutzbereich bzw. dessen Kontur wird mittels der Geometrieachsen definiert, d. h. bezogen auf das Basiskoordinatensystem (BKS) eines Kanals. Damit eine korrekte Schutzbereichsüberwachung in allen Kanälen, in denen der maschinenspezifische Schutzbereich aktiv ist, stattfinden kann, muss das Basiskoordinatensystem (BKS) aller betroffenen Kanäle identisch sein:

- Lage des Koordinatenursprungs bezogen auf den Maschinennullpunkt
- Orientierung der Koordinatenachsen

Bezugspunkt der Konturbeschreibung

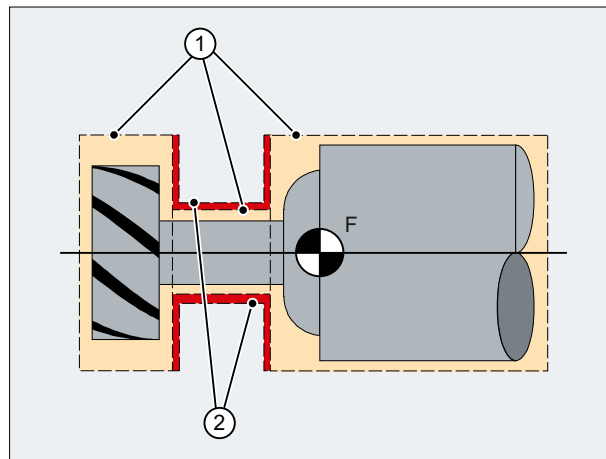
- Werkzeugbezogene Schutzbereiche
Koordinaten für **werkzeugbezogene** Schutzbereiche sind absolut, bezogen auf den **Werkzeugträgerbezugspunkt F**, anzugeben.
- Werkstückbezogene Schutzbereiche
Koordinaten für **werkstückbezogene** Schutzbereiche sind absolut, bezogen auf den Nullpunkt des **Basiskoordinatensystems (BKS)**, anzugeben.

Rotationssymmetrische Schutzbereiche

Bei rotationssymmetrischen Schutzbereichen (z. B. Spindelfutter) muss die Gesamtkontur beschrieben werden, nicht nur die Kontur bis zur Drehmitte.

Werkzeugbezogene Schutzbereiche

Werkzeugbezogene Schutzbereiche müssen immer konvex sein. Falls ein konkaver Schutzbereich gewünscht ist, ist dieser in mehrere konvexe Schutzbereiche zu zerlegen.



- ① Konvexe Schutzbereiche
- ② Konkave Schutzbereiche (**nicht zulässig!**)
- F Werkzeugträgerbezugspunkt

Randbedingungen

Während der Definition eines Schutzbereichs dürfen folgende Funktionen nicht aktiv sein bzw. verwendet werden:

- Werkzeugradiuskorrektur (Fräserradiuskorrektur, Schneidenradiuskorrektur)
- Transformation
- Referenzpunktanfahren (G74)
- Festpunktanfahren (G75)
- Verweilzeit (G4)
- Satzvorlauf-Stopp (STOPRE)
- Programmende (M17, M30)
- M-Funktionen: M0, M1, M2

4.3.2 Schutzbereiche aktivieren/deaktivieren (CPROT, NPROT)

Vorher im Teileprogramm definierte Schutzbereiche können jederzeit aktiviert bzw. für eine spätere Aktivierung durch das PLC-Anwenderprogramm voraktiviert werden. Aktive Schutzbereiche können jederzeit wieder deaktiviert werden.

Bei der Aktivierung bzw. Voraktivierung besteht zudem die Möglichkeit, den Bezugspunkt des Schutzbereichs relativ zu verschieben.

Hinweis

Ein Schutzbereich wird erst nach dem Referenzieren aller Geometrieachsen des Kanals, in dem er aktiviert wurde, berücksichtigt.

Hinweis

Überwachung der Schutzbereiche

Ist kein werkzeugbezogener Schutzbereich aktiv, wird die Werkzeugbahn gegen die werkstückbezogenen Schutzbereiche geprüft.

Ist kein werkstückbezogener Schutzbereich aktiv, findet keine Schutzbereichsüberwachung statt.

Syntax

CPROT (<n>, <Status>, <XMov>, <YMov>, <ZMov>)

NPROT (<n>, <Status>, <XMov>, <YMov>, <ZMov>)

Bedeutung

CPROT:	Vordefinierte Prozedur zum Aktivieren eines kanalspezifischen Schutzbereichs			
NPROT:	Vordefinierte Prozedur zum Aktivieren eines maschinenspezifischen Schutzbereichs			
<n>:	Nummer des Schutzbereichs			
	Datentyp:	INT		
<Status>:	Mit diesem Parameter wird der kanalspezifische Aktivierungsstatus gesetzt			
	Datentyp:	INT		
	Wert:	0	Schutzbereich deaktivieren	
		1	Schutzbereich voraktivieren	
		2	Schutzbereich aktivieren	
3		Schutzbereich voraktivieren mit bedingtem Stopp		

<X Mov>, <Y Mov>, <Z Mov>:	Additive Verschiebungswerte in X/Y/Z-Richtung Die Verschiebung kann in 1, 2 oder 3 Dimensionen erfolgen. Die Verschiebungswerte beziehen sich auf: <ul style="list-style-type: none"> • den Maschinennullpunkt bei werkstückbezogenem Schutzbereich • den Werkzeugträgerbezugspunkt F bei werkzeugbezogenem Schutzbereich
	Datentyp: REAL

Beispiel

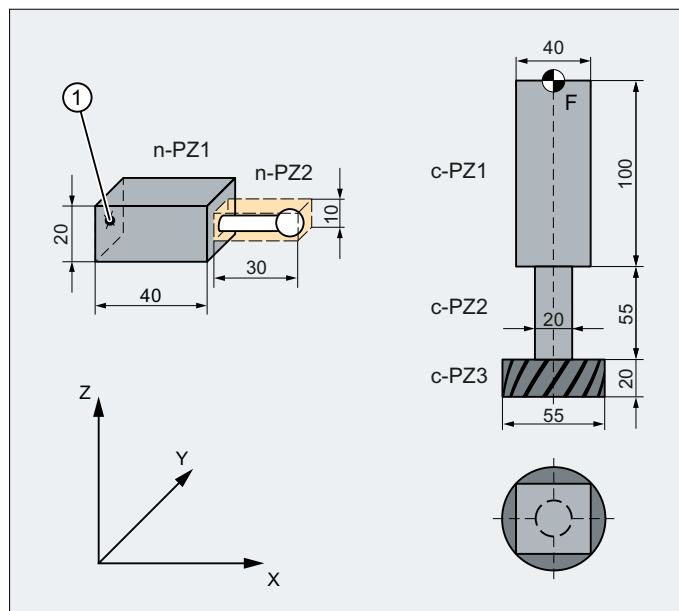
Für eine Fräsmaschine soll eine mögliche Kollision des Fräasers mit dem Messtaster überwacht werden. Die Lage des Messtasters soll bei der Aktivierung durch eine Verschiebung angegeben werden.

Es werden dafür folgende Schutzbereiche definiert:

- Jeweils ein maschinenspezifischer und werkstückbezogener Schutzbereich für den Messtasterhalter (n-PZ1) und für den Messtaster selbst (n-PZ2).
- Jeweils ein kanalspezifischer und werkzeugbezogener Schutzbereich für den Fräserhalter (c-PZ1), den Fräferschaft (c-PZ2) und für den Fräser selbst (c-PZ3).

Die Orientierung aller Schutzbereiche liegt in Z-Richtung.

Die Lage des Bezugspunkts des Messtasters bei der Aktivierung soll bei X = -120, Y = 60 und Z = 80 liegen.



- ① Bezugspunkt für den Schutzbereich des Messtasters
F Werkzeugträgerbezugspunkt

Programmcode	Kommentar
DEF INT PROTZONE	; Definition einer Hilfsvariablen

Programmcode	Kommentar
G17	; Arbeitsebene XY
; Definition der Schutzbereiche:	
NPROTDEF(1,FALSE,3,10,-10)	; Schutzbereich n-PZ1
G01 X0 Y-10	
X40	
Y10	
X0	
Y-10	
EXECUTE (PROTZONE)	
NPROTDEF(2,FALSE,3,5,-5)	; Schutzbereich n-PZ2
G01 X40 Y-5	
X70	
Y5	
X40	
Y-5	
EXECUTE (PROTZONE)	
CPROTDEF(1,TRUE,3,0,-100)	; Schutzbereich c-PZ1
G01 X-20 Y-20	
X20	
Y20	
X-20	
Y-20	
EXECUTE (PROTZONE)	
CPROTDEF(2,TRUE,3,-100,-150)	; Schutzbereich c-PZ2
G01 X0 Y-10	
G03 X0 Y10 J10	
X0 Y-10 J-10	
EXECUTE (PROTZONE)	
CPROTDEF(3,TRUE,3,-150,-170)	; Schutzbereich c-PZ3
G01 X0 Y-27.5	
G03 X0 Y27.5 J27.5	
X0 Y27.5 J-27.5	
EXECUTE (PROTZONE)	
; Aktivierung der Schutzbereiche:	
NPROT(1,2,-120,60,80)	; Schutzbereich n-PZ1 mit Verschiebung aktivieren
NPROT(2,2,-120,60,80)	; Schutzbereich n-PZ2 mit Verschiebung aktivieren
CPROT(1,2,0,0,0)	; Schutzbereich c-PZ1 aktivieren
CPROT(2,2,0,0,0)	; Schutzbereich c-PZ2 aktivieren
CPROT(3,2,0,0,0)	; Schutzbereich c-PZ3 aktivieren

Weitere Informationen

Aktivierungsstatus nach Hochlaufen der Steuerung

Ein Schutzbereich kann bereits nach dem Hochlaufen der Steuerung und dem Referenzieren der Achsen aktiv sein. Dies ist der Fall, wenn für den Schutzbereich die folgende Systemvariable auf TRUE gesetzt ist:

- \$SN_PA_ACTIV_IMMED[<n>] (für maschinenspezifischen Schutzbereich) bzw.
- \$SC_PA_ACTIV_IMMED[<n>] (für kanalspezifischen Schutzbereich)
Der Index "<n>" entspricht der Nummer des Schutzbereichs: 0 = 1. Schutzbereich

Der Schutzbereich wird mit dem Status = 2 und ohne Verschiebung aktiviert.

Mehrfaches Aktivieren eines Schutzbereichs

Ein maschinenspezifischer Schutzbereich kann gleichzeitig in mehreren Kanälen wirksam sein (z. B. Schutzbereich der Pinole bei zwei gegenüberliegenden Schlitten). Die Überwachung der Schutzbereiche erfolgt nur, wenn alle Geometrieachsen referenziert sind.

Ein Schutzbereich ist in einem Kanal nicht gleichzeitig mit verschiedenen Verschiebungen aktivierbar.

Schutzbereichsüberwachung bei aktiver Werkzeugradiuskorrektur

Bei aktiver Werkzeugradiuskorrektur ist eine funktionsfähige Schutzbereichsüberwachung nur möglich, wenn die Ebene der Werkzeugradiuskorrektur identisch ist mit der Ebene der Schutzbereichsdefinitionen.

4.3.3 Überprüfung auf Schutzbereichsverletzung, Arbeitsfeldbegrenzung und Software-Endschalter (CALCPOSI)

Funktion

Die Funktion CALCPOSI überprüft im Werkstückkoordinatensystem (WKS), ob ausgehend von der Startposition die **Geometrieachsen** einen vorgegebenen Weg verfahren können, ohne aktive Begrenzungen zu verletzen. Für den Fall, dass der Verfahrweg aufgrund von Begrenzungen nicht vollständig abgefahren werden kann, werden ein positiver, dezimal codierter Statuswert und der maximal mögliche Verfahrweg zurückgegeben.

Definition

```
INT CALCPOSI (VAR REAL[3] <Start>, VAR REAL[3] <Dist>, VAR REAL[5]
<Limit>, VAR REAL[3] <MaxDist>, BOOL <System>, INT <TestLim>)
```

Syntax

```
<State> = CALCPOSI (VAR <Start>, VAR <Dist>, VAR <Limit>, VAR
<MaxDist>, <System>, <TestLim>)
```

Bedeutung

CALCPOSI (. . .) :	Vordefinierte Funktion für den Test auf Begrenzungsverletzungen bezüglich der Geometrieachsen	
	Vorlaufstopp:	nein
	Alleine im Satz:	ja
<State>: (Teil 1)	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.	
	Datentyp:	INT
	Wertebereich:	$-8 \leq x \leq 100000$
	Wert:	0 Der Fahrweg kann vollständig abgefahren werden
		-1 In <Limit> ist mindestens eine Komponente negativ.
		-2 Fehler bei einer Transformationsberechnung. Beispiel: Der Fahrweg führt durch eine Singularität, so dass die Achspositionen nicht definiert sind.
		-3 Der angegebene Fahrweg <Dist> und der maximal mögliche Fahrweg <MaxDist> sind linear abhängig. Hinweis Kann nur im Zusammenhang mit <TestLim>, Bit 4 == 1 auftreten.
		-4 Die Projektion der in <Dist> enthaltenen Fahrtrichtung auf die Begrenzungsfläche ist der Nullvektor bzw. die Fahrtrichtung steht senkrecht auf der verletzten Begrenzungsfläche. Hinweis Kann nur im Zusammenhang mit <TestLim>, Bit 5 == 1 auftreten.
		-5 In <TestLim> sind Bit 4 == 1 UND Bit 5 == 1
		-6 Mindestens eine Maschinenachse, die für die Überprüfung der Fahrgrenzen betrachtet werden muss, ist nicht referenziert.
		-7 Funktion Kollisionsvermeidung: Ungültige Definition der kinematischen Kette oder der Schutzbereiche.
		-8 Funktion Kollisionsvermeidung: Die Funktion kann wegen Speichermangel nicht ausgeführt werden.

<State>: (Teil 2)	Einerstelle		
	Hinweis Sind gleichzeitig mehrere Grenzen verletzt, wird diejenige gemeldet, die zur stärksten Einschränkung des vorgegebenen Verfahrenswegs führt.		
	Wert:	1	Software-Endschalter begrenzen den Verfahrensweg
		2	Arbeitsfeldbegrenzung begrenzt den Verfahrensweg
		3	Schutzbereiche begrenzen den Verfahrensweg
		4	Funktion Kollisionsvermeidung: Schutzbereiche begrenzen den Verfahrensweg
	Zehnerstelle		
Wert:	1x	Der Anfangswert verletzt die Grenze	
	2x	Die vorgegebene Gerade verletzt die Grenze. Dieser Wert wird auch dann zurückgegeben, wenn der Endpunkt selbst keine Grenze verletzt, auf dem Weg vom Start zum Endpunkt aber eine Verletzung eines Grenzwerts auftreten würde (z. B. Durchfahren eines Schutzbereichs, gekrümmte Software-Endschalter im WKS bei nichtlinearen Transformationen, z. B. Transmit).	
<State>: (Teil 3)	Hunderterstelle		
	Wert:	1xx	UND Einerstelle == 1 oder 2: Der positive Grenzwert ist verletzt.
			UND Einerstelle == 3 ¹⁾ : Es ist ein NC-spez. Schutzbereich verletzt.
		2xx	UND Einerstelle == 1 oder 2: Der negative Grenzwert ist verletzt.
			UND Einerstelle == 3 ¹⁾ : Es ist ein kanalspezifischer Schutzbereich verletzt.
	Tausenderstelle		
<State>: (Teil 4)	Wert:	1xxx	UND Einerstelle == 1 oder 2: Faktor, mit dem die Achsnummer multipliziert wird, die die Grenze verletzt. Die Zählung der Achsen beginnt bei 1. Bezug: <ul style="list-style-type: none">Software-Endschalter: MaschinenachsenArbeitsfeldbegrenzung: Geometrieachsen UND Einerstelle == 3 ¹⁾ : Faktor, mit dem die Nummer des verletzten Schutzbereichs multipliziert wird.
<State>: (Teil 5)	Einhunderttausenderstelle		
	Wert:	0xxxxx	Einhunderttausenderstelle == 0: <Dist> bleibt unverändert
		1xxxxx	In <Dist> wird ein Richtungsvektor zurückgegeben, der die weitere Bewegungsrichtung auf der Begrenzungsfläche definiert. Kann nur bei folgenden Randbedingungen auftreten: <ul style="list-style-type: none">Software-Endschalter oder Arbeitsfeldbegrenzung verletzt (nicht im Startpunkt)Eine Transformation ist nicht aktiv<TestID>, Bit 4 oder Bit 5 == 1

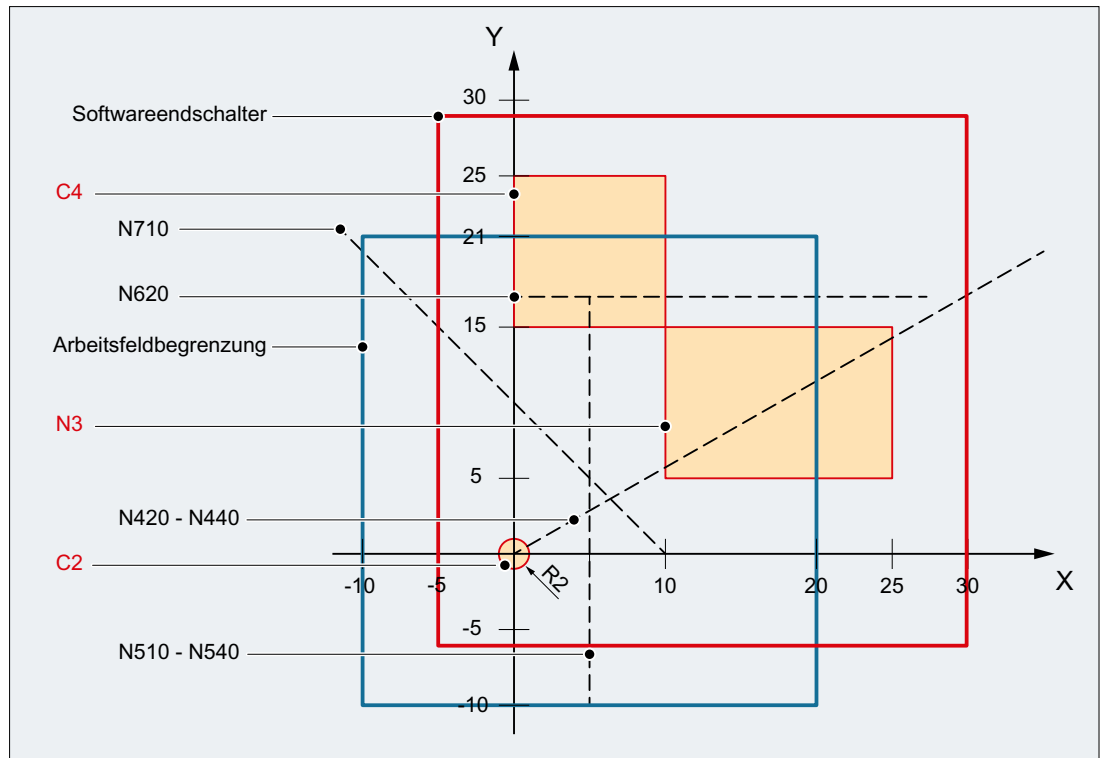
<Start>:	Referenz auf einen Vektor mit den Startpositionen:	
	<ul style="list-style-type: none"> • <Start> [0]: 1. Geometrieachse • <Start> [1]: 2. Geometrieachse • <Start> [2]: 3. Geometrieachse 	
	Parametertyp:	Eingang
	Datentyp:	VAR REAL[3]
	Wertebereich:	-max. REAL-Wert $\leq x[<n>] \leq$ +max. REAL-Wert
<Dist>:	Referenz auf einen Vektor.	
	Eingang: inkrementeller Verfahrensweg	
	<ul style="list-style-type: none"> • <Dist> [0]: 1. Geometrieachse • <Dist> [1]: 2. Geometrieachse • <Dist> [2]: 3. Geometrieachse 	
	Ausgang (nur bei gesetzter Einhunderttausenderstelle im <State>):	
	<Dist> enthält als Ausgangswert einen Einheitsvektor v , der die weitere Verfahrrichtung im WKS definiert.	
	Fall 1: Bildung des Vektors v bei <TestID>, Bit 4 == 1 Die Eingangsvektoren <Dist> und <MaxDist> spannen die Bewegungsebene auf. Diese Ebene wird mit der verletzten Begrenzungsfläche geschnitten. Die Schnittgerade der beiden Ebenen definiert die Richtung des Vektors v . Dabei wird die Orientierung (Vorzeichen) so gewählt, dass der Winkel zwischen dem Eingangsvektor <MaxDist> und v nicht größer als 90 Grad ist.	
	Fall 2: Bildung des Vektors v bei <TestID>, Bit 5 == 1 Der Vektor v ist der Einheitsvektor in Richtung der Projektion des in <Dist> enthaltenen Fahrvektors auf die Begrenzungsfläche. Ist die Projektion des Fahrvektors auf die Begrenzungsfläche der Nullvektor, wird ein Fehler zurückgegeben.	
	Parametertyp:	Ein/Ausgang
	Datentyp:	VAR REAL[3]
	Wertebereich:	-max. REAL-Wert $\leq x[<n>] \leq$ +max. REAL-Wert

<Limit>:	<p>Referenz auf ein Feld der Länge 5.</p> <ul style="list-style-type: none"> • <Limit> [0 - 2]: Mindestabstände der Geometrieachsen zu den Begrenzungen: <ul style="list-style-type: none"> – <Limit> [0]: 1. Geometrieachse – <Limit> [1]: 2. Geometrieachse – <Limit> [2]: 3. Geometrieachse <p>Die Mindestabstände werden eingehalten bei:</p> <ul style="list-style-type: none"> – Arbeitsfeldbegrenzung: Keine Einschränkungen – Software-Endschalter: Wenn keine Transformation aktiv ist, oder eine Transformation aktiv ist, bei der eine eindeutige Zuordnung der Geometrieachsen zu linearen Maschinenachsen möglich ist, z. B. 5-Achs-Transformationen. • <Limit> [3]: Enthält den Mindestabstand für lineare Maschinenachsen, die z. B. auf Grund einer nichtlinearen Transformation keiner Geometrieachse zugeordnet werden können. Dieser Wert wird außerdem als Grenzwert bei der Überwachung der konventionellen Schutzbereiche und der Schutzbereiche der Kollisionsvermeidung verwendet. • <Limit> [4]: Enthält den Mindestabstand für rotatorische Maschinenachsen, die z. B. auf Grund einer nichtlinearen Transformation keiner Geometrieachse zugeordnet werden können. <p>Hinweis Dieser Wert wird nur bei der Überwachung der Software-Endschalter von speziellen Transformationen wirksam.</p> <table border="1" data-bbox="603 1017 1476 1129"> <tr> <td>Parametertyp:</td><td>Eingang</td></tr> <tr> <td>Datentyp:</td><td>VAR REAL[5]</td></tr> <tr> <td>Wertebereich:</td><td>-max. REAL-Wert ≤ x[n] ≤ +max. REAL-Wert</td></tr> </table>	Parametertyp:	Eingang	Datentyp:	VAR REAL[5]	Wertebereich:	-max. REAL-Wert ≤ x[n] ≤ +max. REAL-Wert
Parametertyp:	Eingang						
Datentyp:	VAR REAL[5]						
Wertebereich:	-max. REAL-Wert ≤ x[n] ≤ +max. REAL-Wert						
<MaxDist>:	<p>Referenz auf einen Vektor mit dem inkrementellen Verfahrensweg, bei dem der vorgegebene Mindestabstand von einer Achsgrenze von allen beteiligten Maschinenachsen nicht unterschritten wird:</p> <ul style="list-style-type: none"> • <Dist> [0]: 1. Geometrieachse • <Dist> [1]: 2. Geometrieachse • <Dist> [2]: 3. Geometrieachse <p>Ist der Verfahrensweg nicht eingeschränkt, ist der Inhalt dieses Rückgabeparameters gleich dem Inhalt von <Dist>.</p> <p>Bei <TestID>, Bit 4 == 1: <Dist> und <MaxDist> <MaxDist> und <Dist> müssen als Eingangswerte Vektoren enthalten, die eine Bewegungsebene aufspannt. Die beiden Vektoren müssen voneinander linear unabhängig sein. Der Betrag von <MaxDist> ist beliebig. Zur Berechnung der Bewegungsrichtung siehe die Beschreibung zu <Dist>.</p> <table border="1" data-bbox="603 1581 1476 1691"> <tr> <td>Parametertyp:</td><td>Ausgang</td></tr> <tr> <td>Datentyp:</td><td>VAR REAL[3]</td></tr> <tr> <td>Wertebereich:</td><td>-max. REAL-Wert ≤ x[<n>] ≤ +max. REAL-Wert</td></tr> </table>	Parametertyp:	Ausgang	Datentyp:	VAR REAL[3]	Wertebereich:	-max. REAL-Wert ≤ x[<n>] ≤ +max. REAL-Wert
Parametertyp:	Ausgang						
Datentyp:	VAR REAL[3]						
Wertebereich:	-max. REAL-Wert ≤ x[<n>] ≤ +max. REAL-Wert						

<System>:	Maßsystem (inch / metrisch) für Positions- und Längenangaben (optional)		
	Datentyp:		BOOL
	Wert:	FALSE (Default)	Maßsystem entsprechend des aktuell aktiven G-Befehls aus G-Gruppe 13 (G70, G71, G700, G710). Hinweis Bei aktivem G70 und Grundsystem metrisch oder aktivem G71 und Grundsystem inch werden die Systemvariablen \$AA_IW und \$AA_MW im Grundsystem geliefert und müssen bei Verwendung für CALCPOSI gegebenenfalls umgerechnet werden.
		TRUE	Maßsystem gemäß eingestellttem Grundsystem: MD52806 \$MN_ISO_SCALING_SYSTEM
<TestLim>:	Bitcodierte Auswahl der zu überwachenden Begrenzungen (optional)		
	Datentyp:		INT
	Defaultwert:		Bit 0, 1, 2, 3, 6, 7 == 1 (207)
	Bit	Dezimal	Bedeutung
	0	1	Software-Endschalter
	1	2	Arbeitsfeldbegrenzung
	2	4	Aktivierte konventionelle Schutzbereiche
	3	8	Voraktivierte konventionelle Schutzbereiche
	4	16	Bei verletzten Software-Endschaltern bzw. Arbeitsfeldbegrenzungen in <Dist> die Verfahrrichtung entsprechend Fall 1 (siehe oben) zurückliefern.
	5	32	Bei verletzten Software-Endschaltern bzw. Arbeitsfeldbegrenzungen in <Dist> die Verfahrrichtung entsprechend Fall 2 (siehe oben) zurückliefern.
	6	64	Aktivierte Schutzbereiche der Kollisionsvermeidung
	7	128	Voraktivierte Schutzbereiche der Kollisionsvermeidung
	8	256	Paare von aktivierten und voraktivierten Schutzbereichen der Kollisionsvermeidung
¹⁾ Sind mehrere Schutzbereiche verletzt, wird der Schutzbereich zurückgegeben, der zur stärksten Einschränkung des vorgegebenen Fahrwegs führt.			

Beispiel

Begrenzungen



Im Beispiel sind die wirksamen Software-Endschalter und Arbeitsfeldbegrenzungen in der X-Y-Ebene und folgende drei Schutzbereiche dargestellt:

- C2: Werkzeugbezogener kanalspezifischer Schutzbereich, aktiv, kreisförmig, Radius = 2 mm
- C4: Werkstückbezogener, kanalspezifischer Schutzbereich, voraktiviert, quadratisch, Seitenlänge = 10 mm
- N3: Maschinenspezifischer Schutzbereich, aktiv, rechteckig, Seitenlänge = 10 mm x 15 mm

NC-Programm

Im NC-Programm werden zunächst die Schutzbereiche und Arbeitsfeldbegrenzungen definiert. Anschließend wird die Funktion `CALCPOSI()` mit verschiedenen Parametrierungen aufgerufen.

Programmcode

```

N10 DEF REAL _START[3]
N20 DEF REAL _DIST[3]
N30 DEF REAL _LIMIT[5]
N40 DEF REAL _MAXDIST[3]
N50 DEF INT _PA
N60 DEF INT _STATE

```

Programmcode

```
; Werkzeugbezogener Schutzbereich C2
N70 CPROTDEF(2, TRUE, 0)
N80 G17 G1 X-2 Y0
N90 G3 I2 X2
N100 I-2 X-2
N110 EXECUTE(_PA)

; Werkstückbezogener Schutzbereich C4
N120 CPROTDEF(4, FALSE, 0)
N130 G17 G1 X0 Y15
N140 X10
N150 Y25
N160 X0
N170 Y15
N180 EXECUTE(_PA)

; Maschinenspezifischer Schutzbereich N3
N190 NPROTDEF(3, FALSE, 0)
N200 G17 G1 X10 Y5
N210 X25
N220 Y15
N230 X10
N240 Y5
N250 EXECUTE(_PA)

; Schutzbereiche aktivieren bzw. voraktivieren
N260 CPROT(2, 2, 0, 0, 0)
N270 CPROT(4, 1, 0, 0, 0)
N280 NPROT(3, 2, 0, 0, 0)

; Arbeitsfeldbegrenzungen definieren
N290 G25 XX=-10 YY=-10
N300 G26 XX=20 YY=21
N310 _START[0] = 0.
N320 _START[1] = 0.
N330 _START[2] = 0.
N340 _DIST[0] = 35.
N350 _DIST[1] = 20.
N360 _DIST[2] = 0.
N370 _LIMIT[0] = 0.
N380 _LIMIT[1] = 0.
N390 _LIMIT[2] = 0.
N400 _LIMIT[3] = 0.
N410 _LIMIT[4] = 0.
N420 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST)
N430 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 3)
N440 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 1)
N450 _START[0] = 5.
N460 _START[1] = 17.
N470 _START[2] = 0.
N480 _DIST[0] = 0.
N490 _DIST[1] = -27.
N500 _DIST[2] = 0.
N510 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 14)
N520 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 6)
N530 _LIMIT[1] = 2.
```


Programmcode

```

N540 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 6)
N550 _START[0] = 27.
N560 _START[1] = 17.1
N570 _START[2] = 0.
N580 _DIST[0] = -27.
N590 _DIST[1] = 0.
N600 _DIST[2] = 0.
N610 _LIMIT[3] = 2.
N620 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,12)
N630 _START[0] = 0.
N640 _START[1] = 0.
N650 _START[2] = 0.
N660 _DIST[0] = 0.
N670 _DIST[1] = 30.
N680 _DIST[2] = 0.
N690 TRANS X10
N700 AROT Z45
N710 _STATE = CALCPOSI(_START,_DIST, _LIMIT, _MAXDIST)
; Frames aus N690 und N700 wieder löschen
N720 TRANS
N730 _START[0] = 0.
N740 _START[1] = 10.
N750 _START[2] = 0.
; Vektoren _DIST und _MAXDIST definieren die Bewegungsebene
N760 _DIST[0] = 30.
N770 _DIST[1] = 30.
N780 _DIST[2] = 0.
N790 _MAXDIST[0] = 1.
N800 _MAXDIST[1] = 0.
N810 _MAXDIST[2] = 1.
N820 _STATE = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,17)
N830 M30

```

Ergebnisse von CALCPOSI()

N...	<State>	<MaxDist>[0] Δ X	<MaxDist>[1] Δ Y	Bemerkungen
420	3123	8.040	4.594	N3 wird verletzt.
430	1122	20.000	11.429	Keine Überwachung der Schutzbereiche, Arbeitsfeldbegrenzung wird verletzt.
440	1121	30.000	17.143	Nur noch Überwachung der Software-Endschalter aktiv.
510	4213	0.000	0.000	Startpunkt verletzt C4
520	0000	0.000	-27.000	Voraktivierter C4 wird nicht überwacht. Der vorgegebene Weg kann vollständig verfahren werden.
540	2222	0.000	-25.000	Wegen _LIMIT[1] = 2 wird der Verfahrenweg durch die Arbeitsfeldbegrenzung eingeschränkt.

N...	<State>	<MaxDist>[0] Δ X	<MaxDist>[1] Δ Y	Bemerkungen
620	4223	-13.000	0.000	Abstand zu C4 wegen C2 und _LIMIT[3] insgesamt 4 mm. Abstand C2 → N3 von 0,1 mm führt nicht zur Beschränkung des Verfahrwegs.
710	1221	0.000	21.213	Frame mit Translation und Rotation aktiv. Der zulässige Verfahrweg in _DIST gilt im verschobenen und gedrehten WKS.
820	102121	18.000	18.000	Der Software-Endschalter der Y-Achse wird verletzt. Mit <_TESTLIM> = 17 wird die Berechnung einer Weiterfahrrichtung angefordert. Diese Richtung steht in <_DIST> (0.707, 0.0, 0.707). Sie ist gültig, da in <_STATUS> die Einhunderttausenderstelle gesetzt ist.

Weitere Informationen

Achsstatus "Referenziert"

Alle von `CALCPOSI()` betrachteten Maschinenachsen müssen referenziert sein.

Kreisbezogene Wegangaben

Alle kreisbezogenen Wegangaben werden **immer** als Radiusangabe interpretiert. Dies ist insbesondere bei Planachsen mit aktivierter Durchmesserprogrammierung (`DIAMON` / `DIAM90`) zu berücksichtigen.

Verfahrwegreduzierung

Wird der angegebene Verfahrweg einer Achsen begrenzt, sind im Rückgabewert `<MaxDist>` auch die Verfahrwege der anderen Achsen anteilmäßig reduziert. Der resultierende Endpunkt liegt dadurch weiterhin auf der vorgegebenen Bahn.

Rundachsen

Rundachsen werden nur überwacht, wenn es keine Modulo-Rundachsen sind.

Es ist zulässig, dass für eine oder mehrere der beteiligten Achsen keine Software-Endschalter bzw. Arbeitsfeldbegrenzungen oder Schutzbereiche definiert sind.

Software-Endschalter- und Arbeitsfeldbegrenzungs-Status

Software-Endschalter und Arbeitsfeldbegrenzungen werden nur berücksichtigt, wenn diese bei Ausführung von `CALCPOSI()` aktiv sind. Der Status kann z. B. beeinflusst werden über:

- Maschinendaten: MD21020 \$MC_WORKAREA_WITH_TOOL_RADIUS
- Settingdaten: \$AC_WORKAREA_CS_...
- NC/PLC-Nahtstellensignale: DB31, ... DBX12.2 / 3
- Befehle: `WALIMON` / `WALIMOF`

Software-Endschalter und Transformationen

Bei `CALCPOSI()` können bei verschiedenen kinematischen Transformationen (z. B. `TRANSMIT`) aufgrund von Mehrdeutigkeiten an bestimmten Stellen des Verfahrbereichs die

Positionen der Maschinenachsen (MKS) nicht immer eindeutig aus den Positionen der Geometrieachsen (WKS) bestimmt werden. Im normalen Verfahrbetrieb ergibt sich die Eindeutigkeit in der Regel aus der Vorgeschichte und der Bedingung, dass einer kontinuierlichen Bewegung im WKS eine kontinuierliche Bewegung der Maschinenachsen im MKS entsprechen muss. Bei der Überwachung der Software-Endschalter wird deshalb in derartigen Fällen die zum Ausführungszeitpunkt von `CALCPOSI()` vorliegende Maschinenposition zur Auflösung der Mehrdeutigkeit verwendet.

Hinweis**Vorlaufstopp**

Bei Verwendung von `CALCPOSI()` im Zusammenhang mit Transformationen liegt es in der alleinigen Verantwortung des Anwenders, zur Synchronisation der Maschinenachsposten mit dem Vorlauf vor `CALCPOSI()` einen Vorlaufstopp (`STOPRE`) zu programmieren.

Schutzbereichsabstand und konventionelle Schutzbereiche

Bei konventionellen Schutzbereichen ist **nicht** gewährleistet, dass bei einer Verfahrbewegung auf dem vorgegebenen Verfahrweg der im Parameter `<Limit>[3]` angegebene Sicherheitsabstand gegenüber allen Schutzbereichen eingehalten wird. Es ist nur gewährleistet, dass bei Verlängerung des in `<Dist>` zurückgelieferten Endpunkts um den Sicherheitsabstand in Verfahrrichtung kein Schutzbereich verletzt wird. Die Gerade kann aber in ihrem Verlauf beliebig dicht an einem Schutzbereich vorbei führen.

Schutzbereichsabstand und Schutzbereiche der Kollisionsvermeidung

Bei Schutzbereichen der Kollisionsvermeidung ist gewährleistet, dass bei einer Verfahrbewegung auf dem vorgegebenen Verfahrweg, der im Parameter `<Limit>[3]` angegebene Sicherheitsabstand gegenüber allen Schutzbereichen eingehalten wird.

Der im Parameter `<Limit>[3]` angegebene Sicherheitsabstand wird nur wirksam, wenn gilt:

`<Limit>[3] > (MD10619 $MN_COLLISION_TOLERANCE)`

Ist in Parameter `<TestLim>` Bit 4 gesetzt (Berechnung der weiteren Verfahrrichtung), ist der in `<DIST>` enthaltene Richtungsvektor nur dann gültig, wenn die Einhunderttausenderstelle im Rückgabewert der Funktion (`<State>`) gesetzt ist. Kann eine solche Richtung nicht ermittelt werden, weil entweder Schutzbereiche verletzt wurden oder eine Transformation aktiv ist, bleibt der Eingangswert in `<DIST>` unverändert. Eine weitere Fehlermeldung erfolgt nicht.

4.4 Besonderheiten

4.4.1 Zeitweise Freigabe von Schutzbereichen

Tritt beim Start oder während einer Verfahrbewegung eine Schutzbereichsverletzung auf, kann unter bestimmten Voraussetzungen der Schutzbereich vorübergehend, d. h. zeitweise, zum Durchfahren freigegeben werden. Sowohl in den Betriebsarten AUTOMATIK und MDA als auch in der Betriebsart JOG erfolgt die zeitweise Freigabe von Schutzbereichen über Bedienhandlungen.

Siehe auch:

- Verhalten in den Betriebsarten AUTOMATIK und MDA (Seite 168)
- Verhalten in der Betriebsart JOG (Seite 169)

Hinweis

Eine zeitweise Freigabe ist **nur** bei **werkstückbezogenen** Schutzbereichen möglich!
Werkzeugbezogene Schutzbereiche müssen entweder im Teileprogramm deaktiviert oder über die NC/PLC-Nahtstelle in den Zustand "voraktiviert" versetzt werden.

Die zeitweise Freigabe eines Schutzbereichs ist nach folgenden Ereignissen beendet:

- Hochlauf der Steuerung
- Betriebsarten AUTOMATIK oder MDA: Das Satzende liegt außerhalb des Schutzbereichs
- Betriebsart JOG: Das Ende der Verfahrbewegung liegt außerhalb des Schutzbereichs
- Der Schutzbereich wird aktiviert

4.4.2 Verhalten in den Betriebsarten AUTOMATIK und MDA

In den Betriebsarten AUTOMATIK und MDA werden keine Verfahrbewegungen in oder durch aktive Schutzbereiche freigegeben:

- Eine Verfahrbewegung, die von außerhalb in einen aktiven Schutzbereich hinein führen würde, wird am Satzendpunkt des letzten, außerhalb des Schutzbereichs liegenden Satzes angehalten.
- Eine Verfahrbewegung, die innerhalb eines aktiven Schutzbereichs beginnt, wird nicht gestartet.

Zeitweises Freigeben von Schutzbereichen

Wird in der Betriebsart AUTOMATIK oder MDA eine Verfahrbewegung aufgrund einer Schutzbereichsverletzung angehalten, wird dies dem Bediener durch einen Alarm angezeigt. Entscheidet der Bediener, dass die Verfahrbewegung fortgesetzt werden kann, kann er das Durchfahren von Schutzbereichen freigegeben.

Die Freigabe ist nur temporär und erfolgt über das Auslösen von NC-Start:

DB21, ... DBX7.1 (NC-Start) = 1

Pro verletzten Schutzraum wird ein Alarm angezeigt. Für jeden freizugebenden Schutzraum muss vom Bediener ein NC-Start-Signal ausgelöst werden.

Die Verfahrbewegung wird fortgesetzt, wenn alle Schutzbereiche die zum Anhalten der Verfahrbewegung geführt haben, freigegeben sind.

Fortsetzen einer Verfahrbewegung ohne zeitweises Freigeben

Eine Verfahrbewegung wurde aufgrund einer Schutzbereichsverletzung mit Alarm angehalten. Wird der betroffene Schutzbereich über die NC/PLC-Nahtstelle in den Zustand "voraktiviert" versetzt, kann die Verfahrbewegung mit NC-Start fortgesetzt werden, ohne dass der Schutzbereich zeitweise freigegeben wird.

Stärkere Absicherung der Freigabe von Schutzbereichen

Soll die Freigabe eines Schutzbereichs stärker als nur durch NC-Start abgesichert werden, ist dies vom Maschinenhersteller bzw. Anwender im PLC-Anwenderprogramm auszuführen.

4.4.3 Verhalten in der Betriebsart JOG

Gleichzeitiges Verfahren mehrerer Geometrieachsen

In der Betriebsart JOG können Verfahrbewegungen in mehreren Geometrieachsen gleichzeitig ausgeführt werden. Der Verfahrbereich jeder beteiligten Geometrieachse wird achsspezifisch zum Startzeitpunkt der Verfahrbewegung bezüglich der Verfahrbereichsgrenzen (Software-Endschalter, Arbeitsfeldbegrenzung, etc.) und aktiven Schutzbereiche begrenzt. Eine sichere Überwachung aller aktiven Schutzbereiche kann dabei nicht gewährleistet werden. Als Rückmeldung an den Anwender erfolgt:

- Alarm 10704 "Schutzbereichsüberwachung nicht gewährleistet"
- DB31, ... DBX39.0 (Schutzbereichsüberwachung nicht gewährleistet) = 1

Nach dem Ende der Verfahrbewegungen wird der Alarm automatisch gelöscht.

Liegt die aktuelle Position innerhalb eines aktivierten oder voraktivierten Schutzbereichs, werden folgende Aktionen ausgelöst:

- Alarmmeldung 10702 "NC-Schutzbereich im Handbetrieb verletzt" bzw. 10703 "Kanalspezifischer Schutzbereich im Handbetrieb verletzt" mit Angabe des verletzten Schutzbereichs und der verfahrenen Achse.
- Weitere Verfahrbewegungen werden gesperrt.
- Für den betroffenen Schutzbereich wird das folgende NC/PLC-Nahtstellensignal gesetzt:
DB21, ... DBX276.0 – 277.1 (Maschinenspezifischer Schutzbereich 1 - 10 verletzt) == 1
bzw.
DB21, ... DBX278.0 – 279.1 (Kanalspezifischer Schutzbereich 1 - 10 verletzt) == 1

Zum Fortsetzen siehe Absatz "Zeitweises Freigeben von Schutzbereichen".

Beispiel:

Drei aktivierte Schutzbereiche und gleichzeitiges Verfahren von zwei Geometrieachsen:

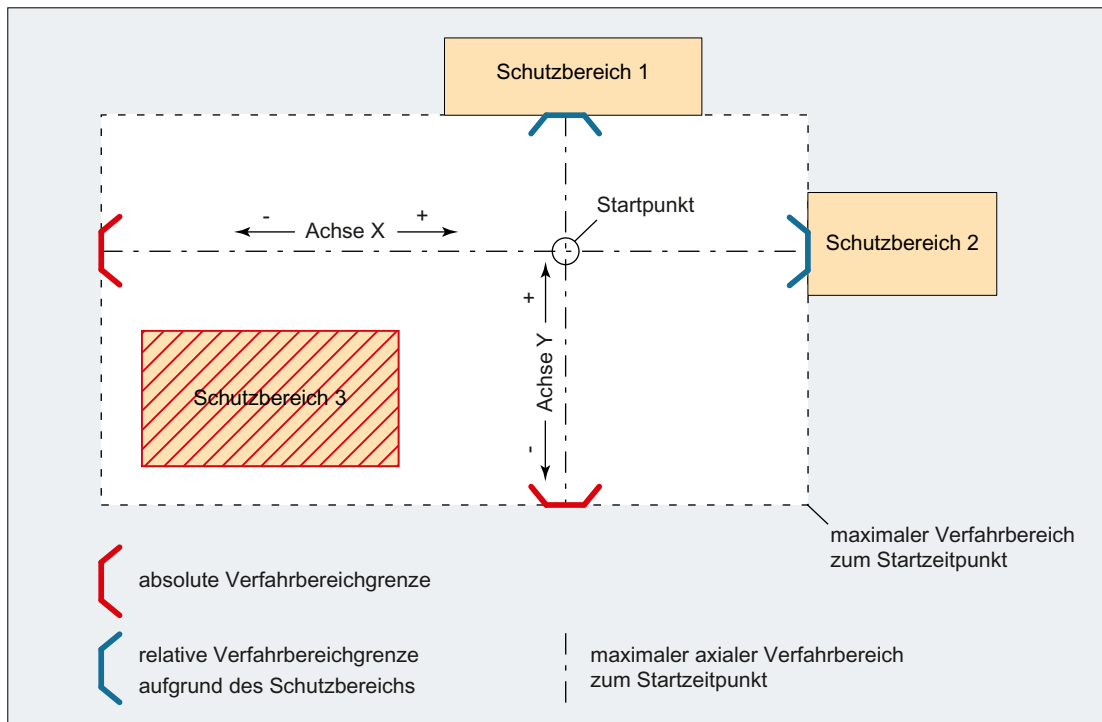


Bild 4-2 Bewegungsbereich der Geometrieachsen zum Startzeitpunkt

Zum Startzeitpunkt der Verfahrbewegungen der Achsen X und Y werden, ausgehend vom Startpunkt, die achsenspezifischen Verfahrbereichsgrenzen ermittelt:

- X Achse
 - positive Verfahrrichtung: Schutzbereich 2
 - negative Verfahrrichtung: absolute Verfahrbereichsgrenze (z. B. Software-Endschalter)
- Y Achse
 - positive Verfahrrichtung: Schutzbereich 1
 - negative Verfahrrichtung: absolute Verfahrbereichsgrenze (z. B. Arbeitsfeldbegrenzung)

Der sich daraus ergebende maximale Verfahrbereich zum Startzeitpunkt berücksichtigt dabei nicht den Schutzbereich 3. Somit wäre eine Schutzbereichsverletzung im Schutzbereich 3 möglich.

Hinweis

Aktivierte und voraktivierte Schutzbereiche werden auch in den manuellen Betriebsarten JOG, INC und DRF überwacht.

Begrenzung der Verfahrbewegung einer Achse

Wird die Verfahrbewegung einer Achse durch das Erreichen eines Schutzbereichs begrenzt, wird Alarm 10706 "NC-Schutzbereich im Handbetrieb erreicht" bzw. 10707 "Kanalspezifischer Schutzbereich im Handbetrieb erreicht" mit Angabe des erreichten Schutzbereichs und der verfahrenen Achse gemeldet. Es ist beim Verfahren einer Achse mit JOG sicher gestellt, dass kein Schutzbereich verletzt ist. (Dieses Verhalten ist analog dem Fahren auf Software-Endschalter bzw. Arbeitsfeldbegrenzung.)

Der Alarm wird gelöscht:

- Beim Fahren einer Achse, die nicht in den Schutzbereich führt.
- Beim Freigeben des Schutzbereichs.
- Beim Hochlaufen der Steuerung.

Wird an einer Schutzbereichsgrenze eine Bewegung Richtung des Schutzbereichs gestartet, wird ebenfalls Alarm 10706 bzw. 10707 gemeldet und die Bewegung wird nicht gestartet.

Zeitweises Freigeben von Schutzbereichen

Wird innerhalb oder auf der Grenze eines aktivierten Schutzbereichs eine Verfahrbewegung gestartet, wird Alarm 10702 "NC-Schutzbereich im Handbetrieb verletzt" bzw. 10703 "Kanalspezifischer Schutzbereich im Handbetrieb verletzt" mit Angabe des verletzten Schutzbereichs und der verfahrenen Achse gemeldet und die Verfahrbewegung nicht gestartet. Die Verfahrbewegung kann ausgeführt werden, wenn der betroffene Schutzbereich dafür zeitweise freigegeben wird. Folgende Aktionen sind dazu auszuführen:

- Erzeugen einer positiven Flanke auf dem NC/PLC-Nahtstellensignal:
DB21, ... DBX1.1 (Schutzbereich freigeben)
- Erneuter Start der gleichen Verfahrbewegung.

Hinweis

Für den zeitweise freigegebenen Schutzbereich wird beim Durchfahren das NC/PLC-Nahtstellensignal "Schutzbereich verletzt" gesetzt:

DB21, ... DBX276.0 – 277.1 (Maschinenspezifischer Schutzbereich 1 - 10 verletzt) == 1
bzw.

DB21, ... DBX278.0 – 279.1 (Kanalspezifischer Schutzbereich 1 - 10 verletzt) == 1

Wird eine Verfahrbewegung gestartet, die nicht in den freigegebenen Schutzbereich führt, wird die Freigabe zurückgesetzt.

Sind durch die Verfahrbewegung weitere Schutzbereiche betroffen, werden pro Schutzbereich weitere Alarme 10702 bzw. 10703 angezeigt. Die in den Alarmen angezeigten Schutzbereiche können durch das Erzeugen weiterer positiver Flanken auf dem NC/PLC-Nahtstellensignal DB21, ... DBX1.1 freigegeben werden.

Verhalten bei Betriebsartenwechsel

Die in der Betriebsart JOG gegebenen zeitweisen Freigaben von Schutzbereichen bleiben nach einem Wechsel in die Betriebsart AUTOMATIK bzw. MDA erhalten. Ebenso bleiben die zeitweisen Freigaben, die in der Betriebsart AUTOMATIK bzw. MDA gegeben wurden, bei einem Wechsel in die Betriebsart JOG erhalten.

Rücksetzen einer Freigabe

Mit dem nächsten Stillstand einer Geometrieachse, bei dem der zeitweise freigegebene Schutzbereich vollständig verlassen wurde, wird die Freigabe intern und an der NC/PLC-Nahtstelle rückgesetzt:

DB21, ... DBX276.0 – 277.1 (Maschinenspezifischer Schutzbereich 1 - 10 verletzt) == 0

bzw.

DB21, ... DBX278.0 – 279.1 (Kanalspezifischer Schutzbereich 1 - 10 verletzt) == 0

4.5 Randbedingungen

Einschränkungen der Schutzbereichsüberwachung

Eine Schutzbereichsüberwachung ist in folgenden Fällen **nicht** möglich:

- Verfahrbewegungen von Orientierungsachsen
- **Feststehende maschinenspezifische** Schutzräume bei Stirnseitentransformation (TRANSMIT) oder Zylindermanteltransformation (TRACYL)
Ausnahme: Rotationssymmetrische Schutzbereiche um die Spindelachse. Dabei darf keine DRF-Verschiebung aktiv sein.
- Gegenseitiges Überwachen von **werkzeugbezogenen** Schutzräumen

Positionierachsen

Bei Positionierachsen wird nur der programmierte Satzendpunkt überwacht. Während der Verfahrbewegung einer Positionierachse wird Alarm 10704 "Schutzbereichsüberwachung nicht gewährleistet" angezeigt.

Achstausch

Bezüglich der Schutzbereiche wird nach einem Achstausch im abgebenden Kanal von der letzten im Kanal angefahrenen Position ausgegangen. Verfahrbewegungen im übernehmenden Kanal werden nicht berücksichtigt. Es ist daher darauf zu achten, dass ein Achstausch nicht aus einer Position mit Schutzraumverletzung erfolgt.

Ist eine für den Achstausch vorgesehene Achse in einem Kanal nicht aktiv, wird die zuletzt im Kanal angefahrne Position der Achse als aktuelle Position angenommen. Wurde diese Achse noch nicht im Kanal verfahren, wird die Position 0,0 angenommen.

Verhalten bei überlagerten Bewegungen

Überlagerte Bewegungen, die im Hauptlauf eingerechnet werden, können von der Satzaufbereitung bezüglich der aktiven Schutzbereiche nicht berücksichtigt werden.

Es erfolgen folgende Reaktionen:

- Alarm 10704 "Schutzbereichsüberwachung nicht gewährleistet"
- DB31, ... DBX39.0 = 1 (Schutzbereichsüberwachung nicht gewährleistet)

4.6 Beispiel

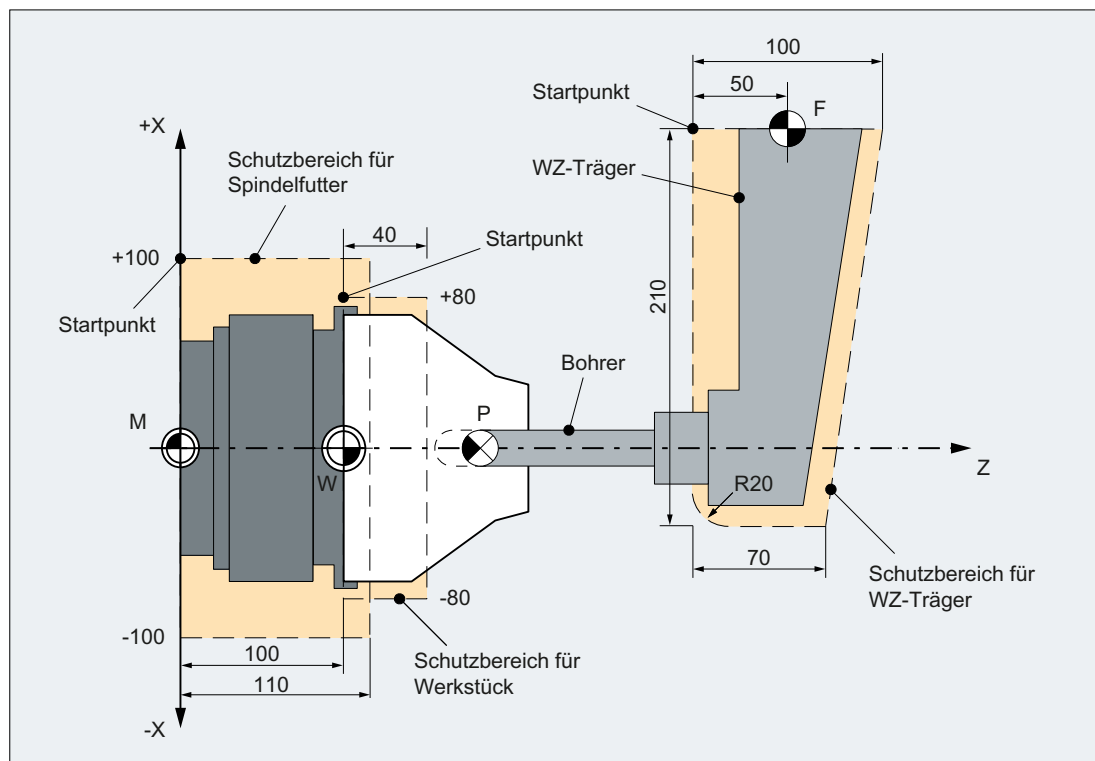
4.6.1 Schutzbereiche an einer Drehmaschine

Für eine Drehmaschine sollen folgende Innenschutzbereiche definiert werden:

- ein maschinenspezifischer und werkstückbezogener Schutzbereich für das Spindelfutter, ohne Begrenzung in der 3. Dimension
- ein kanalspezifischer Schutzbereich für das Werkstück, ohne Begrenzung in der 3. Dimension
- ein kanalspezifischer, werkzeugbezogener Schutzbereich für den Werkzeugträger, ohne Begrenzung in der 3. Dimension

Zur Definition des Schutzbereichs für das Werkstück wird der Werkstücknullpunkt in den Maschinennullpunkt gelegt.

Bei der Aktivierung wird der Schutzbereich um 100 mm in der Z-Achse in positiver Richtung verschoben.



4.6.2 Schutzbereichsdefinition im Teileprogramm

Teileprogramm Ausschnitt zur Schutzbereichsdefinition:

Programmcode	Kommentar
DEF INT AB	
G18	; Arbeitsebene ZX
NPROTDEF(1,FALSE,0,0,0)	; Definitionsbeginn: Schutzbereich für Spindelfutter
G01 X100 Z0	; Konturbeschreibung: Verfahrbewegung an die Kontur
G01 X-100 Z0	; Konturbeschreibung: 1. Konturelement
G01 X-100 Z110	; Konturbeschreibung: 2. Konturelement
G01 X100 Z110	; Konturbeschreibung: 3. Konturelement
G01 X100 Z0	; Konturbeschreibung: 4. Konturelement
EXECUTE(AB)	; Definitionsende: Schutzbereich für Spindelfutter
CPROTDEF(1,FALSE,0,0,0)	; Definitionsbeginn: Schutzbereich für Werkstück
G01 X80 Z0	; Konturbeschreibung: Verfahrbewegung an die Kontur
G01 X-80 Z0	; Konturbeschreibung: 1. Konturelement
G01 X-80 Z40	; Konturbeschreibung: 2. Konturelement
G01 X80 Z40	; Konturbeschreibung: 3. Konturelement
G01 X80 Z0	; Konturbeschreibung: 4. Konturelement

Programmcode	Kommentar
EXECUTE (AB)	; Definitionsende: Schutzbereich für Werkstück
CPROTDEF (2, TRUE, 0, 0, 0)	; Definitionsbeginn: Schutzbereich für Werkzeugträger
G01 X0 Z-50	; Konturbeschreibung: Verfahrbewegung an die Kontur
G01 X-190 Z-50	; Konturbeschreibung: 1. Konturelement
G03 X-210 Z-30 I-20	; Konturbeschreibung: 2. Konturelement
G01 X-210 Z20	; Konturbeschreibung: 3. Konturelement
G01 X0 Z50	; Konturbeschreibung: 4. Konturelement
G01 X0 Z-50	; Konturbeschreibung: 5. Konturelement
EXECUTE (AB)	; Definitionsende: Schutzbereich für Werkzeugträger

4.6.3 Schutzbereichsdefinition mit Systemvariablen

Maschinenspezifischer Schutzbereich für das Spindelfutter

Systemvariable	Wert	Beschreibung
\$SN_PA_ACTIV_IMMED[0]	0	Schutzbereich für Spindelfutter nicht sofort aktiv
\$SN_PA_T_W[0]	0	Schutzbereich für Spindelfutter ist werkstückbezogen
\$SN_PA_ORI[0]	1	Orientierung des Schutzbereichs: 1 = 3. und 1. Geometrieachse
\$SN_PA_LIM_3DIM[0]	0	Art der Begrenzung in der 3. Dimension: 0 = keine Begrenzung
\$SN_PA_PLUS_LIM[0]	0	Wert der Begrenzung in Plus-Richtung in der 3. Dimension
\$SN_PA_MINUS_LIM[0]	0	Wert der Begrenzung in Minus-Richtung in der 3. Dimension
\$SN_PA_CONT_NUM[0]	4	Anzahl der gültigen Konturelemente
\$SN_PA_CONT_TYP[0,0]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Spindelfutter, Konturelement 1
\$SN_PA_CONT_TYP[0,1]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Spindelfutter, Konturelement 2
\$SN_PA_CONT_TYP[0,2]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Spindelfutter, Konturelement 3
\$SN_PA_CONT_TYP[0,3]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Spindelfutter, Konturelement 4
\$SN_PA_CONT_TYP[0,4]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Spindelfutter, Konturelement 5
\$SN_PA_CONT_TYP[0,5]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Spindelfutter, Konturelement 6
\$SN_PA_CONT_TYP[0,6]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Spindelfutter, Konturelement 7
\$SN_PA_CONT_TYP[0,7]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Spindelfutter, Konturelement 8
\$SN_PA_CONT_TYP[0,8]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Spindelfutter, Konturelement 9
\$SN_PA_CONT_TYP[0,9]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Spindelfutter, Konturelement 10

Systemvariable	Wert	Beschreibung
\$SN_PA_CONT_ORD[0,0]	-100	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 1
\$SN_PA_CONT_ORD[0,1]	-100	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 2
\$SN_PA_CONT_ORD[0,2]	100	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 3
\$SN_PA_CONT_ORD[0,3]	100	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 4
\$SN_PA_CONT_ORD[0,4]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 5
\$SN_PA_CONT_ORD[0,5]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 6
\$SN_PA_CONT_ORD[0,6]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 7
\$SN_PA_CONT_ORD[0,7]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 8
\$SN_PA_CONT_ORD[0,8]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 9
\$SN_PA_CONT_ORD[0,9]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 10
\$SN_PA_CONT_ABS[0,0]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 1
\$SN_PA_CONT_ABS[0,1]	110	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 2
\$SN_PA_CONT_ABS[0,2]	110	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 3
\$SN_PA_CONT_ABS[0,3]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 4
\$SN_PA_CONT_ABS[0,4]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 5
\$SN_PA_CONT_ABS[0,5]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 6
\$SN_PA_CONT_ABS[0,6]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 7
\$SN_PA_CONT_ABS[0,7]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 8
\$SN_PA_CONT_ABS[0,8]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 9
\$SN_PA_CONT_ABS[0,9]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 10
\$SN_PA_CENT_ORD[0,0]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 1
\$SN_PA_CENT_ORD[0,1]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 2
\$SN_PA_CENT_ORD[0,2]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 3
\$SN_PA_CENT_ORD[0,3]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 4

Systemvariable	Wert	Beschreibung
\$SN_PA_CENT_ORD[0,4]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 5
\$SN_PA_CENT_ORD[0,5]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 6
\$SN_PA_CENT_ORD[0,6]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 7
\$SN_PA_CENT_ORD[0,7]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 8
\$SN_PA_CENT_ORD[0,8]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 9
\$SN_PA_CENT_ORD[0,9]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Spindelfutter, Konturelement 10
\$SN_PA_CENT_ABS[0,0]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 1
\$SN_PA_CENT_ABS[0,1]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 2
\$SN_PA_CENT_ABS[0,2]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 3
\$SN_PA_CENT_ABS[0,3]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 4
\$SN_PA_CENT_ABS[0,4]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 5
\$SN_PA_CENT_ABS[0,5]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 6
\$SN_PA_CENT_ABS[0,6]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 7
\$SN_PA_CENT_ABS[0,7]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 8
\$SN_PA_CENT_ABS[0,8]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 9
\$SN_PA_CENT_ABS[0,9]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Spindelfutter, Konturelement 10

Kanalspezifischer Schutzbereich für das Werkstück

Systemvariable	Wert	Anmerkung
\$SC_PA_ACTIV_IMMED[0]	0	Schutzbereich für Werkstück nicht sofort aktiv
\$SC_PA_TW[0]	0	Schutzbereich für Werkstück ist werkstückbezogen
\$SC_PA_ORI[0]	1	Orientierung des Schutzbereichs: 1 = 3. und 1. Geometrieachse
\$SC_PA_LIM_3DIM[0]	0	Art der Begrenzung in der 3. Dimension: 0 = keine Begrenzung
\$SC_PA_PLUS_LIM[0]	0	Wert der Begrenzung in Plus-Richtung in der 3. Dimension
\$SC_PA_MINUS_LIM[0]	0	Wert der Begrenzung in Minus-Richtung in der 3. Dimension
\$SC_PA_CONT_NUM[0]	4	Anzahl der gültigen Konturelemente
\$SC_PA_CONT_TYP[0,0]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkstück, Konturelement 1
\$SC_PA_CONT_TYP[0,1]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkstück, Konturelement 2

Systemvariable	Wert	Anmerkung
\$SC_PA_CONT_TYP[0,2]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkstück, Konturelement 3
\$SC_PA_CONT_TYP[0,3]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkstück, Konturelement 4
\$SC_PA_CONT_TYP[0,4]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkstück, Konturelement 5
\$SC_PA_CONT_TYP[0,5]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkstück, Konturelement 6
\$SC_PA_CONT_TYP[0,6]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkstück, Konturelement 7
\$SC_PA_CONT_TYP[0,7]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkstück, Konturelement 8
\$SC_PA_CONT_TYP[0,8]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkstück, Konturelement 9
\$SC_PA_CONT_TYP[0,9]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkstück, Konturelement 10
\$SC_PA_CONT_ORD[0,0]	-80	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 1
\$SC_PA_CONT_ORD[0,1]	-80	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 2
\$SC_PA_CONT_ORD[0,2]	80	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 3
\$SC_PA_CONT_ORD[0,3]	80	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 4
\$SC_PA_CONT_ORD[0,4]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 5
\$SC_PA_CONT_ORD[0,5]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 6
\$SC_PA_CONT_ORD[0,6]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 7
\$SC_PA_CONT_ORD[0,7]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 8
\$SC_PA_CONT_ORD[0,8]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 9
\$SC_PA_CONT_ORD[0,9]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 10
\$SC_PA_CONT_ABS[0,0]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 1
\$SC_PA_CONT_ABS[0,1]	40	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 2
\$SC_PA_CONT_ABS[0,2]	40	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 3
\$SC_PA_CONT_ABS[0,3]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 4
\$SC_PA_CONT_ABS[0,4]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 5
\$SC_PA_CONT_ABS[0,5]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 6

Systemvariable	Wert	Anmerkung
\$SC_PA_CONT_ABS[0,6]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 7
\$SC_PA_CONT_ABS[0,7]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 8
\$SC_PA_CONT_ABS[0,8]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 9
\$SC_PA_CONT_ABS[0,9]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 10
\$SC_PA_CENT_ORD[0,0]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 1
\$SC_PA_CENT_ORD[0,1]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 2
\$SC_PA_CENT_ORD[0,2]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 3
\$SC_PA_CENT_ORD[0,3]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 4
\$SC_PA_CENT_ORD[0,4]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 5
\$SC_PA_CENT_ORD[0,5]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 6
\$SC_PA_CENT_ORD[0,6]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 7
\$SC_PA_CENT_ORD[0,7]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 8
\$SC_PA_CENT_ORD[0,8]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkstück, Konturelement 9
\$SC_PA_CENT_ORD[0,9]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 10
\$SC_PA_CENT_ABS[0,0]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 1
\$SC_PA_CENT_ABS[0,1]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 2
\$SC_PA_CENT_ABS[0,2]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 3
\$SC_PA_CENT_ABS[0,3]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 4
\$SC_PA_CENT_ABS[0,4]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 5
\$SC_PA_CENT_ABS[0,5]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 6
\$SC_PA_CENT_ABS[0,6]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 7
\$SC_PA_CENT_ABS[0,7]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 8
\$SC_PA_CENT_ABS[0,8]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 9
\$SC_PA_CENT_ABS[0,9]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkstück, Konturelement 10

Kanalspezifischer Schutzbereich für den Werkzeugträger

Systemvariable	Wert	Anmerkung
\$SC_PA_ACTIV_IMMED[1]	0	Schutzbereich für Werkzeugträger nicht sofort aktiv
\$SC_PA_TW[1]	3	Schutzbereich für Werkzeugträger ist werkzeugbezogen
\$SC_PA_ORI[1]	1	Orientierung des Schutzbereichs: 1 = 3. und 1. Geometrieachse
\$SC_PA_LIM_3DIM[1]	0	Art der Begrenzung in der 3. Dimension: 0 = keine Begrenzung
\$SC_PA_PLUS_LIM[1]	0	Wert der Begrenzung in Plus-Richtung in der 3. Dimension
\$SC_PA_MINUS_LIM[1]	0	Wert der Begrenzung in Minus-Richtung in der 3. Dimension
\$SC_PA_CONT_NUM[1]	5	Anzahl der gültigen Konturelemente
\$SC_PA_CONT_TYP[1,0]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkzeugträger, Konturelement 1
\$SC_PA_CONT_TYP[1,1]	3	Konturtyp[<i>] : 3 = G3 für Kreiselement gegen den Uhrzeigersinn, Schutzbereich Werkzeugträger, Konturelement 2
\$SC_PA_CONT_TYP[1,2]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkzeugträger, Konturelement 3
\$SC_PA_CONT_TYP[1,3]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkzeugträger, Konturelement 4
\$SC_PA_CONT_TYP[1,4]	1	Konturtyp[<i>] : 1 = G1 für Gerade, Schutzbereich Werkzeugträger, Konturelement 5
\$SC_PA_CONT_TYP[1,5]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkzeugträger, Konturelement 6
\$SC_PA_CONT_TYP[1,6]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkzeugträger, Konturelement 7
\$SC_PA_CONT_TYP[1,7]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkzeugträger, Konturelement 8
\$SC_PA_CONT_TYP[1,8]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkzeugträger, Konturelement 9
\$SC_PA_CONT_TYP[1,9]	0	Konturtyp[<i>] : 0 = nicht definiert, Schutzbereich Werkzeugträger, Konturelement 10
\$SC_PA_CONT_ORD[1,0]	-190	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 1
\$SC_PA_CONT_ORD[1,1]	-210	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 2
\$SC_PA_CONT_ORD[1,2]	-210	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 3
\$SC_PA_CONT_ORD[1,3]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 4
\$SC_PA_CONT_ORD[1,4]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 5
\$SC_PA_CONT_ORD[1,5]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 6
\$SC_PA_CONT_ORD[1,6]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 7
\$SC_PA_CONT_ORD[1,7]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 8
\$SC_PA_CONT_ORD[1,8]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 9

Systemvariable	Wert	Anmerkung
\$SSC_PA_CONT_ORD[1,9]	0	Endpunkt der Kontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 10
\$SSC_PA_CONT_ABS[1,0]	-50	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 1
\$SSC_PA_CONT_ABS[1,1]	-30	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 2
\$SSC_PA_CONT_ABS[1,2]	20	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 3
\$SSC_PA_CONT_ABS[1,3]	50	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 4
\$SSC_PA_CONT_ABS[1,4]	-50	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 5
\$SSC_PA_CONT_ABS[1,5]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 6
\$SSC_PA_CONT_ABS[1,6]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 7
\$SSC_PA_CONT_ABS[1,7]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 8
\$SSC_PA_CONT_ABS[1,8]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 9
\$SSC_PA_CONT_ABS[1,9]	0	Endpunkt der Kontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 10
\$SSC_PA_CENT_ORD[1,0]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 1
\$SSC_PA_CENT_ORD[1,1]	-190	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 2
\$SSC_PA_CENT_ORD[1,2]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 3
\$SSC_PA_CENT_ORD[1,3]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 4
\$SSC_PA_CENT_ORD[1,4]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 5
\$SSC_PA_CENT_ORD[1,5]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 6
\$SSC_PA_CENT_ORD[1,6]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 7
\$SSC_PA_CENT_ORD[1,7]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 8
\$SSC_PA_CENT_ORD[1,8]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 9
\$SSC_PA_CENT_ORD[1,9]	0	Mittelpunkt der Kreiskontur[<i>], Ordinatenwert Schutzbereich Werkzeugträger, Konturelement 10
\$SSC_PA_CENT_ABS[1,0]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 1
\$SSC_PA_CENT_ABS[1,1]	-30	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 2
\$SSC_PA_CENT_ABS[1,2]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 3

Systemvariable	Wert	Anmerkung
\$SC_PA_CENT_ABS[1,3]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 4
\$SC_PA_CENT_ABS[1,4]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 5
\$SC_PA_CENT_ABS[1,5]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 6
\$SC_PA_CENT_ABS[1,6]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 7
\$SC_PA_CENT_ABS[1,7]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 8
\$SC_PA_CENT_ABS[1,8]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 9
\$SC_PA_CENT_ABS[1,9]	0	Mittelpunkt der Kreiskontur[<i>], Abszissenwert Schutzbereich Werkzeugträger, Konturelement 10

4.6.4 Aktivierung der Schutzbereiche

Teileprogrammausschnitt zur Aktivierung der Schutzbereiche für Spindelfutter, Werkstück und Werkzeugträger:

Programmcode	Kommentar
NPROT(1,2,0,0,0)	; Schutzbereich: Spindelfutter
CPROT(1,2,0,0,100)	; Schutzbereich: Werkstück mit Verschiebung um 100 mm in der Z-Achse.
CPROT(2,2,0,0,0)	; Schutzbereich: Werkzeugträger

4.7 Datenlisten

4.7.1 Maschinendaten

4.7.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10618	PROTAREA_GEOAX_CHANGE_MODE	Verhalten beim Transformationswechsel und Geometrieachstausch
18190	MM_NUM_PROTECT_AREA_NCK	Anzahl der verfügbaren maschinenspezifischen Schutzbereiche

4.7.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
28200	MM_NUM_PROTECT_AREA_CHAN (SRAM)	Anzahl der verfügbaren kanalspezifischen Schutzbereiche
28210	MM_NUM_PROTECT_AREA_ACTIVE	Maximale Anzahl der gleichzeitig im Kanal aktivierbaren Schutzbereiche
28212	MM_NUM_PROTECT_AREA_CONTUR	Maximale Anzahl definierbarer Konturelemente im Kanal

4.7.2 Signale

4.7.2.1 Signale an Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Schutzbereiche freigeben	DB21,DBX1.1	DB320x.DBX1.1
Vorschubsperr	DB21,DBX6.0	DB320x.DBX6.0
Maschinenspezifischen Schutzbereich 1 ... 8 aktivieren	DB21,DBX8.0 ... 7	DB320x.DBX8.0 ... 7
Maschinenspezifischen Schutzbereich 9 aktivieren	DB21,DBX9.0	DB320x.DBX9.0
Maschinenspezifischen Schutzbereich 10 aktivieren	DB21,DBX9.1	DB320x.DBX9.1
Kanalspezifischen Schutzbereich 1 ... 8 aktivieren	DB21,DBX10.0 ... 7	DB320x.DBX10.0 ... 7
Kanalspezifischen Schutzbereich 9 aktivieren	DB21,DBX11.0	DB320x.DBX11.0
Kanalspezifischen Schutzbereich 10 aktivieren	DB21,DBX11.1	DB320x.DBX11.1

4.7.2.2 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Maschinenspezifischer Schutzbereich 1 ... 8 voraktiviert	DB21,DBX272.0 ... 7	DB330x.DBX8.0 ... 7
Maschinenspezifischer Schutzbereich 9 voraktiviert	DB21,DBX273.0	DB330x.DBX9.0
Maschinenspezifischer Schutzbereich 10 voraktiviert	DB21,DBX273.1	DB330x.DBX9.1
Kanalspezifischer Schutzbereich 1 ... 8 voraktiviert	DB21,DBX274.0 ... 7	DB330x.DBX10.0 ... 7
Kanalspezifischer Schutzbereich 9 voraktiviert	DB21,DBX275.0	DB330x.DBX11.0
Kanalspezifischer Schutzbereich 10 voraktiviert	DB21,DBX275.1	DB330x.DBX11.1
Maschinenspezifischer Schutzbereich 1 ... 8 verletzt	DB21,DBX276.0 ... 7	DB330x.DBX12.0 ... 7
Maschinenspezifischer Schutzbereich 9 verletzt	DB21,DBX277.0	DB330x.DBX13.0
Maschinenspezifischer Schutzbereich 10 verletzt	DB21,DBX277.1	DB330x.DBX13.1
Kanalspezifischer Schutzbereich 1 ... 8 verletzt	DB21,DBX278.0 ... 7	DB330x.DBX14.0 ... 7
Kanalspezifischer Schutzbereich 9 verletzt	DB21,DBX279.0	DB330x.DBX15.0
Kanalspezifischer Schutzbereich 10 verletzt	DB21,DBX279.1	DB330x.DBX15.1

B1: Bahnsteuerbetrieb, Genauhalt, LookAhead

5.1 Kurzbeschreibung

Genauhalt oder Genauhaltbetrieb

Genauhaltbetrieb ist ein Verfahrensmodus, bei dem am Ende eines jeden Verfahrssatzes alle an der Verfahrbewegung beteiligten Achsen (außer Achsen von satzübergreifenden Verfahrbewegungen) bis zum Stillstand abgebremst werden. Der Satzwechsel zum nachfolgenden Verfahrssatz erfolgt erst, wenn alle an der Verfahrbewegung beteiligten Achsen ihre programmierte Zielposition in Abhängigkeit des gewählten Genauhaltkriteriums erreicht haben.

Bahnsteuerbetrieb

Bahnsteuerbetrieb ist ein Verfahrensmodus, bei dem die NC versucht, die programmierte Bahngeschwindigkeit möglichst konstant beizubehalten. Insbesondere soll dabei das Abbremsen der Bahnachsen an den Satzgrenzen des Teileprogramms vermieden werden.

LookAhead

LookAhead ist eine Funktion zur Optimierung des Bahnsteuerbetriebs.

Um an Werkstücken hohe Oberflächengüten zu erzielen, ist eine gleichmäßige Bearbeitung notwendig. Deshalb sollte es während der Bearbeitung möglichst nicht zu Schwankungen der Bahngeschwindigkeit kommen. Ohne LookAhead betrachtet die NC zur Ermittlung der möglichen Bahngeschwindigkeit nur den unmittelbar auf den aktuellen Verfahrssatz folgenden Verfahrssatz. Beinhaltet der folgende Verfahrssatz nur einen kurzen Verfahrweg, muss die NC die Bahngeschwindigkeit vermindern (Bremsen im aktuellen Verfahrssatz), um eventuell rechtzeitig am Satzende des Folgesatzes anhalten zu können.

Durch "Vorausschauen" über eine parametrierbare Anzahl von dem aktuellen Verfahrssatz folgenden Verfahrssätzen kann mit LookAhead eine unter Umständen wesentlich höhere Bahngeschwindigkeit erzielt werden, da der NC jetzt wesentlich mehr Verfahrssätze bzw. Verfahrweg zur Berechnung zur Verfügung steht.

Dadurch ergeben sich folgende Vorteile:

- Bearbeitung mit durchschnittlich höherer Bahngeschwindigkeit
- Verbesserung der Oberflächengüte durch Vermeidung von Brems- und Beschleunigungsvorgängen

Glättung der Bahngeschwindigkeit

"Glättung der Bahngeschwindigkeit" ist eine Funktion speziell für Anwendungen, die eine möglichst gleichmäßige Bahngeschwindigkeit erfordern (z. B. Hochgeschwindigkeitsfräsen im Formenbau). Dazu wird bei der Glättung der Bahngeschwindigkeit auf Brems- und Beschleunigungsvorgänge verzichtet, die zu hochfrequenten Anregungen von Maschinenresonanzen führen würden.

Dadurch ergeben sich folgende Vorteile:

- Verbesserung der Oberflächengüte und der Stückzeit durch Vermeidung von Anregungen von Maschinenresonanzen
- Konstanter Verlauf der Bahngeschwindigkeit bzw. Schnittgeschwindigkeiten durch Vermeidung von "überflüssigen" Beschleunigungsvorgängen, d. h. Beschleunigungsvorgänge, die keinen großen Gewinn hinsichtlich der Programmlaufzeit bewirken.

Anpassung der Bahndynamik

Die "Anpassung der Bahndynamik" ist neben der "Glättung der Bahngeschwindigkeit" eine weitere Funktion zur Vermeidung hochfrequenter Anregungen von Maschinenresonanzen bei gleichzeitiger Optimierung der Bahndynamik. Dazu werden hochfrequente Änderungen der Bahngeschwindigkeit automatisch mit geringeren Ruck- oder Beschleunigungswerten als den in den Maschinendaten parametrisierten Dynamikgrenzwerten durchgeführt.

Somit wirken bei niederfrequenten Änderungen der Bahngeschwindigkeit die vollen Dynamikgrenzwerte, während bei hochfrequenten Änderungen durch die automatische Dynamikanpassung nur die reduzierten Dynamikgrenzwerte wirken.

Dynamikmodus für Bahninterpolation

Zur Optimierung der Bahndynamik zählen auch die Technologie-spezifischen Dynamikeinstellungen, die für unterschiedliche Bearbeitungstechnologien (u. a. Gewindebohren, Schruppen, Schlichten) voreingestellt sind und im Teileprogramm durch Aufruf des jeweiligen Dynamikmodus aktiviert werden können.

Freiformflächenmodus

Jegliche Schwankung der Krümmung bzw. Torsion führt zu einer Veränderung der Bahngeschwindigkeit. Dadurch erhält man in der Regel bei der Bearbeitung von Freiformflächen-Werkstücken unnötige Brems- und Beschleunigungsvorgänge, die sich negativ auf die Qualität der Werkstückoberflächen auswirken können.

Für die Freiformflächenbearbeitung stehen daher folgende Funktionen zur Verfügung:

- Funktion "Freiformflächenmodus: Grundfunktionen"
Damit wird die Festlegung des Verlaufs der Bahngeschwindigkeit "unempfindlicher" gegenüber Schwankungen der Krümmung und Torsion.
- Funktion "Freiformflächenmodus: Erweiterungsfunktion"
Diese Erweiterung der LookAhead-Standardfunktionalität dient zur Berechnung des Bahngeschwindigkeitsprofils bei der Freiformflächenbearbeitung.

Die Vorteile des Freiformflächenmodus liegen in einer gleichmäßigeren Werkstückoberfläche und einer geringeren Belastung der Maschine.

Kompression von Linearsätzen

Nach Abschluss der Konstruktion eines Werkstücks mit einem CAD/CAM-System übernimmt dieses gewöhnlich auch die Generierung des entsprechenden Teileprogramms zur Erzeugung der Werkstückoberfläche. Dabei verwenden die meisten CAD/CAM-Systeme zur Beschreibung auch gekrümmter Abschnitte der Werkstückoberfläche Linearsätze. Zur Einhaltung der erforderlichen Konturgenauigkeit sind dabei gewöhnlich sehr viele Stützstellen notwendig. Demzufolge ergeben sich viele lineare Verfahrssätze mit meist sehr kurzen Verfahrwegen.

Durch den Einsatz einer "Kompressor-Funktion" erfolgt eine nachträgliche Annäherung an die durch die Linearsätze vorgegebene Kontur durch Polynom-Sätze. Dabei wird eine parametrierbare Anzahl von Linearsätzen durch einen Polynom-Satz ersetzt.

Vorteile:

- Reduzierung der Anzahl von Verfahrssätzen
- Erhöhung der Bahngeschwindigkeit
- Erhöhung der Oberflächengüte
- Stetige Satzübergänge

Kompression kurzer Spline-Sätze

Ein Spline definiert eine Kurve, die aus Polynomen 2. oder 3. Grades zusammengesetzt wird. Mit der Spline-Interpolation kann die Steuerung aus nur wenigen vorgegebenen Stützpunkten einer Sollkontur einen glatten Kurvenverlauf erzeugen.

Die Vorteile der Spline-Interpolation gegenüber der Linear-Interpolation sind:

- Reduzierung der Anzahl von benötigten Teileprogrammsätzen zur Beschreibung einer gekrümmten Kontur.
- Weicher, Mechanik-schonender Kurvenverlauf auch beim Übergang zwischen den Teileprogrammsätzen.

Die Nachteile der Spline-Interpolation gegenüber der Linear-Interpolation sind:

- Für eine Spline-Kurve kann kein exakter Kurvenverlauf, sondern nur ein Toleranzband vorgegeben werden, innerhalb dessen die Spline-Kurve liegen soll.

Wie bei der Linear-Interpolation können auch bei der Aufbereitung von Splines so kurze Sätze entstehen, dass zur Interpolation dieser Spline-Sätze die Bahngeschwindigkeit reduziert werden muss. Dies ist auch dann der Fall, wenn der Spline eigentlich eine lange, glatte Kurve darstellt. Mit der Funktion "Kompression kurzer Spline-Sätze" können die Spline-Sätze so zusammengefasst werden, dass die resultierende Satzlänge ausreichend groß ist und nicht zu einer Verringerung der Bahngeschwindigkeit führt.

5.2 Genauhaltbetrieb

Genauhalt oder Genauhaltbetrieb

Genauhalt oder Genauhaltbetrieb ist ein Verfahrensmodus, bei dem am Ende eines jeden Verfahrssatzes alle an der Verfahrbewegung beteiligten Bahnachsen und Zusatzachsen, die nicht satzübergreifend verfahren, zum Satzende bis zum Stillstand abgebremst werden. Der Satzwechsel zum nachfolgenden Verfahrssatz erfolgt erst, wenn alle an der Verfahrbewegung beteiligten Achsen ihre programmierte Zielposition in Abhängigkeit der gewählten Genauhaltbedingung erreicht haben.

Daraus ergibt sich folgendes Verhalten:

- Durch das Abbremsen der Achsen und die Wartezeit bis zum Erreichen des Zustandes "Genauhalt" für alle beteiligten Maschinenachsen verlängert sich die Programmlaufzeit gegenüber dem Bahnsteuerbetrieb erheblich.
- Bei Genauhaltbetrieb während der Bearbeitung kann es zu Freischneidmarken auf der Werkstückoberfläche kommen.

Genauhaltbedingung

Folgende Genauhaltbedingung können eingestellt werden:

- "Genauhalt grob"
- "Genauhalt fein"
- "Interpolator-Ende"

Anwendung

Der Genauhaltbetrieb ist immer dann zu verwenden, wenn die programmierte Kontur exakt abgefahren werden muss.

Aktivierung

In einem Programm kann der Genauhaltbetrieb über folgende Befehle modal oder satzweise vorgegeben werden:

Befehl	Bedeutung
G60	Der Genauhaltbetrieb ist ab dem aktuellen Satz modal wirksam.
G9	Der Genauhaltbetrieb ist im aktuellen Satz wirksam.

Genauhaltbedingung "Genauhalt grob" und "Genauhalt fein"

Die Genauhaltbedingung "grob" bzw. "fein" ist von einer Maschinenachse erfüllt, wenn ihr aktueller Schleppabstand kleiner bzw. gleich dem im Maschinendatum parametrisierten Toleranzbereich um die Sollposition ist:

- MD36000 \$MA_STOP_LIMIT_COARSE (Genauhalt grob)
- MD36010 \$MA_STOP_LIMIT_FINE (Genauhalt fein)

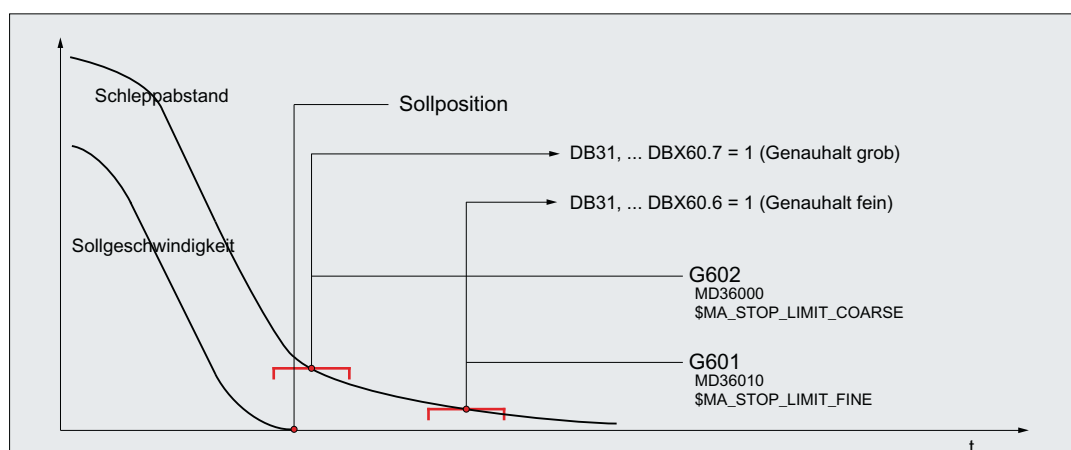


Bild 5-1 Toleranzfenster der Genauhaltbedingungen

Hinweis

Die Toleranzfenster der Genauhaltbedingungen "Genauhalt grob" und "Genauhalt fein" sollten so parametrisiert werden, dass folgende Forderung erfüllt ist:

"Genauhalt grob" > "Genauhalt fein"

Genauhaltbedingung "Interpolator-Ende"

Bei Genauhaltbedingung "Interpolator-Ende" erfolgt der Satzwechsel zum nachfolgenden Verfahrssatz, sobald alle an der Verfahrbewegung beteiligten Bahnachsen und Zusatzachsen, die nicht satzübergreifend verfahren, Sollwertbezogen ihre im Verfahrssatz programmierte Position erreicht haben. D. h., der Interpolator hat den Verfahrssatz abgefahren.

Die Istposition bzw. der Schleppabstand der beteiligten Maschinenachsen wird bei Genauhaltbedingung "Interpolator-Ende" nicht betrachtet. Dadurch kann es, abhängig von der Dynamik der Maschinenachsen, zu einem im Vergleich zu den Genauhaltbedingungen "Genauhalt grob" und "Genauhalt fein" größeren Verschleifen der Kontur an den Satzübergängen kommen.

Aktivierung der programmierbaren Genauhaltbedingungen

Die programmierbaren Genauhaltbedingungen werden durch folgende Befehle aktiviert:

Befehl	Genauhaltbedingung
G601	Genauhalt fein
G602	Genauhalt grob
G603	Interpolator-Ende

Satzwechsel in Abhängigkeit der aktiven Genauhaltbedingung

Das nachfolgende Bild veranschaulicht den Zeitpunkt des Satzwechsels in Abhängigkeit von der gewählten Genauhaltbedingung.

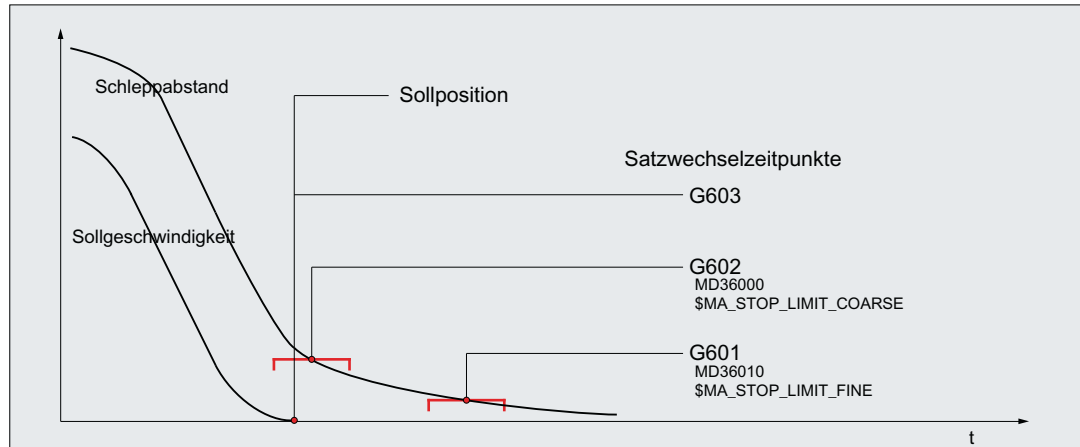


Bild 5-2 Satzwechsel in Abhängigkeit der aktiven Genauhaltbedingung

Bewertungsfaktor für Genauhaltbedingungen

Eine Parametersatz-abhängige Bewertung der Genauhaltbedingungen kann über das folgende achsspezifische Maschinendatum vorgegeben werden:

MD36012 \$MA_STOP_LIMIT_FACTOR[<Parametersatz>] = <Wert>

Mit dem Bewertungsfaktor werden die Werte folgender Maschinendaten beaufschlagt:

- MD36000 \$MA_STOP_LIMIT_COARSE
- MD36010 \$MA_STOP_LIMIT_FINE
- MD36030 \$MA_STANDSTILL_POS_TOL

Anwendungsbeispiele

- Anpassen des Positionierverhaltens an veränderte Massenverhältnisse, z. B. nach Getriebeumschaltung.
- Verkürzen der Positionierzeit in Abhängigkeit von verschiedenen Bearbeitungszuständen, z. B. Schruppen, Schlichten.

Parametrierbare Vorgabe der wirksamen Genauhaltbedingungen

Die Genauhaltbedingungen für die Befehle der 1. G-Gruppe können fest vorgegeben werden. Programmierte Genauhaltbedingungen werden dadurch unwirksam.

Die Genauhaltbedingungen können **unabhängig voneinander** für folgende Befehle eingestellt werden:

- Eilgang G0
- Alle anderen Befehle der 1. G-Gruppe

Das Einstellen der Genauhaltbedingung erfolgt kanalspezifisch über das nachfolgend dezimalcodierte Maschinendatum:

MD20550 \$MC_EXACT_POS_MODE = <Z><E>

Z	E	Wirksame Genauhaltbedingung
0	0	Programmierte Genauhaltbedingung
1	1	G601 (Genauhaltfenster fein)
2	2	G602 (Genauhaltfenster grob)
3	3	G603 (Interpolator-Ende)
E (Einerstelle): Einstellen der Genauhaltbedingung für Eilgang.		
Z (Zehnerstelle): Einstellen der Genauhaltbedingung für alle anderen Befehle der 1. G-Gruppe.		

Beispiel

MD20550 \$MC_EXACT_POS_MODE = 02

- <E> = 2: Beim Verfahren mit Eilgang wirkt unabhängig von der Programmierung immer Genauhaltbedingung G602 (Genauhaltfenster grob).
- <Z> = 0: Beim Verfahren mit allen anderen Befehlen der 1. G-Gruppe wirkt die programmierte Genauhaltbedingung.

Satzwechselverhalten bei Satzübergängen von G0 ↔ Nicht-G0 im Bahnsteuerbetrieb

Bei Bahnsteuerbetrieb kann das Satzwechselverhalten zwischen Eilgang- und Nicht-Eilgangsätzen (G0 ↔ Nicht-G0) über das nachfolgende Maschinendatum eingestellt werden:

MD20552 \$MC_EXACT_POS_MODE_G0_TO_G1 = <Wert>

<Wert>	Bedeutung
0	Kein zusätzlicher Stopp am Satzübergang.
1	Stopp am Satzübergang. Das Verhalten entspricht G601 (Genauhaltfenster fein)
2	Stopp am Satzübergang. Das Verhalten entspricht G602 (Genauhaltfenster grob)
3	Stopp am Satzübergang. Das Verhalten entspricht G603 (Interpolator-Ende)
4	Kein Stopp am Satzübergang. Bei Bahnsteuerbetrieb wird bei Satzwechseln von G0 → Nicht-G0 im G0-Satz vorausschauend der aktuelle Wert der Vorschubkorrektur des nachfolgenden Nicht-G0-Satzes berücksichtigt. Abhängig von der Achsdynamik und der Bahnlänge des aktuellen Satzes erfolgt der Satzwechsel mit der exakten bzw. bestmöglich angepassten Geschwindigkeit des Folgesatzes.
5	Kein Stopp am Satzübergang. Bei Bahnsteuerbetrieb wird bei Satzwechseln von G0 → Nicht-G0 und Nicht-G0 → G0 vorausschauend der aktuelle Wert der Vorschubkorrektur (G0 → Nicht-G0) bzw. der Eilgangkorrektur (Nicht-G0 → G0) des nachfolgenden Satzes berücksichtigt. Abhängig von der Achsdynamik und der Bahnlänge des aktuellen Satzes erfolgt der Satzwechsel mit der exakten bzw. bestmöglich angepassten Geschwindigkeit des Folgesatzes.

5.3 Bahnsteuerbetrieb

5.3.1 Allgemeine Funktionalität

Bahnsteuerbetrieb

Im Bahnsteuerbetrieb wird die Bahngeschwindigkeit am Satzende zum Satzwechsel **nicht** auf eine Geschwindigkeit abgebremst, die ein Erreichen des Genauhaltkriteriums ermöglicht. Ziel ist dagegen, ein größeres Abbremsen der Bahnachsen am Satzwechselpunkt zu vermeiden, um mit möglichst gleicher Bahngeschwindigkeit in den nächsten Satz zu wechseln. Um dieses Ziel zu erreichen, wird mit Anwahl des Bahnsteuerbetriebs zusätzlich die Funktion "LookAhead" aktiviert.

Bahnsteuerbetrieb bewirkt, dass knickförmige Satzübergänge durch lokale Änderungen des programmierten Verlaufs tangential gestaltet bzw. geglättet werden. Das Ausmaß der Änderung relativ zum programmierten Verlauf kann durch die Kriterien des Überlastfaktors oder Überschleifens eingegrenzt werden.

Bahnsteuerbetrieb bewirkt:

- eine Verrundung der Kontur.
- kürzere Bearbeitungszeiten durch fehlende Brems- und Beschleunigungsvorgänge, die für das Erreichen des Genauhaltkriteriums benötigt werden.
- bessere Schnittbedingungen durch den gleichmäßigeren Geschwindigkeitsverlauf.

Bahnsteuerbetrieb ist sinnvoll, wenn:

- eine Kontur möglichst zügig abgefahren werden soll (z. B. mit Eilgang).
- der exakte Verlauf im Rahmen eines Fehlerkriteriums vom programmierten abweichen darf, um einen durchgehend stetigen Verlauf zu erzeugen.

Bahnsteuerbetrieb ist nicht sinnvoll, wenn:

- eine Kontur exakt abgefahren werden soll.
- absolute Geschwindigkeitskonstanz erforderlich ist.

Impliziter Genauhalt

In einigen Fällen muss im Bahnsteuerbetrieb ein Genauhalt erzeugt werden, um Folgeaktionen ausführen zu können. In diesen Situationen wird die Bahngeschwindigkeit auf Null abgebremst.

- Werden Hilfsfunktionen vor der Verfahrbewegung ausgegeben, so wird der vorhergehende Satz erst mit dem Erreichen des angewählten Genauhaltkriteriums beendet.
- Werden Hilfsfunktionen nach der Verfahrbewegung ausgegeben, so werden diese nach Interpolator-Ende des Satzes ausgegeben.
- Enthält ein ausführbarer Satz (z. B. Starten einer Positionierachse) keine Verfahrinformation für die Bahnachsen, so wird der vorhergehende Satz mit dem Erreichen des angewählten Genauhaltkriteriums beendet.

- Wird eine Positionierachse zur Geometrieachse deklariert, so wird mit Programmierung der Geometrieachse der vorhergehende Satz mit Interpolator-Ende abgeschlossen.
- Wird eine Synchronachse programmiert, die zuletzt als Positionierachse oder als Spindel programmiert war (Grundstellung der Zusatzachse ist Positionierachse), so wird der vorhergehende Satz mit Interpolator-Ende beendet.
- Wird die Transformation gewechselt, so wird der zuvor bearbeitete Satz mit dem aktiven Genauhaltkriterium abgeschlossen.
- Ein Satz wird mit Interpolator-Ende beendet, falls im Folgesatz die Umschaltung des Beschleunigungsprofils BRISK/SOFT erfolgt (siehe Kapitel "B2: Beschleunigung (Seite 265)").
- Wird im Teileprogramm die Funktion "Zwischenspeicher leeren" programmiert, so wird der vorhergehende Satz mit dem Erreichen des angewählten Genauhaltkriteriums beendet.

Geschwindigkeit = 0 im Bahnsteuerbetrieb

Unabhängig vom impliziten Genauhalt wird die Bahnbewegung am Satzende auf Geschwindigkeit Null abgebremst, wenn:

- Positionierachsen mit der Anweisung `POS` programmiert sind und deren Verfahrzeit länger als die Verfahrzeit der Bahnachsen ist. Der Satzwechsel erfolgt mit Erreichen des "Genauhalt fein" der Positionierachse.
- die Zeit für die Positionierung einer Spindel, die mit der Anweisung `SPOS` programmiert wurde, länger dauert als die Verfahrzeit der Bahnachsen. Der Satzwechsel erfolgt mit Erreichen des "Genauhalt fein" der positionierenden Spindel.
- im aktuellen Satz Geometrieachsen verfahren und im folgenden Satz stattdessen Synchronachsen verfahren, bzw. im aktuellen Satz Synchronachsen verfahren und im nächsten Satz stattdessen Geometrieachsen verfahren werden.
- eine Synchronisation erforderlich wird.

5.3.2 Geschwindigkeitsabsenkung gemäß Überlastfaktor

Funktion

Die Funktion senkt im Bahnsteuerbetrieb die Bahngeschwindigkeit soweit ab, dass unter Wahrung der Beschleunigungsgrenze und unter Berücksichtigung eines Überlastfaktors der nichttangentialer Satzübergang in einem Interpolatortakt überfahren werden kann.

Mit der Geschwindigkeitsabsenkung werden bei nichttangentialem Konturverlauf am Satzübergang axiale Geschwindigkeitssprünge erzeugt. Diese werden auch von den mitfahrenden Synchronachsen nachvollzogen. Mit dem Geschwindigkeitssprung wird vermieden, dass die Bahngeschwindigkeit zu null abgesenkt wird. Der Sprung wird dann ausgeführt, wenn die axiale Geschwindigkeit mit der Achsbeschleunigung auf eine Geschwindigkeit reduziert wurde, ab der dann mit dem Sprung auf den neuen Sollwert gelangt werden kann. Die Sprunghöhe des Sollwerts kann mittels des Kriteriums Überlastfaktor begrenzt werden. Da die Sprunghöhe achsbezogen ist, wird am Satzübergang die kleinste Sprunghöhe der beim Satzwechsel aktiven Bahnachsen berücksichtigt.

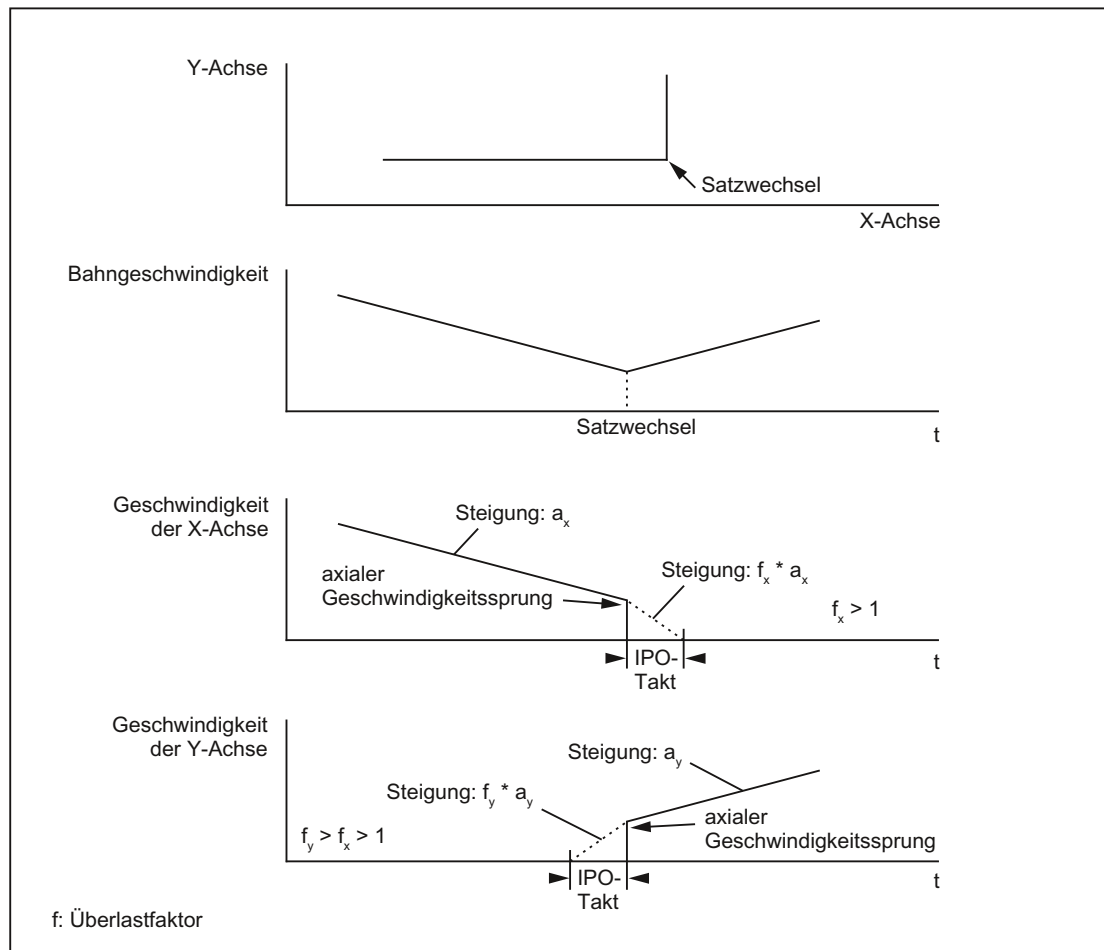


Bild 5-3 Axiale Geschwindigkeitsänderung am Satzübergang

Bei nahezu tangentialem Satzübergang wird die Bahngeschwindigkeit nicht abgesenkt, falls die zulässigen Achsbeschleunigungen nicht überschritten werden. Damit wird erreicht, dass sehr kleine Knicke in der Kontur (z. B. $0,5^\circ$) direkt überfahren werden.

Überlastfaktor

Der Überlastfaktor begrenzt den Geschwindigkeitssprung der Maschinenachse am Satzübergang. Damit der Geschwindigkeitssprung die Achsbelastbarkeit nicht überschreitet, wird der Sprung aus der Beschleunigung der Achse abgeleitet.

Der Überlastfaktor gibt an, um welches Maß die Beschleunigung der Maschinenachse (MD32300 \$MA_MAX_AX_ACCEL) für einen IPO-Takt überschritten werden darf.

Der Geschwindigkeitshub ergibt sich wie folgt:

Geschwindigkeitshub = Achsbeschleunigung * (Überlastfaktor - 1) * Interpolator-Takt

Der Überlastfaktor ist hinterlegt im Maschinendatum:

MD32310 \$MA_MAX_ACCEL_OVL_FACTOR (Überlastfaktor für axiale Geschwindigkeitssprünge)

Faktor 1.0 bedeutet, dass nur tangentielle Übergänge mit endlicher Geschwindigkeit überfahren werden können. Bei allen anderen Übergängen wird sollwertseitig auf Geschwindigkeit 0 abgebremst. Das Verhalten entspricht der Funktionalität "Genauhalt mit Interpolator-Ende". Da dies für den Bahnsteuerbetrieb nicht erwünscht ist, ist der Faktor größer 1.0 einzustellen.

Hinweis

Für die Inbetriebnahme ist zu beachten, dass der Faktor dann herabzusetzen ist, wenn die Maschine bei knickförmigem Satzübergang zu Schwingungen angeregt wird und Überschleifen nicht verwendet werden soll.

Durch Setzen des folgenden Maschinendatums werden Satzübergänge unabhängig vom eingestelltem Überlastfaktor mit G641 / G642 überschleifen:

MD20490 \$MC_IGNORE_OVL_FACTOR_FOR_ADIS

Aktivierung / Deaktivierung

Der Bahnsteuerbetrieb mit Geschwindigkeitsabsenkung gemäß Überlastfaktor kann in jedem NC-Teileprogrammsatz durch den modal wirksamen Befehl G64 aktiviert werden.

Eine Unterbrechung ist möglich durch Anwahl des satzweise wirksamen Genauhalts G9.

Der Bahnsteuerbetrieb G64 kann deaktiviert werden durch Anwahl von:

- Modalen Genauhalt G60
- Überschleifen G641, G642, G643, G644 oder G645

Impliziter Bahnsteuerbetrieb

Können im Bahnsteuerbetrieb mit Überschleifen G641 wegen sehr kurzer Weglängen der Sätze (z. B. nulltaktige Sätze) keine Überschleifsätze eingefügt werden, wird auf Bahnsteuerbetrieb G64 umgeschaltet.

5.3.3 Überschleifen**Funktion**

Die Funktion "Überschleifen" fügt entlang einer programmierten Kontur (Bahnachsen) an nicht stetigen (knickförmigen) Satzübergängen Zwischensätze (Überschleifsätze) ein, sodass der sich daraus ergebende neue Satzübergang stetig (tangential) verläuft.

Synchronachsen

Überschleifen berücksichtigt neben den Geometrie- auch alle Synchronachsen. Allerdings kann bei parallelen Verfahren von Bahn- und Synchronachsen nicht für beide Achstypen gleichzeitig ein stetiger Satzübergang erzeugt werden. Für Synchronachsen wird dann zu Gunsten der Bahnachsen, die immer exakt verfahren werden, nur ein annähernd stetiger Satzübergang erzeugt.

Überschleifen bei G64

Überschleifen erfolgt auch, wenn zum Einhalten der Dynamikgrenzen am Satzübergang eine Geschwindigkeit erforderlich wird, welche bei G64 die zulässige Geschwindigkeit am Satzübergang unterschreitet. Siehe Kapitel "Geschwindigkeitsabsenkung gemäß Überlastfaktor (Seite 193)" Absatz "Überlastfaktor".

Auswirkung auf Synchronisationsbedingungen

Durch Überschleifen werden die programmierten Sätze, zwischen denen der oder die Überschleifsätze eingefügt werden, verkürzt. Die programmierte Satzgrenze verschwindet dabei und steht dann für etwaige Synchronisierungsbedingungen (z. B. Hilfsfunktionsausgabe parallel zur Bewegung, Stopp am Satzende) nicht mehr als Kriterium zur Verfügung.

Hinweis

Es wird empfohlen, bei Verwendung der Funktion "Überschleifen" Synchronisierungsbedingungen auf das Satzende des Satzes vor der Überschleifstelle zu beziehen, nicht auf das Ende des eingefügten Überschleifsatzes. Der Folgesatz wurde dann noch nicht begonnen und bei einem Stopp am Satzende kann die Kontur des Folgesatzes noch manuell geändert werden.

Ausnahmen

In folgenden Fällen erfolgt am Satzübergang, hier beispielhaft zwischen den Sätzen N10 nach N20, **kein** Überschleifen, d.h. es wird kein Überschleifsatz eingefügt:

Implizites Anhalten der Verfahrbewegung

Mögliche Ursachen:

- Hilfsfunktionsausgabe vor der Verfahrbewegung von N20 aktiv
- N20 enthält keine Verfahrbewegung von Bahnachsen
- In N20 verfährt zum ersten Mal eine Achse als Bahnachse, die zuvor Positionierachse war
- In N20 verfährt zum ersten Mal eine Achse als Positionierachse, die zuvor Bahnachse war
- In N10 werden Geometrieachsen verfahren, in N20 nicht
- In N20 werden Geometrieachsen verfahren, in N10 nicht
- Aktivierung von Gewindeschneiden G33 in N20
- Wechsel von BRISK und SOFT
- Transformationsrelevante Achsen sind nicht vollständig der Bahnbewegung zugeordnet (z. B. Pendelachsen, Positionierachsen).

Das Einfügen des Überschleifsatzes würde die Teileprogrammbearbeitung überproportional verlangsamen

Mögliche Ursachen:

- Ein Programm bzw. Programmabschnitt besteht aus einer Vielzahl sehr kurzer Verfahrssätze (≈ 1 Interpolatortakt / Verfahrssatz; da jeder Verfahrssatz mindestens einen Interpolatortakt benötigt, würden die eingefügten Zwischensätze die Bearbeitungszeit nahezu verdoppeln)
- G64 (Bahnsteuerbetrieb ohne Überschleifen) ohne Geschwindigkeitsreduzierung beim Satzwechsel aktiv
- Der parametrierte Überlastfaktor (MD32310 \$MA_MAX_ACCEL_OVL_FACTOR) erlaubt das Verfahren der programmierten Kontur ohne dass die Bahngeschwindigkeit reduziert werden muss. Siehe dazu auch: MD20490 \$MC_IGNORE_OVL_FACTOR_FOR_ADIS

Bahnparameter verhindern das Überschleifen

Mögliche Ursachen:

- G641 (Bahnsteuerbetrieb mit Überschleifen nach Wegkriterium) ist aktiv, aber Eilgang ist aktiv (G0) UND ADISPOS == 0 (Überschleifabstand bei G0)
- G641 (Bahnsteuerbetrieb mit Überschleifen nach Wegkriterium) ist aktiv, aber Eilgang ist **nicht** aktiv UND ADIS == 0 (Überschleifabstand für Bahnfunktionen G1, G2, G3, ...)
- G642 oder G643 (Bahnsteuerbetrieb mit Überschleifen unter Einhaltung definierter Toleranzen) ist aktiv, aber alle Toleranzen == Null

N10 oder N20 enthält keine Verfahrbewegung (Nullsatz)

Normalerweise werden keine Nullsätze erzeugt. Ausnahmen:

- Aktive Synchronaktion
- Programmsprünge

Auswirkung auf Synchronisationsbedingungen

Beim Überschleifen werden die programmierten Sätze, zwischen denen die Überschleifkontur eingefügt wird, verkürzt. Die ursprünglich programmierte Satzgrenze verschwindet dabei und steht dann für etwaige Synchronisierungsbedingungen (z. B. Hilfsfunktionsausgabe parallel zur Bewegung, Stopp am Satzende) nicht mehr zur Verfügung.

Hinweis

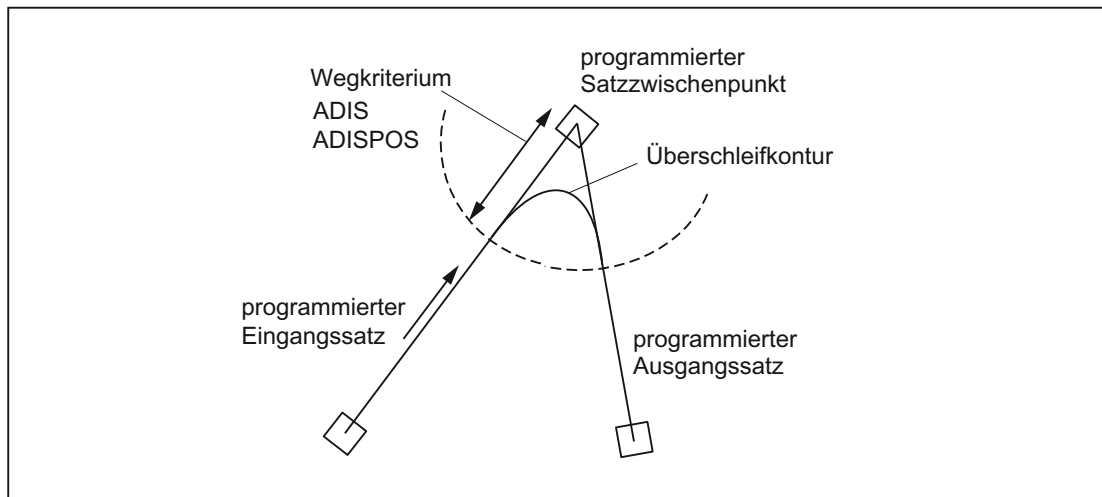
Es wird empfohlen, bei Verwendung der Funktion "Überschleifen" Synchronisierungsbedingungen auf das Satzende des Satzes vor der Überschleifstelle zu beziehen, nicht auf das Ende des eingefügten Überschleifsatzes. Der Folgesatz wurde dann noch nicht begonnen und bei einem Stopp auf Satzende kann die Kontur des Folgesatzes noch geändert werden.

5.3.3.1 Überschleifen nach Wegkriterium (G641)

Funktion

Beim Bahnsteuerbetrieb mit Überschleifen nach Wegkriterium wird die Größe des Verrundungsbereichs durch die Wegkriterien ADIS und ADISPOS beeinflusst.

Die Wegkriterien ADIS und ADISPOS beschreiben die Strecke, die der Überschleifsatz vor dem Satzende frühestens beginnen darf, bzw. die Strecke nach Satzende, in der der Überschleifsatz beendet sein muss.



Hinweis

Spitze Ecken erzeugen stark gekrümmte Überschleifkurven und haben damit auch eine entsprechende Geschwindigkeitsreduzierung zur Folge.

Hinweis

ADISPOS wird wie ADIS gehandhabt, ist aber speziell nur für die Bewegungsart Eilgang G0 einzusetzen.

Wirksamkeit des Wegkriteriums

- ADIS bzw. ADISPOS müssen programmiert werden. Ist die Voreinstellung "Null" verhält sich G641 wie G64.
- Sind nicht beide aufeinanderfolgende Sätze Eilgang G0, so gilt der kleinere Überschleifabstand.

- Wird ein sehr kleiner Wert für ADIS verwendet, so ist zu beachten, dass die Steuerung sicherstellt, dass jeder interpolierte Satz - auch ein Überschleifzwischenatz - mindestens einen Interpolationspunkt enthält. Die maximale Bahngeschwindigkeit wird damit auf ADIS/ Interpolatortakt begrenzt.
- Unabhängig von ADIS und ADISPOS wird der Überschleifbereich durch die Satz-Weglänge begrenzt.
Bei Sätzen mit kurzen Wegstrecken (Strecke $< 4 * ADIS$ bzw. $4 * ADISPOS$) wird der Überschleifabstand reduziert, damit ein abfahrbarer Teil des ursprünglichen Satzes noch erhalten bleibt. Die verbleibende Länge ist vom Achsverlauf abhängig und beträgt ca. 60% des Wegs, welcher in dem Satz noch abgefahren werden soll. ADIS bzw. ADISPOS werden somit auf die verbleibenden 40% des zu verfahrenen Wegs reduziert. Durch diesen Algorithmus wird weiterhin verhindert, dass bei einer sehr kleinen Konturänderung ein Überschleifsatz eingefügt wird. In diesem Fall wird solange auf Bahnsteuerbetrieb G64 umgeschaltet, bis Überschleifsätze wieder eingefügt werden können.

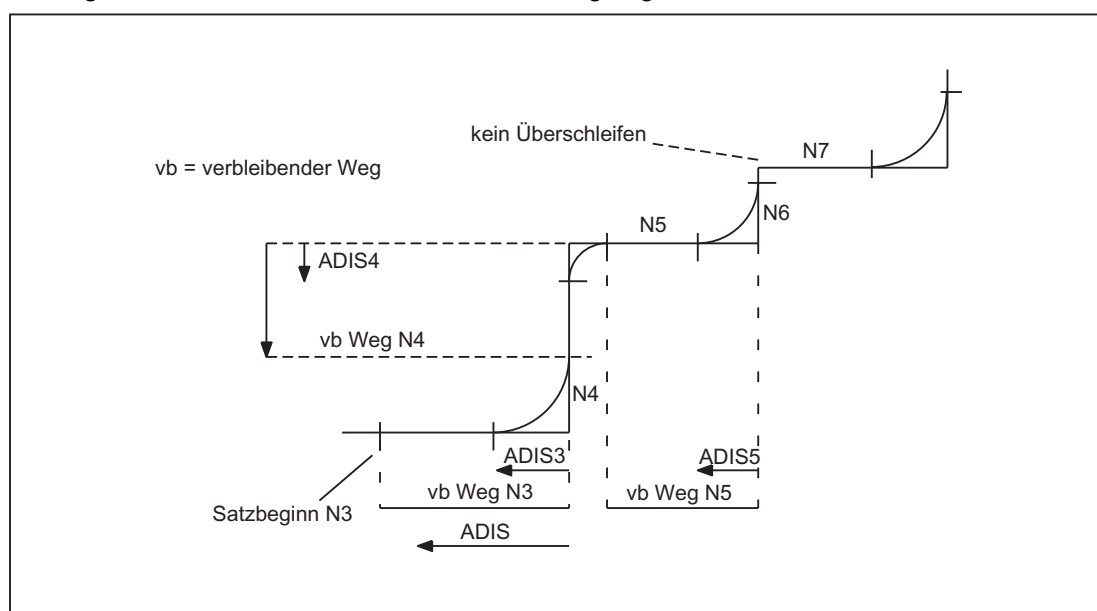


Bild 5-4 Bahnverlauf mit Begrenzung von ADIS

Aktivierung / Deaktivierung

Der Bahnsteuerbetrieb mit Überschleifen nach Wegkriterium kann in jedem NC-Teileprogrammsatz durch den modal wirksamen Befehl G641 aktiviert werden. Vor oder mit der Aktivierung sollten die Wegkriterien ADIS/ADISPOS angegeben werden.

Eine Unterbrechung ist möglich durch Anwahl des satzweise wirksamen Genauhalts G9.

Der Bahnsteuerbetrieb mit Überschleifen nach Wegkriterium (G641) kann deaktiviert werden durch Anwahl von:

- Modalen Genauhalt (G60)
- Bahnsteuerbetrieb G64, G642, G643, G644 oder G645

Programmbeispiel

Programmcode	Kommentar
N1 G641 Y50 F10 ADIS=0.5	; Bahnsteuerbetrieb mit Überschleifen nach Wegkriterium aktivieren (Überschleifabstand: 0,5 mm).
N2 X50	
N3 X50.7	
N4 Y50.7	
N5 Y51.4	
N6 Y51.0	
N7 X52.1	

5.3.3.2 Überschleifen unter Einhaltung definierter Toleranzen (G642/G643)

Funktion

Beim Bahnsteuerbetrieb mit Überschleifen unter Einhaltung definierter Toleranzen erfolgt das Überschleifen im Normalfall unter Einhaltung der maximal erlaubten Bahnabweichung. Anstelle dieser achsspezifischen Toleranz kann aber auch die Einhaltung der maximalen Konturabweichung (Konturtoleranz) oder der maximalen Winkelabweichung der Werkzeugorientierung (Orientierungstoleranz) konfiguriert werden.

Aktivierung

Der Bahnsteuerbetrieb mit Überschleifen unter Einhaltung definierter Toleranzen kann in jedem NC-Teileprogrammsatz durch den modal wirksamen Befehl G642 bzw. G643 aktiviert werden.

Eine Unterbrechung ist möglich durch Anwahl des satzweise wirksamen Genauhalts G9.

Der Bahnsteuerbetrieb mit Überschleifen unter Einhaltung definierter Toleranzen (G642/G643) kann deaktiviert werden durch Anwahl von:

- Modalen Genauhalt (G60)
- Bahnsteuerbetrieb G64, G641, G644 oder G645

Unterschiede G642 - G643

Die Funktionen G642 und G643 weisen im Überschleifverhalten folgende Unterschiede auf:

G642	G643
Bei G642 wird der Überschleifweg aus dem kürzesten Überschleifweg aller Achsen bestimmt. Dieser Wert wird bei der Erzeugung eines Überschleifsatzes berücksichtigt.	Bei G643 kann der Überschleifweg jeder Achse unterschiedlich sein. Die Überschleifwege werden achsspezifisch satzintern berücksichtigt (⇒ kein eigener Überschleifsatz).
Bei G642 bestimmt sich der Überschleifbereich aus der kleinsten Toleranzvorgabe.	Sehr unterschiedliche Vorgaben für die Konturtoleranz und die Orientierungstoleranz können sich nur bei G643 auswirken.

Parametrierung

Maximale Bahnabweichung

Die beim Überschleifen mit G642/G643 maximal erlaubte Bahnabweichung wird für jede Achse eingestellt im Maschinendatum:

MD33100 \$MA_COMPRESS_POS_TOL

Konturtoleranz und Orientierungstoleranz

Die Konturtoleranz und die Orientierungstoleranz werden eingestellt in den kanalspezifischen Settingdaten:

SD42465 \$SC_SMOOTH_CONTUR_TOL (Maximale Konturabweichung)

SD42466 \$SC_SMOOTH_ORI_TOL (Maximale Winkelabweichung der Werkzeugorientierung)

Die Settingdaten können im NC-Programm programmiert und dadurch für jeden Satzübergang anders vorgegeben werden.

Hinweis

Das Settingdatum SD42466 \$SC_SMOOTH_ORI_TOL ist nur bei aktiver Orientierungstransformation wirksam.

Überschleifverhalten

Das Überschleifverhalten mit G642 und G643 wird konfiguriert über das Maschinendatum:

MD20480 \$MC_SMOOTHING_MODE (Verhalten des Überschleifens mit G64x)

Die Einerstellen (**E**) definieren das Verhalten bei G643, die Zehnerstellen (**Z**) das Verhalten bei G642:

Wert E bzw. Z	Bedeutung
0	Alle Achsen: Überschleifen unter Einhaltung der maximal erlaubten Bahnabweichung: MD33100 \$MA_COMPRESS_POS_TOL
1	Geometrieachsen: Überschleifen unter Einhaltung der Konturtoleranz: SD42465 \$SC_SMOOTH_CONTUR_TOL Restliche Achsen: Überschleifen unter Einhaltung der maximal erlaubten Bahnabweichung: MD33100 \$MA_COMPRESS_POS_TOL
2	Geometrieachsen: Überschleifen unter Einhaltung der Orientierungstoleranz: SD42466 \$SC_SMOOTH_ORI_TOL Restliche Achsen: Überschleifen unter Einhaltung der maximal erlaubten Bahnabweichung: MD33100 \$MA_COMPRESS_POS_TOL

Wert E bzw. Z	Bedeutung
3	Geometrieachsen: Überschleifen unter Einhaltung der Konturtoleranz und Orientierungstoleranz: SD42465 \$SC_SMOOTH_CONTUR_TOL SD42466 \$SC_SMOOTH_ORI_TOL Restliche Achsen: Überschleifen unter Einhaltung der maximal erlaubten Bahnabweichung: MD33100 \$MA_COMPRESS_POS_TOL
4	Alle Achsen: Es wird die mit ADIS bzw. ADISPOS programmierte Überschleiflänge verwendet (wie bei G641). Die Vorgaben der achsspezifischen Toleranz bzw. der Kontur- und Orientierungstoleranz werden ignoriert.

Profil für die Grenzgeschwindigkeit

Die Benutzung eines Geschwindigkeitsprofils beim Überschleifen unter Einhaltung definierter Toleranzen wird über die Hunderterstelle von MD20480 gesteuert:

Wert	Bedeutung
< 100:	Innerhalb des Überschleifbereichs wird ein Profil der Grenzgeschwindigkeit berechnet, wie es sich aus den vorgegebenen maximalen Werten für Beschleunigung und Ruck der beteiligten Achsen bzw. der Bahn ergibt. Dies kann zu einem Ansteigen der Bahngeschwindigkeit in dem Überschleifbereich führen, und damit zu einem Beschleunigen der beteiligten Achsen.
≥100:	Für Überschleifsätze mit G641/G642 wird kein Profil der Grenzgeschwindigkeit berechnet. Es wird nur eine konstante Grenzgeschwindigkeit festgelegt. Damit wird verhindert, dass beim Überschleifen mit G641/G642 die beteiligten Achsen im Überschleifbereich eventuell beschleunigt werden. Diese Einstellung kann jedoch unter Umständen, insbesondere bei großen Überschleifbereichen, dazu führen, dass in Überschleifsätzen mit zu kleiner Geschwindigkeit gefahren wird.
1xx:	Kein Geschwindigkeitsprofil für G641
2xx:	Kein Geschwindigkeitsprofil für G642

Hinweis

Siehe auch MD28530 \$MC_MM_PATH_VELO_SEGMENTS (Anzahl der Speicherelemente z. Begrenzung der Bahngeschwindigkeit)

Randbedingungen

Einschränkung für Schutzbereiche bei aktiver Radius-Korrektur und einer Werkzeugorientierung:

Eine Werkzeugorientierung, die nicht senkrecht zu einer der drei Grundebenen des Basis-Koordinatensystems steht, berücksichtigt zwar die Radius-Korrektur, die Schutzbereiche werden jedoch nicht in die entsprechende Ebene gedreht.

Für G643 hat unbedingt zu gelten:

MD28530 \$MC_MM_PATH_VELO_SEGMENTS > 0 (Anzahl der Speicherelemente zur Begrenzung der Bahngeschwindigkeit)

Ist diese Bedingung erfüllt, so muss für alle Achsen gelten:

MD35240 \$MC_ACCEL_TYPE_DRIVE = FALSE (Beschleunigungskennlinie DRIVE für Achsen Ein / Aus)

5.3.3.3 Überschleifen mit maximal möglicher Achsdynamik (G644)

Funktion

Bei diesem Modus des Bahnsteuerbetriebs mit Überschleifen steht die maximal mögliche Dynamik der Achsen im Vordergrund.

Hinweis

Das Überschleifen mit G644 ist nur möglich, wenn:

- alle beteiligten Achsen nur eine lineare Bewegung in den beiden betrachteten Sätzen enthalten.
- **keine** kinematische Transformation aktiv ist.

Falls eine beteiligte Achse ein Polynom enthält (Polynom programmiert, Spline aktiv, Kompressor aktiv) oder eine kinematische Transformation aktiv ist, wird der Satzübergang mit G642 überschliften.

Aktivierung

Der Bahnsteuerbetrieb mit Überschleifen mit maximal möglicher Achsdynamik kann in jedem NC-Teileprogrammsatz durch den modal wirksamen Befehl G644 aktiviert werden.

Eine Unterbrechung ist möglich durch Anwahl des satzweise wirksamen Genauhalts G9.

Der Bahnsteuerbetrieb mit Überschleifen mit maximal möglicher Achsdynamik (G644) kann deaktiviert werden durch Anwahl von:

- Modalen Genauhalt (G60)
- Bahnsteuerbetrieb G64, G641, G642, G643 oder G645

Parametrierung

Das Überschleifverhalten mit G644 wird konfiguriert über die Tausender- und Zehntausenderstelle im Maschinendatum:

MD20480 \$MC_SMOOTHING_MODE (Verhalten des Überschleifens mit G64x)

Wert	Bedeutung
Tausenderstelle:	
0xxx:	Beim Überschleifen mit G644 werden die mit dem folgenden Maschinendatum angegebenen maximalen Abweichungen jeder Achse eingehalten: MD33100 \$MA_COMPRESS_POS_TOL Falls die Dynamik der Achse es zulässt, wird dabei evtl. die vorgegebene Toleranz nicht ausgenutzt.
1xxx:	Vorgabe des maximalen Überschleifwegs durch Programmierung von ADIS=... bzw. ADISPOS=... (wie bei G641).
2xxx:	Vorgabe der maximal auftretenden Frequenzen jeder Achse im Überschleifbereich mit dem Maschinendatum: MD32440 \$MA_LOOKAH_FREQUENCY (Glättungsfrequenz bei Look Ahead) Der Überschleifbereich wird so festgelegt, dass bei der Überschleifbewegung keine Frequenzen auftreten, welche die vorgegebene maximale Frequenz überschreiten.
3xxx:	Jede Achse, die einen Geschwindigkeitssprung an einer Ecke hat, fährt mit maximal möglicher Dynamik (maximale Beschleunigung und maximaler Ruck) um die Ecke. SOFT: Wenn SOFT aktiv ist, wird die maximale Beschleunigung und der maximale Ruck jeder Achse eingehalten. BRISK: Wenn BRISK aktiv ist, wird nur die maximale Beschleunigung eingehalten, der Ruck wird nicht begrenzt. Bei dieser Einstellung werden also weder die maximal auftretenden Abweichungen noch der maximale Überschleifweg kontrolliert. Die sich ergebenden Abweichungen bzw. Überschleifwege bestimmen sich allein aus den dynamischen Grenzwerten der jeweiligen Achse und der aktuellen Bahngeschwindigkeit.
4xxx:	Wie bei 0xxx werden die mit dem folgenden Maschinendatum angegebenen maximalen Abweichungen jeder Achse eingehalten: MD33100 \$MA_COMPRESS_POS_TOL Im Unterschied zu 0xxx wird die vorgegebene Toleranz nach Möglichkeit auch ausgenutzt. Die Achse erreicht daher nicht ihre maximal mögliche Dynamik.
5xxx:	Wie bei 1xxx wird der maximale Überschleifweg durch Programmierung von ADIS=... bzw. ADISPOS=... vorgegeben. Im Unterschied zu 1xxx wird der vorgegebene Überschleifweg nach Möglichkeit auch ausgenutzt. Die beteiligten Achsen erreichen dann eventuell nicht ihre maximal mögliche Dynamik.
Zehntausenderstelle	
0xxxx	Die Geschwindigkeitsprofile der Achsen werden im Überschleifbereich bei BRISK ohne Ruckbegrenzung und bei SOFT mit Ruckbegrenzung bestimmt.
1xxxx	Die Geschwindigkeitsprofile der Achsen werden im Überschleifbereich immer mit Ruckbegrenzung bestimmt, unabhängig davon ob BRISK oder SOFT aktiv ist.

Bei der Vorgabe der maximalen axialen Abweichungen (MD33100 \$MA_COMPRESS_POS_TOL) bzw. des maximalen Überschleifabstandes (ADIS/ADISPOS) wird der zur Verfügung stehende Überschleifweg normalerweise nicht ausgenutzt, wenn es die Dynamik der beteiligten Achsen erlauben. Dadurch hängt die Länge des Überschleifwegs vom aktiven Bahnvorschub ab. Bei kleineren Bahngeschwindigkeiten erhält

man geringere Abweichungen von der programmierten Kontur. Es kann jedoch eingestellt werden, dass in diesen Fällen nach Möglichkeit die vorgegebenen maximalen axialen Abweichungen bzw. der vorgegebene Überschleifabstand ausgenutzt wird. In diesem Fall sind dann die Abweichungen von der programmierten Kontur unabhängig vom programmierten Bahnvorschub.

Hinweis

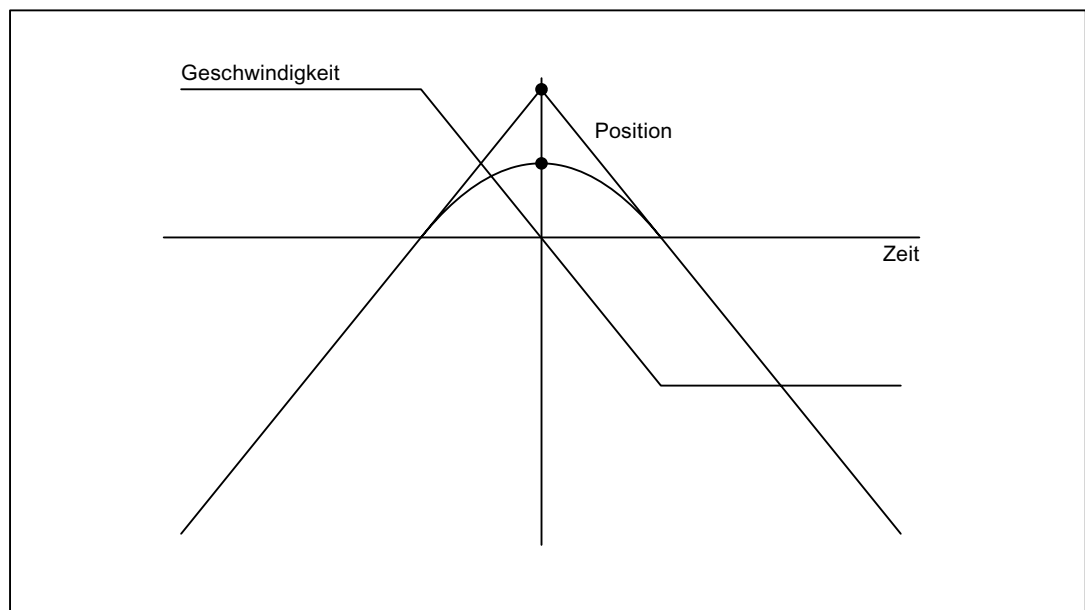
Außer den bisher genannten kann zusätzlich folgende Begrenzung wirksam werden:

Der Überschleifabstand kann maximal die Hälfte der Länge der beteiligten Originalsätze betragen.

Ruckbegrenzung

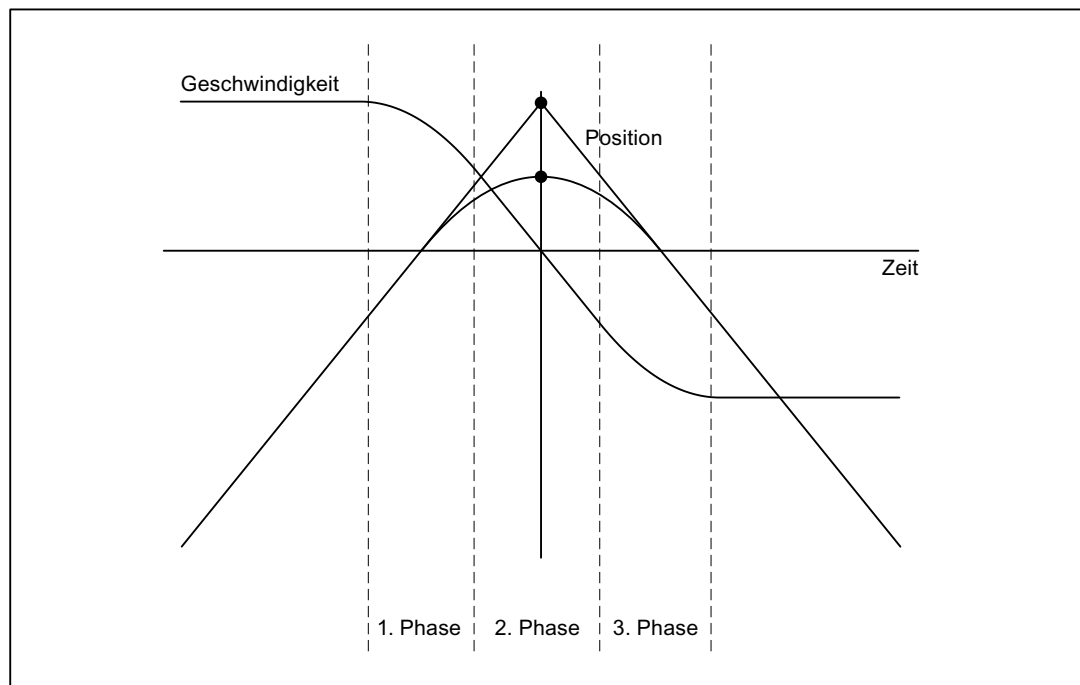
Die Glättung des Geschwindigkeitssprungs jeder Achse und damit die Form des Überschleifwegs hängt davon ab, ob eine Interpolation mit oder ohne Ruckbegrenzung durchgeführt wird.

Ohne Ruckbegrenzung erreicht die Beschleunigung jeder Achse im gesamten Überschleifbereich ihren Maximalwert:



Mit Ruckbegrenzung wird der Ruck jeder Achse im Überschleifbereich auf ihren jeweiligen Maximalwert begrenzt. Demzufolge besteht die Überschleifbewegung im Allgemeinen aus 3 Phasen:

- **1. Phase**
In der 1. Phase wird die maximale Beschleunigung jeder Achse aufgebaut. Dabei ist der Ruck konstant und gleich dem maximal möglichen Ruck der jeweiligen Achse.
- **2. Phase**
Die 2. Phase wird mit der maximal erlaubten Beschleunigung durchfahren.
- **3. Phase**
In der 3. Phase schließlich wird die Beschleunigung jeder Achse mit dem maximal erlaubten Ruck wieder auf Null abgebaut.



5.3.3.4 Überschleifen tangentialer Satzübergänge (G645)

Funktion

Bei diesem Modus des Bahnsteuerbetriebs mit Überschleifen werden auch bei tangentialen Satzübergängen Überschleifsätze gebildet, wenn der Krümmungsverlauf der Originalkontur in mindestens einer Achse einen Sprung aufweist.

Die Überschleifbewegung wird hierbei so festgelegt, dass alle beteiligten Achsen keinen Sprung in der Beschleunigung erfahren und die parametrisierten maximalen Abweichungen zur Originalkontur (MD33120 \$MA_PATH_TRANS_POS_TOL) nicht überschritten werden.

Bei knickförmigen, nicht-tangentialen Satzübergängen ist das Überschleifverhalten wie bei G642 (siehe Kapitel "Überschleifen unter Einhaltung definierter Toleranzen (G642/G643) (Seite 200)").

Aktivierung / Deaktivierung

Der Bahnsteuerbetrieb mit Überschleifen tangentialer Satzübergänge kann in jedem NC-Teileprogrammsatz durch den modal wirksamen Befehl G645 aktiviert werden.

Eine Unterbrechung ist möglich durch Anwahl des satzweise wirksamen Genauhalts G9.

Der Bahnsteuerbetrieb mit Überschleifen tangentialer Satzübergänge (G645) kann deaktiviert werden durch Anwahl von:

- Modalen Genauhalt (G60)
- Bahnsteuerbetrieb G64, G641, G642, G643 oder G644

Vergleich G642 - G645

Beim Überschleifen mit G642 werden nur Satzübergänge überschleift, die eine Ecke darstellen, d. h. die Geschwindigkeit mindestens einer Achse hat einen Sprung. Ist ein Satzübergang jedoch tangential, aber die Krümmung macht einen Sprung, wird mit G642 kein Überschleifsatz eingefügt. Wird dieser Satzübergang mit endlicher Geschwindigkeit überfahren, so erfahren die Achsen einen mehr oder weniger großen Sprung der Beschleunigung, der (bei aktiver Ruckbegrenzung!) den parametrisierten Grenzwert (MD32432 \$MA_PATH_TRANS_JERK_LIM) nicht überschreiten darf. Abhängig von der Höhe des Grenzwerts wird die Bahngeschwindigkeit am Satzübergang dadurch evtl. stark verringert. Diese Einschränkung wird durch die Verwendung von G645 vermieden, da hier die Überschleifbewegung so festgelegt wird, dass keine Beschleunigungssprünge auftreten.

Parametrierung

Mit dem folgenden Maschinendatum wird für jede Achse die maximal erlaubte Bahnabweichung beim Überschleifen mit G645 angegeben:

MD33120 \$MA_PATH_TRANS_POS_TOL

Dieser Wert ist nur relevant für tangentiale Satzübergänge, die nicht beschleunigungsstetig sind. Beim Überschleifen von knickförmigen, nicht-tangentialen Satzübergängen wird (wie bei G642) der Toleranzwert aus MD33100 \$MA_COMPRESS_POS_TOL wirksam.

Siehe auch

Freiformflächenmodus: Grundfunktionen (Seite 232)

5.3.3.5 Überschleifen und Repositionieren (REPOS)

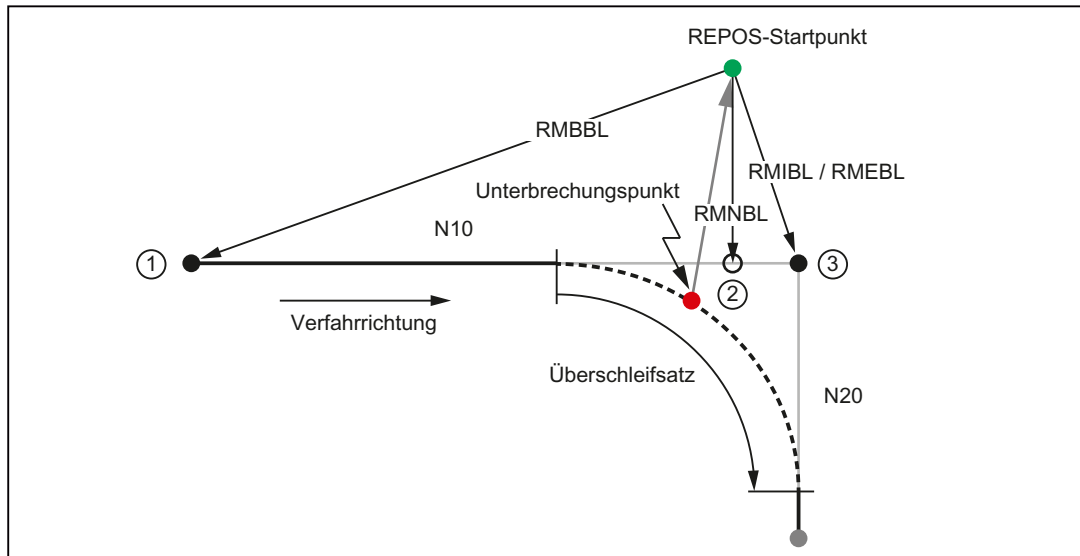
Wird eine Bearbeitung im Bereich der Überschleifkontur unterbrochen, kann durch einen REPOS-Vorgang **nicht** wieder direkt an die Überschleifkontur positioniert werden. In diesem Fall kann nur an die **programmierte** Kontur positioniert werden.

Beispiel

Programmiert: Zwei Verfahrssätze N10 und N20 mit programmiertem Überschleifen G641.

Im Überschleifbereich erfolgt eine Unterbrechung der Verfahrbewegung. Anschließend werden die Achsen z. B. manuell auf den REPOS-Startpunkt verfahren. Abhängig vom

gewählten REPOS-Mode erfolgt das Wiederanfahren an die Kontur an den Punkten ①, ② oder ③.



- RMBBL Wiederanfahren an den Anfang des unterbrochenen Verfahrstrahles
- RMIBL Wiederanfahren an die Unterbrechungsstelle
- RMEBL Wiederanfahren an das Ende des unterbrochenen Verfahrstrahles
- RMNBL Wiederanfahren an den nächstliegenden Konturpunkt
- ① Satzanfang N10
- ② Zum REPOS-Startpunkt nächstliegender Konturpunkt
- ③ Satzende N10 / Satzanfang N20

5.3.4 LookAhead

5.3.4.1 Standardfunktionalität

Funktion

LookAhead ist eine im Bahnsteuerbetrieb (G64, G64x) aktive Funktion, die über den aktuellen Satz hinaus für mehrere NC-Teilprogrammätze eine vorausschauende Geschwindigkeitsführung ermittelt.

Hinweis

LookAhead ist nur für Bahnachsen verfügbar, **nicht** für Spindeln und Positionierachsen.

Beinhaltet ein Teilprogramm aufeinanderfolgende Sätze mit sehr kleinen Bahnwegen, dann wird ohne LookAhead pro Satz nur eine Geschwindigkeit erreicht, die zum Satzende ein Abbremsen der Achsen unter Wahrung der Beschleunigungsgrenzen ermöglicht. Dies bedeutet, dass die programmierte Geschwindigkeit überhaupt nicht erreicht wird. Mit

LookAhead ist es dagegen möglich, bei annähernd tangentialen Satzübergängen die Beschleunigungs- und Bremsphase über mehrere Sätze zu realisieren und somit bei kleinen Wegen einen höheren Vorschub zu erzielen.

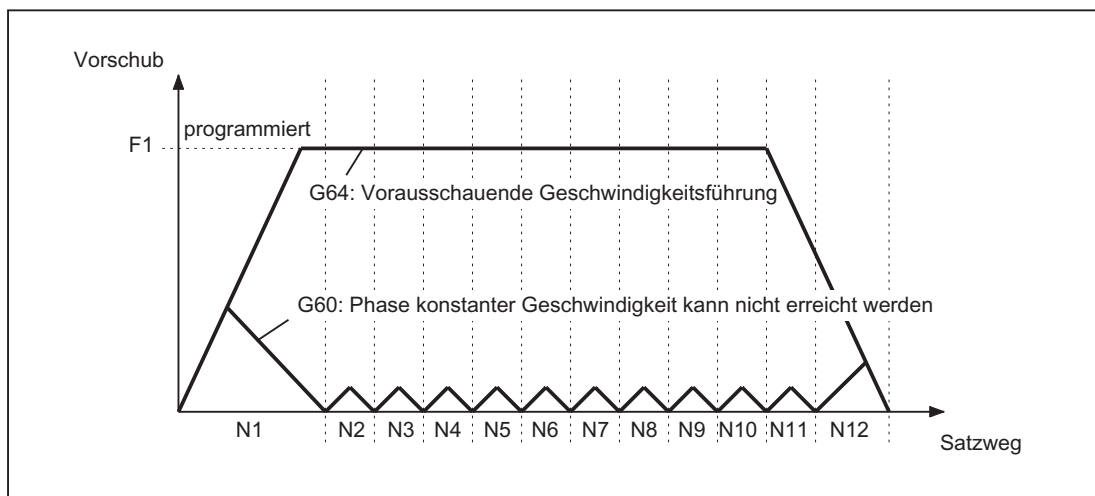


Bild 5-5 Geschwindigkeitsführung bei kurzen Wegstrecken und Genauhalt G60 bzw. Bahnsteuerbetrieb G64 mit LookAhead

Auf Geschwindigkeitbeschränkungen wird vorausschauend so abgebremst, dass eine Verletzung der Beschleunigungs- und der Geschwindigkeitsgrenze vermieden wird.

LookAhead berücksichtigt planbare Geschwindigkeitsbeschränkungen wie:

- Genauhalt am Satzende
- Geschwindigkeitsbegrenzung im Satz
- Beschleunigungsbegrenzung im Satz
- Geschwindigkeitsbegrenzung am Satzübergang
- Synchronisieren mit Satzwechsel am Satzübergang

Funktionsweise

LookAhead analysiert satzbezogen die planbaren Geschwindigkeitsbeschränkungen und legt dementsprechend die benötigten Bremsrampenprofile fest. Die Vorausschau wird automatisch an Satzlänge, Bremsvermögen und zulässige Bahngeschwindigkeit angepasst.

Aus Sicherheitsgründen wird die Geschwindigkeit am Satzende des letzten vorbereiteten Satzes zunächst zu 0 angenommen, da der anschließende Satz sehr klein oder ein Genauhaltsatz sein könnte und die Achsen zum Satzendpunkt Stillstand erreicht haben sollen.

Bei einer Folge von Sätzen mit hoher Sollgeschwindigkeit und sehr kurzen Wegstrecken kann in den einzelnen Sätzen die Geschwindigkeit, abhängig vom aktuellen vorausgeschauten Geschwindigkeitswert, erhöht werden, um die geforderte Sollgeschwindigkeit zu erreichen und anschließend wieder reduziert werden, damit die Geschwindigkeit am Satzendpunkt des letzten vorausschauenden Folgesatz 0 werden kann. Man erhält dadurch ein sägezahnförmiges Geschwindigkeitsprofil (siehe folgendes Bild), welches durch Herabsetzen der Sollgeschwindigkeit oder durch Vergrößern der vorausschauenden Satzanzahl vermieden werden kann.

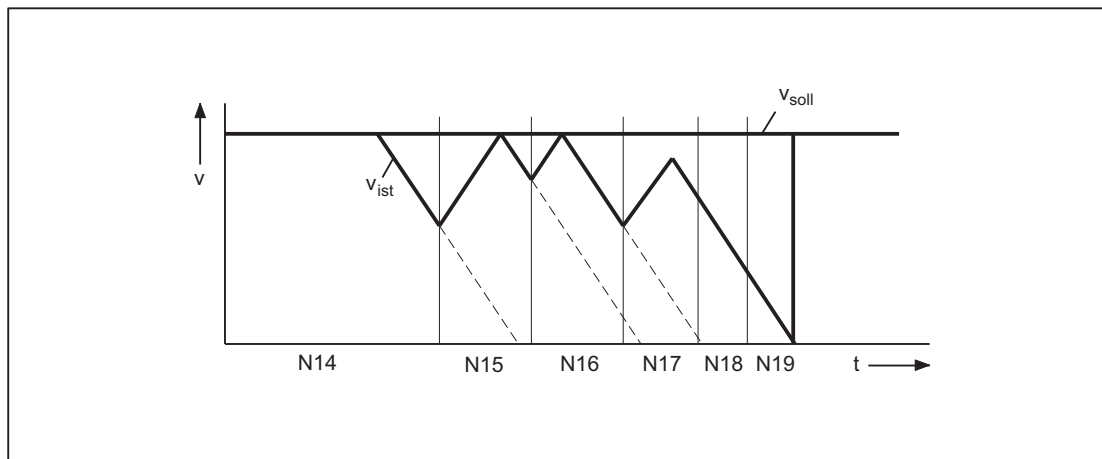


Bild 5-6 Beispiel für satzübergreifende Geschwindigkeitsführung (Zahl der vorausschauenden Sätze = 2)

Aktivierung / Deaktivierung

LookAhead wird aktiviert durch Anwahl des Bahnsteuerbetriebs G64, G641, G642, G643, G644 oder G645.

Eine Unterbrechung ist möglich durch Anwahl des satzweise wirksamen Genauhalts G09.

LookAhead wird deaktiviert durch Anwahl des modalen Genauhalt (G60).

Parametrierung

Satzanzahl

Um im Bahnsteuerbetrieb sicher zu fahren, muss der Vorschub über mehrere Sätze angepasst werden. Die Anzahl der vorausschauenden Sätze wird steuerungsintern automatisch ermittelt, wird optional aber über ein Maschinendatum begrenzt. Die Standardeinstellung ist "1" und bedeutet, dass LookAhead die Geschwindigkeitsführung nur für den Folgesatz berücksichtigt.

Da LookAhead vor allem bei (relativ zum Bremsweg) kurzen Sätzen wichtig ist, ist für das vorausschauende Bremsen die benötigte Satzanzahl von Interesse. Es ist ausreichend, eine Weglänge gleich dem Bremsweg zu betrachten, der nötig ist, um aus der maximalen Geschwindigkeit Stillstand zu erreichen.

Für eine Maschine mit einer niedrigen Achsbeschleunigung von $a = 1 \text{ m/s}^2$ und einem dazu hohen Vorschub von $v_{\text{Bahn}} = 10 \text{ m/min}$ erhält man mit einer erreichbaren Blockzykluszeit der Steuerung von $TB = 10 \text{ ms}$ eine für die Steuerung folgende Satzanzahl $n_{\text{LookAhead}}$ von:

$$n_{\text{LookAhead}} = \text{Bremsweg} / \text{Satzlänge} = (v_{\text{Bahn}}^2 / (2a)) / (v_{\text{Bahn}} * TB) = 9$$

Unter den angegebenen Bedingungen ist es nicht sinnvoll, den Vorschub über 10 Sätze anzupassen. Die angegebene Satzanzahl für die Vorausschau der LookAhead-Funktion verändert den LookAhead-Algorithmus und den Speicheraufwand nicht.

Da in einem Programm sehr oft die Bearbeitungsgeschwindigkeit kleiner der Maximalgeschwindigkeit ist, würde man mehr Sätze vorausschauen wie nötig ist, was

zusätzliche unnötige Rechenleistung beanspruchen würde. Daher wird die benötigte Satzanzahl aus der Geschwindigkeit abgeleitet, die sich aus folgender Multiplikation ergibt:

- Programmierte Geschwindigkeit * MD12100 \$MN_OVR_FACTOR_LIMIT_BIN
(bei Verwendung eines **binär**kodierten Vorschub-Override-Schalters)
- Programmierte Geschwindigkeit * MD12030 \$MN_OVR_FACTOR_FEEDRATE[30]
(bei Verwendung eines **gray**kodierten Vorschub-Override-Schalters)

Der Wert von MD12100 bzw. der 31. Override-Wert von MD12030 legt dabei die Dynamik-Reserven fest, die die Geschwindigkeitsführung für eine Überhöhung des Bahnvorschubs bereithält.

Hinweis

Der 31. Override-Wert von MD12030 sollte dem höchsten tatsächlich verwendeten Override-Faktor entsprechen.

Hinweis

Die vorausschauende Satzanzahl wird durch die mögliche Anzahl der NC-Sätze im IPO-Puffer begrenzt.

Geschwindigkeitsprofile

Neben den festen planbaren Geschwindigkeitsbeschränkungen kann LookAhead zusätzlich auch die programmierte Geschwindigkeit miteinbeziehen. Damit ist es möglich, über den aktuellen Satz hinaus vorausschauend die geringere Geschwindigkeit zu erreichen.

- **Ermittlung der Folgesatzgeschwindigkeit**
Ein mögliches Geschwindigkeitsprofil enthält die Ermittlung der Folgesatzgeschwindigkeit. Anhand von Informationen aus dem aktuellen und dem folgenden NC-Satz wird ein Geschwindigkeitsprofil berechnet, aus dem wiederum die erforderlichen Geschwindigkeitsreduzierungen für den aktuellen Override abgeleitet werden. Der ermittelte Maximalwert des Geschwindigkeitsprofils wird durch die maximale Bahngeschwindigkeit begrenzt.
Mit der Funktion ist es möglich, Override-berücksichtigend im aktuellen Satz eine Geschwindigkeitsreduzierung zu starten, so dass zu Beginn des Folgesatzes dessen geringere Geschwindigkeit erreicht sein kann. Dauert die Reduzierung der Geschwindigkeit länger als die Verfahrzeit des aktuellen Satzes, so wird im Folgesatz die Geschwindigkeit weiter verringert. Die Geschwindigkeitsführung wird immer nur für den Folgesatz berücksichtigt.
Aktiviert wird die Funktion über das Maschinendatum:
MD20400 \$MC_LOOKAH_USE_VELO_NEXT_BLOCK

Wert	Bedeutung
TRUE	Funktion aktiv
FALSE	Funktion nicht aktiv

- **Festlegung von Override-Eckwerten**

Ist das Geschwindigkeitsprofil der Folgesatzgeschwindigkeit nicht ausreichend, weil z. B. sehr hohe Override-Werte (z. B. 200 %) verwendet werden bzw. konstante Schnittgeschwindigkeit G96/G961 aktiv ist und somit die Geschwindigkeit im Folgesatz immer noch reduziert werden muss, so bietet LookAhead eine Möglichkeit an, die programmierte Geschwindigkeit über mehrere NC-Sätze vorausschauend zu reduzieren: Mittels Festlegung von Override-Eckwerten berechnet sich LookAhead für jeden Eckwert ein begrenzendes Geschwindigkeitsprofil. Von diesen Profilen werden die erforderlichen Geschwindigkeitsreduzierungen für den aktuellen Override abgeleitet.

Der ermittelte Maximalwert des Geschwindigkeitsprofils wird durch die maximale Bahngeschwindigkeit begrenzt.

Der obere Eckwert sollte den Geschwindigkeitsbereich abdecken, der durch den Maximalwert des folgenden Maschinendatums erreicht wird:

MD12030 \$MN_OVR_FACTOR_FEEDRATE[n] (Bewertung des Bahnvorschub-Override-Schalters)

Er kann auch erreicht werden durch den Wert des Maschinendatums:

MD12100 \$MN_OVR_FACTOR_LIMIT_BIN (Begrenzung bei binärkodiertem Override-Schalter)

Damit kann eine Reduzierung der Geschwindigkeit in den Satz hinein, in dem sie programmiert ist, vermieden werden.

Sind auch schon bei 100 %-Override deutliche satzübergreifende Geschwindigkeitsreduzierungen erforderlich, so sollte auch im unteren Override-Bereich ein Eckwert gesetzt werden.

Die Anzahl der verwendeten Override-Eckwerte pro Kanal werden angegeben im Maschinendatum:

MD20430 \$MC_LOOKAH_NUM_OVR_POINTS (Anzahl der Override-Schalter-Eckwerte bei LookAhead)

Die dazugehörigen Eckwerte werden hinterlegt im Maschinendatum:

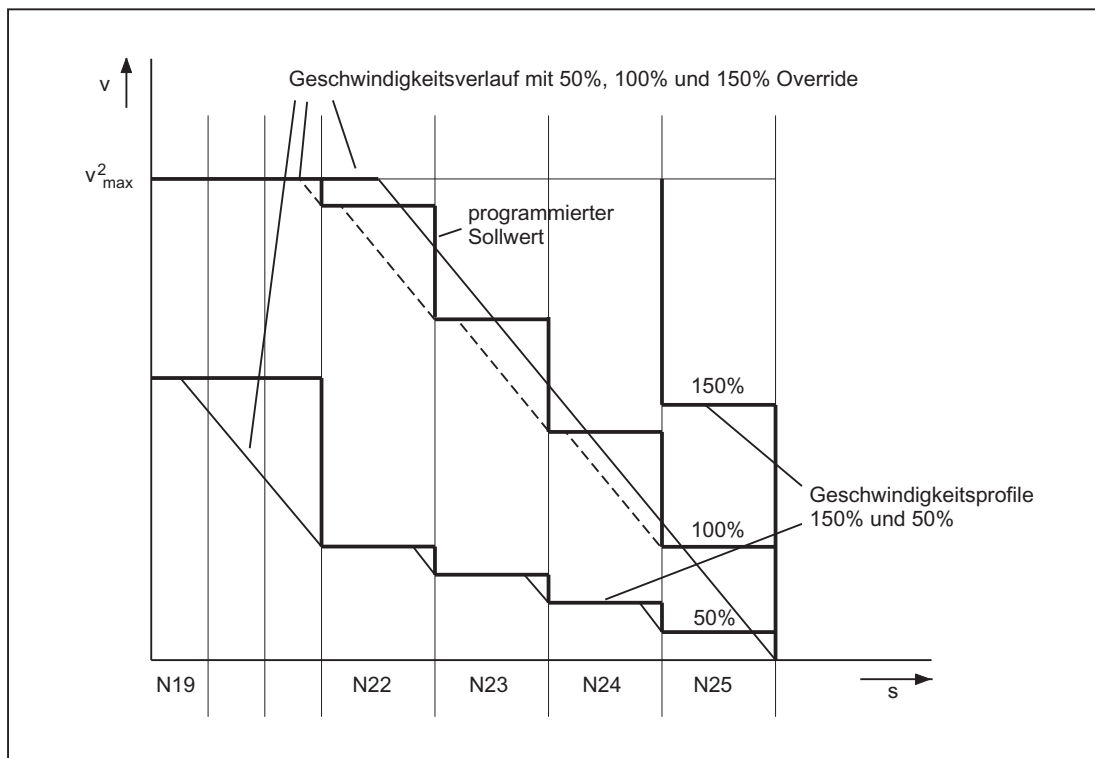
MD20440 \$MC_LOOKAH_OVR_POINTS (Override-Schalter-Eckwerte bei LookAhead)

Beispiel:

Begrenzende Geschwindigkeitsverläufe mit:

- Override = 50 %, 100 % bzw. 150 %
- vorausschauender Satzanzahl = 4
- MD20430 \$MC_LOOKAH_NUM_OVR_POINTS = 2
- MD20440 \$MC_LOOKAH_OVR_POINTS = 1.5, 0.5

– MD20400 \$MC_LOOKAH_USE_VELO_NEXT_BLOCK = 1



Eine Kombination beider Verfahren (Ermittlung der Folgesatzgeschwindigkeit und Festlegung von Override-Eckwerten) zur Ermittlung der Geschwindigkeitsprofile ist möglich und in der Regel auch sinnvoll, weil bereits mit den vobesetzten Maschinendaten für diese Funktionen der größte Bereich der Override-abhängigen Geschwindigkeitsbeschränkungen abgedeckt ist.

Hinweis

Ist keines der beiden Verfahren aktiviert, so wird die Sollgeschwindigkeit immer erst im aktuellen Satz angefahren.

Hinweis

Planbare Geschwindigkeitsbeschränkungen begrenzen die Override-abhängigen Geschwindigkeitsbeschränkungen.

Entlastungsfaktor bei Blockzyklus-Problemen

Blockzyklusprobleme treten auf, wenn die Verfahrnlänge der abzuarbeitenden NC-Sätze so kurz ist, dass die LookAhead-Funktion die Maschinengeschwindigkeit reduzieren muss, um der Satzaufbereitung genügend Zeit zur Verfügung zu stellen. In dieser Situation kann ein ständiges Abbremsen und Beschleunigen der Bahnbewegung auftreten.

Derartige Geschwindigkeits-Schwankungen können durch Angabe eines Entlastungsfaktors gedämpft werden:

MD20450 \$MC_LOOKAH_RELIEVE_BLOCK_CYCLE (Entlastungsfaktor für die Blockzykluszeit)

Randbedingungen

Achsspezifischer Vorschub-Halt / Achssperre

Achsspezifischer Vorschub-Halt und achsspezifische Achssperre werden von LookAhead nicht berücksichtigt.

Soll eine Achse interpoliert werden, die aber andererseits per achsspezifischem Vorschub-Halt oder Achsen-Sperre stehenbleiben soll, so hält LookAhead die Bahnbewegung nicht vor dem betreffenden Satz an, sondern bremst **im** Satz ab.

Stört dieses Verhalten, so kann über die PLC ein **achsspezifischer** Vorschub-Halt auf eine **kanalspezifische** Vorschubsperrung umgelegt und damit die Bahn sofort angehalten werden (siehe auch Kapitel "Funktion (Seite 89)").

5.3.4.2 Freiformflächenmodus: Erweiterungsfunktion

Funktion

Die Funktion "Freiformflächenmodus: Erweiterungsfunktion" ist eine Erweiterung der LookAhead-Standardfunktionalität und dient zur Berechnung des Bahngeschwindigkeitsprofils bei der Freiformflächenbearbeitung (siehe auch Kapitel "Freiformflächenmodus: Grundfunktionen (Seite 232)").

Ihr Einsatz optimiert den Bahnsteuerbetrieb wie folgt:

- Symmetrie zwischen Beschleunigungs- und Bremsprofil
- Gleichmäßiger Beschleunigungsvorgang auch bei sich ändernder Ruck- bzw. Beschleunigungsbegrenzung
- Gleichmäßiger Beschleunigungsvorgang von Sollgeschwindigkeitsprofilen unabhängig davon, inwieweit sie mit der vorgegebenen Dynamikbegrenzung anfahrbar sind oder nicht
- Vorausschauendes Bremsen auf niedrigere Sollgeschwindigkeiten

Die Gleichmäßigkeit und die Einhaltung der Dynamikbegrenzung gewährleisten, dass die Sollgeschwindigkeitsprofile zu einem homogenen Geschwindigkeitsprofil auf dem Teil geglättet werden. Damit wird sichergestellt, dass sich Schleppfehler möglichst gering auf die Oberflächengüte auswirken.

Vorteile

- Gleichmäßigere Oberfläche des Werkstücks
- Geringere Belastung der Maschine

Anwendungen

Die Funktion "Freiformflächenmodus: Erweiterungsfunktion" wird zur Bearbeitung von Werkstücken verwendet, die vorwiegend aus Freiformflächen bestehen.

Hinweis

Bei Standardbearbeitungen werden keine besseren Ergebnisse erzielt, weshalb in solchen Fällen die Standardfunktionalität von LookAhead verwendet werden sollte.

Wirksamkeit

Die Funktion ist nur wirksam:

- in der Betriebsart: AUTOMATIK
- im Beschleunigungsmodus "Beschleunigung mit Ruckbegrenzung (SOFT)"

Parametrierung

Arbeitsspeicher

Der Speicher für die Funktion "Freiformflächenmodus: Erweiterungsfunktion" wird konfiguriert durch das Maschinendatum:

MD28533 \$MC_MM_LOOKAH_FFORM_UNITS = <Wert>

Der benötigte Speicher ist abhängig vom Teileprogramm, den Satzlängen, der Achsdynamik sowie einer aktiven kinematischen Transformation.

Als Richtwert für die Freiformflächenbearbeitung gilt folgende Einstellung: MD28533 = 18

Hinweis

MD28533 sollte wegen dem zusätzlichen Speicherbedarf nur für die Kanäle gesetzt werden, in denen auch Freiformflächen bearbeitet werden.

Anzahl der NC-Sätze im IPO-Puffer

In der Regel empfiehlt es sich, für die Verwendung der Funktion "Freiformflächenmodus: Erweiterungsfunktion" die konfigurierte Anzahl der NC-Sätze im Interpolationspuffer deutlich zu erhöhen:

MD28060 \$MC_MM_IPO_BUFFER_SIZE > 100

Ein zu geringer Satzspeicher verringert ggf. die Gleichmäßigkeit des Bahngeschwindigkeitsprofils.

Aktivierung / Deaktivierung

Die Funktion kann für jeden Dynamikmodus (siehe Kapitel "Dynamikmodus für Bahninterpolation (Seite 230)") unabhängig ein- oder ausgeschaltet werden:

MD20443 \$MC_LOOKAH_FFORM[<n>]= <Wert>

Index <n>	Dynamikmodus	<Wert>	Freiformflächenmodus: Erweiterungsfunktion
0	Standard-Dynamikeinstellungen (DYNNORM)	0	aus
		1	ein
1	Positionierbetrieb, Gewindebohren (DYNPOS)	0	aus
		1	ein
2	Schruppen (DYNROUGH)	0	aus
		1	ein
3	Schlichten (DYNSEMIFIN)	0	aus
		1	ein
4	Feinschlichten (DYNFINISH)	0	aus
		1	ein

Typischerweise ist die Funktion "Freiformflächenmodus: Erweiterungsfunktion" nur aktiv, wenn auch die Funktion "Freiformflächenmodus: Grundfunktionen" aktiv ist. Die Einstellungen in MD20443 \$MC_LOOKAH_FFORM[<n>] sollten daher mit den Einstellungen in MD20606 \$MC_PREPDYN_SMOOTHING_ON[<n>] übereinstimmen.

In den Dynamikmodi, in denen die Funktion "Freiformflächenmodus: Erweiterungsfunktion" ausgeschaltet ist, ist die LookAhead-Standardfunktionalität aktiv.

Programmierung

In der Regel wird die Funktion "Freiformflächenmodus: Erweiterungsfunktion" durch einen Wechsel des Dynamikmodus im Teileprogramm wirksam.

Beispiel

Folgende Parametrierung ist gegeben:

MD20443 \$MC_LOOKAH_FFORM[0] = 0

MD20443 \$MC_LOOKAH_FFORM[1] = 0

MD20443 \$MC_LOOKAH_FFORM[2] = 1

MD20443 \$MC_LOOKAH_FFORM[3] = 1

MD20443 \$MC_LOOKAH_FFORM[4] = 1

Wechsel des Dynamikmodus im Teileprogramm:

Programmcode	Kommentar
N10 DYNPOS	; Dynamikmodus DYNPOS einschalten. Im Dynamikmodus DYNPOS ist die LookAhead-Standardfunktionalität aktiv.
...	
N100 G17 G54 F10000	

Programmcode	Kommentar
N101 DYNFINISH	; Dynamikmodus DYNFINISH einschalten. Im Dynamikmodus DYNFINISH ist die Funktion "Freiformflächenmodus: Erweiterungsfunktionen" aktiv.
N102 SOFT G642	
N103 X-0.274 Y149.679 Z100.000 G0	
N104 COMPCAD	
...	
N1009 Z4.994 G01	
N10010 X.520 Y149.679 Z5.000	
N10011 X10.841 Y149.679 Z5.000	
N10012 X11.635 Y149.679 Z5.010	
N10013 X12.032 Y149.679 Z5.031	
M30	

Hinweis

Beim Wechsel zwischen der LookAhead-Standardfunktionalität und der Funktion "Freiformflächenmodus: Erweiterungsfunktion" bzw. umgekehrt wird der Bahnsteuerbetrieb durch einen Interpolator-Stopp unterbrochen.

Randbedingungen

Automatische Funktionsumschaltung

Die Anwendung folgender Funktionen bewirken eine automatische Umschaltung auf die LookAhead-Standardfunktionalität:

- Gewindeschneiden/Gewindebohren (G33, G34, G35, G331, G332, G63)
- Bahn-Leitwertkopplung
- Stanzen, Nibbeln
- Kartesisches PTP-Fahren

Danach wird die Funktion "Freiformflächenmodus: Erweiterungsfunktion" automatisch wieder eingeschaltet.

Verwendung der Befehle der G-Gruppe 15 (Vorschubtypen)

Folgende Vorschubtypen sind in Verbindung mit der Funktion "Freiformflächenmodus: Erweiterungsfunktion" nicht zu empfehlen:

- Umdrehungsvorschub (G95, G96, G97, ...)
- Zeitreziproker Vorschub (G93)

Verwendung von FLIN

Das Vorschubprofil FLIN ist in Verbindung mit der Funktion "Freiformflächenmodus: Erweiterungsfunktion" nicht nutzbar.

Einfluss von Vorschubkorrekturen

Vorschubkorrekturen (über Maschinensteuertafel, \$AC_OVR, ...) können die Verfahrszeit gegenüber der LookAhead-Standardfunktionalität merklich verlängern.

Wechselwirkung mit Eilgangbewegung (G0)

In die Freiformflächenbearbeitung eingestreute G0-Sätze schalten die LookAhead-Funktionalität nicht um (von der Funktion "Freiformflächenmodus: Erweiterungsfunktion" auf die LookAhead-Standardfunktionalität oder umgekehrt). D. h. obwohl mit G0 die Standard-Dynamikeinstellung (DYNNORM) wirksam ist, wird dadurch nicht automatisch auch die für DYNNORM voreingestellte LookAhead-Standardfunktionalität (→ MD20443 \$MC_LOOKAH_FFORM) wirksam. Durch das Beibehalten der aktuell aktiven LookAhead-Funktionalität wird ein homogeneres Geschwindigkeitsprofil erreicht, zumal in der Regel die G0- und Polynomsätze durch Überschleifen glatt verbunden sind.

5.4 Dynamikanpassungen

5.4.1 Glättung der Bahngeschwindigkeit

Einleitung

Die Geschwindigkeitsführung nutzt die vorgegebene Achsdynamik aus. Wenn der programmierte Vorschub nicht erreicht werden kann, wird die Bahngeschwindigkeit an den parametrisierten axialen Grenzwerten und den Grenzwerten der Bahn (Geschwindigkeit, Beschleunigung, Ruck) geführt. Damit kann es zu häufigen Brems- und Beschleunigungsvorgängen auf der Bahn kommen.

Findet bei einer Bearbeitung mit hoher Bahngeschwindigkeit ein kurzzeitiger Beschleunigungsvorgang statt, der nach sehr kurzer Zeit wieder zu einem Bremsvorgang führt, so wird die Bearbeitungszeit hierdurch nicht deutlich reduziert. Diese Beschleunigungsvorgänge können jedoch zu unerwünschten Erscheinungen führen, z. B. wenn Maschinenresonanzen angeregt werden.

Bei einigen Anwendungen im Formenbau, speziell beim Hochgeschwindigkeitsfräsen, ist dagegen eine gleichmäßige Bahngeschwindigkeit wünschenswert. In diesen Fällen kann es daher sinnvoll sein, zugunsten einer ruhigeren Bahngeschwindigkeit auf kurzzeitige Beschleunigungsvorgänge zu verzichten.

Funktion

Bei aktiver Funktion "Glättung der Bahngeschwindigkeit" ist zugunsten einer ruhigeren Bahngeschwindigkeitsführung ein Glättungsfaktor wirksam, der den maximal zulässigen Produktivitätsverlust bestimmt: Beschleunigungsvorgänge, die weniger als dieser Faktor zu einer kürzeren Programmlaufzeit beitragen, werden nicht durchgeführt. Betrachtet werden dabei nur Beschleunigungsvorgänge, deren Frequenzen oberhalb der parametrierbaren Grenzfrequenzen der beteiligten Achsen liegen.

Vorteile:

- Vermeidung von Anregungen möglicher Maschinenresonanzen aufgrund von ständigen, kurzzeitigen Brems- und Beschleunigungsvorgängen (im Bereich weniger IPO-Takte).
- Vermeidung ständig variierender Schnittgeschwindigkeiten durch Beschleunigungsvorgänge, die keinen großen Gewinn für die Programmlaufzeit bewirken.

Hinweis

Die Glättung der Bahngeschwindigkeit bewirkt keinen Konturfehler.

Schwankungen der Achsgeschwindigkeit aufgrund von Krümmungen in der Kontur bei konstanter Bahngeschwindigkeit können weiterhin auftreten und werden mit dieser Funktion nicht reduziert.

Schwankungen der Bahngeschwindigkeit aufgrund der Vorgabe eines neuen Vorschubs werden ebenfalls nicht verändert. Dies liegt in der Verantwortung des Teileprogrammerstellers.

Voraussetzungen

- Die Glättung der Bahngeschwindigkeit ist nur wirksam im Bahnsteuerbetrieb mit LookAhead über mehrere Sätze bei SOFT und BRISK. Die Glättung ist **nicht** wirksam bei G0.
- Die Taktzeiten der Steuerung müssen so parametrisiert sein, dass der Vorlauf genügend Sätze aufbereiten kann, um einen Beschleunigungsvorgang analysieren zu können.

Aktivierung / Deaktivierung

Die Funktion "Glättung der Bahngeschwindigkeit" wird aktiviert/deaktiviert mit dem Maschinendatum:

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR (Glättungsfaktor bei LookAhead)

Wert	Bedeutung
0.0	Glättung der Bahngeschwindigkeit nicht aktiv (Standardeinstellung)
> 0	Glättung der Bahngeschwindigkeit aktiv

Eine Änderung der MD-Einstellung wird erst durch NEW CONF wirksam.

Parametrierung**Glättungsfaktor**

Der Glättungsfaktor wird eingestellt über das kanalspezifische Maschinendatum:

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR (Glättungsfaktor bei LookAhead)

Der Prozentwert definiert, wieviel ein Bearbeitungsschritt ohne Beschleunigungen/Verzögerungen länger sein darf als der entsprechende Schritt mit Ausführung der Beschleunigungen/Verzögerungen.

Dies wäre ein "worst case"-Wert, wenn alle Beschleunigungsvorgänge innerhalb des Teileprogramms, außer der ersten Anfahrbewegung, weggeglättet werden würden. Die tatsächliche Verlängerung wird in jedem Fall kleiner sein, möglicherweise sogar 0, falls das Kriterium für keinen Beschleunigungsvorgang anspricht. Es können also durchaus Werte von 50 bis 100 % eingetragen werden, ohne eine deutliche Verlängerung der Bearbeitungszeit zu erhalten.

Berücksichtigung des programmierten Vorschubs

Die Glättung der Bahngeschwindigkeit kann mit oder ohne Berücksichtigung des programmierten Vorschubs durchgeführt werden. Die Auswahl erfolgt über das Maschinendatum:

MD20462 \$MC_LOOKAH_SMOOTH_WITH_FEED (Bahnglättung mit programmiertem Vorschub)

Wert	Bedeutung
0	Programmierter Vorschub wird nicht berücksichtigt.
1	Programmierter Vorschub wird berücksichtigt (Standardeinstellung).

Bei Berücksichtigung des programmierten Vorschubs wird der vorgegebene Glättungsfaktor (siehe MD20460) besser eingehalten, wenn der Override auf 100 % steht.

Achsspezifische Grenzfrequenzen

Die achsspezifischen Grenzfrequenzen werden festgelegt über das Maschinendatum:

MD32440 \$MA_LOOKAH_FREQUENCY (Glättungsfrequenz bei LookAhead)

Beschleunigungs- und Bremsvorgänge, die mit einer höheren Frequenz ablaufen, werden abhängig von der Parametrierung der folgenden Maschinendaten geglättet oder in der Dynamik reduziert:

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR (Glättungsfaktor bei LookAhead)

MD20465 \$MC_ADAPT_PATH_DYNAMIC (Adaption der Bahndynamik)

Weitere Informationen zu MD20465 siehe Kapitel "Anpassung der Bahndynamik (Seite 222)".

Hinweis

Werden Schwingungen in der Mechanik einer Achse angeregt und ist deren Frequenz bekannt, so sollte MD32440 kleiner als diese Frequenz eingestellt werden.

Die benötigten Resonanzfrequenzen können z. B. über die integrierten Messfunktionen ermittelt werden.

Funktionsweise

Aus den an der Bahn beteiligten Achsen wird intern das Minimum von MD32440 ermittelt ($= f_{\text{Bahn}}$). Für die Glättung werden nur Beschleunigungsvorgänge betrachtet, bei denen die Anfangs- oder Endgeschwindigkeit dieser Bewegung innerhalb folgender Zeit wieder erreicht wird:

$$t = t_2 - t_1 = 2 / f_{\text{Bahn}}$$

Auf diese Beschleunigungsvorgänge wird verzichtet, wenn die sich dadurch ergebende Verlängerung der Bearbeitungszeit den über den Glättungsfaktor (MD20460) vorgegebenen Grenzwert nicht überschreitet.

Beispiel

Folgende Parametrierung ist gegeben:

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR = 10 %

MD32440 \$MA_LOOKAH_FREQUENCY[AX1] = 20 Hz

MD32440 \$MA_LOOKAH_FREQUENCY[AX2] = 20 Hz

MD32440 \$MA_LOOKAH_FREQUENCY[AX3] = 10 Hz

An der Bahn sind die 3 Achsen X = AX1, Y = AX2, Z = AX3 beteiligt.

Das Minimum vom MD32440 dieser 3 Achsen beträgt somit 10 Hz. Daraufhin werden Beschleunigungsvorgänge untersucht, die in einem Zeitraum $t_2 - t_1 = 2 / 10 \text{ Hz} = 200 \text{ ms}$ ablaufen. Die Zeit t_2 ist diejenige Zeit, zu der nach einem Beschleunigungsvorgang von der Geschwindigkeit v_1 ausgehend diese Geschwindigkeit v_1 wieder erreicht wird. Nur dieser Bereich wird auch für die Verlängerung der Bearbeitungszeit betrachtet.

Wenn der Zeitraum $t_2 - t_1$ größer als 200 ms ist oder wenn die zusätzliche Programmbearbeitungszeit $t_3 - t_2$ mehr als 10 % (= MD20460) von $t_2 - t_1$ beträgt, ergibt sich folgender Zeitverlauf:

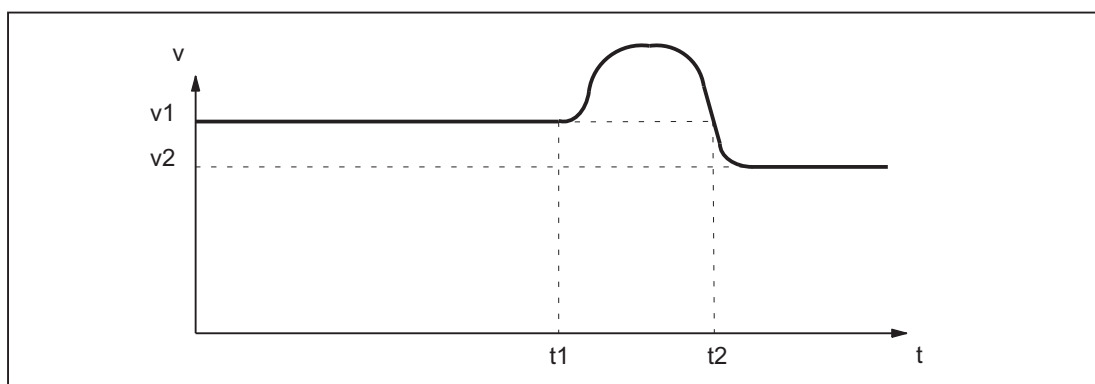


Bild 5-7 Verlauf der zeitoptimalen Bahngeschwindigkeit (ohne Glättung)

Wenn dagegen der Zeitraum $t_2 - t_1$ kleiner als 200 ms ist und wenn die zusätzliche Programmbearbeitungszeit $t_3 - t_2$ maximal 10 % von $t_2 - t_1$ beträgt, dann ergibt sich dieser Zeitverlauf:

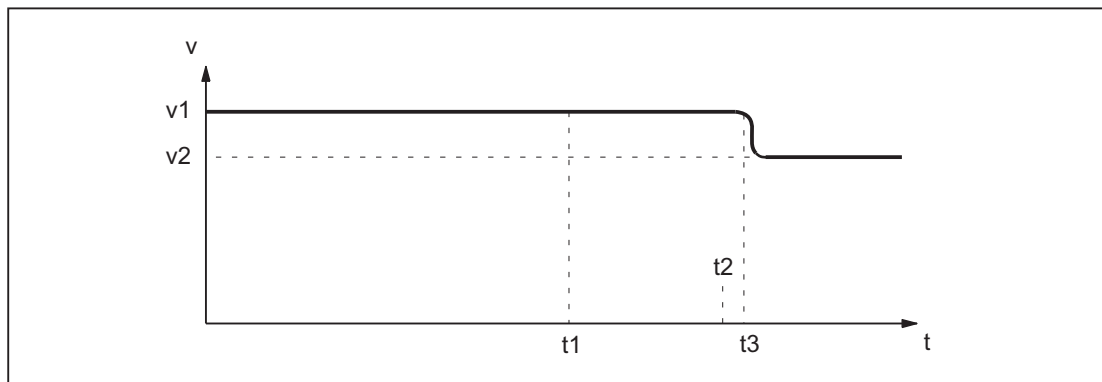


Bild 5-8 Verlauf der geglätteten Bahngeschwindigkeit

5.4.2 Anpassung der Bahndynamik

Funktion

Hochdynamische Beschleunigungs- und Bremsvorgänge während der Bearbeitung können zur Anregung von mechanischen Schwingungen von Maschinenelementen und in Folge zu einer Verminderung der Oberflächengüte des Werkstücks führen.

Mit der Funktion "Anpassung der Bahndynamik" kann die Dynamik der Beschleunigungs- und Bremsvorgänge an die Maschinengegebenheiten angepasst werden.

Hinweis

Die Funktion "Anpassung der Bahndynamik" betrachtet nicht die Brems- und Beschleunigungsvorgänge der einzelnen an der Bahn beteiligten Achsen, sondern immer der resultierenden Bahn. Daher kann es auch bei einem konstanten Bahngeschwindigkeitsverlauf durch un stetige Konturverläufe oder kinematische Transformationen zu bezüglich der Anregung von mechanischen Schwingungen kritischen Brems- und Beschleunigungsvorgängen der Achsen kommen.

Wirksamkeit

Die Funktion "Anpassung der Bahndynamik" ist nur während Bahnbewegungen wirksam:

- Bahnsteuerbetrieb (G64, G64x)
Im Bahnsteuerbetrieb wird die optimale Wirkung der Dynamikanpassung bei einem wirksamen Override von 100% erreicht. Deutliche Abweichungen von diesem Wert oder Funktionen, durch die die Bahnachsen abgebremst werden müssen (z. B. Hilfsfunktionsausgaben an die PLC), vermindern die gewünschte Wirkung stark.
- Genauhalt (G60)

Unter folgenden Bedingungen ist die Funktion "Anpassung der Bahndynamik" auch bei Bahnbewegungen **nicht** wirksam:

- Programmierem Eilgang (G0)
- Veränderungen des Override-Werts
- Stoppanforderungen während der Bewegung (z. B. NC-Stop, NC-Reset)
- Funktion "Geschwindigkeitsabhängige Bahnbeschleunigung" (DRIVE) ist aktiv

Aktivierung / Deaktivierung

Die Funktion wird aktiviert/deaktiviert mit dem Maschinendatum:

MD20465 \$MC_ADAPT_PATH_DYNAMIC (Adaption der Bahndynamik)

Wert	Bedeutung
= 1.0	Dynamikanpassung nicht aktiv (Standardeinstellung)
> 1.0	Dynamikanpassung aktiv

Durch die Aktivierung wird im Bahnsteuerbetrieb intern **immer die Funktion "Glättung der Bahngeschwindigkeit" mitaktiviert** (siehe Kapitel "Glättung der Bahngeschwindigkeit (Seite 218)").

Falls der Glättungsfaktor (MD20460 \$MC_LOOKAH_SMOOTH_FACTOR) auf 0 % eingestellt ist (= Funktion deaktiviert; Voreinstellung!), wird als Ersatz ein Glättungsfaktor von 100 % verwendet. Bei einem von 0 % verschiedenen Glättungsfaktor wird der eingestellte Wert wirksam.

Parametrierung

Anpassfaktor der Bahndynamik

Durch den Anpassfaktor der Bahndynamik werden kurzzeitige Veränderungen der Bahngeschwindigkeit mit verminderten Dynamikgrenzwerten durchgeführt.

Der Anpassfaktor ist kanalspezifisch einstellbar:

- Für Verfahrbewegungen mit Beschleunigung ohne Ruckbegrenzung (**BRISK**) über:
MD20465 \$MC_ADAPT_PATH_DYNAMIC[0]
→ Der Anpassfaktor wirkt auf die Beschleunigung.
- Für Verfahrbewegungen mit Beschleunigung mit Ruckbegrenzung (**SOFT**) über:
MD20465 \$MC_ADAPT_PATH_DYNAMIC[1]
→ Der Anpassfaktor wirkt auf den Ruck.

Achsspezifische Grenzfrequenzen

Die Dynamikbegrenzung soll nur bei den Brems- und Beschleunigungsvorgängen wirken, die mechanische Schwingungen größer einer bestimmten Grenzfrequenz auslösen und dadurch zu einer Anregung von Maschinenresonanzen führen.

Die Grenzfrequenz, ab der die Dynamikbegrenzung einsetzen soll, wird achsspezifisch vorgegeben über das Maschinendatum:

MD32440 \$MA_LOOKAH_FREQUENCY (Glättungsfrequenz bei LookAhead)

Weitere Informationen siehe Kapitel "Glättung der Bahngeschwindigkeit (Seite 218)".

Funktionsweise

Bei der Bearbeitung ermittelt die Steuerung zyklisch über alle an der Bahn beteiligten Achsen das Minimum aller Grenzfrequenzen als die für die Dynamikanpassung relevante Grenzfrequenz (f) und berechnet daraus das relevante Zeitfenster (t_{adapt}):

$$t_{\text{adapt}} = 1 / f$$

Die Größe des relevanten Zeitfensters t_{adapt} bestimmt das weitere Verhalten:

1. Die erforderliche Zeit für die Geschwindigkeitsänderung ist kleiner als t_{adapt} :
Die Beschleunigungen werden reduziert um einen Faktor > 1 und \leq dem Wert im Maschinendatum:
MD20465 ADAPT_PATH_DYNAMIC (Adaption der Bahndynamik)
Durch die geringere Beschleunigung verlängert sich die Zeit für die Geschwindigkeitsänderung.
Folgende Fälle werden unterschieden:
 - Die Beschleunigung wird mit einem Wert kleiner als MD20465 so verringert, dass der Vorgang t_{adapt} [s] dauert. Die erlaubte Reduktion muss nicht voll ausgenutzt werden.
 - Die Beschleunigungszeit wird mit dem Wert im MD20465 verringert. Der Vorgang dauert trotz der geringeren Beschleunigung kürzer als t_{adapt} . Die erlaubte Reduktion wurde voll ausgenutzt.
2. Die erforderliche Zeit für die Geschwindigkeitsänderung ist größer als t_{adapt} :
Es ist keine Dynamikanpassung erforderlich.

Beispiel

Das folgende Beispiel soll die Auswirkung der Funktion "Anpassung der Bahndynamik" auf Verfahrbewegungen mit Beschleunigung ohne Ruckbegrenzung (BRISK) veranschaulichen.

Folgende Parametrierung ist gegeben:

MD20465 \$MC_ADAPT_PATH_DYNAMIC[0] = 1.5	
MD20460 \$MC_LOOKAH_SMOOTH_FACTOR = 1.0	
MD32440 \$MA_LOOKAH_FREQUENCY[AX1] = 20 Hz	$T_{\text{AX1}} = 1/20 \text{ Hz} = 50 \text{ ms}$
MD32440 \$MA_LOOKAH_FREQUENCY[AX2] = 10 Hz	$T_{\text{AX2}} = 1/10 \text{ Hz} = 100 \text{ ms}$
MD32440 \$MA_LOOKAH_FREQUENCY[AX3] = 20 Hz	$T_{\text{AX3}} = 1/20 \text{ Hz} = 50 \text{ ms}$

Hinweis

Zur Verdeutlichung der Auswirkung der Dynamikanpassung ist der Wert für den Glättungsfaktor (MD20460) auf "1" gesetzt, wodurch die mitaktivierte Funktion "Glättung der Bahngeschwindigkeit" praktisch nicht aktiv ist.

An der Bahn sind die 3 Achsen $X = \text{AX1}$, $Y = \text{AX2}$, $Z = \text{AX3}$ beteiligt.

Bei Bahnbewegungen, an denen die Achse AX2 beteiligt ist, werden alle Brems- und Beschleunigungsvorgänge angepasst, die weniger als T_{AX2} dauern würden.

Sind bei Bahnbewegungen nur die Achsen AX1 und/oder AX3 beteiligt, werden alle Brems- und Beschleunigungsvorgänge angepasst, die weniger als $T_{\text{AX1}} = T_{\text{AX3}}$ dauern würden.

Das relevante Zeitfenster wird in den folgenden Bildern mit t_{adapt} bezeichnet.

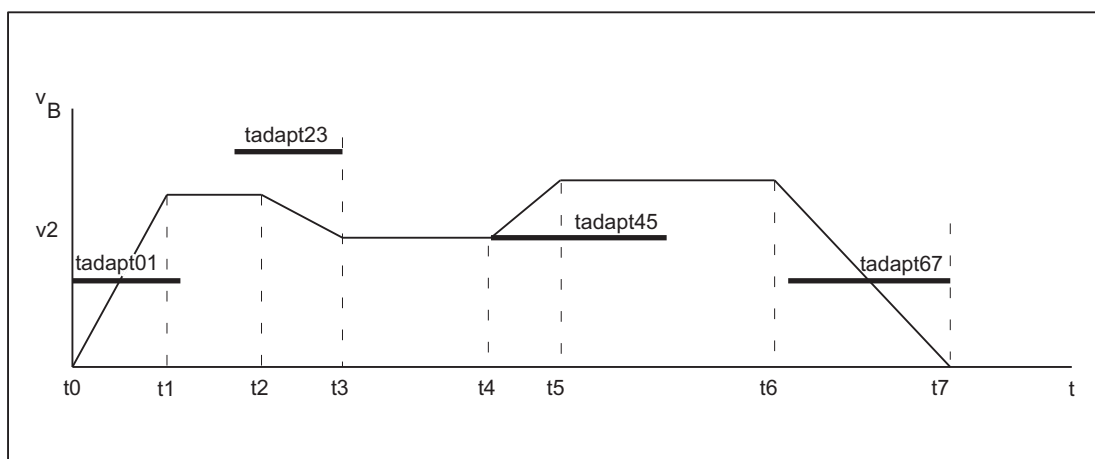


Bild 5-9 Zeitoptimaler Bahngeschwindigkeitsverlauf ohne Glättung und Dynamikanpassung

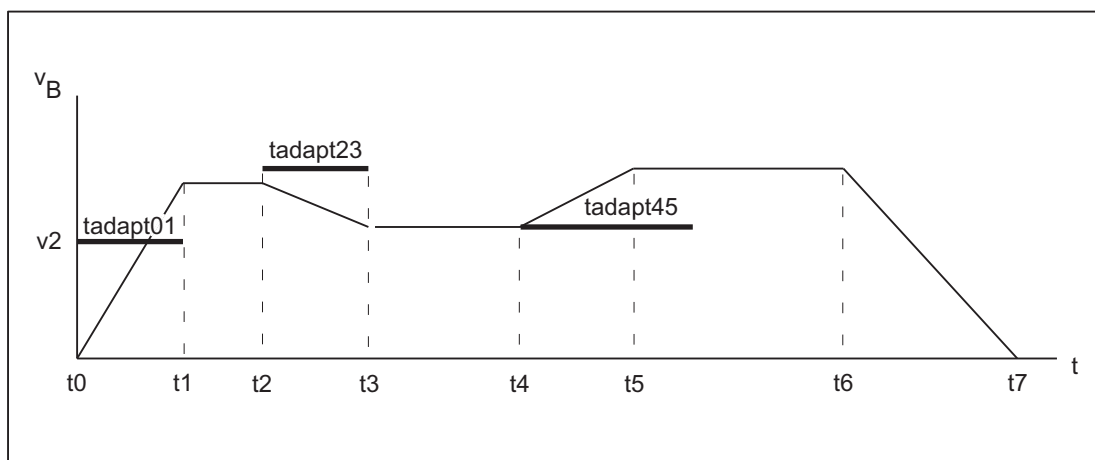


Bild 5-10 Bahngeschwindigkeitsverlauf mit Anpassung der Bahndynamik

Intervalle $t_0 - t_1$ und $t_2 - t_3$:

Der Beschleunigungsvorgang zwischen $t_0 - t_1$ und der Bremsvorgang zwischen $t_2 - t_3$ werden durch eine Anpassung der Beschleunigung auf die Zeit t_{adapt01} bzw. t_{adapt23} zeitlich verlängert.

Intervall $t_4 - t_5$:

Der Beschleunigungsvorgang zwischen $t_4 - t_5$ wird mit einer um den maximalen Anpassfaktor 1,5 reduzierten Beschleunigung durchgeführt. Dennoch ist der Beschleunigungsvorgang vor der Zeit t_{adapt45} beendet.

Intervall $t_6 - t_7$:

Der Bremsvorgang zwischen $t_6 - t_7$ bleibt unverändert, da er länger als t_{adapt67} dauert.

5.4.3 Ermittlung der Dynamikgrenzwerte

Die Inbetriebnahme der Funktion "Anpassung der Bahndynamik" erfordert neben der Ermittlung der Eigenfrequenz der Bahnachsen zur Parametrierung der achsspezifischen Grenzfrequenzen (MD32440 \$MA_LOOKAH_FREQUENCY) auch die Ermittlung der Dynamikgrenzwerte für Geschwindigkeit, Beschleunigung und Ruck.

Vorgehensweise

Im Folgenden ist die Ermittlung der Dynamikgrenzwerte für das Verfahren der Bahnachsen mittels Beschleunigung mit Ruckbegrenzung (SOFT) beschrieben. Das Vorgehen kann sinngemäß auf den Fall der Beschleunigung ohne Ruckbegrenzung (BRISK) übertragen werden.

1. Deaktivieren Sie die Funktion "Anpassung der Bahndynamik":
MD20465 \$MC_ADAPT_PATH_DYNAMIC [1] = 1
2. Untersuchen Sie das Positionierverhalten der einzelnen Bahnachsen bei unterschiedlichen Verfahrgeschwindigkeiten. Stellen Sie dabei den Ruck so ein, dass die gewünschte Positioniertoleranz eingehalten wird.

Hinweis

Je höher die Verfahrgeschwindigkeit ist, von der aus der Positioniervorgang begonnen wird, desto höher kann im Allgemeinen der Ruck eingestellt werden.

3. Übernehmen Sie den für die unkritischste Verfahrgeschwindigkeit ermittelten maximal zulässigen Ruck:
MD32431 \$MA_MAX_AX_JERK (Maximaler Ruck)
4. Ermitteln Sie für alle Bahnachsen den Faktor F_{APD} mit:
 $F_{APD} = (\text{größter ermittelter Ruck}) / (\text{kleinster ermittelter Ruck})$

Hinweis

Der kleinste ermittelte Ruck ist der Wert für den Ruck bei der kritischsten Verfahrgeschwindigkeit.

5. Tragen Sie den über alle Bahnachsen ermittelten größten Faktor F_{APD} als Wert für den Anpassfaktor der Bahndynamik ein:
MD20465 \$MC_ADAPT_PATH_DYNAMIC [1] = F_{APD}

5.4.4 Zusammenwirken der Funktionen "Glättung der Bahngeschwindigkeit" und "Anpassung der Bahndynamik"

Die folgenden Beispiele sollen das Zusammenwirken der Funktionen "Glättung der Bahngeschwindigkeit" und "Anpassung der Bahndynamik" im Bahnsteuerbetrieb veranschaulichen.

Beispiel 1

Beschleunigungsmodus: BRISK

An der Bahn sind die 3 Achsen X = AX1, Y = AX2, Z = AX3 beteiligt.

Folgende Parametrierung ist gegeben:

MD20465 \$MC_ADAPT_PATH_DYNAMIC[0] = 3

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR = 80.0

MD32440 \$MA_LOOKAH_FREQUENCY[AX1] = 20 $T_{AX1} = 1/20 \text{ Hz} = 50 \text{ ms}$

MD32440 \$MA_LOOKAH_FREQUENCY[AX2] = 20 $T_{AX2} = 1/20 \text{ Hz} = 50 \text{ ms}$

MD32440 \$MA_LOOKAH_FREQUENCY[AX3] = 20 $T_{AX3} = 1/20 \text{ Hz} = 50 \text{ ms}$

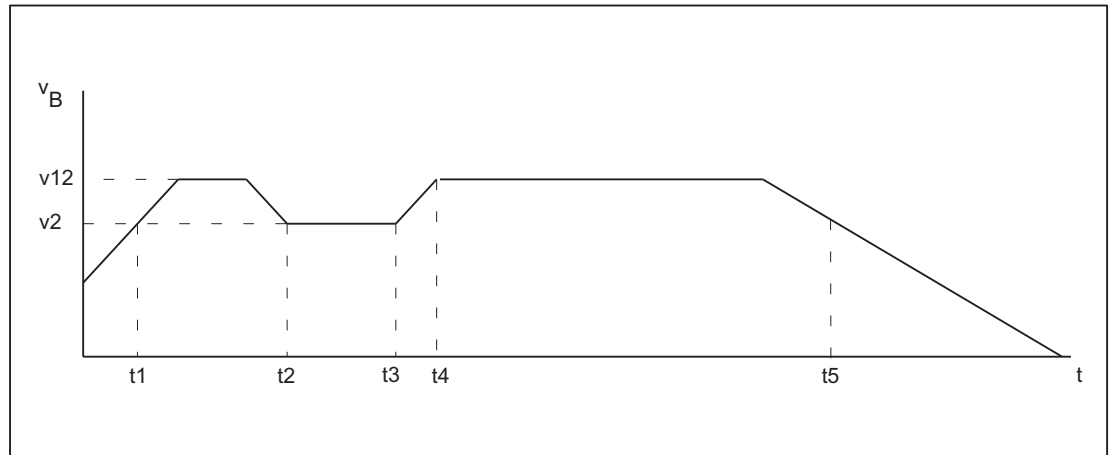


Bild 5-11 Zeitoptimaler Bahngeschwindigkeitsverlauf ohne Glättung und Dynamikanpassung

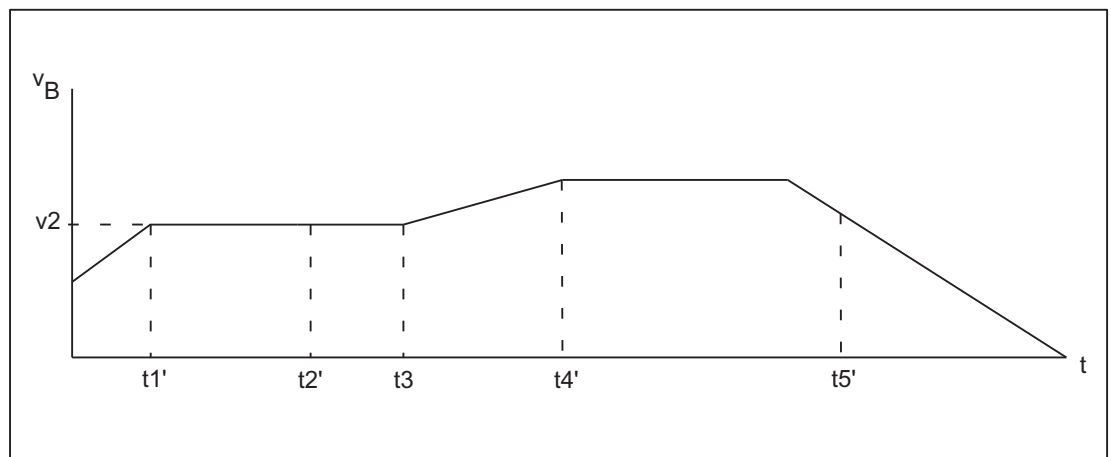


Bild 5-12 Bahngeschwindigkeitsverlauf mit Glättung der Bahngeschwindigkeit und Anpassung der Bahndynamik

Auswirkungen der Glättung der Bahngeschwindigkeit:

- Intervall $t_1 - t_2$: Der Beschleunigungs- und Bremsvorgang zwischen $t_1 - t_2$ entfällt, da die Verlängerung der Bearbeitungszeit ohne den Beschleunigungsvorgang auf v_{12} kleiner als die sich mittels Glättungsfaktor von 80 % ergebende Zeit ist.
- Intervall $t_3 - t_5$: Der Beschleunigungs- und Bremsverlauf zwischen $t_3 - t_5$ erfüllt diese Bedingung nicht oder dauert länger als die parametrisierte Glättungszeit $T_{AXn} = 2/20 \text{ Hz} = 100 \text{ ms}$.

Auswirkungen der Dynamikanpassung:

- Intervall $t_3 - t_4$: Der Beschleunigungsvorgang zwischen $t_3 - t_4$ ist kürzer als $\text{MIN}(T_{AXn}) = 1/20 \text{ Hz} = 50 \text{ ms}$ und wird daher mit einer um den Anpassfaktor 3 geringeren Beschleunigung durchgeführt.
- Intervall bis t_1 : Der nach der Bahnglättung verbleibende Beschleunigungsvorgang bis t_1 wird durch die Dynamikanpassung auf den Zeitraum bis t_1' gedehnt.

Hinweis

Das Beispiel zeigt, dass diejenigen Beschleunigungs- oder Bremsvorgänge, die von der Glättung der Bahngeschwindigkeit nicht eliminiert werden, anschließend noch durch die Anpassung der Bahndynamik optimiert werden können. Daher sollten immer möglichst beide Funktionen aktiviert werden.

Beispiel 2

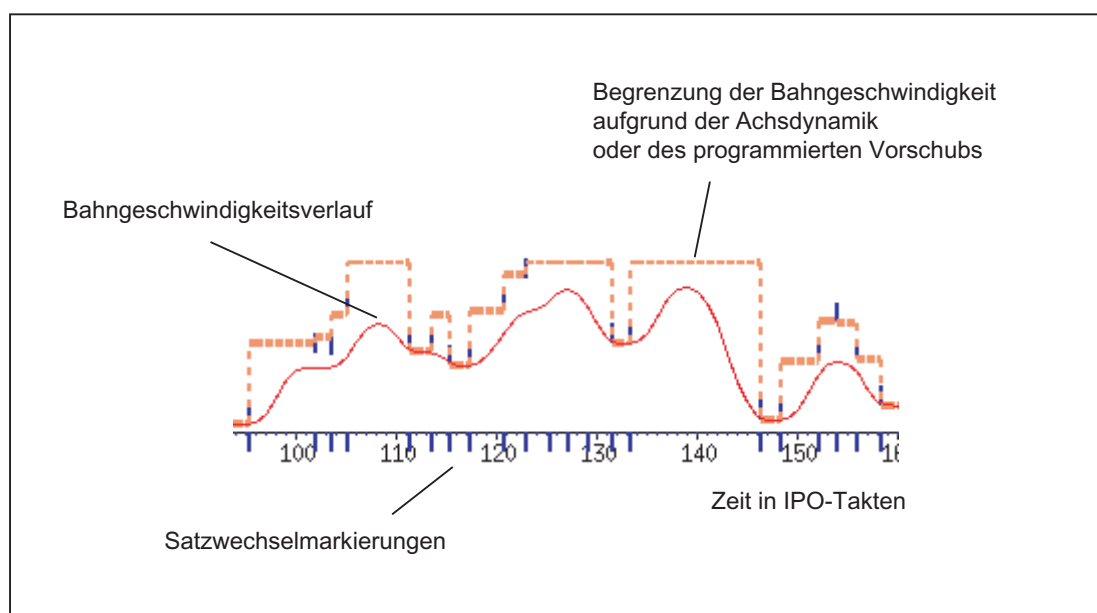
Beschleunigungsmodus: SOFT

An der Bahn sind die 3 Achsen $X = AX1$, $Y = AX2$, $Z = AX3$ beteiligt.

Folgende Parametrierung ist gegeben:

```
MD20465 $MC_ADAPT_PATH_DYNAMIC[1] = 1
MD20460 $MC_LOOKAH_SMOOTH_FACTOR = 0.0
MD32440 $MA_LOOKAH_FREQUENCY[AX1] = 10      TAX1 = 1/20 Hz = 100 ms
MD32440 $MA_LOOKAH_FREQUENCY[AX2] = 10      TAX2 = 1/20 Hz = 100 ms
MD32440 $MA_LOOKAH_FREQUENCY[AX3] = 20      TAX3 = 1/20 Hz = 50 ms
```

Dies führt zu einem zeitoptimalen Bahngeschwindigkeitsverlauf ohne Glättung der Bahngeschwindigkeit und ohne Anpassung der Bahndynamik:

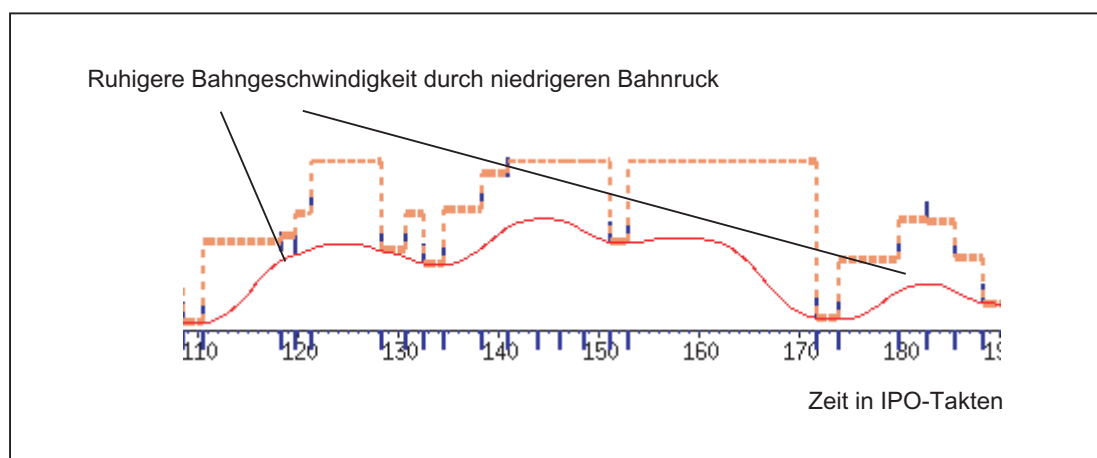


Die Parametrierung wird wie folgt geändert:

MD20465 \$MC_ADAPT_PATH_DYNAMIC[1] = 4

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR = 1.0

Daraus ergibt sich ein Bahngeschwindigkeitsverlauf mit Anpassung der Bahndynamik und minimaler und damit fast abgeschalteter Glättung der Bahngeschwindigkeit:

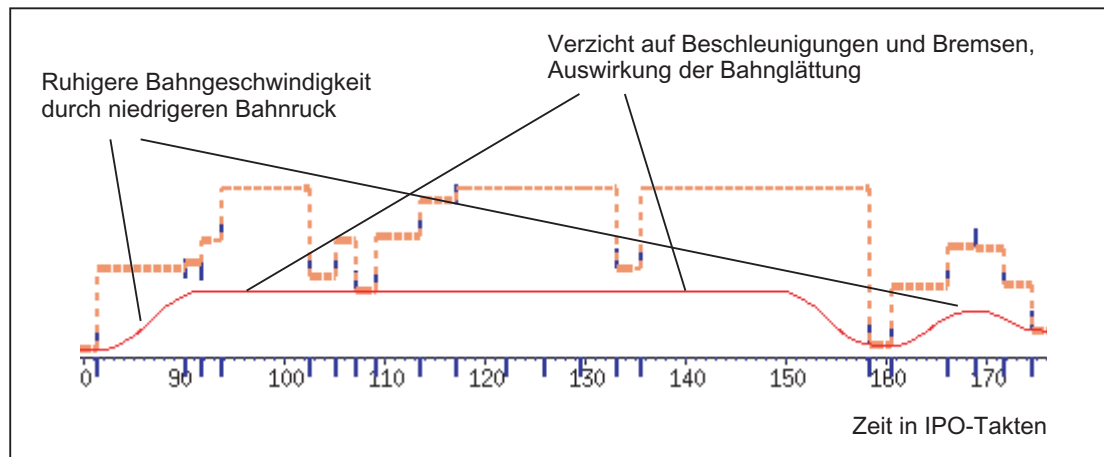


Der Glättungsfaktor wird statt auf 1 % auf 0 % eingestellt (entspricht der Voreinstellung!):

MD20460 \$MC_LOOKAH_SMOOTH_FACTOR = 0.0

Mit dieser Parametrierung wird ein Glättungsfaktor von 100 % wirksam.

Daraus ergibt sich ein Bahngeschwindigkeitsverlauf mit Glättung der Bahngeschwindigkeit und Anpassung der Bahndynamik:



5.4.5 Dynamikmodus für Bahninterpolation

Funktion

Technologie-spezifische Dynamikeinstellungen können in Maschinendaten hinterlegt und im Teileprogramm über die Befehle der G-Gruppe 59 (Dynamikmodus für Bahninterpolation) aktiviert werden.

Befehl	Aktiviert die Dynamikeinstellungen für:
DYNNORM	Standard-Dynamikeinstellungen
DYNPOS	Positionierbetrieb, Gewindebohren
DYNROUGH	Schruppen
DYNSEMIFIN	Schlichten
DYNFINISH	Feinschlichten

Hinweis

Mit den Befehlen der G-Gruppe 59 (Dynamikmodus für Bahninterpolation) wird ausschließlich die Dynamik der **Bahnachsen** bestimmt. Sie haben keinen Einfluss auf:

- Positionierachsen
- PLC-Achsen
- Kommandoachsen
- Bewegungen aufgrund von Achskopplungen
- Überlagerte Bewegungen mit Handrad
- JOG-Bewegungen
- Referenzpunktfahren (G74)
- Festpunktfahren (G75)
- Eilgangbewegung (G0)

Für diese Achsbewegungen ist immer die Standard-Dynamikeinstellung (DYNNORM) wirksam.

Anwendung

Durch Umschalten der Dynamikeinstellungen ist es z. B. möglich, ein Werkstück beim Schruppen zeitoptimiert und beim Schlichten oberflächenoptimiert zu bearbeiten.

Parametrierung**Achsspezifische Dynamikeinstellungen:**

- MD32300 \$MA_MAX_AX_ACCEL[<n>] (Achsbeschleunigung)
- MD32310 \$MA_MAX_ACCEL_OVL_FACTOR[<n>] (Überlastfaktor für axiale Geschwindigkeitssprünge)
- MD32431 \$MA_MAX_AX_JERK[<n>] (maximaler axialer Ruck bei Bahnbewegung)
- MD32432 \$MA_PATH_TRANS_JERK_LIM[<n>] (maximaler axialer Ruck am Satzübergang im Bahnsteuerbetrieb)
- MD32433 \$MA_SOFT_ACCEL_FACTOR[<n>] (Skalierung der Beschleunigungsbegrenzung bei SOFT)

Kanalspezifische Dynamikeinstellungen:

- MD20600 \$MC_MAX_PATH_JERK[<n>] (Bahnbezogener Maximalruck)
- MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL[<n>] (Einfluss der Bahnkrümmung auf die Bahnbeschleunigung)
- MD20603 \$MC_CURV_EFFECT_ON_PATH_JERK[<n>] (Einfluss der Bahnkrümmung auf den Bahnruck)

mit Index <n> = 0 für DYNNORM
 1 für DYNPOS

- 2 für DYNROUGH
- 3 für DYNSEMIFIN
- 4 für DYNFINISH

Hinweis

Das **Beschreiben** der Maschinendaten **ohne Index** setzt den gleichen Wert in alle Feldelemente des betreffenden Maschinendatums.

Das **Lesen** der Maschinendaten **ohne Index** liefert immer den Wert des Feldes mit dem Index 0.

G-Befehle unterdrücken

Es wird empfohlen, nicht zur Verwendung vorgesehene G-Befehle der G-Gruppe 59 (Dynamikmodus für Bahninterpolation) über das folgende Maschinendatum zu unterdrücken:

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[<n>] (Liste umprojektierter NC-Befehle)

Bei Verwendung eines unterdrückten G-Befehls wird ein Alarm angezeigt. Dadurch wird verhindert, dass nicht-parametrierte Maschinendaten wirksam werden.

Beispiel

Mit folgenden Einstellungen können die G-Befehle `DYNPOS` und `DYNSEMIFIN` unterdrückt werden:

- MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[0]="DYNPOS"
- MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[1]=" "
- MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[2]="DYNSEMIFIN"
- MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[3]=" "

Literatur

Ausführliche Informationen zur Programmierung der G-Befehle der G-Gruppe 59 (Dynamikmodus für Bahninterpolation) finden sich in:

Literatur:

Programmierhandbuch Arbeitsvorbereitung; Kapitel: Bahnverhalten

5.4.6 Freiformflächenmodus: Grundfunktionen

Einleitung

Bei Anwendungen im Werkzeug- und Formenbau ist es wichtig, dass möglichst gleichmäßige Oberflächen auf dem Werkstück entstehen. Diese Anforderung ist meistens wichtiger als die Genauigkeit der Werkstückoberfläche.

Ungleichmäßige Werkstückoberflächen können z. B. folgende Ursachen haben:

- Das Teileprogramm zur Fertigung des Werkstücks enthält eine ungleichmäßige Geometrie. Dies betrifft vor allem den Verlauf der Krümmung und der Torsion.

Hinweis

Die Krümmung k einer Kontur ist das Inverse des Radius r des anschließenden Kreises in einem Konturpunkt ($k = 1/r$). Die Torsion ist die Änderung der Krümmung (1. Ableitung).

Durch die ungleichmäßige Geometrie werden bei der Abarbeitung des Teileprogramms die Dynamikgrenzen der Maschine erreicht und es kommt zu unnötigen Brems- und Beschleunigungsvorgängen. Dies führt je nach Größe des effektiven Nachlaufs der Achsen zu unterschiedlichen Konturabweichungen.

- Unnötige Brems- und Beschleunigungsvorgänge können zur Anregung von Schwingungen der Maschine führen, die als unerwünschte Marken auf dem Werkstück sichtbar werden.

Zur Behebung dieser Ursachen gibt es verschiedene Möglichkeiten:

- CAD-/CAM-System verwenden
Von CAD-/CAM-Systemen generierte Teileprogramme enthalten einen sehr gleichmäßigen Verlauf der Krümmung und der Torsion, so dass es nicht zu unnötigen Reduzierungen der Bahngeschwindigkeit kommen kann.
- Maximale Bahngeschwindigkeit so bestimmen, dass ungewollte geometrische Schwankungen des Verlaufs der Krümmung und Torsion keinen Einfluss haben.

Funktion

Mit der Funktion "Freiformflächenmodus: Grundfunktionen" kann die Festlegung der Begrenzung der Bahngeschwindigkeit unempfindlicher gegenüber kleinen geometrischen Schwankungen der Krümmung bzw. Torsion gemacht werden, ohne dass die dynamischen Begrenzungen der Maschine bezüglich der Beschleunigung und des Rucks der Achsen überschritten werden.

Dadurch ergeben sich folgende Vorteile:

- Gleichmäßigerer Verlauf der Bahngeschwindigkeit
- Gleichmäßigere Oberfläche des Werkstücks
- Verringerung der Bearbeitungszeit (falls die Dynamik der Maschine dies zulässt)

Anwendungen

Die Funktion wird zur Bearbeitung von Werkstücken verwendet, die vorwiegend aus Freiformflächen bestehen.

Voraussetzungen

Die Funktion ist nur aktivierbar, wenn bei der Speicherkonfiguration der notwendige Speicher reserviert wird:

MD28610 \$MC_MM_PREPDYN_BLOCKS = 10

Der eingetragene Wert gibt die Anzahl von Sätzen vor, die bei der Festlegung der Bahngeschwindigkeit (Geschwindigkeitspräparation) berücksichtigt werden sollen. Ein sinnvoller Wert ist "10".

Hat MD28610 den Wert "0", dann werden zur Festlegung der maximalen Bahngeschwindigkeit eines Satzes nur die jeweiligen Bewegungen der Achsen in diesem Satz berücksichtigt. Wird bei der Festlegung der Bahngeschwindigkeit auch die Geometrie in benachbarten Sätzen berücksichtigt (Wert > 0), erhält man einen gleichmäßigeren Verlauf der Bahngeschwindigkeit.

Aktivierung / Deaktivierung

Die Funktion kann für jeden Dynamikmodus (siehe Kapitel "Dynamikmodus für Bahninterpolation (Seite 230)") unabhängig ein- oder ausgeschaltet werden:

MD20606 \$MC_PREPDYN_SMOOTHING_ON[<n>] = <Wert>

Index <n>	Dynamikmodus	<Wert>	Freiformflächenmodus: Grundfunktionen
0	Standard-Dynamikeinstellungen (DYNNORM)	0	aus
		1	ein
1	Positionierbetrieb, Gewindebohren (DYNPOS)	0	aus
		1	ein
2	Schruppen (DYNROUGH)	0	aus
		1	ein
3	Schlichten (DYNSEMIFIN)	0	aus
		1	ein
4	Feinschlichten (DYNFINISH)	0	aus
		1	ein

Hinweis

Die Funktion sollte wegen dem zusätzlichen Speicherbedarf nur in den relevanten Bearbeitungskanälen aktiviert werden.

Parametrierung

Veränderung des Konturabtastfaktors

Bei der Interpolation von gekrümmten Konturen ist der dabei entstehende Sekantenfehler von folgenden Faktoren abhängig:

- Krümmung
- Interpolatortakt (Anzeige im MD10071 \$MN_IPO_CYCLE_TIME)
- Geschwindigkeit, mit der die betreffende Kontur abgefahren wird

Der maximal mögliche Sekantenfehler ist für jede Achse festgelegt im Maschinendatum:

MD33100 \$MA_COMPRESS_POS_TOL (Maximale Abweichung bei Kompression)

Wenn der eingestellte Interpolatortakt nicht ausreichend klein ist, kann es vorkommen, dass bei stärker gekrümmten Konturen die maximale Bahngeschwindigkeit reduziert wird. Dies ist

notwendig, damit auch in diesem Fall die Oberfläche des Werkstücks noch ausreichend genau hergestellt wird.

Durch die Veränderung des Konturabtastfaktors ist es möglich, das Zeitintervall, mit dem eine gekrümmte Kontur im Interpolator abgetastet wird (Konturabtastzeit), unterschiedlich zum Interpolatorakt einzustellen. Eine Konturabtastzeit niedriger als der Interpolatorakt kann so bei stärker gekrümmten Konturen eine Reduzierung der Bahngeschwindigkeit verhindern.

Der Konturabtastfaktor wird eingestellt mit dem Maschinendatum:

MD10682 \$MN_CONTOUR_SAMPLING_FACTOR

Die wirksame Konturabtastzeit berechnet sich daraus wie folgt:

$$T_s = f * T_1$$

mit: T_s = wirksame Konturabtastzeit
 T_1 = Interpolatorakt
 f = Kontur-Abtastfaktor (Wert aus MD10682)

Der Standardwert des Konturabtastfaktors ist "1", d. h. die Konturabtastzeit ist gleich dem Interpolatorakt.

Der Konturabtastfaktor kann sowohl größer als auch kleiner "1" sein.

Wird ein Wert kleiner "1" eingestellt, kann man erreichen, dass die Überprüfung der Genauigkeit der Konturabtastung außer Kraft gesetzt wird.

Die eingestellte Abtastzeit darf die projektierte minimale Konturabtastzeit nicht unterschreiten:

MD10680 \$MN_MIN_CONTOUR_SAMPLING_TIME

Hinweis

MD10680 ist für jedes Steuerungsmodell spezifisch eingestellt und nicht veränderbar.

Programmierung

Abhängig von der Einstellung im Maschinendatum

MD20606 \$MC_PREPDYN_SMOOTHING_ON kann die Funktion "Freiformflächenmodus: Grundfunktionen" im Teileprogramm durch einen Wechsel des aktiven Dynamikmodus ein- oder ausgeschaltet werden.

Beispiel:

Durch die Parametrierung von MD20606 \$MC_PREPDYN_SMOOTHING_ON[2-4] = 1 und MD20606 \$MC_PREPDYN_SMOOTHING_ON[0-1] = 0 kann die Funktion über die Befehle DYNROUGH, DYNSEMIFIN und DYNFINISH eingeschaltet und über die Befehle DYNNORM und DYNPOS ausgeschaltet werden.

Siehe auch

Überschleifen tangentialer Satzübergänge (G645) (Seite 206)

Geschwindigkeitsabhängige Ruckadaption (achsspezifisch) (Seite 290)

Freiformflächenmodus: Erweiterungsfunktion (Seite 214)

5.5 Kompressor-Funktionen

5.5.1 Kompression von Linear-, Kreis- und Eilgangsätzen

5.5.1.1 Funktion

CAD/CAM-Systeme erzeugen zur Beschreibung von komplexen Konturen eine große Anzahl von Linear- und Kreissätzen mit zum Teil sehr kurzen Bahnlängen. Die maximal mögliche Bahngeschwindigkeit wird dabei häufig durch die Satzwechselzeit begrenzt. Ab einer bestimmten Bahngeschwindigkeit können dann vom Vorlauf nicht mehr schnell genug neue Verfahrssätze aufbereitet und in den Hauptlauf eingewechselt werden.

Kompression von Linearsätzen

Kompressor-Funktionen ersetzen aufeinander folgende Linearsätze durch Polynom-Sätze mit möglichst großen Bahnlängen unter Einhaltung einer parametrierbaren Konturgenauigkeit. Dadurch ergeben sich folgende Vorteile:

- Reduzierung der Anzahl von Verfahrssätzen
- Erhöhung der Bahngeschwindigkeit
- Erhöhung der Oberflächengüte
- Stetige Satzübergänge

In der folgenden Tabelle sind die zur Verfügung stehenden Kompressor-Funktionen mit ihren wichtigsten Eigenschaften aufgelistet:

Kompressor	Funktion	Stetigkeit an Satzübergängen	Hinweise zur Anwendung
COMPON	COMPON erzeugt aus bis zu 10 aufeinander folgenden Linearsätzen der Form "G01 X... Y... Z... F..." einen Polynom-Satz.	geschwindigkeitsstetig	
COMPCURV	wie COMPON	geschwindigkeits- und beschleunigungsstetig	

Kompressor	Funktion	Stetigkeit an Satzübergängen	Hinweise zur Anwendung
COMPCAD	COMPCAD kann aus beliebig vielen aufeinander folgenden Linearsätzen einen Polynom-Satz erzeugen.	geschwindigkeits- und beschleunigungsstetig	COMPCAD ist sehr rechenzeit- und speicherplatzintensiv. Es wird daher empfohlen, COMPCAD nur dort einzusetzen, wo Maßnahmen zur Oberflächenverbesserung im CAD/CAM-Programm nicht zum Erfolg führen.
COMPSURF	wie COMPCAD	geschwindigkeits- und beschleunigungsstetig	Mit COMPSURF können noch bessere Ergebnisse als mit COMPCAD erzielt werden. Ebenso wie COMPCAD ist COMPSURF jedoch sehr rechenzeit- und speicherplatzintensiv. Durch den Einsatz von COMPSURF können z. B. bei schräg abgezeilten Schlichtprogrammen, schlechter Datenqualität und/oder unregelmäßiger Punkteverteilung im CAD/CAM-Programm deutlich bessere Werkstückoberflächen erhalten werden. Außerdem ermöglicht COMPSURF eine richtungsunabhängige identische abschaltbare Glättung der Fräsbahnen, was beim bidirektionalen Fräsen die Oberflächengüte deutlich erhöht.

Wenn in und zwischen den zu komprimierenden Verfahrssätzen Befehle programmiert werden, die keine Verfahrensanweisungen sind (z. B. Hilfsfunktionsausgaben), wird der Kompressionsvorgang unterbrochen.

Die maximal tolerierbare Abweichung der berechneten Bahn zu den programmierten Positionen kann für alle Kompressor-Funktionen über Maschinendaten vorgegeben werden (siehe "Parametrierung (Seite 238)"). Anders als bei COMPON und COMPCURV werden die parametrisierten Toleranzen bei COMPCAD und COMPSURF in benachbarten Bahnen nicht in unterschiedlichen Richtungen ausgenutzt.

Kompression von Kreis- und Eilgangsätzen

Außer der Kompression von Linearsätzen bieten alle Kompressor-Funktionen die Möglichkeit, auch Eilgangsätze (G0-Sätze) zu komprimieren. Die Kompression von Kreissätzen ist dagegen nur mit der Kompressor-Funktion COMPCAD möglich.

Die Kompression von Kreis- und/oder Eilgangsätzen wird kanalspezifisch eingestellt über die **Hunderterstelle** im Maschinendatum:

MD20482 \$MC_COMPRESSOR_MODE = <Wert>

Wert	Bedeutung
0xx	Kreissätze und G0-Sätze werden nicht komprimiert. Dies ist kompatibel zu früheren SW-Versionen.
1xx	Kreissätze werden von COMPCAD linearisiert und komprimiert. Vorteil: Die Kompressor-Funktion arbeitet genauer und erzeugt dadurch i. d. R. bessere Oberflächen. Nachteil: Die Kompressor-Funktion wird empfindlicher gegenüber Defekten in den NC-Programmen. Aus Kompatibilitätsgründen kann es deshalb notwendig sein, die Einstellung 0xx beizubehalten.

5.5 Kompressor-Funktionen

Wert	Bedeutung
2xx	G0-Sätze werden komprimiert, evtl. wird dabei eine andere Toleranz wirksam (siehe "Toleranzfaktor für G0-Bewegungen (Seite 256)"). Vorteil: Durch die mögliche Einstellung einer größeren Toleranz und der Kompression von G0-Zustellbewegungen können diese schneller und flüssiger abgefahren werden.
3xx	Kombination der beiden vorhergehenden Möglichkeiten: sowohl Kreissätze als auch G0-Sätze werden komprimiert.

Aktivierung/Deaktivierung

Die Kompressor-Funktionen COMPON, COMPCURV, COMPCAD und COMPSURF werden mit den G-Befehlen der G-Gruppe 30 ein-/ausgeschaltet (siehe "Programmierung (Seite 247)").

5.5.1.2 Inbetriebnahme

Parametrierung

Achsspezifische Maschinendaten

Nummer	Bezeichner \$MA_	Bedeutung
MD33100	COMPRESS_POS_TOL	Maximal erlaubte achsspezifische Bahnabweichung bei Kompression

Kanalspezifische Maschinendaten

Nummer	Bezeichner \$MC_	Bedeutung
MD20170	COMPRESS_BLOCK_PATH_LIMIT	Maximale Verfahrslänge eines komprimierbaren NC-Satzes
MD20171	SURF_BLOCK_PATH_LIMIT	Maximale Verfahrslänge eines komprimierbaren NC-Satzes bei Kompression mit COMPSURF
MD20172	COMPRESS_VELO_TOL	Maximal erlaubte Abweichung des Bahnvorstubs bei Kompression
MD20173	SURF_VELO_TOL	Maximal erlaubte Abweichung des Bahnvorstubs bei Kompression mit COMPSURF
MD20482	COMPRESSOR_MODE	Arbeitsweise des Kompressors
MD20484	COMPRESSOR_PERFORMANCE	Kompressorleistung
MD20485	COMPRESS_SMOOTH_FACTOR	Glättungsfaktor bei Kompression mit COMPCAD für den jeweiligen Dynamikmodus
MD20486	COMPRESS_SPLINE_DEGREE	Spline-Grad bei Kompression mit COMPCAD für den jeweiligen Dynamikmodus
MD20487	COMPRESS_SMOOTH_FACTOR_2	Glättungsfaktor für Rundachsen bei Kompression mit COMPCAD für den jeweiligen Dynamikmodus

Nummer	Bezeichner \$MC_	Bedeutung
MD28071	MM_NUM_SURF_LEVELS	Dimensionierung der Funktion COMPSURF (DRAM)
MD28072	MM_MAXNUM_SURF_GROUPS	Dimensionierung der Funktion COMPSURF bzgl. Achsgruppen (DRAM)

Kanalspezifische Settingdaten

Nummer	Bezeichner \$SC_	Bedeutung
SD42470	CRIT_SPLINE_ANGLE	Ecken-Grenzwinkel bei Kompression mit COMP-CAD
SD42471	MIN_CURV_RADIUS	Minimaler Krümmungsradius bei Kompression mit COMPCAD
SD42472	MIN_SURF_RADIUS	Minimaler Krümmungsradius bei Kompression mit COMPSURF
SD42473	ACTNUM_SURF_GROUPS	Anzahl Achsgruppen bei COMPSURF
SD42475	COMPRESS_CONTUR_TOL	Maximal erlaubte Konturabweichung bei Kompression
SD42476	COMPRESS_ORI_TOL	Maximale Abweichung der Werkzeugorientierung bei Kompression Hinweis: Nur bei aktiver Orientierungstransformation!
SD42477	COMPRESS_ORI_ROT_TOL	Maximale Abweichung der Werkzeugdrehung bei Kompression Hinweis: Nur bei 6-Achs-Maschinen mit drehbarem Werkzeug!

Hinweis

Ecken-Grenzwinkel und Kompressorfunktion COMPCAD

Der über das Settingdatum SD42470 \$SC_CRIT_SPLINE_ANGLE eingestellte Ecken-Grenzwinkel für die Kompressor-Funktion COMPCAD dient nur als ungefähres Maß zur Eckenerkennung. Der Kompressor erkennt aufgrund von Plausibilitätsbetrachtungen auch flachere Satzübergänge als Ecken und größere Winkel als Ausreißer.

Einstellempfehlungen

Kompressor-Funktionen haben eine wichtige Bedeutung im Werkzeug- und Formenbau. Entsprechende Einstellempfehlungen relevanter Maschinen- und Settingdaten finden sich in folgender Tabelle.

Tabelle 5-1 Einstellempfehlungen für den Werkzeug- und Formenbau mit Advanced Surface / Top Surface

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
10131	\$MN_SUPPRESS_SCREEN_REFRESH	Verhalten der Anzeige-Aktualisierung bei Vorlauf-Überlast	0		
10200	\$MN_INT_INCR_PER_MM	Interne Rechenfeinheit Linearachse	100000		10000 bei Exportvariante
10210	\$MN_INT_INCR_PER_DEG	Interne Rechenfeinheit Rundachse	= MD10200		10000 bei Exportvariante
10682	\$MN_CONTOUR_SAMPLING_FACTOR	Kontur Abtastfaktor	1		Voreinstellung
10708	\$MN_SERUPRO_MASK	Satzsuchlauf Modi	Bit 5 = 1		Mind. 20hex
11450	\$MN_SEARCH_RUN_MODE	Satzsuchlauf Parametrierung	Bit 6 = 1		Mind. 40hex
18360	\$MN_MM_EXT_PROG_BUFFER_SIZE	Maximaler Nachladespeicher beim Abarbeiten von Extern	500 ... 2000	2000	
18362	\$MN_MM_EXT_PROG_NUM	Anzahl externer Programme, die gleichzeitig bearbeitet werden können.	2 ... 4	4	COMPCAD: Mindestens Wert 2 verwenden. Größere Werte verwenden, wenn das Hauptprogramm auch von Extern abgearbeitet wird. Hinweis: Bei der Einstellung ist zu beachten, dass MD18362 ein Multiplikator für MD18360 ist!
20150	\$MC_GCODE_RESET_VALUES [3]	Löschstellung der G-Gruppe 4	3		FIFOCTRL
20150	\$MC_GCODE_RESET_VALUES [19]	Löschstellung der G-Gruppe 20	2		SOFT

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
20150	\$MC_GCODE_RESET_VALUES [44]	Löschstellung der G-Gruppe 45	2		UPATH (bei 5-Achsbearbeitung)
20150	\$MC_GCODE_RESET_VALUES [48]	Löschstellung der G-Gruppe 49	1		CP
20150	\$MC_GCODE_RESET_VALUES [50]	Löschstellung der G-Gruppe 50	2		ORIAXES (bei 5-Achsbearbeitung)
20170	\$MC_COMPRESS_BLOCK_PATH_LIMIT	Maximale Verfahr­länge eines komprimierbaren NC-Satzes	20		
20171	\$MC_SURF_BLOCK_PATH_LIMIT	Maximale Verfahr­länge eines komprimierbaren NC-Satzes bei Kompression mit COMPSURF	-	0	Voreinstellung
20172	\$MC_COMPRESS_VELO_TOL	Maximal erlaubte Abweichung des Bahn­vorschubs bei Kompression	1000		Voreinstellung
20173	\$MC_SURF_VELO_TOL	Maximal erlaubte Abweichung des Bahn­vorschubs bei Kompression mit COMPSURF	-	1000	Voreinstellung
20443	\$MC_LOOKAH_FFORM [0-1]	Aktivieren des er­weiterten LookA-head im Dynamik­modus DYN-NORM / DYNPOS	0		Voreinstellung
20443	\$MC_LOOKAH_FFORM [2-4]	Aktivieren des er­weiterten LookA-head im Dynamik­modus DYN-ROUGH / DYNSE-MIFIN / DYNFI-NISH	1		
20455	\$MC_LOOKAH_FUNCTION_MASK	Sonderfunktio­nen des LookA-head	3		

5.5 Kompressor-Funktionen

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
20470	\$MC_CPREC_WITH_FFW	Programmierbare Konturgenauigkeit	3		Die programmierbare Konturgenauigkeit ist auch bei aktiver Vorsteuerung wirksam. Die programmierte Konturtoleranz CTOL bestimmt den erlaubten Konturfehler. Hinweis: Ggf. kann der Wert auf "2" gesetzt werden, damit über SD42450 \$SC_CONTPREC "große Toleranzen (0,1 mm)" realisiert werden können.
20476	\$MC_ORISON_STEP_LENGTH	Bahnlänge für Satzunterteilung bei ORISON	0,5		Voreinstellung
20478	\$MC_ORISON_MODE	Modus der Orientierungsglättung	100		Die programmierten Sätze werden unterteilt, sodass die Orientierungsglättung insgesamt einen homogenen Verlauf der Orientierung erzeugen kann. Es werden nur solche Sätze unterteilt, in denen ein Kompressor aktiv ist.
20480	\$MC_SMOOTHING_MODE	Verhalten des Überschleifens bei G64x	x0x		Voreinstellung
20482	\$MC_COMPRESSOR_MODE	Arbeitsweise des Kompressors	300		Sollte es mit der dieser Einstellung Probleme mit den Geschwindigkeitssollwerten geben, dann sollte stattdessen der Wert "100" verwendet werden.

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
20485	\$MC_COMPRESS_SMOOTH_FACTOR [0-1]	Glättungsfaktor bei Kompression mit COMPCAD im Dynamikmodus DYN-NORM / DYNPOS	0		Voreinstellung
20485	\$MC_COMPRESS_SMOOTH_FACTOR [2-4]	Glättungsfaktor bei Kompression mit COMPCAD im Dynamikmodus DYN-ROUGH / DYNSE-MIFIN / DYNFINISH	0,0001		Soll nicht wirken.
20486	\$MC_COMPRESS_SPLINE_DEGREE [0-1]	Spline-Grad bei Kompression mit COMPCAD im Dynamikmodus DYN-NORM / DYNPOS	3		Voreinstellung
20486	\$MC_COMPRESS_SPLINE_DEGREE [2-4]	Spline-Grad bei Kompression mit COMPCAD im Dynamikmodus DYN-ROUGH / DYNSE-MIFIN / DYNFINISH	5		
20487	\$MC_COMPRESS_SMOOTH_FACTOR_2 [0-1]	Glättungsfaktor für Rundachsen bei Kompression mit COMPCAD im Dynamikmodus DYN-NORM / DYNPOS	0		
20487	\$MC_COMPRESS_SMOOTH_FACTOR_2 [2-4]	Glättungsfaktor für Rundachsen bei Kompression mit COMPCAD im Dynamikmodus DYN-ROUGH / DYNSE-MIFIN / DYNFINISH	0,5		
20490	\$MC_IGNORE_OVL_FACTOR_FOR_ADIS	Beeinflussung von G642, G645	1		
20560	\$MC_G0_TOLERANCE_FACTOR	G0-Toleranzfaktor bei Kompression, G645, OST, ORISON	3		

5.5 Kompressor-Funktionen

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
20600	\$MC_MAX_PATH_JERK [0-4]	Bahnruck im jeweiligen Dynamikmodus	10000		Soll nicht wirken.
20602	\$MC_CURV_EFFECT_ON_PATH_ACCEL [0-1]	Verhältnis von translatorischer Beschleunigung zu Zentripetalbeschleunigung im Dynamikmodus DYN-NORM / DYNPOS	0		Soll nicht wirken.
20602	\$MC_CURV_EFFECT_ON_PATH_ACCEL [2]	Verhältnis von translatorischer Beschleunigung zu Zentripetalbeschleunigung im Dynamikmodus DYNROUGH	0 (ggf. durch Kreisformtest ermitteln)	0,65	Kreisbeschleunigung vor allem bei "großen" Maschinen begrenzen!
20602	\$MC_CURV_EFFECT_ON_PATH_ACCEL [3]	Verhältnis von translatorischer Beschleunigung zu Zentripetalbeschleunigung im Dynamikmodus DYNSEMIFIN		0,6	
20602	\$MC_CURV_EFFECT_ON_PATH_ACCEL [4]	Verhältnis von translatorischer Beschleunigung zu Zentripetalbeschleunigung im Dynamikmodus DYNFINISH		0,5	
20606	\$MC_PREPDYN_SMOOTHING_ON [0-1]	Aktivieren der Glättung der Krümmung im Dynamikmodus DYN-NORM / DYNPOS	0		Voreinstellung
20606	\$MC_PREPDYN_SMOOTHING_ON [2-4]	Aktivieren der Glättung der Krümmung im Dynamikmodus DYN-ROUGH / DYNSEMIFIN / DYNFINISH	1		
22430	\$MC_FGROUP_PATH_MODE	Verhalten der Bahngeschwindigkeit in singulären Situationen	3		Bei 5-Achs-Bearbeitung unbedingt erforderlich!

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
22440	\$MC_FGROUP_PATH_RATIO	Wegverhältnis zum Einsatz der FGROUP-Singularitätenstrategie	1		Voreinstellung
28060	\$MC_MM_IPO_BUFFER_SIZE	Anzahl der NC-Sätze im IPO-Puffer	150	300	
28070	\$MC_MM_NUM_BLOCKS_IN_PREP	Anzahl der Sätze für die Satzaufbereitung	80	500	
28071	\$MC_MM_NUM_SURF_LEVELS	Dimensionierung der Funktion COMPSURF	-	6	
28072	\$MC_MM_MAXNUM_SURF_GROUPS	Dimensionierung der Funktion COMPSURF bzgl. Achsgruppen	-	2	
28520	\$MC_MM_MAX_AXISPOLY_PER_BLOCK	Maximale Anzahl der Achspolynome pro Satz	5	15	
28530	\$MC_MM_PATH_VELO_SEGMENTS	Anzahl Speicherelemente zur Begrenzung der Bahngeschwindigkeit	5	8	
28533	\$MC_MM_LOOKAH_FFORM_UNITS	Speicher für den erweiterten LookAhead	18		
28540	\$MC_MM_ARCLENGTH_SEGMENTS	Anzahl Speicherelemente zur Darstellung der Bogenlängenfunktion	10	20	
28590	\$MC_MM_ORISON_BLOCKS	Maximale Anzahl Sätze, über die die Orientierung geglättet wird	100		Bei 5-Achsbearbeitung! Abhängig von SD42678 \$SC_ORISON_TOL muss der Wert ggf. vervielfacht werden.
28610	\$MC_MM_PREPDYN_BLOCKS	Anzahl Sätze zur Geschwindigkeitsvorbereitung	10		
29000	\$OC_LOOKAH_NUM_CHECKED_BLOCKS	Anzahl LookAhead-Sätze	150	300	Muss mit dem Wert in MD28060 identisch sein!

5.5 Kompressor-Funktionen

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
42450	\$SSC_CONTPREC	Konturgenauigkeit für CPRE-CON bei MD20470 \$MC_CPREC_WITH_FFW = 1 oder 2	0,1		Bei MD20470 = 3 ist der Wert in SD42450 nicht wirksam!
42470	\$SSC_CRIT_SPLINE_ANGLE	Ecken-Grenzwinkel bei Kompression mit COMPCAD	36		Voreinstellung Der Wert sollte größer 30° sein.
42471	\$SSC_MIN_CURV_RADIUS	Minimaler Krümmungsradius bei Kompression mit COMPCAD	1		Voreinstellung Empfohlener Einstellungsbereich: 0,3 ... 3
42472	\$SSC_MIN_SURF_RADIUS [0]	Glätten in Achsgruppe <n> (Geometrieachsen) Kontur	-		Voreinstellung = 1 Wird durch CYCLE832 beschrieben. 1 = Glättung ein 0 = Glättung aus
42472	\$SSC_MIN_SURF_RADIUS [1]	Glätten in Achsgruppe <n> (Orientierungsachsen) Orientierung	-		Voreinstellung = 1 Wird durch CYCLE832 beschrieben. 1 = Glättung ein 0 = Glättung aus
42472	\$SSC_MIN_SURF_RADIUS [2]	Glätten in Achsgruppe <n> (Hilfsachsen)	-	1	
42473	\$SSC_ACTNUM_SURF_GROUPS	Anzahl Achsgruppen bei COMPSURF	-	2	
42500	\$SSC_IS_MAX_PATH_ACCEL	Begrenzung der Bahnbeschleunigung über SD	10000		
42502	\$SSC_IS_SD_MAX_PATH_ACCEL	Aktivierung der Bahnbeschleunigung über SD	0 (FALSE)		
42510	\$SSC_SD_MAX_PATH_JERK	Begrenzung des Bahnricks über SD	10000		

MD	Name	Beschreibung	Empfohlener Wert		Kommentar
			Advanced Surface	Top Surface	
42512	\$SC_IS_SD_MAX_PATH_JERK	Aktivierung des Bahnruks über SD	0 (FALSE)		
42678	\$SC_ORISON_TOL	Toleranz zur Glättung der Orientierung mit ORISON	0,1		Bei 5-Achsbearbeitung! Wert ist abhängig von Bearbeitung/Zyklus, wird durch OTOL überstimmt! Empfehlung: 0 für Umfangsfräsen 0,1 für Stirnfräsen

5.5.1.3 Programmierung

NC-Satz-Kompression ein-/ausschalten (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF)

Die Funktionen zur Kompression von Linearsätzen (und abhängig von der Parametrierung auch Kreis- und/oder Eilgangsätzen) werden mit den G-Befehlen der G-Gruppe 30 ein-/ausgeschaltet. Die Befehle sind modal wirksam.

Syntax

```
COMPON / COMPCURV / COMPCAD / COMPSURF
...
COMPOF
```

Bedeutung

COMPON:	Einschalten der Kompressor-Funktion COMPON
COMPCURV:	Einschalten der Kompressor-Funktion COMPCURV
COMPCAD:	Einschalten der Kompressor-Funktion COMPCAD
COMPSURF:	Einschalten der Kompressor-Funktion COMPSURF
COMPOF:	Ausschalten der aktuell aktiven Kompressor-Funktion

Hinweis

Zur zusätzlichen Verbesserung der Oberflächengüte kann die Überschleiffunktion G642 und die Ruckbegrenzung SOFT verwendet werden. Diese Befehle sind am Programmanfang zu schreiben.

Beispiel: COMPCAD

Programmcode	Kommentar
N10 G00 X30 Y6 Z40	
N20 G1 F10000 G642	; Einschalten: Überschleiffunktion G642
N30 SOFT	; Einschalten: Ruckbegrenzung SOFT
N40 COMPCAD	; Einschalten: Kompressor-Funktion COMPCAD
N50 STOPFIFO	
N24050 Z32.499	; 1. Verfahrssatz
N24051 X41.365 Z32.500	; 2. Verfahrssatz
...	
N99999 X... Z...	; letzter Verfahrssatz
COMPOF	; Kompressor-Funktion aus.
...	

5.5.1.4 Randbedingungen**Orientierungstransformation (TRAORI)**

Die Kompressor-Funktionen können bei aktiver Orientierungstransformation auch Bewegungssätze zur Werkzeugorientierung und Werkzeugdrehung komprimieren.

Literatur:

Funktionshandbuch Sonderfunktionen; Kapitel "F2: Mehrachstransformationen" > "Orientierung" > "Komprimierung der Orientierung"

Satzsuchlauf mit Berechnung

Wenn bei Satzsuchlauf **Typ 2** oder **Typ 4** (Satzsuchlauf **mit Berechnung** an ...) der Zielsatz in einem Programmabschnitt liegt, in dem eine Kompressor-Funktion aktiv ist, werden beim Wiederauffahren an die Kontur Positionen auf der durch den Kompressor berechneten Bahn angefahren. Diese Positionen müssen nicht exakt mit den Positionen auf der im Teileprogramm programmierten Bahn übereinstimmen.

Teileprogrammsätze, die durch die Komprimierung ersetzt werden, können im Satzsuchlauf nicht als Zielsatz gefunden werden. Es wird Alarm 15370 "Suchziel nicht gefunden" ausgegeben.

5.5.2 Kompression kurzer Spline-Sätze**Funktion**

Bei der Erzeugung von Spline-Sätzen zur Beschreibung von komplexen Konturen durch CAD/CAM-Systeme entstehen zwischen Spline-Sätzen mit großen Bahnlängen auch immer wieder Spline-Sätze mit sehr kurzen Bahnlängen. Diese zwingen die Steuerung zu einer signifikanten Reduzierung der Bahngeschwindigkeit. Die Funktionen zur Kompression kurzer Spline-Sätze erzeugen neue Spline-Sätze so, dass möglichst große Bahnlängen entstehen.

Verfügbarkeit

System	Verfügbarkeit
SINUMERIK 840D sl	Standard (Grundumfang)
SINUMERIK 828D	Option

Inbetriebnahme

Aktivierung

Die Kompression kurzer Spline-Sätze kann für folgende Spline-Arten aktiviert werden:

- BSPLINE
- BSPLINE / ORICURVE
- CSPLINE

Die Aktivierung erfolgt über das kanalspezifische Maschinendatum:

MD20488 \$MC_SPLINE_MODE, Bit <n> = <Wert>

Bit	<Wert>	Spline-Art	Kompression kurzer Spline-Sätze
0	0	BSPLINE	nicht aktiv
	1	BSPLINE	aktiv
1	0	BSPLINE / ORICURVE	nicht aktiv
	1	BSPLINE / ORICURVE	aktiv
2	0	CSPLINE	nicht aktiv
	1	CSPLINE	aktiv

Randbedingungen

- Wenn in und zwischen den zu komprimierenden Verfahrssätzen Befehle programmiert sind, die keine Verfahrenweisungen sind, z. B. Hilfsfunktionsausgaben, können die Spline-Sätze nicht zusammengefasst werden.
- Die maximale Anzahl hintereinander liegender Spline-Sätzen, die in einem Programmabschnitt zusammengefasst werden können, hängt von der parametrisierten Größe des Satzspeichers in der Satzaufbereitung ab:
MD28070 \$MC_MM_NUM_BLOCKS_IN_PREP (Anzahl der Sätze für die Satzaufbereitung)

Beispiel

Um beim Abarbeiten der Verfahrssätze eine höhere Bahngeschwindigkeit erreichen zu können, wird die Kompression für kurze Spline-Sätze für die BSPLINE-Interpolation aktiviert:

MD20488 \$MC_SPLINE_MODE, Bit 0 = 1

Programmcode	Kommentar
N10 G1 G64 X0 Y0 Z0 F1000	; Grundstellung
N20 G91 F10000 BSPLINE	; Einschalten: B-Spline

Programmcode	Kommentar
N30 X0.001 Y0.001 Z0.001	; Ab hier: Zusammenfassung kurzer Spline-Sätze
N40 X0.001 Y0.001 Z0.001	
...	

5.6 Kontur-/Orientierungstoleranz

5.6.1 Inbetriebnahme

5.6.1.1 Parametrierung

Maschinendaten

Konturtoleranz / Orientierungstoleranz

MD33100 \$MA_COMPRESS_POS_TOL[<Achse>] = <Wert> (Maximale Abweichung bei Kompression)

Über das achsspezifische Maschinendatum wird die maximal erlaubte Konturabweichung (Konturtoleranz) bzw. Winkelabweichung der Werkzeugorientierung (Orientierungstoleranz) der jeweiligen Achse eingestellt. Das Maschinendatum wirkt bei folgenden Funktionen:

- Überschleifen: G642, G643, G644, G645
- Kompressor: COMPON, COMPCURV, COMPCAD, COMPSURF
- Je größer der Wert ist, umso mehr kurze Sätze können in einen langen Satz komprimiert werden.

Das Maschinendatum wirkt nicht bei Überschleiffunktion G641. Bei G641 wirkt die mit ADIS bzw. ADISPOS programmierbare Wegdistanz zum Satzübergang.

Überschleif-Modus

MD20480 \$MC_SMOOTHING_MODE (Verhalten des Überschleifens mit G64x)

Kompressor-Modus

MD20482 \$MC_COMPRESSOR_MODE (Modus des Kompressors)

Überschleifen G645

MD33120 \$MA_PATH_TRANS_POS_TOL (Maximale Konturabweichung bei Überschleifen G645)

Wirksam beim Überschleifen von tangentialen-, aber nicht krümmungsstetigen Satzübergängen (z. B. Kreis - Gerade)

Settingdaten

Kanalspezifische Konturtoleranz

SD42465 \$SC_SMOOTH_CONTUR_TOL (Maximale Konturabweichung)

Kanalspezifische Orientierungstoleranz

SD42466 \$SC_SMOOTH_ORI_TOL (Maximale Winkelabweichung der Werkzeugorientierung)

Kanalspezifische Orientierungstoleranz bei Überschleifen mit OST

SD42676 \$SC_ORI_SMOOTH_TOL (Toleranz zur Glättung der Orientierung beim Überschleifen)

Kanalspezifische Orientierungstoleranz bei Glättung der Orientierung mit ORISON

SD42678 \$SC_ORISON_TOL (Toleranz zur Glättung der Orientierung)

5.6.2 Programmierung

5.6.2.1 Kontur-/Orientierungstoleranz programmieren (CTOL, OTOL, ATOL)

Mit den Adressen CTOL, OTOL und ATOL können die über Maschinen- und Settingdaten parametrisierten Bearbeitungstoleranzen für Kompressor-Funktionen, Überschleifen und Orientierungsglättung im Teileprogramm angepasst werden.

Die programmierten Toleranzwerte gelten, bis sie neu programmiert oder durch Zuweisung eines negativen Werts gelöscht werden. Sie werden ferner gelöscht bei Programmende bzw. Reset. Nach dem Löschen sind wieder die parametrisierten Toleranzwerte wirksam.

Syntax

```
CTOL=<Value>  
OTOL=<Value>  
ATOL[<Axis>]=<Value>
```

Bedeutung

CTOL:	Adresse zum Programmieren der Konturtoleranz		
	Anwendungsbereich:	<ul style="list-style-type: none"> • alle Kompressor-Funktionen • alle Überschleifarten außer G641 und G644 	
	Vorlaufstopp:	nein	
	Wirksamkeit:	modal	
	<Value>:	Der Wert für die Konturtoleranz ist eine Längenangabe.	
		Typ:	REAL
		Einheit:	Inch/mm (abhängig von der aktuellen Einstellung der Maßangabe)
		Wertebereich:	<div>≥ 0: Toleranzwert</div> <div>< 0: Löschen des programmierten Toleranzwerts ⇒ Es gilt wieder der im Maschinen- bzw. Settingdatum parametrisierte Toleranzwert.</div>
OTOL:	Adresse zum Programmieren der Orientierungstoleranz		
	Anwendungsbereich:	<ul style="list-style-type: none"> • alle Kompressor-Funktionen • Orientierungsglättung ORISON • alle Überschleifarten außer G641, G644 und OSD 	
	Vorlaufstopp:	nein	
	Wirksamkeit:	modal	
	<Value>:	Der Wert für die Orientierungstoleranz ist eine Winkelangabe.	
		Typ:	REAL
		Einheit:	Grad
		Wertebereich:	<div>≥ 0: Toleranzwert</div> <div>< 0: Löschen des programmierten Toleranzwerts ⇒ Es gilt wieder der im Maschinen- bzw. Settingdatum parametrisierte Toleranzwert.</div>

ATOL:	Adresse zum Programmieren einer achsspezifischen Toleranz		
	Anwendungsbereich:	<ul style="list-style-type: none"> • alle Kompressor-Funktionen • Orientierungsglättung ORISON • alle Überschleifarten außer G641, G644 und OSD 	
	Vorlaufstopp:	nein	
	Wirksamkeit:	modal	
	<Axis>:	Name der Kanalachse, für welche die programmierte Toleranz wirken soll	
	<Value>:	Der Wert für die Achstoleranz ist je nach Achstyp (Linear- oder Rundachse) eine Längen- oder Winkelangabe.	
		Typ:	REAL
		Einheit:	für Linearachsen: Inch/mm (abhängig von der aktuellen Einstellung der Maßangabe)
			für Rundachsen: Grad
	Wertebereich:	≥ 0:	Toleranzwert
		< 0:	Löschen des programmierten Toleranzwerts ⇒ Es gilt wieder der im Maschinen- bzw. Settingdatum parametrisierte Toleranzwert.

Hinweis

Die mit CTOL und OTOL programmierten kanalspezifischen Toleranzwerte sind höherprior als die mit ATOL programmierten achsspezifischen Toleranzwerte.

Hinweis**Skalierende Frames**

Skalierende Frames wirken auf die programmierten Toleranzen in gleicher Weise wie auf die Achspositionen, d. h. die relative Toleranz bleibt gleich.

Beispiel

Programmcode	Kommentar
COMPCAD G645 G1 F10000	; Kompressor-Funktion COMPCAD aktivieren.
X... Y... Z...	; Hier wirken die Maschinen- und Settingdaten.
X... Y... Z...	
X... Y... Z...	
CTOL=0.02	; Ab hier wirkt eine Konturtoleranz von 0,02 mm.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
ASCALE X0.25 Y0.25 Z0.25	; Ab hier wirkt eine Konturtoleranz von 0,005 mm.
X... Y... Z...	

Programmcode	Kommentar
X... Y... Z...	
X... Y... Z...	
CTOL=-1	; Ab hier wirken wieder Maschinen- und Settingdaten.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

5.6.2.2 Kontur-/Orientierungstoleranz programmieren (CTOL, OTOL, ATOL): Weitere Informationen

Systemvariablen

Lesen mit Vorlaufstopp

Über folgende Systemvariablen sind im Teileprogramm und Synchronaktion die aktuell wirksamen Toleranzen lesbar:

- **\$AC_CTOL**
Kanalspezifische Konturtoleranz, die bei der Aufbereitung des aktuellen Hauptlaufsatzes wirksam war.
Wenn keine Konturtoleranz wirksam ist, liefert \$AC_CTOL die Wurzel aus der Summe der Quadrate der Toleranzen der Geometrieachsen.
- **\$AC_OTOL**
Kanalspezifische Orientierungstoleranz, die bei der Aufbereitung des aktuellen Hauptlaufsatzes wirksam war.
Wenn keine Orientierungstoleranz wirksam ist, liefert \$AC_OTOL während einer aktiven Orientierungstransformation die Wurzel aus der Summe der Quadrate der Toleranzen der Orientierungsachsen, ansonsten den Wert "-1".
- **\$AA_ATOL[<Achse>]**
Achsspezifische Konturtoleranz, die bei der Aufbereitung des aktuellen Hauptlaufsatzes wirksam war.
Wenn eine Konturtoleranz aktiv ist, liefert \$AA_ATOL[<Geometrieachse>] die Konturtoleranz geteilt durch die Wurzel aus der Anzahl der Geometrieachsen.
Wenn eine Orientierungstoleranz und eine Orientierungstransformation aktiv ist, liefert \$AA_ATOL[<Orientierungsachse>] die Orientierungstoleranz geteilt durch die Wurzel aus der Anzahl der Orientierungsachsen.

Hinweis

Wenn keine Toleranzwerte programmiert wurden, dann sind die \$A-Variablen nicht differenziert genug, um die Toleranzen der einzelnen Funktionen zu unterscheiden.

Solche Fälle können auftreten, wenn die Maschinen- und Settingdaten unterschiedliche Toleranzen für Kompressor-Funktionen, Überschleifen und Orientierungsglättung einstellen. Die Systemvariablen liefern dann den größten Wert, der bei den gerade aktiven Funktionen auftritt. Wenn z. B. eine Kompressor-Funktion mit Orientierungstoleranz 0,1° und eine Orientierungsglättung ORISON mit 1° aktiv ist, liefert \$AC_OTOL den Wert "1". Wird die Orientierungsglättung ausgeschaltet, liefert \$AC_OTOL den Wert "0,1".

Lesen ohne Vorlaufstopp

Über folgende Systemvariablen sind im Teileprogramm die aktuell wirksamen Toleranzen lesbar:

- \$P_CTOL
Aktuell wirksame kanalspezifische Konturtoleranz.
- \$P_OTOL
Aktuell wirksame kanalspezifische Orientierungstoleranz.
- \$PA_ATOL
Aktuell wirksame achsspezifische Konturtoleranz.

Randbedingungen

Die mit CTOL, OTOL und ATOL programmierten Toleranzen wirken auch auf Funktionen, die indirekt von diesen Toleranzen abhängen:

- Die Begrenzung des Sehnenfehlers bei der Sollwertberechnung
- Die Grundfunktionen des Freiformflächenmodus

Folgende Überschleif-Funktionen werden durch die Programmierung von CTOL, OTOL und ATOL **nicht** beeinflusst:

- Überschleifen der Orientierung mit OSD
OSD verwendet keine Toleranz, sondern eine Distanz zum Satzübergang.
- Überschleifen mit G644
G644 dient nicht der Bearbeitung, sondern der Optimierung von Werkzeugwechseln und anderen Bewegungen ohne Bearbeitung.
- Überschleifen mit G645
G645 verhält sich fast immer wie G642 und verwendet somit die programmierten Toleranzen. Nur an tangentialen Satzübergängen mit einem Krümmungssprung, z. B. einem tangentialen Kreis-Gerade-Übergang, wird der Toleranzwert aus dem Maschinendatum MD33120 \$MA_PATH_TRANS_POS_TOL verwendet. Der Überschleifweg kann sich an diesen Stellen nämlich auch auf der Außenseite der programmierten Kontur befinden, wo viele Anwendungen weniger tolerant sind. Außerdem genügt in der Regel eine kleine, fest eingestellte Toleranz zum Ausgleich der Krümmungsänderungen, über die sich der NC-Programmierer keine Gedanken machen muss.

5.7 Toleranzfaktor für G0-Bewegungen

5.7.1 Funktion

Mit dem G0-Toleranzfaktor können die Toleranzen für G0-Bewegungen (Eilgang, Zustellbewegungen) unterschiedlich zu den Werkstückbearbeitungstoleranzen eingestellt werden.

Vorteil

Größere Toleranzen für G0-Bewegungen (G0-Toleranzfaktor > 1) ermöglichen ein schnelleres Verfahren von G0-Sätzen.

Voraussetzungen

Der G0-Toleranzfaktor wird nur wirksam, wenn folgende Bedingungen erfüllt sind:

- Eine der folgenden Funktionen ist aktiv:
 - Kompressorfunktion COMPON, COMPCURV, COMPCAD oder COMPSURF
 - Überschleiffunktion G642 oder G645
 - Orientierungsüberschleifen OST
 - Orientierungsglättung ORISON
 - Glättung bei bahnrelativer Orientierung ORIPATH
- Im Teileprogramm folgen mehrere (≥ 2) G0-Sätze aufeinander.
Bei einem einzelnen G0-Satz wird der G0-Toleranzfaktor nicht wirksam, da beim Übergang von einer Nicht-G0-Bewegung zu einer G0-Bewegung (und umgekehrt) grundsätzlich die "**kleinere Toleranz**" (Werkstückbearbeitungstoleranz) wirkt!

5.7.2 Inbetriebnahme

5.7.2.1 Parametrierung

Maschinendaten

G0-Toleranzfaktor

Der G0-Toleranzfaktor wird kanalspezifisch eingestellt über das Maschinendatum:

MD20560 \$MC_G0_TOLERANCE_FACTOR (Toleranz-Faktor für G0)

Der G0-Toleranzfaktor kann sowohl größer als auch kleiner 1.0 sein. Ist der Faktor gleich 1.0 (Standardwert), sind für G0-Bewegungen dieselben Toleranzen wirksam wie für Nicht-G0-Bewegungen. Im Normalfall wird der G0-Toleranzfaktor > 1.0 eingestellt.

5.7.3 Programmierung

5.7.3.1 Toleranzfaktor für G0-Bewegungen programmieren (STOLF)

Der mittels Maschinendatum (MD20560 \$MC_G0_TOLERANCE_FACTOR) eingestellte Toleranzfaktor für G0-Bewegungen kann im Teileprogramm durch Programmierung von STOLF angepasst werden. Der Wert im Maschinendatum wird dabei nicht verändert. Nach Reset bzw. Teileprogrammende wird wieder der im Maschinendatum eingestellte G0-Toleranzfaktor wirksam.

Syntax

STOLF=<Value>

Bedeutung

STOLF:	Adresse zum Programmieren eines Toleranzfaktors für G0-Bewegungen		
	Anwendungsbereich:	<ul style="list-style-type: none"> • Kompressor-Funktionen COMPON, COMPCURV, COMPCAD und COMPSURF • Überschleiffunktionen G642 und G645 • Orientierungsüberschleifen OST • Orientierungsglättung ORISON • Glättung bei bahnrelativer Orientierung ORIPATH 	
	Vorlaufstopp:	nein	
	Wirksamkeit:	modal	
	<Value>:	G0-Toleranzfaktor Der G0-Toleranzfaktor kann sowohl größer als auch kleiner 1.0 sein. Ist der Faktor gleich 1.0 (Standardwert), sind für G0-Bewegungen dieselben Toleranzen wirksam wie für Nicht-G0-Bewegungen. Im Normalfall wird der G0-Toleranzfaktor > 1.0 eingestellt.	
	Typ:	REAL	
	Einheit:	Inch/mm (abhängig von der aktuellen Einstellung der Maßangabe)	
	Wertebereich:	≥ 0: Toleranzwert < 0: Löschen des programmierten Toleranzwerts ⇒ Es gilt wieder der im Maschinendatum parametrisierte Toleranzwert.	

Beispiel

Programmcode	Kommentar
COMPCAD G645 G1 F10000	; Kompressor-Funktion COMPCAD
X... Y... Z...	; Hier wirken die Maschinen- und Settingdaten.
X... Y... Z...	
X... Y... Z...	

Programmcode	Kommentar
G0 X... Y... Z...	
G0 X... Y... Z...	; Hier wirkt das Maschinendatum \$MC_G0_TOLERANCE_FACTOR (z.B. =3), also eine Überschleiftoleranz von: \$MC_G0_TOLERANCE_FACTOR * \$MA_COMPRESS_POS_TOL
CTOL=0.02	
STOLF=4	
G1 X... Y... Z...	; Ab hier wirkt eine Konturtoleranz von 0,02 mm.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; Ab hier wirkt ein G0-Toleranzfaktor von 4, also eine Konturtoleranz von 0,08 mm.

Systemvariablen

Der im Teileprogramm bzw. im aktuellen IPO-Satz wirksame G0-Toleranzfaktor ist über Systemvariablen lesbar.

- In Synchronaktionen oder mit Vorlauf-Stopp im Teileprogramm über die Systemvariable:

\$AC_STOLF Aktiver G0-Toleranzfaktor
G0-Toleranzfaktor, der bei der Aufbereitung des aktuellen Hauptlaufsatzes wirksam war.

- Ohne Vorlauf-Stopp im Teileprogramm über die Systemvariable:

\$P_STOLF Programmierter G0-Toleranzfaktor

Ist im aktiven Teileprogramm kein Wert mit STOLF programmiert, dann liefern diese beiden Systemvariablen den durch MD20560 \$MC_G0_TOLERANCE_FACTOR eingestellten Wert.

Ist in einem Satz kein Eilgang (G0) aktiv, dann liefern diese Systemvariablen immer den Wert 1.

5.8 RESET-Verhalten

MD20150

Durch Reset (Kanal- oder BAG-Reset) wird für alle G-Gruppen die kanalspezifisch parametrisierte Grundstellung wirksam:

MD20150 \$MC_GCODE_RESET_VALUES (Löschstellung der G-Gruppen)

Bezüglich "Bahnsteuerbetrieb, Genauhalt, LookAhead" sind folgende G-Gruppen relevant:

- Gruppe 10: Genauhalt - Bahnsteuerbetrieb
- Gruppe 12: Satzwechselkriterium bei Genauhalt
- Gruppe 21: Beschleunigungsprofil

- Gruppe 30: NC-Satz-Kompression
- Gruppe 59: Dynamikmodus für Bahninterpolation

Ausführliche Informationen zur Einstellung der Grundstellungen siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)".

5.9 Randbedingungen

5.9.1 Satzwechsel und Positionierachsen

Werden in einem Teileprogramm Bahnachsen im Bahnsteuerbetrieb verfahren, können parallel dazu verfahrenende Positionierachsen sowohl das Verhalten der Bahnachsen als auch den Satzwechsel beeinflussen.

Eine ausführliche Beschreibung der Positionierachsen findet sich in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2)

5.9.2 Satzwechselverzögerung

Auch wenn für alle im Teileprogrammsatz verfahrenenden Bahn- und Zusatzachsen die jeweils spezifischen Satzwechselkriterien erfüllt sind, kann es aufgrund anderweitiger nicht erfüllter Bedingungen und / oder aktiver Funktionen zu zeitlichen Verzögerungen des Satzwechsels kommen.

Beispiele:

- Fehlende Hilfsfunktionsquittierung durch die PLC
- Nicht vorhandene Folgesätze
- Aktive Funktion "Zwischenspeicher leeren"

Auswirkungen

Wenn während des Bahnsteuerbetriebs ein Satzwechsel nicht durchgeführt werden kann, dann werden alle in diesem Teileprogrammsatz programmierten Achsen (außer satzübergreifend verfahrenende Zusatzachsen) angehalten. Dabei treten keine Konturfehler auf.

Durch das Anhalten der Bahnachsen **während der Bearbeitung** kann es zum Entstehen von Freischneidmarken auf der Werkstückoberfläche kommen.

5.10 Datenlisten

5.10.1 Maschinendaten

5.10.1.1 Allgemeine Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10110	PLC_CYCLE_TIME_AVERAGE	Mittlere PLC-Quittierungszeit
10680	MIN_CONTOUR_SAMPLING_TIME	Minimale Kontur-Abtastzeit
10682	CONTOUR_SAMPLING_FACTOR	Konturabtastfaktor
10712	NC_USER_CODE_CONF_NAME_TAB	Liste umprojektierter NC-Befehle
12030	OVR_FACTOR_FEEDRATE	Bewertung des Bahnvorschub-Override-Schalters
12100	OVR_FACTOR_LIMIT_BIN	Begrenzung bei binärkodiertem Override-Schalter
18360	MM_EXT_PROG_BUFFER_SIZE	FIFO-Buffer Größe für Abarbeiten von extern (DRAM)

5.10.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20150	GCODE_RESET_VALUES	Löschstellung der Gruppen
20170	COMPRESS_BLOCK_PATH_LIMIT	Maximale Verfahrnlänge eines komprimierbaren NC-Satzes
20171	SURF_BLOCK_PATH_LIMIT	Maximale Verfahrnlänge eines komprimierbaren NC-Satzes bei Kompression mit COMPSURF
20172	COMPRESS_VELO_TOL	Maximal erlaubte Abweichung des Bahnvorschubs bei Kompression
20173	SURF_VELO_TOL	Maximal erlaubte Abweichung des Bahnvorschubs bei Kompression mit COMPSURF
20400	LOOKAH_USE_VELO_NEXT_BLOCK	LookAhead Folgesatzgeschwindigkeit
20430	LOOKAH_NUM_OVR_POINTS	Anzahl der Override-Schalter-Eckwerte bei LookAhead
20440	LOOKAH_OVR_POINTS	Override-Schalter-Eckwerte bei LookAhead
20443	LOOKAH_FFORM	Aktivierung des erweiterten LookAhead
20450	LOOKAH_RELIEVE_BLOCK_CYCLE	Entlastungsfaktor für die Blockzykluszeit
20455	LOOKAH_FUNCTION_MASK	Sonderfunktionen des LookAhead
20460	LOOKAH_SMOOTH_FACTOR	Glättungsfaktor bei LookAhead
20462	LOOKAH_SMOOTH_WITH_FEED	Glättung mit programmierten Vorschub
20465	ADAPT_PATH_DYNAMIC	Adaption der Bahndynamik
20480	SMOOTHING_MODE	Verhalten des Überschleifens mit G64x
20482	COMPRESSOR_MODE	Arbeitsweise des Kompressors
20484	COMPRESSOR_PERFORMANCE	Kompressorleistung
20485	COMPRESS_SMOOTH_FACTOR	Glättungsfaktor bei Kompression mit COMPCAD für den jeweiligen Dynamikmodus

Nummer	Bezeichner: \$MC_	Beschreibung
20486	COMPRESS_SPLINE_DEGREE	Spline-Grad bei Kompression mit COMPCAD für den jeweiligen Dynamikmodus
20487	COMPRESS_SMOOTH_FACTOR_2	Glättungsfaktor für Rundachsen bei Kompression mit COMPCAD für den jeweiligen Dynamikmodus
20488	SPLINE_MODE	Einstellung für Spline-Interpolation
20490	IGNORE_OVL_FACTOR_FOR_ADIS	G641/G642 unabhängig vom Überlastfaktor
20550	EXACT_POS_MODE	Genauhaltbedingungen bei G0/G1
20552	EXACT_POS_MODE_G0_TO_G1	Genauhaltkriterium bei Eilgangübergängen im Bahnsteuerbetrieb
20560	G0_TOLERANCE_FACTOR	G0-Toleranzfaktor
20600	MAX_PATH_JERK	Bahnbezogener Maximalruck
20602	CURV_EFFECT_ON_PATH_ACCEL	Einfluss der Bahnkrümmung auf Bahndynamik
20603	CURV_EFFECT_ON_PATH_JERK	Einfluss der Bahnkrümmung auf Bahnruck
20606	PREPDYN_SMOOTHING_ON	Aktivierung der Glättung der Krümmung
28060	MM_IPO_BUFFER_SIZE	Anzahl der NC-Sätze im IPO-Buffer (DRAM)
28070	MM_NUM_BLOCKS_IN_PREP	Anzahl der NC-Sätze für die Satzaufbereitung (DRAM)
28071	MM_NUM_SURF_LEVELS	Dimensionierung der Funktion COMPSURF (DRAM)
28072	MM_MAXNUM_SURF_GROUPS	Dimensionierung der Funktion COMPSURF bzgl. Achsgruppen (DRAM)
28520	MM_MAX_AXISPOLY_PER_BLOCK	Maximale Anzahl Achspolynome pro Satz
28530	MM_PATH_VELO_SEGMENTS	Anzahl der Speicherelemente zur Begrenzung der Bahngeschwindigkeit im Satz
28533	MM_LOOKAH_FFORM_UNITS	Speicher für den erweiterten LookAhead
28540	MM_ARCLENGTH_SEGMENTS	Anzahl der Speicherelemente zur Darstellung der Bogenlängenfunktion pro Satz
28610	MM_PREPDYN_BLOCKS	Anzahl Sätze zur Geschwindigkeitspräparation

5.10.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
32310	MAX_ACCEL_OVL_FACTOR	Überlastfaktor für axiale Geschwindigkeitssprünge
32431	MAX_AX_JERK	Maximaler axialer Ruck bei Bahnbewegung
32432	PATH_TRANS_JERK_LIM	Maximaler axialer Ruck am Satzübergang im Bahnsteuerbetrieb
32433	SOFT_ACCEL_FACTOR	Skalierung der Beschleunigungsbegrenzung bei SOFT
32434	G00_ACCEL_FACTOR	Skalierung der Beschleunigungsbegrenzung bei G00
32435	G00_JERK_FACTOR	Skalierung der axialen Ruckbegrenzung bei G00
32440	LOOKAH_FREQUENCY	Glättungsgrenzfrequenz bei LookAhead
33100	COMPRESS_POS_TOL	Maximale Abweichung bei Kompression
33120	PATH_TRANS_POS_TOL	Maximale Abweichung beim Überschleifen mit G645
35240	ACCEL_TYPE_DRIVE	Beschleunigungskennlinie DRIVE für Achsen Ein/Aus
36000	STOP_LIMIT_COARSE	Genauhalt grob

5.10 Datenlisten

Nummer	Bezeichner: \$MA_	Beschreibung
36010	STOP_LIMIT_FINE	Genauhalt fein
36012	STOP_LIMIT_FACTOR	Faktor Genauhalt grob/fein und Stillstandsüberwachung
36020	POSITIONING_TIME	Verzögerungszeit Genauhalt fein

5.10.2 Settingdaten

5.10.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42465	SMOOTH_CONTUR_TOL	Maximale Konturabweichung beim Überschleifen
42466	SMOOTH_ORI_TOL	Maximale Abweichung der Werkzeugorientierung beim Überschleifen
42470	CRIT_SPLINE_ANGLE	Ecken-Grenzwinkel bei Kompression mit COMPCAD
42471	MIN_CURV_RADIUS	Minimaler Krümmungsradius bei Kompression mit COMPCAD
42472	MIN_SURF_RADIUS	Minimaler Krümmungsradius bei Kompression mit COMPSURF
42473	ACTNUM_SURF_GROUPS	Anzahl Achsgruppen bei COMPSURF
42475	COMPRESS_CONTUR_TOL	Maximal erlaubte Konturabweichung bei Kompression
42476	COMPRESS_ORI_TOL	Maximale Abweichung der Werkzeugorientierung bei Kompression Hinweis: Nur bei aktiver Orientierungstransformation!
42477	COMPRESS_ORI_ROT_TOL	Maximale Abweichung der Werkzeugdrehung bei Kompression Hinweis: Nur bei 6-Achs-Maschinen mit drehbarem Werkzeug!
42676	ORI_SMOOTH_TOL	Toleranz zur Glättung der Orientierung beim Überschleifen
42678	ORISON_TOL	Toleranz zur Glättung der Orientierung

5.10.3 Signale

5.10.3.1 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Alle Achsen stehen	DB21,DBX36.3	DB330x.DBX4.3

5.10.3.2 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Position erreicht mit Genauhalt grob	DB31,DBX60.6	DB390x.DBX0.6
Position erreicht mit Genauhalt fein	DB31,DBX60.7	DB390x.DBX0.7

B2: Beschleunigung

6.1 Kurzbeschreibung

6.1.1 Allgemeine Informationen

Funktionsumfang

Die Funktionsbeschreibung umfasst die Teilfunktionen:

- Beschleunigung
- Ruck
- Geknickte Beschleunigungskennlinie

Beschleunigung und Ruck

Durch achs- und kanalspezifisch parametrierbare Maximalwerte, sowie in Teileprogrammen und Synchronaktionen programmierbare Beschleunigungsprofile, dynamischen Anpassungen und Begrenzungen, lassen sich die wirksame Beschleunigung und Ruck optimal an die Maschine und die jeweilige Bearbeitungssituation anpassen.

Geknickte Beschleunigungskennlinie

Durch die geknickte Beschleunigungskennlinie lässt sich für Maschinenachsen mit Motor, insbesondere Schrittmotoren, mit einem stark drehzahlabhängigen Drehmomentenverlauf, die Beschleunigung so einstellen, dass eine optimale Auslastung des Motors bei gleichzeitigem Schutz vor Überlastung erzielt wird.

6.1.2 Merkmale

Beschleunigung

Achsspezifische Funktionen:

- Parametrierbarer Maximalwert der Beschleunigung
- Über Teileprogrammanweisung anwählbares Beschleunigungsprofil: Beschleunigung ohne Ruckbegrenzung (**BRISKA**)
- Vorgabe des Maximalwertes per Teilprogrammanweisung (**ACC**)
- Eigener Maximalwert zum Verfahren mit programmiertem Eilgang (**G00**).
- Eigener Maximalwert zum Verfahren mit aktiver Ruckbegrenzung
- Beschleunigungsüberhöhung bei nicht tangentialen Satzübergängen

Kanalspezifische Funktionen:

- Über Teileprogrammanweisung anwählbares Beschleunigungsprofil: Beschleunigung ohne Ruckbegrenzung (**BRISK**)
- Parametrierbare Konstantfahrzeit zur Vermeidung von extremen Beschleunigungssprüngen
- Parametrierbare Beschleunigungsreserve für überlagerte Verfahrbewegungen
- Einstellbare Beschleunigungsbegrenzung
- Einstellbare Beschleunigung für spezifische Echtzeitereignisse
- Parametrierbare Beschleunigungsreserve für die Radialbeschleunigung

Ruck

Achsspezifische Funktionen:

- Über Teileprogrammanweisung anwählbares Beschleunigungsprofil: Beschleunigung mit Ruckbegrenzung (**SOFTA**)
- Parametrierbarer Maximalwert des Rucks für Einzelachsinterpolation
- Parametrierbarer Maximalwert des Rucks für Bahninterpolation

Kanalspezifische Funktionen:

- Über Teileprogrammanweisung anwählbares Beschleunigungsprofil: Beschleunigung mit Ruckbegrenzung (**SOFT**)
- Einstellbare Begrenzung des Rucks
- Einstellbarer Bahnruck für spezifische Echtzeitereignisse
- Eigener Maximalwert zum Verfahren mit programmiertem Eilgang (**G00**)
- Rucküberhöhung bei nicht krümmungsstetigen Satzübergängen

Geknickte Beschleunigungskennlinie

Eine geknickte Beschleunigungskennlinie wird über folgende Kenndaten parametriert:

- Maximale Geschwindigkeit v_{\max}
- Maximale Beschleunigung a_{\max}
- Reduziergeschwindigkeit v_{red}
- Reduzierbeschleunigung a_{red}
- Verlauf der Beschleunigungsreduktion (konstant, hyperbolisch, linear)

6.2 Funktionen

6.2.1 Beschleunigung ohne Ruckbegrenzung (BRISK/BRISKA) (kanal-/achsspezifisch)

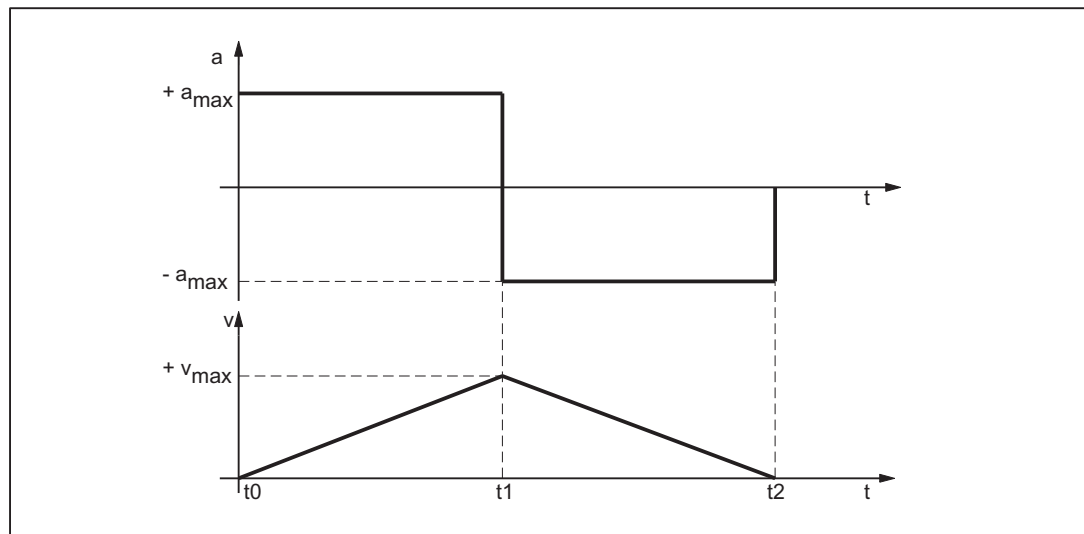
6.2.1.1 Allgemeine Informationen

Allgemeine Informationen

Bei Beschleunigung ohne Ruckbegrenzung (Ruck = unendlich) wird sofort mit dem Maximalwert beschleunigt. Bezüglich einer Beschleunigung mit Ruckbegrenzung ergeben sich folgende Unterschiede:

- **Vorteile**
Kürzere Bearbeitungszeiten bei gleichen Maximalwerten für Geschwindigkeit und Beschleunigung.
- **Nachteile**
Höhere Belastung der Maschinenmechanik und Gefahr der Anregung von hochfrequenten, schlecht regelbaren, mechanischen Schwingungen.

Beschleunigungsprofil



a_{\max} : Maximalwert der Beschleunigung

v_{\max} : Maximalwert der Geschwindigkeit

t : Zeit

Bild 6-1 Prinzipieller Verlauf von Geschwindigkeit und Beschleunigung bei sprungförmigem Beschleunigungsprofil

Aus dem obigen Bild lassen sich folgende Eigenschaften des Beschleunigungsprofils erkennen:

- Zeitpunkt: t_0
Beschleunigungssprung von 0 auf $+a_{\max}$
- Intervall: $t_0 - t_1$
Konstante Beschleunigung mit $+a_{\max}$; lineare Zunahme der Geschwindigkeit
- Zeitpunkt: t_1
Beschleunigungssprung von $2 * a_{\max}$ beim unmittelbaren Umschalten von Beschleunigen auf Bremsen

Hinweis

Der Beschleunigungssprung kann normalerweise durch Vorgabe einer Konstantfahrzeit (siehe Kapitel "Konstantfahrzeit (kanalspezifisch) (Seite 270)") vermieden werden.

- Intervall: $t_1 - t_2$
Konstante Beschleunigung mit $-a_{\max}$; lineare Abnahme der Geschwindigkeit

6.2.1.2 Parametrierung

Maximale axiale Beschleunigung für Bahnbewegungen

Die maximale axiale Beschleunigung für Bahnbewegungen kann Technologie-spezifisch für jede Maschinenachse über folgendes Maschinendatum eingestellt werden:

MD32300 \$MA_MAX_AX_ACCEL[<Parametersatz-Index>]

mit <Parametersatz-Index> = 0, 1, 2 ... (max. Parametersatz-Nummer - 1)

Zu den Technologie-spezifischen Parametersätzen siehe Kapitel "Dynamikmodus für Bahninterpolation (Seite 230)".

Die Bahnparameter werden von der Bahnplanung des Vorlaufs so berechnet, dass die parametrisierten Maximalwerte der an der Bahn beteiligten Maschinenachsen nicht überschritten werden.

Hinweis

Eine Überschreitung des Maximalwerts für spezifische Bearbeitungssituationen ist möglich (siehe Kapitel "Beschleunigungsanpassung (ACC) (achsspezifisch) (Seite 272)" und "Bahnbeschleunigung für Echtzeitereignisse (kanalspezifisch) (Seite 275)").

Maximale axiale Beschleunigung für Positionierachsbewegungen

Bei Positionierachsbewegungen wird abhängig vom eingestellten Positionierachsdynamikmodus einer der beiden folgenden Maximalwerte wirksam:

- MD32300 \$MA_MAX_AX_ACCEL [0] (maximale axiale Beschleunigung bei Bahnbewegungen im Dynamikmodus DYNNORM)
- MD32300 \$MA_MAX_AX_ACCEL [1] (maximale axiale Beschleunigung bei Bahnbewegungen im Dynamikmodus DYNPOS)

Der Positionierachsdynamikmodus wird eingestellt im NC-spezifischen Maschinendatum:

MD18960 \$MN_POS_DYN_MODE = <Modus>

<Modus>	Bedeutung
0	wirksame maximale axiale Beschleunigung: MD32300 \$MA_MAX_AX_ACCEL[0]
1	wirksame maximale axiale Beschleunigung: MD32300 \$MA_MAX_AX_ACCEL[1]

Maximale axiale Beschleunigung für JOG-Bewegungen

Für den JOG-Betrieb kann für jede Maschinenachse ein JOG-spezifischer Beschleunigungsmaximalwert projiziert werden (siehe Kapitel "Beschleunigung und Ruck bei JOG-Bewegungen (Seite 304)").

6.2.1.3 Programmierung

Bahnbeschleunigung ohne Ruckbegrenzung (BRISK)

Syntax

BRISK

Funktionalität

Über die Teileprogrammanweisung `BRISK` wird das Beschleunigungsprofil "ohne Ruckbegrenzung" für die Bahnbeschleunigung angewählt.

G-Gruppe: 21

Wirksamkeit: modal

Reset-Verhalten

Durch Reset wird die kanalspezifisch parametrisierte Grundstellung wirksam:

MD20150 \$MC_GCODE_RESET_VALUES[20]

Randbedingungen

Wird in einem Teileprogrammen das Beschleunigungsprofil während der Bearbeitung gewechselt (BRISK / SOFT) erfolgt ein Genauhalt am Satzende.

Einzelachsbeschleunigung ohne Ruckbegrenzung (BRISKA)

Syntax

BRISKA (*Achse*{*Achse*})

Funktion

Über die Teileprogrammanweisung BRISKA wird das Beschleunigungsprofil "ohne Ruckbegrenzung" für Einzelachsbewegungen (JOG, JOG/INC, Positionierachse, Pendelachse, etc.) angewählt.

G-Gruppe: -

Wirksamkeit: modal

Achse:

- Wertebereich: Achsname der Kanalachsen

Achsspezifische Grundstellung

Beschleunigung ohne Ruckbegrenzung kann als achsspezifische Grundstellung für Einzelachsbewegungen vorgegeben werden:

MD32420 \$MA_JOG_AND_POS_JERK_ENABLE = FALSE

Reset-Verhalten

Durch Reset wird die achsspezifisch parametrisierte Grundstellung wirksam:

MD32420 \$MA_JOG_AND_POS_ENABLE

6.2.2 Konstantfahrzeit (kanalspezifisch)

6.2.2.1 Allgemeine Informationen

Übersicht

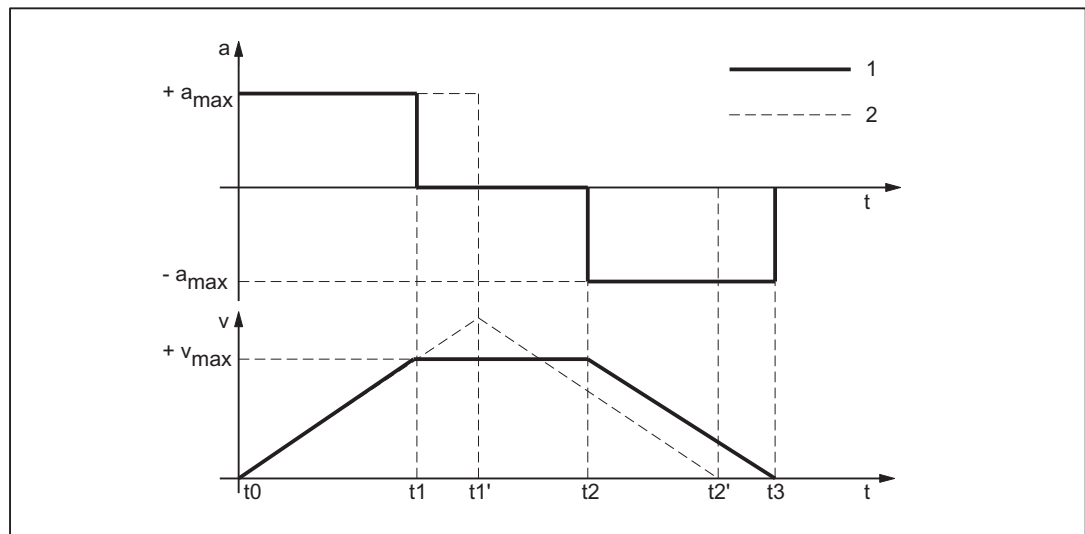
Bei Beschleunigung ohne Ruckbegrenzung tritt beim Wechsel von Beschleunigung und Bremsen ein Beschleunigungssprung von $2 \cdot a_{\max}$ auf. Zur Vermeidung dieses Beschleunigungssprungs kann eine kanalspezifische Konstantfahrzeit parametrisiert werden. Die Konstantfahrzeit legt die Zeit fest, mit der zwischen Beschleunigungs- und Bremsphase mit konstanter Geschwindigkeit verfahren wird:

MD20500 \$MC_CONST_VELO_MIN_TIME (Minimale Zeit mit konstanter Geschwindigkeit)

Hinweis

Die Konstantfahrzeit wirkt nicht bei:

- Aktiver Funktion: LookAhead
- In Verfahrssätzen mit einer Verfahrzeit kleiner bzw. gleich dem Interpolatorakt.



- 1: Verlauf mit Konstantfahrzeit
 2: Verlauf ohne Konstantfahrzeit
 a_{\max} : Maximalwert der Beschleunigung
 v_{\max} : Maximalwert der Geschwindigkeit
 t: Zeit

Bild 6-2 Prinzipieller Verlauf bei sprunghafter Beschleunigung

Aus dem obigen Bild lässt sich die Wirkung der Konstantfahrzeit erkennen:

- Zeitpunkt: t_1
Ende der Beschleunigungsphase mit Beschleunigungssprung $1 \cdot a_{\max}$
- Intervall: $t_1 - t_2$
Beschleunigung 0; Konstante Geschwindigkeit über die parametrisierte Konstantfahrzeit
- Zeitpunkt: t_2
Beginn der Bremsphase mit Beschleunigungssprung $1 \cdot a_{\max}$

Die Zeitpunkte t_0 , t_1' und t_2' kennzeichnen den entsprechenden Verlauf, der sich ohne Konstantfahrzeit ergeben hätte.

6.2.2.2 Parametrierung

Die Parametrierung der Konstantfahrzeit erfolgt kanalspezifisch über Maschinendaten:

MD20500 \$MC_CONST_VELO_MIN_TIME
(Minimale Zeit mit konstanter Geschwindigkeit)

6.2.3 Beschleunigungsanpassung (ACC) (achsspezifisch)

6.2.3.1 Allgemeine Informationen

Funktion

Über den Befehl **ACC** kann achsspezifisch, die in den Beschleunigungs-spezifischen Maschinendaten parametrisierte und aktuell wirkende Maximalbeschleunigung der Achse, reduziert werden. Die Reduzierung erfolgt über einen prozentualen Faktor, der bei der Programmierung des Befehls mit angegeben wird.

Die Reduzierung der maximal möglichen Beschleunigung einer Achse in einem Bearbeitungsabschnitt kann z.B. angewandt werden, um mechanische Schwingungen aufgrund hoher Bearbeitungskräfte zu vermeiden.

Wirksamkeit

Die Beschleunigungsreduzierung mit **ACC** **wirkt** bei allen Interpolationsarten in den Betriebsarten AUTOMATIK, MDA und bei Probelaufvorschub.

Die Beschleunigungsreduzierung **wirkt nicht**:

- In Betriebsart JOG
- Während Maschinenfunktion REF (Referenzpunktfahren).
- Wenn aufgrund eines erkannten Fehlers die Achse mit Schnellstopp (Sollwert = 0) stillgesetzt wird

6.2.3.2 Programmierung

Syntax

ACC [<Achse>]=<Reduzierfaktor>

ACC [SPI (<Spindelnummer>)] =<Reduzierfaktor>

ACC (S<Spindelnummer>) =<Reduzierfaktor>

Bedeutung

<ACC>:	Befehl zur achsspezifischen Reduzierung, der aus den Maschinendaten abgeleiteten, aktuellen maximal möglichen Beschleunigung	
	Wirksamkeit:	Modal
<Achse>:	Kanalachsname der Bahnachse	
	Datentyp:	AXIS
	Wertebereich:	Kanalachsnamen

SPI (<Spindelnummer>):	Die Funktion SPI (...) konvertiert die Spindelnummer in den entsprechenden Kanalachsenamen.
S<Spindelnummer>:	Spindelname im Kanal

Weitere Informationen

Systemvariable

Die aktuell im Kanal wirkende, mit ACC eingestellte, Beschleunigungsreduzierung, kann achsspezifisch gelesen werden über:

\$AA_ACC [<Achse>]

Reset-Verhalten

Die mit ACC eingestellte Beschleunigungsreduzierung kann über Kanal-Reset oder Programmende hinaus erhalten bleiben. Die Parametrierung erfolgt über:

MD32320 \$MA_DYN_LIMIT_RESET_MASK

6.2.4 Beschleunigungsreserve (kanalspezifisch)

6.2.4.1 Allgemeine Informationen

Allgemeine Information

Im Normalfall nutzt der Vorlauf für die Bahnbeschleunigung die parametrisierten Maximalwerte der Maschinenachsen zu 100% aus. Um eine Beschleunigungsreserve für überlagerte Bewegungen z.B. im Rahmen der Funktion "Schnellabheben von der Kontur" zu erhalten, kann die Bahnbeschleunigung um einen parametrierbaren Faktor vermindert werden. Bei einem Faktor von z.B. 0,2 wird dann vom Vorlauf die maximal mögliche Bahnbeschleunigung nur zu 80% ausgenutzt. 20% verbleiben als Beschleunigungsreserve für überlagerte Bewegungen.

6.2.4.2 Parametrierung

Die Parametrierung der Beschleunigungsreserve erfolgt kanalspezifisch über Maschinendatum:

MD20610 \$MC_ADD_MOVE_ACCEL_RESERVE
(Beschleunigungsreserve für überlagerte Bewegungen)

6.2.5 Begrenzung der Bahnbeschleunigung (kanalspezifisch)

6.2.5.1 Allgemeine Informationen

Allgemeine Informationen

Um auf die jeweilige Bearbeitungssituationen flexibel reagieren zu können, kann die vom Vorlauf berechnete Bahnbeschleunigung kanalspezifisch über Settingdaten begrenzt werden:

SD42500 \$SC_SD_MAX_PATH_ACCEL (Maximale Bahnbeschleunigung)

Der im Settingdatum vorgegebene Wert wird nur dann berücksichtigt, wenn er kleiner ist als die vom Vorlauf berechnete Bahnbeschleunigung.

Die Begrenzung muss kanalspezifisch über Settingdaten freigegeben werden:

SD42502 \$SC_IS_SD_MAX_PATH_ACCEL = TRUE

6.2.5.2 Parametrierung

Die Parametrierung erfolgt kanalspezifisch über Settingdaten:

SD42500 \$SC_SD_MAX_PATH_ACCEL (Maximale Bahnbeschleunigung)

SD42502 \$SC_IS_SD_MAX_PATH_ACCEL (Aktivierung der Begrenzung der Bahnbeschleunigung)

6.2.5.3 Programmierung

Begrenzungswert

Syntax

`$SC_SD_MAX_PATH_ACCEL` = *Begrenzungswert*

Funktionalität

Die Begrenzung der Bahnbeschleunigung kann durch Programmierung des Settingdatums angepasst werden.

Begrenzungswert:

- Wertebereich: ≥ 0
- Einheit: m/s^2

Anwendbarkeit:

- Teileprogramm
- Statische Synchronaktion

Ein/Ausschalten

Syntax

`$SC_IS_SD_MAX_PATH_ACCEL = Wert`

Funktionalität

Die Begrenzung der Bahnbeschleunigung kann durch Programmierung des Settingdatums ein/ausgeschaltet werden.

Parameter: *Wert*

- Wertebereich: TRUE, FALSE

Anwendbarkeit:

- Teileprogramm
- Statische Synchronaktion

6.2.6 Bahnbeschleunigung für Echtzeitereignisse (kanalspezifisch)

6.2.6.1 Allgemeine Informationen

Allgemeine Informationen

Um keinen Kompromiss zwischen bearbeitungsoptimaler Beschleunigung einerseits und zeitoptimaler Beschleunigung bei folgenden Echtzeitereignissen:

- NC-Stop / NC-Start
- Änderungen des Vorschub-Overrides
- Änderung der Geschwindigkeitsvorgabe für die "sicher reduzierte Geschwindigkeit" im Rahmen der Funktion "Safety Integrated"

andererseits eingehen zu müssen, kann für die genannten Echtzeitereignisse die Bahnbeschleunigung über eine kanalspezifische Systemvariable vorgegeben werden:

`$AC_PATHACC = Bahnbeschleunigung`

Die Bahnbeschleunigung für Echtzeitereignisse ist nur für die Zeitdauer der Geschwindigkeitsänderung aufgrund eines der genannten Echtzeitereignisse wirksam.

Begrenzung

Übersteigt die angegebene Bahnbeschleunigung das Leistungsvermögen der an der Bahn beteiligten Maschinenachsen, wird die Bahnbeschleunigung Steuerungs-intern derart begrenzt, dass die resultierende axiale Beschleunigung (a_{res}) höchstens das 2-fache des parametrisierten axialen Maximalwertes (a_{max}) beträgt.

$$a_{\text{res}} = 2 * a_{\text{max}}, \text{ mit } a_{\text{max}} = \text{MD32300 \$MA_MAX_AX_ACCEL}$$

Hinweis

Die Bahnbeschleunigung für Echtzeitereignisse wird ohne Rücksicht auf die Radialbeschleunigung zugelassen.

Wirksamkeit

Wirkt	Die Bahnbeschleunigung für Echtzeitereignisse wirkt in den Betriebsarten AUTOMATIK und MDA nur im Zusammenhang mit folgenden Echtzeitereignissen: <ul style="list-style-type: none"> • NC-Stop / NC-Start • Override-Änderungen • Änderung der Geschwindigkeitsvorgabe für die "sicher reduzierte Geschwindigkeit" im Rahmen der Funktion "Safety Integrated"
Wirkt nicht	Die Bahnbeschleunigung für Echtzeitereignisse wirkt nicht bei Änderungen der Bahngeschwindigkeit, die sich aufgrund der Bahnplanung im Vorlauf des Kanals ergeben, wie z. B. Konturkrümmungen, Ecken, kinematische Begrenzungen der Transformation, etc. Die Bahnbeschleunigung für Echtzeitereignisse wirkt nicht, wenn der programmierte Wert kleiner ist als die für den betroffenen Bahnabschnitt vom Vorlauf berechnete Bahnbeschleunigung.

Programmierung

Zur Programmierung der Systemvariablen im Teileprogramm oder Synchronaktion siehe Kapitel "Programmierung (Seite 276)".

6.2.6.2 Programmierung**Syntax**

$\$AC_PATHACC = \text{Bahnbeschleunigung}$

Funktionalität

Über die kanalspezifische Systemvariable wird die Bahnbeschleunigung für Echtzeitereignisse vorgegeben.

Parameter: *Bahnbeschleunigung*

- Wertebereich: Bahnbeschleunigung ≥ 0
- Einheit: m/s^2

Ausschalten $\$AC_PATHACC = 0$

Anwendbarkeit:

- Teileprogramm
- statischen Synchronaktion

Reset-Verhalten

Bei Reset wird die Bahnbeschleunigung für Echtzeitereignisse ausgeschaltet.

Randbedingungen

Durch Programmierung von `$AC_PATHACC` im Teileprogramm wird implizit ein Vorlaufstopp mit Reorg ausgelöst (`STOPRE`).

6.2.7 Beschleunigung bei programmiertem Eilgang (G00) (achsspezifisch)

6.2.7.1 Allgemeine Informationen

Oftmals muss die Beschleunigung für die an der Bearbeitung beteiligten Maschinenachsen aufgrund der bearbeitungsspezifischen Randbedingungen niedriger eingestellt werden als es der Leistungsfähigkeit der Maschine entspricht.

Zum zeitoptimalen Verfahren der Maschinenachsen bei programmiertem Eilgang (Teileprogrammanweisung `G00`), kann ein eigener Maximalwert der achsspezifischen Beschleunigung parametrierbar werden.

Einrichtebetriebsart JOG

Die Beschleunigung bei Eilgangüberlagerung in der Einrichtebetriebsart JOG wird durch diese Funktionalität nicht beeinflusst.

6.2.7.2 Parametrierung

Die Parametrierung des Maximalwerts der achsspezifischen Beschleunigung bei programmiertem Eilgang (`G00`) erfolgt über das achsspezifische Maschinendatum:

MD32434 `$MA_G00_ACCEL_FACTOR`
(Skalierung der Beschleunigungsbegrenzung bei `G00`)

Daraus ergibt sich der von der Bahnplanung im Vorlauf berücksichtigte Maximalwert der achsspezifischen Beschleunigung bei programmiertem Eilgang (G00) zu:

Beschleunigung[Achse] =
MD32300 \$MA_MAX_AX_ACCEL * MD32434 \$MA_G00_ACCEL_FACTOR

6.2.8 Beschleunigung bei aktiver Ruckbegrenzung (SOFT/SOFTA) (achsspezifisch)

6.2.8.1 Allgemeine Informationen

Funktion

Beim Beschleunigen mit Ruckbegrenzung entsteht bei gleichem Maximalwert der Beschleunigung ein gewisser Zeitverlust bezogen auf ein Beschleunigen ohne Ruckbegrenzung. Zum Ausgleich dieses Zeitverlustes, kann für das Verfahren der Maschinenachsen bei aktiver Ruckbegrenzung (SOFT/SOFTA), ein eigener Maximalwert der achsspezifische Beschleunigung parametrieren werden.

Die Parametrierung des Maximalwerts der Beschleunigung bei aktiver Ruckbegrenzung erfolgt über einen Faktor bezogen auf den achsspezifischen Maximalwert. Daraus ergibt sich der von der Bahnplanung im Vorlauf berücksichtigte Maximalwert der achsspezifischen Beschleunigung bei aktiver Ruckbegrenzung zu:

Beschleunigung[Achse] =
MD32300 \$MA_MAX_AX_ACCEL * MD32433 \$MA_SOFT_ACCEL_FACTOR

6.2.8.2 Parametrierung

Die Parametrierung des Maximalwerts der Beschleunigung bei aktiver Ruckbegrenzung (SOFT/SOFTA) erfolgt über das achsspezifische Maschinendatum:

MD32434 \$MA_SOFT_ACCEL_FACTOR
(Skalierung der Beschleunigungsbegrenzung bei SOFT)

6.2.9 Beschleunigungsüberhöhung bei nicht tangentialen Satzübergängen (achsspezifisch)

6.2.9.1 Allgemeine Informationen

Funktion

Bei nicht tangentialen Satzübergängen (Ecken) muss von der Steuerung zur Einhaltung der parametrisierten Achsdynamik die Verbahrbewegung der Geometrieachsen unter Umständen stark abgebremst werden. Zur Verminderung bzw. Vermeidung des Abbremsens an nicht tangentialen Satzübergängen kann eine höhere achsspezifische Beschleunigung zugelassen werden.

Die Parametrierung der Beschleunigungsüberhöhung erfolgt über einen Faktor bezogen auf den achsspezifischen Maximalwert. Daraus ergibt sich der von der Bahnplanung im Vorlauf berücksichtigte Maximalwert der achsspezifischen Beschleunigung bei nicht tangentialen Satzübergängen zu:

$$\text{Beschleunigung[Achse]} = \text{MD32300 \$MA_MAX_AX_ACCEL} * \text{MD32310 \$MA_MAX_ACCEL_OVL_FACTOR}$$

6.2.9.2 Parametrierung

Die Parametrierung der Beschleunigungsüberhöhung bei nicht tangentialen Satzübergängen erfolgt über das achsspezifische Maschinendatum:

MD32310 \$MA_MAX_ACCEL_OVL_FACTOR
(Überlastfaktor für Geschwindigkeitssprünge)

6.2.10 Beschleunigungsreserve für die Radialbeschleunigung (kanalspezifisch)

6.2.10.1 Allgemeine Informationen

Übersicht

An gekrümmten Konturen wirkt neben der Bahnbeschleunigung (Tangentialbeschleunigung) zusätzlich die Radialbeschleunigung. Wird diese bei der Parametrierung der Bahnparameter nicht berücksichtigt, kann die wirksame axiale Beschleunigung während Beschleunigungs- oder Bremsvorgängen auf der gekrümmten Kontur kurzfristig bis zum 2-fachen des Maximalwertes betragen.

$$\begin{aligned} \text{Wirksame axiale Beschleunigung} &= \\ \text{Anteil der Bahnbeschleunigung} + \text{Anteil der Radialbeschleunigung} &= \\ 2 * (\text{MD32300 \$MA_MAX_AX_ACCEL}) & \end{aligned}$$

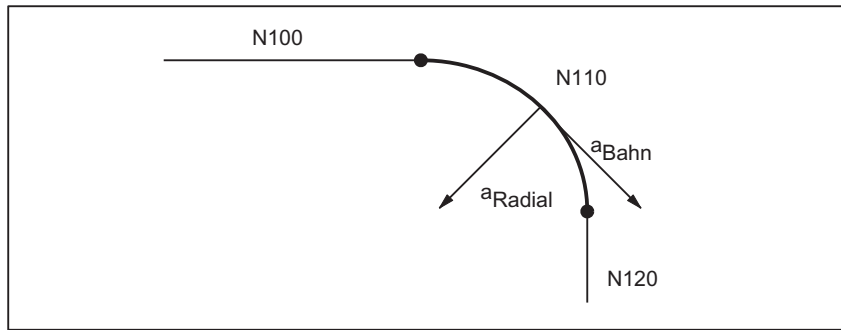


Bild 6-3 Radial- und Bahnbeschleunigung an gekrümmten Konturen

Über das kanalspezifische Maschinendatum:

MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL

(Einfluss der Bahnkrümmung auf die Bahndynamik)

kann der Anteil der achsspezifischen Beschleunigung eingestellt werden, der als Beschleunigungsreserve für die Radialbeschleunigung berücksichtigt werden soll.

Bei einem Wert von z.B. 0.75 werden 75% der achsspezifischen Beschleunigung für die Radialbeschleunigung und 25% für die Bahnbeschleunigung zur Verfügung gestellt.

Allgemein berechnen sich die entsprechenden Maximalwerte zu:

Radialbeschleunigung =

MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL * MD32300 \$MA_MAX_AX_ACCEL

Bahnbeschleunigung =

(1 - MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL) * MD32300 \$MA_MAX_AX_ACCEL

Beispiel

Folgende Maschinenparameter sind gegeben:

- MD32300 \$MA_MAX_AX_ACCEL für alle Geometrieachsen: 3 m/s
- Maximale Bahngeschwindigkeit bei einem Bahnradius von 10 mm aufgrund mechanischer Gegebenheiten an der Maschine: 5 m/min

Die Radialbeschleunigung berechnet sich daraus zu:

$$a_{\text{Radial}} = \frac{v_{\text{Bahn}}^2 [\text{m/min}]}{r [\text{mm}] * 3.6 [\text{m/s}^2]} = \frac{5^2}{10 * 3.6} = 0.694 \text{ m/s}^2$$

Daraus ergibt sich die Einstellung für die Beschleunigungsreserve zu:

$$\text{MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL} = \frac{a_{\text{Radial}} [\text{m/s}^2]}{\text{MD32300 \$MA_MAX_AX_ACCEL} [\text{m/s}^2]} = \frac{0.694}{3} \approx 0.23$$

Linearsätze

Bei Linearsätzen (Geradeninterpolation) ohne aktive kinematische Transformation wirkt die genannte Beschleunigungsreserve nicht.

6.2.10.2 Parametrierung

Die Parametrierung des Anteils an der maximalen Achsbeschleunigung, der als Beschleunigungsreserve für die Radialbeschleunigung an gekrümmten Konturen berücksichtigt werden soll, erfolgt über das kanalspezifische Maschinendatum:

MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL
(Einfluss der Bahnkrümmung auf die Bahndynamik)

6.2.11 Ruckbegrenzung bei Bahninterpolation (SOFT) (kanalspezifisch)

6.2.11.1 Allgemeine Informationen

Übersicht

Im Rahmen der im weiteren Verlauf beschriebenen Funktionalität wird immer von stetiger Beschleunigung, d. h. von Beschleunigung mit Ruckbegrenzung (Ruck = endlicher Wert) als Beschleunigungsprofil ausgegangen. Bei Beschleunigung mit Ruckbegrenzung wird die Beschleunigung von 0 bis zum Maximalwert linear interpoliert.

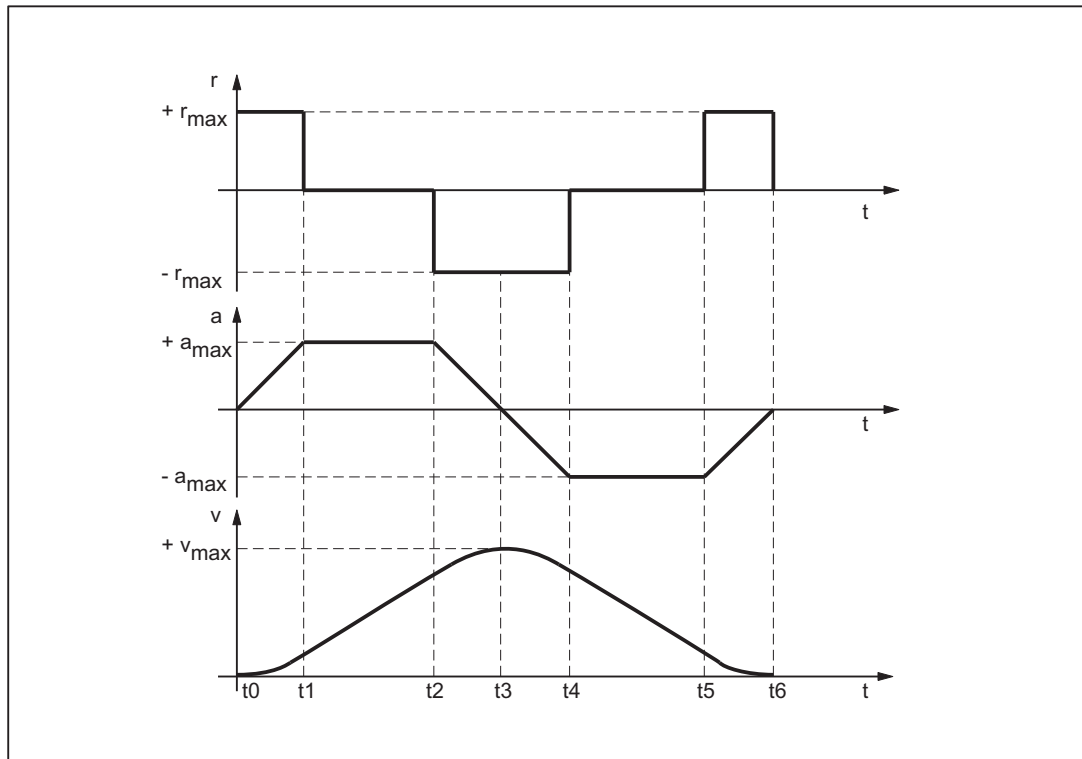
Vorteile

Geringe Belastung der Maschinenmechanik und kaum Gefahr der Anregung von hochfrequenten, schlecht regelbaren, mechanischen Schwingungen aufgrund der stetigen Beschleunigungserhöhung.

Nachteile

Längere Bearbeitungszeiten bei gleichen Maximalwerten für Geschwindigkeit und Beschleunigung gegenüber dem sprungförmigen Beschleunigungsprofil.

Beschleunigungsprofil



r_{\max} : Maximalwert des Rucks
 a_{\max} : Maximalwert der Beschleunigung
 v_{\max} : Maximalwert der Geschwindigkeit
 t : Zeit

Bild 6-4 Prinzipieller Verlauf von Ruck, Beschleunigung und Geschwindigkeit bei ruckbegrenztem Beschleunigungsprofil

Aus dem obigen Bild lassen sich folgende Eigenschaften des Beschleunigungsprofils erkennen:

- Intervall: $t_0 - t_1$
Konstanter Ruck mit $+r_{\max}$; lineare Zunahme der Beschleunigung; quadratische Zunahme der Geschwindigkeit
- Intervall: $t_1 - t_2$
Konstante Beschleunigung mit $+a_{\max}$; lineare Zunahme der Geschwindigkeit
- Intervall: $t_2 - t_3$
Konstanter Ruck mit $-r_{\max}$; lineare Abnahme der Beschleunigung; quadratische Abnahme der Geschwindigkeitserhöhung bis zum Erreichen des Maximalwertes $+v_{\max}$
- Intervall: $t_3 - t_4$
Konstanter Ruck mit $-r_{\max}$; lineare Zunahme der Bremsbeschleunigung; quadratische Abnahme der Geschwindigkeit

- Intervall: $t_4 - t_5$
Konstante Bremsbeschleunigung mit $-a_{\max}$; lineare Abnahme der Geschwindigkeit
- Intervall: $t_5 - t_6$
Konstanter Ruck mit $+r_{\max}$; lineare Abnahme der Bremsbeschleunigung; quadratische Abnahme der Geschwindigkeitsverminderung bis zum Stillstand $v = 0$

6.2.11.2 Parametrierung

Maximalwert des Rucks für Bahnbewegungen (achsspezifisch)

Der maximale axiale Ruck für Bahnbewegungen kann Technologie-spezifisch für jede Maschinenachse über folgendes Maschinendatum eingestellt werden:

MD32431 \$MA_MAX_AX_JERK[<Parametersatz-Index>]

mit <Parametersatz-Index> = 0, 1, 2 ... (max. Parametersatz-Nummer - 1)

Zu den Technologie-spezifischen Parametersätzen siehe Kapitel "Dynamikmodus für Bahninterpolation (Seite 230)".

Die Bahnparameter werden von der Bahnplanung des Vorlaufs so berechnet, dass die parametrisierten Maximalwerte der an der Bahn beteiligten Maschinenachsen nicht überschritten werden.

Hinweis

Eine Überschreitung des Maximalwerts für spezifische Bearbeitungssituationen ist möglich (siehe Kapitel "Bahnruck für Echtzeitereignisse (kanalspezifisch) (Seite 287)").

Maximalwert des Rucks für Bahnbewegungen (kanalspezifisch)

Neben der achsspezifischen Einstellung kann der Maximalwert des Rucks auch als kanalspezifischer Bahnparameter über folgendes Maschinendatum vorgegeben werden:

MD20600 \$MC_MAX_PATH_JERK (Bahnbezogener Maximalruck)

Um eine gegenseitige Beeinflussung von achs- und kanalspezifischen Maximalwerten des Rucks auszuschließen, ist der kanalspezifische Maximalwert auf einen Wert größer den axialen Maximalwerten einzustellen.

6.2.11.3 Programmierung

Syntax

SOFT

Funktionalität

Über die Teileprogrammanweisung `SOFT` wird das Beschleunigungsprofil mit Ruckbegrenzung für die Verfahrbewegungen der Geometrieachsen im Kanal angewählt.

G-Gruppe: 21

Wirksamkeit: modal

Reset-Verhalten

Durch Reset wird die kanalspezifisch parametrisierte Grundstellung wirksam:

MD20150 \$MC_GCODE_RESET_VALUES[20]

Randbedingungen

Wird in einem Teileprogramm der Beschleunigungsmodus während der Bearbeitung gewechselt (*BRISK* ↔ *SOFT*) erfolgt auch bei Bahnsteuerbetrieb am Übergang ein Satzwechsel mit Genauhalt am Satzende.

6.2.12 Ruckbegrenzung bei Einzelachsinterpolation (SOFTA) (achsspezifisch)

6.2.12.1 Parametrierung

Grundeinstellung der axialen Ruckbegrenzung

Die Beschleunigung mit Ruckbegrenzung kann als axiale Grundstellung vorgegeben werden:

MD32420 \$MA_JOG_AND_POS_JERK_ENABLE== TRUE

Maximaler axialer Ruck für Positionierachsbewegungen

Beim Verfahren von Positionierachsen bei aktiver Ruckbegrenzung wird der Wert aus einem der folgenden Maschinendaten als maximaler axialer Ruck wirksam:

- MD32430 \$MA_JOG_AND_POS_MAX_JERK (maximaler axialer Ruck bei Positionierachsbewegungen)
- MD32431 \$MA_MAX_AX_JERK [0] (maximaler axialer Ruck bei Bahnbewegungen im Dynamikmodus DYNNORM)
- MD32431 \$MA_MAX_AX_JERK [1] (maximaler axialer Ruck bei Bahnbewegungen im Dynamikmodus DYNPOS)

Welches Maschinendatum zur Anwendung kommt, wird bestimmt durch den eingestellten Positionierachsdynamikmodus:

MD18960 \$MN_POS_DYN_MODE = <Modus>

<Modus>	Bedeutung
	Als maximaler axialer Ruck bei Positionierachsbewegungen wirkt:
0	MD32430 \$MA_JOG_AND_POS_MAX_JERK Bei aktivem G75 (Festpunkt anfahren): MD32431 \$MA_MAX_AX_JERK[0]
1	MD32431 \$MA_MAX_AX_JERK[1]

Maximaler axialer Ruck für JOG-Bewegungen

Für den JOG-Betrieb kann für jede Maschinenachse ein JOG-spezifischer Ruck-Maximalwert projektiert werden (siehe Kapitel "Beschleunigung und Ruck bei JOG-Bewegungen (Seite 304)").

6.2.12.2 Programmierung

Syntax

`SOFTA (Achse {Achse})`

Funktionalität

Über die Teileprogrammanweisung `SOFTA` wird Beschleunigung mit Ruckbegrenzung für Einzelachsbewegungen (Positionierachse, Pendelachse, etc.) angewählt.

G-Gruppe: -

Wirksamkeit: modal

Achse:

- Wertebereich: Achsname der Kanalachsen

Achsspezifische Grundstellung

Beschleunigung mit Ruckbegrenzung kann als achsspezifische Grundstellung für Einzelachsbewegungen vorgegeben werden:

`MD32420 $MA_JOG_AND_POS_JERK_ENABLE = TRUE`

Reset-Verhalten

Durch Reset wird die achsspezifisch parametrisierte Grundstellung wirksam:

`MD32420 $MA_JOG_AND_POS_ENABLE`

6.2.13 Begrenzung des Bahnricks (kanalspezifisch)

6.2.13.1 Allgemeine Informationen

Übersicht

Um auf die jeweilige Bearbeitungssituationen flexibel reagieren zu können, kann der vom Vorlauf berechnete Bahnrick kanalspezifisch über Settingdaten begrenzt werden:

`SD42510 $SC_SD_MAX_PATH_JERK` (Maximaler Bahnrick)

Der im Settingdatum vorgegebene Wert wird im Kanal nur dann berücksichtigt, wenn er kleiner ist als der vom Vorlauf berechnete Bahnrick.

Die Begrenzung muss kanalspezifisch über Settingdaten freigegeben werden:

SD42512 \$SC_IS_SD_MAX_PATH_JERK = TRUE

6.2.13.2 Parametrierung

Die Parametrierung erfolgt kanalspezifisch über Settingdaten:

SD42510 \$SC_SD_MAX_PATH_JERK (Maximaler Bahnrick)

SD42512 \$SC_IS_SD_MAX_PATH_JERK
(Aktivierung der Begrenzung des Bahnricks)

6.2.13.3 Programmierung

Maximaler Bahnrick

Syntax

\$SC_SD_MAX_PATH_JERK = *Ruckwert*

Funktionalität

Die Begrenzung des Bahnricks kann durch Programmierung des Settingdatums angepasst werden.

Ruckwert:

- Wertebereich: ≥ 0
- Einheit: m/s^3

Anwendbarkeit:

- Teileprogramm
- Statische Synchronaktion

Ein/Ausschalten

Syntax

\$SC_IS_SD_MAX_PATH_JERK = *Wert*

Funktionalität

Die Begrenzung des Bahnricks kann durch Programmierung des Settingdatums ein/ ausgeschaltet werden.

Parameter: *Wert*

- Wertebereich: TRUE, FALSE

Anwendbarkeit:

- Teileprogramm
- Statische Synchronaktion

6.2.14 Bahnruck für Echtzeitereignisse (kanalspezifisch)

6.2.14.1 Allgemeine Informationen

Übersicht

Um keinen Kompromiss zwischen bearbeitungsoptimalem Ruck einerseits und zeitoptimalem Ruck bei folgenden Echtzeitereignissen:

- NC-Stop / NC-Start
- Änderungen des Vorschub-Overrides
- Änderung der Geschwindigkeitsvorgabe für die "sicher reduzierte Geschwindigkeit" im Rahmen der Funktion "Safety Integrated"

andererseits eingehen zu müssen, kann für die genannten Echtzeitereignisse der Bahnruck über eine kanalspezifische Systemvariable vorgegeben werden:

$\$AC_PATHJERK = \text{Bahnruck}$

Der Bahnruck für Echtzeitereignisse ist nur für die Zeitdauer der Geschwindigkeitsänderung aufgrund eines der genannten Echtzeitereignisse wirksam.

Begrenzung

Da der Ruck keine für den Antrieb relevante physikalische Größe darstellt, erfolgt **keine** Begrenzung des vorgegebenen Rucks.

Wirksamkeit

Wirkt	Der Bahnrick für Echtzeitereignisse wirkt in den Betriebsarten AUTOMATIK und MDA nur im Zusammenhang mit folgenden Echtzeitereignissen: <ul style="list-style-type: none"> • NC-Stop / NC-Start • Override-Änderungen • Änderung der Geschwindigkeitsvorgabe für die "sicher reduzierte Geschwindigkeit" im Rahmen der Funktion "Safety Integrated"
Wirkt nicht	Der Bahnrick für Echtzeitereignisse wirkt nicht bei Änderungen der Bahngeschwindigkeit, die sich aufgrund der Bahnplanung im Vorlauf des Kanals ergeben wie z.B. Konturkrümmungen, Ecken, kinematische Begrenzungen der Transformation, etc. Der Bahnrick für Echtzeitereignisse wirkt nicht, wenn der programmierte Wert kleiner ist als der für den betroffenen Bahnabschnitt vom Vorlauf berechnete Bahnrick.

Programmierung

Um für Echtzeitereignisse den Ruck entsprechend der Beschleunigung zu setzen, kann die Systemvariable folgendermaßen gesetzt werden:

$\$AC_PATHJERK = \$AC_PATHACC / \text{Glättungszeit}$

- $\$AC_PATHACC$: Bahnbeschleunigung [m/s²]
Glättungszeit: frei wählbar z. B. 0,02 s

Zur Programmierung der Systemvariablen im Teileprogramm oder Synchronaktion siehe Kapitel "Programmierung (Seite 288)".

6.2.14.2 Programmierung**Syntax**

$\$AC_PATHJERK = \text{Ruckwert}$

Funktionalität

Über die kanalspezifische Systemvariable wird der Bahnrick für Echtzeitereignisse vorgegeben.

Ruckwert:

- Wertebereich: Bahnrick ≥ 0
- Einheit: m/s³

Anwendbarkeit:

- Teileprogramm
- statischen Synchronaktion

Reset-Verhalten

Bei Reset wird die Funktion ausgeschaltet.

Randbedingungen

Durch Programmierung von \$AC_PATHJERK im Teileprogramm wird implizit ein Vorlaufstopp mit Reorg ausgelöst (STOPRE).

6.2.15 Ruck bei programmiertem Eilgang (G00) (achsspezifisch)

6.2.15.1 Allgemeine Informationen

Übersicht

Oftmals muss der maximale Ruck für die an der Bearbeitung beteiligten Maschinenachsen aufgrund der bearbeitungsspezifischen Randbedingungen niedriger eingestellt werden als es der Leistungsfähigkeit der Maschine entspricht.

Zum zeitoptimalen Verfahren der Maschinenachsen bei programmiertem Eilgang (Teileprogrammanweisung G00), kann ein eigener Maximalwert des achsspezifischen Rucks parametrierbar werden.

Einrichtebetriebsart JOG

Der Ruck bei Eilgangüberlagerung in der Einrichtebetriebsart JOG wird durch diese Funktionalität nicht beeinflusst.

6.2.15.2 Parametrierung

Die Parametrierung des Maximalwerts des achsspezifischen Rucks bei programmiertem Eilgang (G00) erfolgt über das achsspezifische Maschinendatum:

MD32434 \$MA_G00_ACCEL_FACTOR
(Skalierung der Beschleunigungsbegrenzung bei G00)

Daraus ergibt sich der von der Bahnplanung im Vorlauf berücksichtigte Maximalwert des achsspezifischen Rucks bei programmiertem Eilgang (G00) zu:

Ruck[Achse] =
MD32431 \$MA_MAX_AX_JERK * MD32435 \$MA_G00_JERK_FACTOR

6.2.16 Rucküberhöhung bei nicht krümmungsstetigen Satzübergängen (achsspezifisch)

6.2.16.1 Allgemeine Informationen

Übersicht

Bei nicht krümmungsstetigen Satzübergängen (z. B. Gerade > Kreis) muss von der Steuerung zur Einhaltung der parametrisierten Achsdynamik die Verfahrbewegung der Geometrieachsen unter Umständen stark abgebremst werden. Zur Verminderung bzw. Vermeidung des Abbremsens an nicht krümmungsstetigen Satzübergängen kann ein höherer achsspezifischer Ruck zugelassen werden.

Die Parametrierung der Rucküberhöhung erfolgt über einen eigenen achsspezifischen Maximalwert.

6.2.16.2 Parametrierung

Die Parametrierung der Rucküberhöhung bei nicht krümmungsstetigen Satzübergängen erfolgt über das achsspezifische Maschinendatum:

MD32432 \$MA_PATH_TRANS_JERK_LIM
(Rucküberhöhung bei nicht krümmungsstetigen Satzübergängen)

6.2.17 Geschwindigkeitsabhängige Ruckadaption (achsspezifisch)

Funktion

Die Bahndynamik ergibt sich aus den parametrisierten, konstanten axialen Maximalwerte für Geschwindigkeit, Beschleunigung und Ruck der an der Bahn beteiligten Achsen:

- MD32000 \$MA_MAX_AX_VELO (max. Achsgeschwindigkeit)
- MD32300 \$MA_MAX_AX_ACCEL (max. Achsbeschleunigung)
- MD32431 \$MA_MAX_AX_JERK (max. axialer Ruck bei Bahnbewegung)

Bei **Konturen mit nicht konstanter Krümmung** (Torsion), wie sie z.B. im Zusammenhang mit Freiformflächen vorkommen, können sich, hauptsächlich aufgrund des axialen Rucks, Schwankungen der Bahngeschwindigkeit, besonders im oberen Geschwindigkeitsbereich, ergeben. Die Schwankungen der Bahngeschwindigkeit führen zu Beeinträchtigungen der Oberflächengüte.

Durch eine geschwindigkeitsabhängige Erhöhung des zulässigen axialen Rucks, vermindert sich bei Konturen mit nicht konstanter Krümmung der Einfluss des axialen Rucks auf die Bahngeschwindigkeit. Bei entsprechender Parametrierung können die Schwankungen der Bahngeschwindigkeit u.U. vollständig vermieden werden.

Die geschwindigkeitsabhängige Erhöhung des zulässigen axialen Rucks hat keinen Einfluss auf die maximal mögliche Bahnbeschleunigung und Bahnrick. Diese ergeben sich auch bei aktiver Ruckadaption weiter aus den in den Maschinendaten parametrisierten, konstanten axialen Maximalwerte.

Da bei Linearbewegungen sind sowohl Krümmung als auch Torsion gleich Null sind, hat die geschwindigkeitsabhängige Ruckadaption bei Linearbewegungen keine Auswirkungen.

Verfügbarkeit

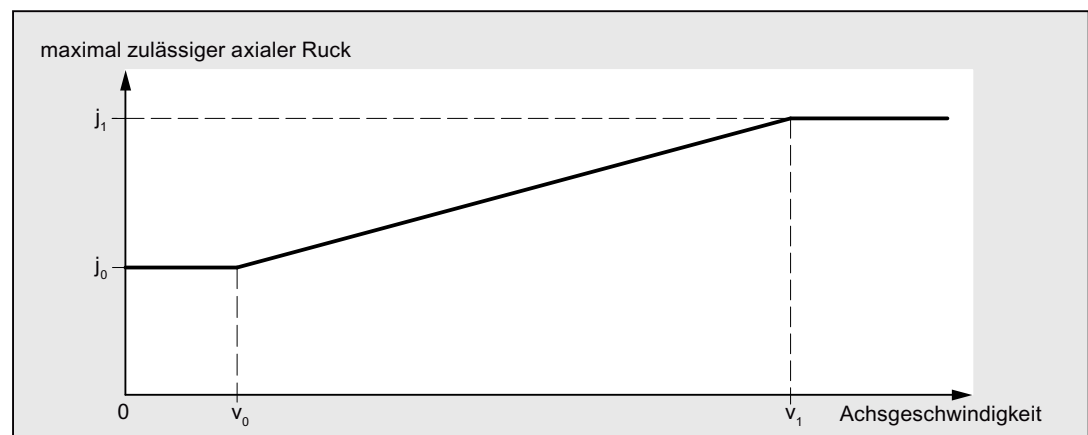
Die Funktion "Geschwindigkeitsabhängige Ruckadaption" ist unabhängig von der Funktion "Freiformflächenmodus: Grundfunktionen (Seite 232)" verfügbar.

Parametrierung

Die Parametrierung der Funktion "Geschwindigkeitsabhängige Ruckadaption" erfolgt achsspezifisch mit folgenden Maschinendaten:

- MD32437 \$MA_AX_JERK_VEL0[<n>] = <Schwellwert_{unten}>
Untere Geschwindigkeitsschwelle der Ruckadaption. Ab dieser Geschwindigkeit wird geschwindigkeitsabhängige Ruckadaption wirksam.
Die untere Geschwindigkeitsschwelle ist über den Index n, für jeden Dynamikmodus (siehe Kapitel "Dynamikmodus für Bahninterpolation (Seite 230)") separat einstellbar:
- MD32438 \$MA_AX_JERK_VEL1[<n>] = <Schwellwert_{oben}>
Obere Geschwindigkeitsschwelle, der Ruckadaption. Bei dieser Geschwindigkeit erreicht der geschwindigkeitsabhängige Ruck seinen mit MD32439 \$MA_MAX_AX_JERK_FACTOR parametrisierten Maximalwert j_{\max} .
Die obere Geschwindigkeitsschwelle ist über den Index n, für jeden Dynamikmodus (siehe Kapitel "Dynamikmodus für Bahninterpolation (Seite 230)") separat einstellbar:
- MD32439 \$MA_MAX_AX_JERK_FACTOR = <Faktor>
Faktor zur Parametrierung des maximalen geschwindigkeitsabhängigen Rucks j_{\max} bei Erreichen der oberen Geschwindigkeitsschwelle MD32438 \$MA_AX_JERK_VEL1[<n>]:
$$j_{\max} = (\text{MD32431 } \$MA_MAX_AX_JERK) * (\text{MD32439 } \$MA_MAX_AX_JERK_FACTOR)$$

Bei einem Wert > 1.0 ist die geschwindigkeitsabhängige Ruckadaption aktiv.
Bei einem Wert = 1.0, ist die geschwindigkeitsabhängige Ruckadaption inaktiv.



v_0 : MD32437 \$MA_AX_JERK_VEL0

v_1 : MD32438 \$MA_AX_JERK_VEL1

j_0 : MD32431 \$MA_MAX_AX_JERK

j_1 : MD32439 \$MA_MAX_AX_JERK_FACTOR * MD32431 \$MA_MAX_AX_JERK

Bild 6-5 Axialer Ruck als Funktion der Achsgeschwindigkeit

Hinweis

Die geschwindigkeitsabhängige Ruckadaption wird nur aktiv, wenn:

MD32439 \$MA_MAX_AX_JERK_FACTOR > 1.0

Beispiel

Beispiel für Parametrierung:

- MD32437 \$MA_AX_JERK_VELO = 3000 mm/min
- MD32438 \$MA_AX_JERK_VELO1 = 6000 mm/min
- MD32439 \$MA_MAX_AX_JERK_FACTOR[AX1] = 2.0
- MD32439 \$MA_MAX_AX_JERK_FACTOR[AX2] = 3.0
- MD32439 \$MA_MAX_AX_JERK_FACTOR[AX3] = 1.0

Auswirkung

- Die geschwindigkeitsabhängige Ruckadaption wird für die 1. und 2. Achse aktiv, während die Funktion für die 3. Achse nicht aktiv wird.
- Für Achsgeschwindigkeiten im Bereich 0 bis 3000 mm/min wird der parametrierte Ruck wirksam.
- Für Achsgeschwindigkeiten im Bereich 3000 mm/min bis 6000 mm/min wird der maximale Ruck linear erhöht.
- Der maximal erlaubte Ruck der 1. Achse wird dabei für Achsgeschwindigkeiten größer als 6000 mm/min um den Faktor 2 erhöht, bei der 2. Achse um den Faktor 3.
- Die parametrierten Werte gelten in jedem Dynamikmodus.

6.2.18 Ruckfilter (achsspezifisch)

6.2.18.1 Allgemeine Informationen

Übersicht

In einigen Anwendungsfällen, z. B. beim Fräsen von Freiformflächen, kann es vorteilhaft sein, die Lagesollwertverläufe der Maschinenachsen zu glätten. Dadurch lassen sich höhere Oberflächengüten durch Verminderung der Anregungen von mechanischen Schwingungen an der Maschine erreichen.

Zum Glätten des Lagesollwertverlaufs einer Maschinenachse kann unabhängig von den kanal- und achsspezifischen Ruckbegrenzungen, die auf Interpolatorebene berücksichtigt werden, ein Ruckfilter auf Lagereglerebene aktiviert werden.

Die Wirkung des Ruckfilters muss möglichst ausgeprägt sein, ohne dabei die Konturgenauigkeit unzulässig zu beeinträchtigen, und zusätzlich ein möglichst

"symmetrisches" Glättungsverhalten aufweisen. D. h. fährt man dieselbe Kontur vorwärts und rückwärts ab, sollte der vom Filter verrundete Konturverlauf in beiden Richtungen möglichst ähnlich sein.

Zur optimalen Anpassung des Ruckfilters an die Gegebenheiten an der Maschine stehen verschiedenen Filter-Modi zur Verfügung:

- Filter 2. Ordnung (PT2)
- Gleitende Mittelwertbildung
- Bandsperre

Modus: Filter 2. Ordnung

Der Filter-Modus "Filter 2. Ordnung" erfüllt als einfacher Tiefpass-Filter die oben genannten Anforderungen nur bei relativ kleinen Filterzeitkonstanten (ca. 10 ms). Bei größeren Zeitkonstanten ergeben sich schnell unzulässige Konturabweichungen. Die Filterwirkung ist relativ gering.

Dieser Filter-Modus bietet Vorteile wenn sehr große Filterzeitkonstanten benötigt werden und die Konturgenauigkeit eine untergeordnete Rolle spielt (z. B. Positionierachsen).

Aus historischen Gründen ist dieser Filter-Modus voreingestellt.

Modus: Gleitende Mittelwertbildung

Mit der gleitenden Mittelwertbildung als Filter-Modus können bei geringen Konturabweichungen Filterzeitkonstanten im Bereich von 20 - 40 ms eingestellt werden. Die Glättungswirkung ist weitgehend symmetrisch.

Die Anzeige des berechneten Faktors der Regelkreisverstärkung (K_V -Faktor) in der Bedienoberfläche zeigt kleinere Werte an als anhand der Filterwirkung angemessen wäre. Die Konturgenauigkeit ist höher als der angezeigte K_V -Faktor erwarten lässt.

Bei einer Umstellung von Filter-Modus "Filter 2. Ordnung" nach "Gleitende Mittelwertbildung" kann sich daher der angezeigte K_V -Faktor verkleinern (bei gleicher Filterzeitkonstanten), obwohl die Konturgenauigkeit besser wird.

Modus: Bandsperre

Bei der Bandsperre handelt es sich um einen Filter 2. Ordnung in Zähler und Nenner:

$$H(s) = \frac{\frac{s^2}{(2 \cdot \pi \cdot f_Z)^2} + \frac{2 \cdot s \cdot D_Z}{(2 \cdot \pi \cdot f_Z)}}{\frac{s^2}{(2 \cdot \pi \cdot f_N)^2} + \frac{2 \cdot s \cdot D_N}{2 \cdot \pi \cdot f_N}}$$

mit:

- f_Z : Zähler-Eigenfrequenz
- f_N : Nenner-Eigenfrequenz
- D_Z : Zähler-Dämpfung
- D_N : Nenner-Dämpfung

Da erwartet wird, dass eine schwingfähige Filtereinstellung ohnehin nicht zu brauchbaren Ergebnissen führt, steht wie beim Tiefpassfilter (PT2) Filter-Modus "Filter 2. Ordnung" (PT2) des Ruckfilters keine Einstellmöglichkeit für die Nenner-Dämpfung D_N zur Verfügung. Die Nenner-Dämpfung D_N ist fest auf den Wert 1 eingestellt.

Die Bandsperre kann auf 2 verschiedene Arten parametrisiert werden:

- Echte Bandsperre
- Bandsperre mit zusätzlicher Betragsanhebung/-absenkung bei hohen Frequenzen

Echte Bandsperre

Die echte Bandsperre ergibt sich, wenn Zähler- und Nenner-Eigenfrequenz identisch gewählt werden:

$$f_Z = f_N = f_{\text{Sperr}} \text{ (Sperrfrequenz)}$$

Wählt man die Zähler-Dämpfung = 0, ergibt sich für die Sperrfrequenz vollständige Auslöschung. Die 3dB-Bandbreite ergibt sich dann zu:

$$f_{\text{3dB-Bandbreite}} = 2 \cdot f_{\text{Sperr}}$$

Soll keine komplette Auslöschung erfolgen, sondern nur eine Absenkung um einen Faktor k , so ist die Zählerdämpfung entsprechend k zu wählen. Die obige Formel für die 3dB-Bandbreite gilt dann nicht.

Bandsperre mit zusätzlicher Betragsanhebung/-absenkung bei hohen Frequenzen

In diesem Fall werden Zähler- und Nenner-Eigenfrequenz unterschiedlich eingestellt. Die Zähler-Eigenfrequenz legt dabei die Sperrfrequenz fest.

Durch Wahl einer kleineren/größeren Nenner-Eigenfrequenz als der Zähler-Eigenfrequenz wird bei hohen Frequenzen der Amplitudengang entsprechend angehoben/abgesenkt. Eine Amplitudengang-Anhebung bei hohen Frequenzen ist in den meisten Fällen vertretbar, da die Regelstrecke i. A. selbst Tiefpasscharakter besitzt, d. h. bei hohen Frequenzen der Amplitudengang ohnehin abfällt.

Randbedingungen

Wird die Zähler-Eigenfrequenz zu groß gewählt, wird der Filter abgeschaltet. Die Grenzfrequenz $f_{Z_{\text{max}}}$ ist dabei abhängig vom Lagereglertakt:

$$f_{Zmax} = \frac{1}{2 * \pi * T_{Zmin}} = \frac{1}{2 * \pi * T_{Lageregeltakt}} \quad (\text{Shannon-Theorem})$$

6.2.18.2 Parametrierung

Aktivierung

Die Aktivierung des Ruckfilters erfolgt über das Maschinendatum:

MD32400 \$MA_AX_JERK_ENABLE (Axiale Ruckbegrenzung)

Der Ruckfilter ist in jeder Betriebs- und Interpolationsart aktiv.

Filter-Modus

Die Auswahl des Filter-Modus erfolgt über das Maschinendatum:

MD32402 \$MA_AX_JERK_MODE

Wert	Filter-Modus
1	Filter 2. Ordnung
2	Gleitende Mittelwertbildung
3	Bandsperre

Zeitkonstante

Die Zeitkonstante für den axialen Ruckfilter wird eingestellt mit dem Maschinendatum:

MD32410 \$MA_AX_JERK_TIME

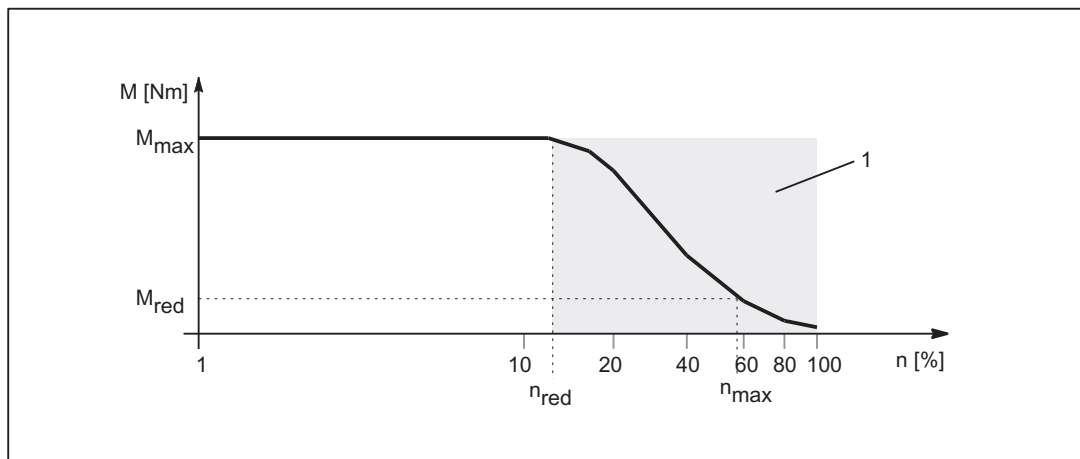
Der Ruckfilter ist nur dann wirksam, wenn die Zeitkonstante größer ist als ein Lageregeltakt.

6.2.19 Geknickte Beschleunigungskennlinie

6.2.19.1 Anpassung an die Motorkennlinie

Funktion

Verschiedene Motortypen, insbesondere Schrittmotoren, weisen einen stark drehzahlabhängigen Drehmomentenverlauf mit einem steilen Abfall des Drehmoments im oberen Drehzahlbereich auf. Zur optimalen Ausnutzung der Motorkennlinie ist es erforderlich, die Beschleunigung ab einer bestimmten Drehzahl zu reduzieren.



1: Bereich des Drehmomentabfalls

n_{red} : Drehzahl ab der mit reduziertem Drehmoment gerechnet wird

n_{\max} : Maximale Drehzahl

M_{\max} : Maximales Drehmoment

M_{red} : Drehmoment bei n_{\max} (entspricht Reduzierbeschleunigung)

Bild 6-6 Drehmomentkennlinie eines Motors mit stark drehzahlabhängigem Drehmomentverlauf

Nachbildung des Drehmomentverlaufs

Zur Nachbildung des Drehmomentverlaufs der Motorkennlinie können über folgendes Maschinendatum:

MD35242 \$MA_ACCEL_REDUCTION_TYPE = *Verlauf*

verschiedene Kennlinientypen ausgewählt werden:

0	= Konstanter Verlauf
1	= Hyperbolischer Verlauf
2	= Linearer Verlauf

Die folgenden Bilder zeigen die typischen Geschwindigkeits- und Beschleunigungskennlinien des jeweiligen Kennlinientyps:

Konstanter Verlauf

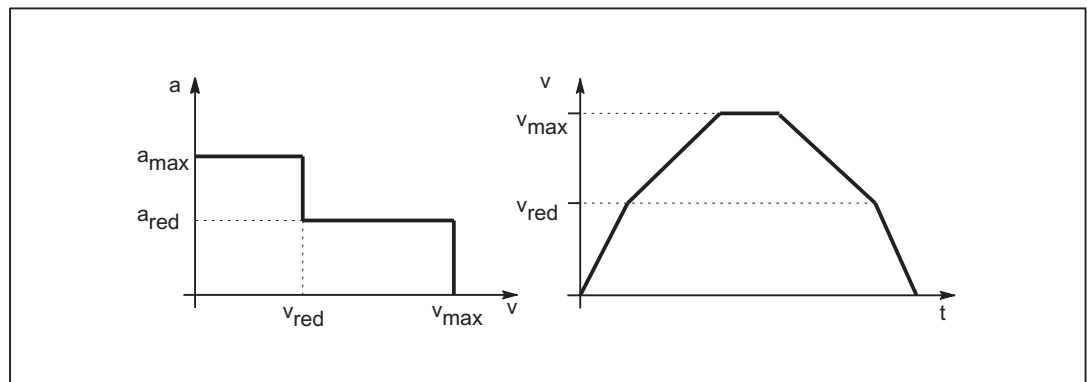


Bild 6-7 Beschleunigungs- und Geschwindigkeitsverlauf bei Beschleunigungsreduktion: 0 = konstant

Hyperbolischer Verlauf

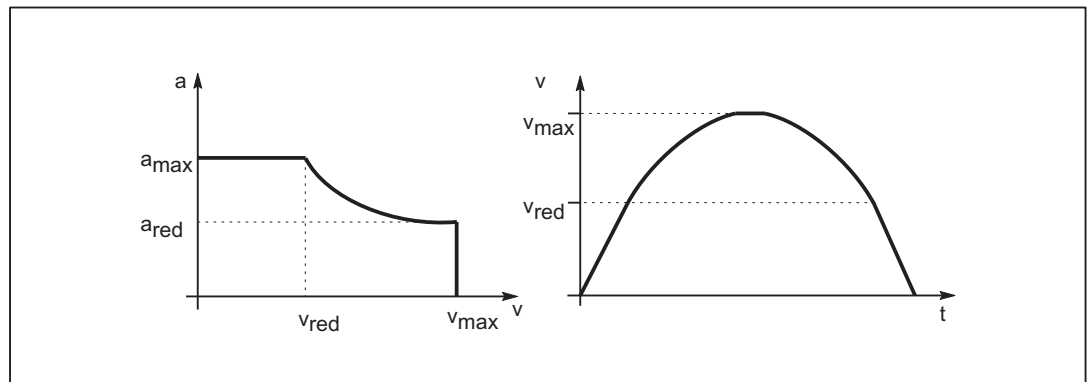


Bild 6-8 Beschleunigungs- und Geschwindigkeitsverlauf bei Beschleunigungsreduktion: 1 = hyperbolisch

Linearer Verlauf

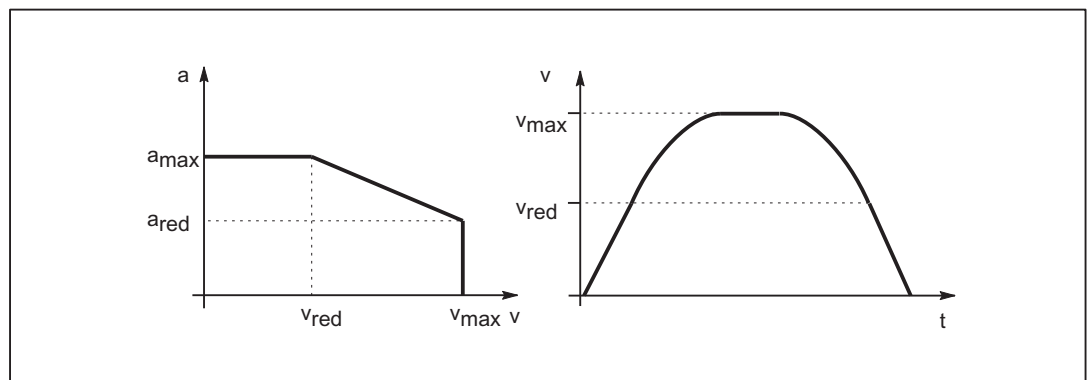


Bild 6-9 Beschleunigungs- und Geschwindigkeitsverlauf bei Beschleunigungsreduktion: 2 = linear

Die Eckdaten der Kennlinien ergeben sich zu:

$$\begin{aligned}v_{\max} &= \$MA_MAX_AX_VELO \\v_{\text{red}} &= \$MA_ACCEL_REDUCTION_SPEED_POINT * \$MA_MAX_AX_VELO \\a_{\max} &= \$MA_MAX_AX_ACCEL \\a_{\text{red}} &= (1 - \$MA_ACCEL_REDUCTION_FACTOR) * \$MA_MAX_AX_ACCEL\end{aligned}$$

6.2.19.2 Auswirkungen auf die Bahnbeschleunigung

Funktion

Die Kennlinie der Bahnbeschleunigung ergibt sich aus den Kennlinientypen der an der Bahn beteiligten Achsen. Werden Achsen mit unterschiedlichen Kennlinientypen gemeinsam interpoliert, bestimmt der restriktivste Reduktionstyp das Beschleunigungsprofil der Bahnbeschleunigung.

Es gilt folgende Reihenfolge der Prioritäten, mit 1. = Höchste Priorität:

1. Beschleunigungsreduktion: 0 = konstanter Verlauf
2. Beschleunigungsreduktion: 1 = hyperbolischer Verlauf
3. Beschleunigungsreduktion: 2 = linearer Verlauf
4. Keine Beschleunigungsreduktion wirksam
Keine wirksame Beschleunigungsreduktion ergibt sich z. B. durch:
MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT = 1
und / oder
MD35230 \$MA_ACCEL_REDUCTION_FACTOR = 0

Hinweis

Maschinenachsen mit Schrittmotor- und Gleichstromantrieb können gemeinsam interpoliert werden.

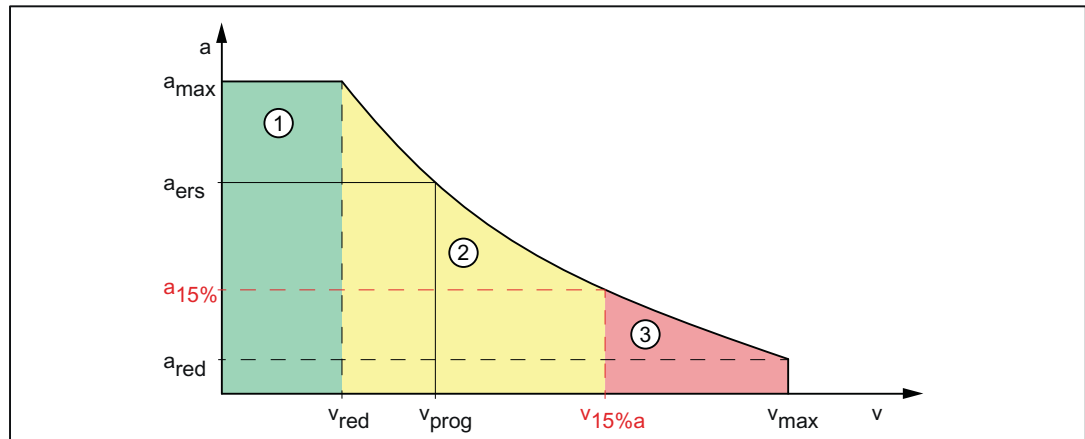
6.2.19.3 Ersatzkennlinie

Funktion

Kann die programmierte Bahn nicht mit der in den Maschinendaten der beteiligten Achsen parametrisierten Beschleunigungskennlinie gefahren werden, z. B. bei aktiver kinematischer Transformation und den Beschleunigungsarten *BRISK* oder *SOFT*, wird eine Ersatzkennlinie durch Reduzierung der dynamischen Grenzwerte erzeugt. Die reduzierten dynamischen Grenzwerte werden dabei so berechnet, dass sich als Ersatzkennlinie ein optimierter Kompromiss zwischen maximaler Geschwindigkeit und konstanter Beschleunigung ergibt.

Ersatzkennlinie bei linearen Bahnabschnitten

Wird eine Bahngeschwindigkeit programmiert, die größer der Geschwindigkeit ist bei der das Beschleunigungsvermögen noch 15 % des Maximalwertes beträgt ($v_{15\%a}$), wird auf diesen Wert begrenzt. Somit stehen in jeder Bearbeitungssituation mindestens 15 % des maximalen Beschleunigungsvermögens bzw. Drehmoments des Motors zur Verfügung.



- ① Normalbereich
- ② Reduzierbereich
- ③ Gesperrter Bereich

a_{ers} Konstante Beschleunigung der Ersatzkennlinie

$a_{15\%}$ Minimale konstante Beschleunigung

$$a_{15\%} = 0,15 \cdot (a_{max} - a_{red}) + a_{red}$$

v_{ers} Geschwindigkeit der Ersatzkennlinie

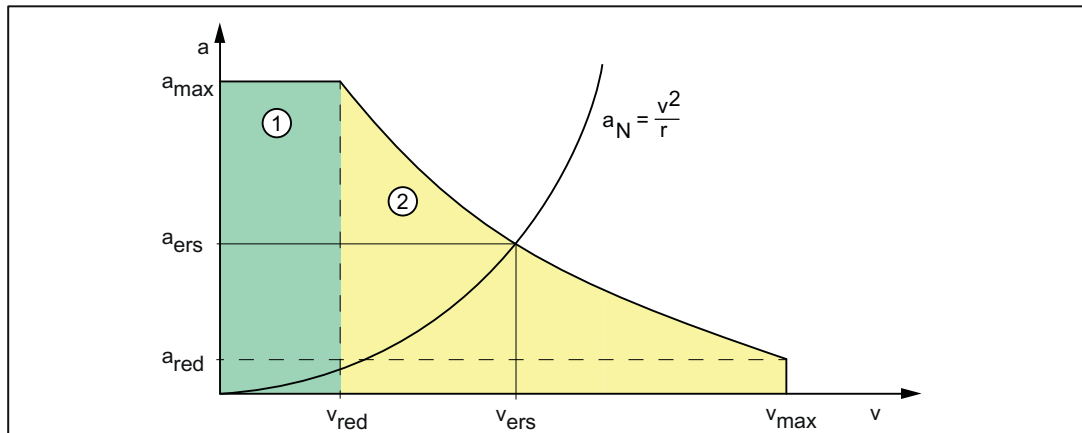
v_{prog} Programmierte Geschwindigkeit

$v_{15\%a}$ Geschwindigkeit bei $a_{15\%}$

Bild 6-10 Ersatzkennlinie der Bahn: Lineare Bahn

Ersatzkennlinie bei gekrümmten Bahnabschnitten

Bei gekrümmten Bahnabschnitten werden Normalen- und Tangentialbeschleunigung gemeinsam betrachtet. Die Bahngeschwindigkeit wird dabei soweit reduziert, dass nur maximal 25 % des geschwindigkeitsabhängigen Beschleunigungsvermögens der Achsen für die Normalenbeschleunigung benötigt wird. Die verbleibenden 75 % des Beschleunigungsvermögens werden für die Tangentialbeschleunigung, also dem Bremsen bzw. Beschleunigen auf der Bahn, reserviert.



- ① Normalbereich
- ② Reduzierbereich
- a_N Normalenbeschleunigung
- a_{ers} Konstante Beschleunigung der Ersatzkennlinie
- v_{ers} Geschwindigkeit der Ersatzkennlinie
- r Bahnradius

Bild 6-11 Ersatzkennlinie der Bahn: Gekrümmte Bahn

Satzübergänge bei Bahnsteuerbetrieb

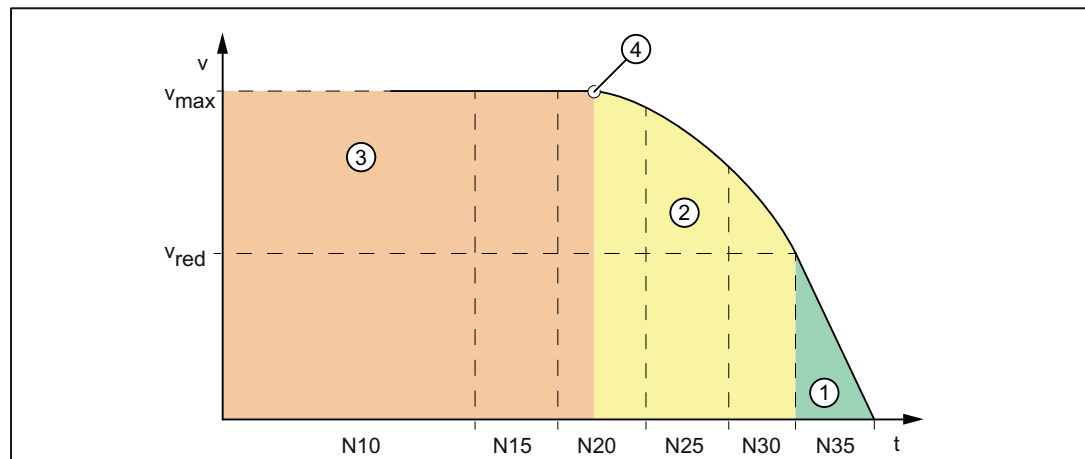
Bei aktivem Bahnsteuerbetrieb kommt es an nichttangentialen Satzübergängen beim Verfahren mit der programmierten Bahngeschwindigkeit zu axialen Geschwindigkeitssprüngen.

Die Bahngeschwindigkeit wird dann derart geführt, dass beim Satzübergang kein axialer Geschwindigkeitsanteil größer der Reduziergeschwindigkeit v_{red} ist.

Bremsrampe bei Bahnsteuerbetrieb und LookAhead

Bei aufeinander folgenden Verfahrenssätzen mit kurzen Bahnwegen kann sich ein Beschleunigungs- oder Bremsvorgang über mehrere Teileprogrammsätzen erstrecken.

Die Funktion "LookAhead" berücksichtigt dabei auch den parametrisierten geschwindigkeitsabhängigen Verlauf der Beschleunigung.



- ① Normalbereich $\Rightarrow a = a_{\max}$
- ② Reduzierbereich $\Rightarrow a < a_{\max}$
- ③ Konstantfahrbereich $\Rightarrow a = 0 \text{ m/s}^2$
- ④ Bremsseinsatzpunkt
- v_{red} Reduziergeschwindigkeit
- v_{max} Maximale Geschwindigkeit
- N_x Verfahrenssatz mit Satznummer N_x

Bild 6-12 Bremsvorgang mit LookAhead

6.2.19.4 Parametrierung

Die Aktivierung der geknickten Beschleunigungskennlinie erfolgt Maschinenachs-spezifisch über das Maschinendatum:

MD35240 \$MA_ACCEL_TYPE_DRIVE = TRUE

Die Parametrierung der geknickten Beschleunigungskennlinie erfolgt achsspezifisch über die Maschinendaten:

MD32000 \$MA_MAX_AX_VELO (Maximale Achsgeschwindigkeit)

MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT
(Drehzahl für reduzierte Beschleunigung)

MD35230 \$MA_ACCEL_REDUCTION_FACTOR (Reduzierte Beschleunigung)

MD32300 \$MA_MAX_AX_ACCEL (Maximale Achsbeschleunigung)

MD35242 \$MA_ACCEL_REDUCTION_TYPE
(Art der Beschleunigungsreduktion: 0=konstant, 1=hyperbolisch, 2=linear)

6.2.19.5 Programmierung

Kanalspezifisches Einschalten (DRIVE)

Syntax

DRIVE

Funktionalität

Über die Teileprogrammanweisung `DRIVE` wird die geknickte Beschleunigungskennlinie für die Bahnbeschleunigung aktiviert

G-Gruppe: 21

Wirksamkeit: modal

Reset-Verhalten

Durch Reset wird die kanalspezifisch parametrierte Grundstellung wirksam:

MD20150 \$MC_GCODE_RESET_VALUES[20]

Abhängigkeiten

Ist für eine Maschinenachsen die geknickten Beschleunigungskennlinie parametriert, wird sie standardmäßig nach diesem Beschleunigungsprofil verfahren.

Wird für einen bestimmten Bahnabschnitt durch Teileprogrammanweisung `SOFT` oder `BRISK` das wirksame Beschleunigungsprofil umgeschaltet, wird statt der geknickten Beschleunigungskennlinie eine entsprechende Ersatzkennlinie mit reduzierten dynamischen Grenzwerten verwendet.

Mit erneuter Programmierung von `DRIVE` kann die geknickte Beschleunigungskennlinie wieder aktiviert werden.

Achsspezifisches Einschalten (DRIVEA)

Syntax

DRIVEA (*Achse* {*Achse*})

Funktionalität

Über die Teileprogrammanweisung wird die geknickte Beschleunigungskennlinie achsspezifisch für alle Einzelachs-Interpolationen (Positionierachse, Pendelachse, etc.) eingeschaltet.

G-Gruppe: -

Wirksamkeit: modal

Achse:

- Wertebereich: Achsname der Kanalachsen

Reset-Verhalten

Durch Reset wird die kanalspezifisch parametrisierte Grundstellung wirksam:

MD20150 \$MC_GCODE_RESET_VALUES[20]

Abhängigkeiten

Ist für eine Maschinenachse die geknickte Beschleunigungskennlinie parametrisiert, wird sie standardmäßig nach diesem Beschleunigungsprofil verfahren.

Wird durch die Teileprogrammanweisung `SOFTA` oder `BRISKA` achsspezifisch das wirksame Beschleunigungsprofil umgeschaltet, wird statt der geknickten Beschleunigungskennlinie eine entsprechende Ersatzkennlinie verwendet.

Mit Programmierung von `DRIVEA` kann achsspezifisch wieder auf die geknickte Beschleunigungskennlinie umgeschaltet werden.

6.2.19.6 Randbedingungen

Einzelachs-Interpolation

Ab Aktivierung der geknickten Beschleunigungskennlinie wird bei Einzelachs-Interpolationen (Positionierachse, Pendelachse, manuelles Verfahren, etc.) die Maschinenachse ausschließlich im Modus `DRIVEA` verfahren.

Ein Umschalten des Beschleunigungsprofils über folgende Teileprogrammanweisungen ist nicht möglich:

- Sprungförmige Beschleunigung (`BRISKA`)
- Ruckbegrenzte Beschleunigung (`SOFTA`)

Bahn-Interpolation

Ist für eine an einer programmierten Bahn beteiligten Maschinenachse die geknickte Beschleunigungskennlinie parametrisiert ohne dass die Teileprogrammanweisung `DRIVE` aktiv ist, wird für die Bahn eine Ersatzkennlinie mit reduzierten dynamischen Grenzwerten bestimmt.

Kinematische Transformation

Im Zusammenhang mit einer aktiven kinematischen Transformation wird die geknickte Beschleunigungskennlinie nicht berücksichtigt. Steuerungsintern wird auf Beschleunigung ohne Ruckbegrenzung (`BRISK`) umgeschaltet und für die Bahnbeschleunigung eine Ersatzkennlinie wirksam.

6.2.20 Beschleunigung und Ruck bei JOG-Bewegungen

Um im JOG-Betrieb störende Maschinenruckbewegungen zu vermeiden, können für JOG-Bewegungen eigene achsspezifische Beschleunigungs- und Ruckbegrenzungswerte vorgegeben werden.

Darüber hinaus ist es möglich, Beschleunigung und Ruck für das manuelle Verfahren von Geometrie- und Orientierungsachsen kanalspezifisch zu begrenzen. Dies ermöglicht eine bessere Handhabbarkeit für Kinematiken, die kartesische Bewegungen vollständig über Rundachsen erzeugen (Roboter).

6.2.20.1 Parametrierung

Achsspezifische Begrenzung von Beschleunigung und Ruck

Maximale achsspezifische Beschleunigung bei JOG-Bewegungen

Die maximale achsspezifische Beschleunigung bei JOG-Bewegungen kann für jede Maschinenachse vorgegeben werden über das Maschinendatum:

MD32301 \$MA_JOG_MAX_ACCEL

Bei MD32301 = 0 ist statt des JOG-spezifischen Maximalwerts der Wert aus MD32300 \$MA_MAX_AX_ACCEL wirksam.

Maximaler achsspezifischer Ruck bei JOG-Bewegungen

Der maximale achsspezifische Ruck bei JOG-Bewegungen kann für jede Maschinenachse vorgegeben werden über das Maschinendatum:

MD32436 \$MA_JOG_MAX_JERK

Bei MD32436 = 0 ist statt des JOG-spezifischen Maximalwerts der Wert aus MD32430 \$MA_JOG_AND_POS_MAX_JERK wirksam.

Hinweis

MD32436 \$MA_JOG_MAX_JERK ist nur wirksam, wenn für die zu verfahrenen Maschinenachsen die achsspezifische Ruckbegrenzung im JOG-Betrieb freigegeben wurde:

MD32420 \$MA_JOG_AND_POS_JERK_ENABLE [<Achse>] == TRUE

Dies ist auch über die Programmierung von SOFTA (<Achse1>, <Achse2>, . . .) im Teileprogramm möglich.

Kanalspezifische Begrenzung von Beschleunigung und Ruck

Maximale Beschleunigung beim manuellen Verfahren von Geometrieachsen

Die maximale Beschleunigung beim manuellen Verfahren von Geometrieachsen kann kanalspezifisch vorgegeben werden über das Maschinendatum:

MD21166 \$MC_JOG_ACCEL_GEO [<Geometrieachse>]

mit <Geometrieachse> = 0, 1, 2

Bei MD21166 = 0 ist statt der kanalspezifischen Beschleunigungsbegrenzung der achsspezifische Grenzwert aus MD32301 \$MA_JOG_MAX_ACCEL wirksam.

Hinweis

Für MD21166 \$MC_JOG_ACCEL_GEO [<Geometrieachse>] gibt es keine direkte Begrenzung auf MD32300 \$MA_MAX_AX_ACCEL.

Hinweis

Bei einer aktiven Transformation bestimmt MD32300 \$MA_MAX_AX_ACCEL die maximal mögliche achsspezifische Beschleunigung.

Maximaler Ruck beim manuellen Verfahren von Geometrieachsen

Der maximale Ruck beim manuellen Verfahren von Geometrieachsen im Beschleunigungsmodus SOFT (Beschleunigung mit Ruckbegrenzung) kann kanalspezifisch vorgegeben werden über das Maschinendatum:

MD21168 \$MC_JOG_JERK_GEO [<Geometrieachse>]

mit <Geometrieachse> = 0, 1, 2

Bei MD21168 = 0 ist statt der kanalspezifischen Ruckbegrenzung der achsspezifische Grenzwert aus MD32436 \$MA_JOG_MAX_JERK wirksam.

Hinweis

MD21168 \$MC_JOG_JERK_GEO ist nur wirksam, wenn für die zugrunde liegenden Maschinenachsen die achsspezifische Ruckbegrenzung im JOG-Betrieb freigegeben wurde:

MD32420 \$MA_JOG_AND_POS_JERK_ENABLE [<Achse>] == TRUE

Maximaler Ruck beim manuellen Verfahren von Orientierungsachsen

Der maximale Ruck beim manuellen Verfahren von Orientierungsachsen kann kanalspezifisch vorgegeben werden über das Maschinendatum:

MD21158 \$MC_JOG_JERK_ORI [<Orientierungsachse>]

Damit MD21158 wirksam werden kann, muss die kanalspezifische Ruckbegrenzung für das manuelle Verfahren von Orientierungsachsen über das folgende Maschinendatum freigegeben werden:

MD21159 \$MC_JOG_JERK_ORI_ENABLE == TRUE

Hinweis

Orientierungsachsen sind von den Maschinendaten MD32301 \$MA_JOG_MAX_ACCEL und MD32436 \$MA_JOG_MAX_JERK nicht betroffen.

6.2.20.2 Randbedingungen

Wegüberlagerung / Überlagerte Bewegungen

Bei Wegüberlagerung / überlagerten Bewegungen (z. B. DRF) sind die JOG-spezifischen Maximalwerte für Beschleunigung und Ruck (MD32301 \$MA_JOG_MAX_ACCEL und MD32436 \$MA_JOG_MAX_JERK) **nicht** wirksam. Es wirken stattdessen die Werte für Positionierachsbewegungen:

- Beschleunigung:
 - MD32300 \$MA_MAX_AX_ACCEL [0] (maximale achsspezifische Beschleunigung bei Bahnbewegungen im Dynamikmodus DYNNORM)
Siehe auch Kapitel "Beschleunigung ohne Ruckbegrenzung (BRISK/BRISKA) (kanal-/achsspezifisch) (Seite 267)".
- Ruck
 - MD32430 \$MA_JOG_AND_POS_MAX_JERK (maximaler achsspezifischer Ruck bei Positionierachsbewegungen)
bzw. (bei G75):
 - MD32431 \$MA_MAX_AX_JERK [0] (maximaler achsspezifischer Ruck bei Bahnbewegungen im Dynamikmodus DYNNORM)

Siehe auch Kapitel "Ruckbegrenzung bei Einzelachsinterpolation (SOFTA) (achsspezifisch) (Seite 284)".

Hinweis

Für den JOG-Betrieb ist immer nur der Dynamikmodus DYNNORM wirksam.

Verhalten beim manuellen Verfahren von Geometrieachsen bei aktiver Rotation

Beim manuellen Verfahren von Geometrieachsen im Beschleunigungsmodus SOFT (Beschleunigung mit Ruckbegrenzung) wird der Wert aus MD32436 \$MA_JOG_MAX_JERK bzw. MD32430 \$MA_JOG_AND_POS_MAX_JERK auch bei aktiver Rotation oder aktivem orientierbaren Werkzeugträger verwendet.

Teileprogrammanweisung SOFTA / BRISKA / DRIVEA

Die Teileprogrammanweisung SOFTA (<Achse1>, <Achse2>, ...) wirkt auch im JOG-Betrieb, d. h. für die angegebenen Achsen ist beim Verfahren im JOG-Betrieb der maximale achsspezifische Ruck aus MD32436 \$MA_JOG_MAX_JERK wirksam (genau wie beim Setzen von MD32420 \$MA_JOG_AND_POS_JERK_ENABLE [<Achse>] == TRUE).

Hinweis

Die Teileprogrammanweisung SOFT hat dagegen keine Auswirkung auf den JOG-Betrieb.

Wie SOFTA sind auch die Teileprogrammanweisungen BRISKA und DRIVEA im JOG-Betrieb wirksam, d. h. die Beschleunigung erfolgt ohne Ruckbegrenzung, auch wenn für die

betroffenen Maschinenachsen MD32420 \$MA_JOG_AND_POS_JERK_ENABLE auf "TRUE" gesetzt ist.

Hinweis

Das manuelle Verfahren von Orientierungsachsen wird von BRISKA/SOFTA/DRIVEA nicht beeinflusst.

6.3 Beispiele

6.3.1 Beschleunigung

6.3.1.1 Bahngeschwindigkeitsverlauf

Das folgende Beispiel zeigt anhand der programmierten Verfahrbewegungen und Aktionen eines Teileprogrammausschnitts den Verlauf der Bahngeschwindigkeit.

Teileprogrammausschnitt

Programmcode

```
; Synchronaktion: Beschleunigungsumschaltung abhängig von schnellem Eingang
1 ($A_IN[1]):
N53 ID=1 WHEN $A_IN[1] == 1 DO $AC_PATHACC = 2.*$MA_MAX_AX_ACCEL[X]
; Synchronaktion: Test-Override-Profil (Simulation von externen
Eingriffen):
N54 ID=2 WHENEVER ($AC_TIMEC > 16) DO $AC_OVR=10
N55 ID=3 WHENEVER ($AC_TIMEC > 30) DO $AC_OVR=100
; Anfahren
N1000 GO X0 Y0 BRISK
N1100 TRANS Y=-50
N1200 AROT Z=30 G642
; Kontur
N2100 X0 Y0
N2200 X = 70 G1 F10000 RNDM=10 ACC[X]=30 ACC[Y]=30
N2300 Y = 70
N2400 X0
N2500 Y0
```

- 1: Beschleunigen auf 100% Bahngeschwindigkeit (F10000) gemäß Beschleunigungsvorgabe: ACC (N2200...)
- 2: Bremsen auf 10% Bahngeschwindigkeit aufgrund der Override-Änderung (\$AC_OVR) gemäß Echtzeitbeschleunigung \$AC_PATHACC (N53/N54...)
- 3: Beschleunigen auf 100% Bahngeschwindigkeit aufgrund der Override-Änderung (\$AC_OVR) gemäß Echtzeitbeschleunigung \$AC_PATHACC (N53/N55...)
- 4: Bremsen auf Satzendgeschwindigkeit für den Verrundungs-Zwischensatz gemäß Beschleunigungsvorgabe: ACC (N2200...)
- 5: Geschwindigkeitsbegrenzung aufgrund der Verrundungen (siehe 9)
- 6: Beschleunigen auf 100% Bahngeschwindigkeit (\$AC_OVR) gemäß Beschleunigungsvorgabe: ACC (N2300...)
- 7: Abbremsen aufgrund der Override-Änderung mit Beschleunigung gemäß Echtzeitbeschleunigung \$AC_PATHACC (N53/N54...)
- 8: Beschleunigen auf 100% Bahngeschwindigkeit aufgrund der Override-Änderung (\$AC_OVR) gemäß Echtzeitbeschleunigung \$AC_PATHACC (N53/N55...)
- 9: Steuerungsintern eingefügter Zwischensatz aufgrund der programmierten Verrundung (RNDM) (N2200...)

Bild 6-13 Umschalten zwischen im Vorlauf bestimmter Bahnbeschleunigung und Echtzeitbeschleunigung

6.3.2 Ruck

6.3.2.1 Bahngeschwindigkeitsverlauf

Das folgende Beispiel zeigt anhand der programmierten Verfahrbewegungen und Aktionen eines Teileprogrammausschnitts den Verlauf der Bahngeschwindigkeit und des Rucks.

Teileprogramm Ausschnitt

```
; Einstellen von Bahnbeschleunigung und Bahnruck bei externem Eingriff:
N0100 $AC_PATHACC = 0.0
N0200 $AC_PATHJERK = 4.0 * ($MA MAX AX JERK[X] + $MA MAX AX JERK[Y]) / 2.0
```


Programmcode

; Synchronaktion: Variieren des Overrides (Simulation von externen Eingriffen)

N0530 ID=1 WHENEVER (\$AC_TIMEC > 16) DO \$AC_OVR=10

N0540 ID=2 WHENEVER (\$AC_TIMEC > 30) DO \$AC_OVR=100

; Anfahren

N1000 G0 X0 Y0 SOFT

N1100 TRANS Y=-50

N1200 AROT Z=30 G642

; Kontur

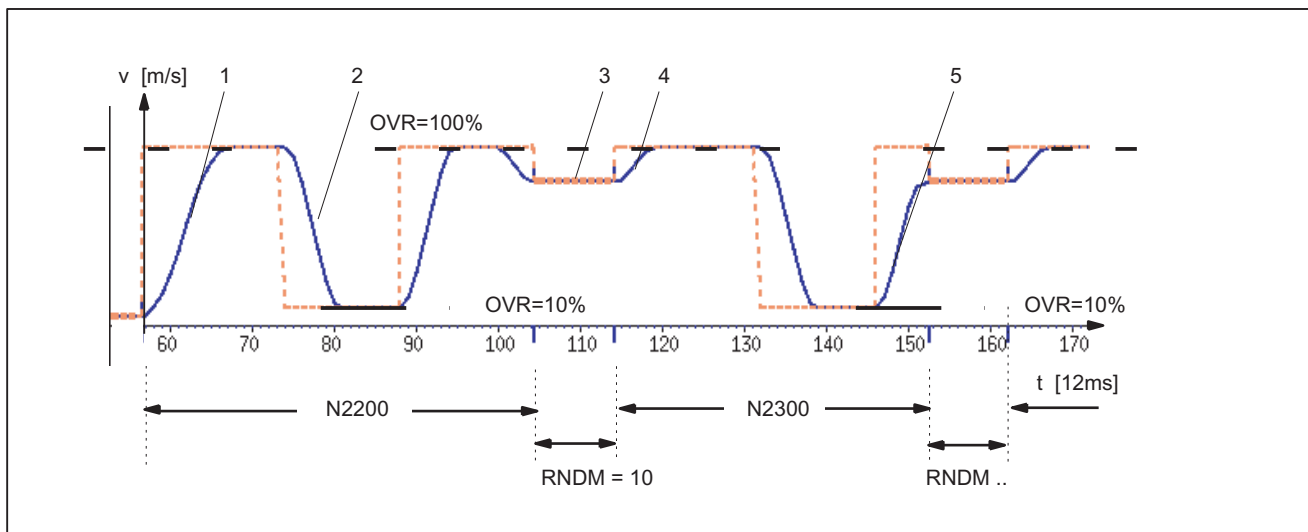
N2100 X0 Y0

N2200 X = 70 G1 F10000 RNDM=10

N2300 Y = 70

N2400 X0

N2500 Y0

Verlauf von Bahngeschwindigkeit und Ruck**Beschleunigungsprofil: SOFT**

- 1: Ruck gemäß \$MA_MAX_AX_JERK[.]
- 2: Ruck gemäß \$AC_PATHJERK
- 3: Ruck gemäß \$MA_MAX_AX_JERK[.] (Anfahren der Satzengeschwindigkeit)
- 4: Geschwindigkeitsbegrenzung durch Kreisbogen
- 5: Ruck gemäß \$AC_PATHJERK

Bild 6-14 Umschalten zwischen im Vorlauf bestimmtem Bahnrick und \$AC_PATHJERK

6.3.3 Beschleunigung und Ruck

Das folgende Beispiel zeigt den Verlauf von Geschwindigkeit und Beschleunigung der X-Achse aufgrund der programmierten Verfahrbewegungen des Teileprogrammausschnittes. Außerdem welche der Geschwindigkeits- und Beschleunigungs-relevanten Maschinendaten für welchen Abschnitt der Kontur maßgeblich sind.

Teileprogrammausschnitt

Programmcode	Kommentar
N90 F5000 SOFT G64	; Bahnsteuerbetrieb, rückbegrenzte Beschleunigung
N100 G0 X0 Y0 Z0	; Eilgang
N110 G1 X10	; Gerade
N120 G3 CR=5 X15 Y5	; Kreisbogen, Radius 5 mm, Satzübergang: tangential
N130 G3 CR=10 X5 Y15	; Kreisbogen, Radius 10 mm, Satzübergang: tangential
N140 G1 X-5 Y17.679	; Gerade, 15° Knick

Kontur

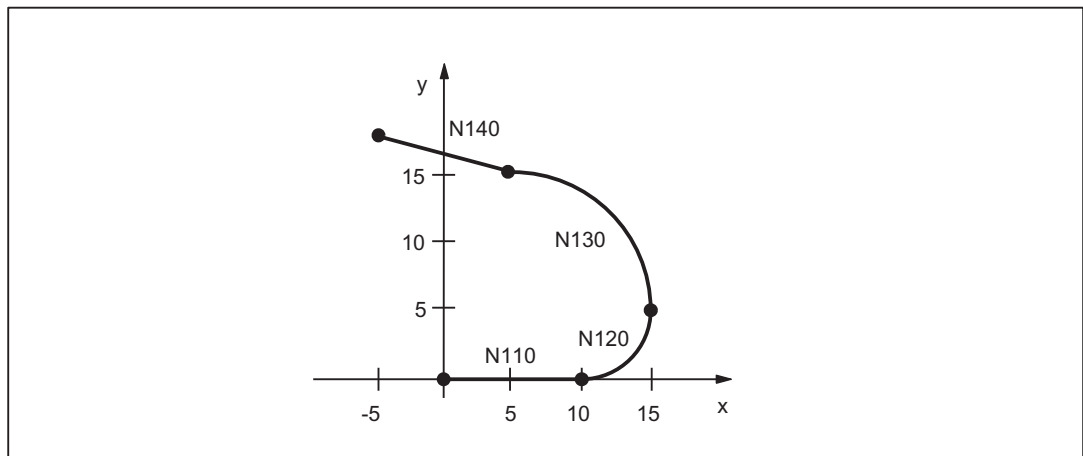


Bild 6-15 Kontur des Teileprogrammausschnitts

Geschwindigkeits- und Beschleunigungsverlauf

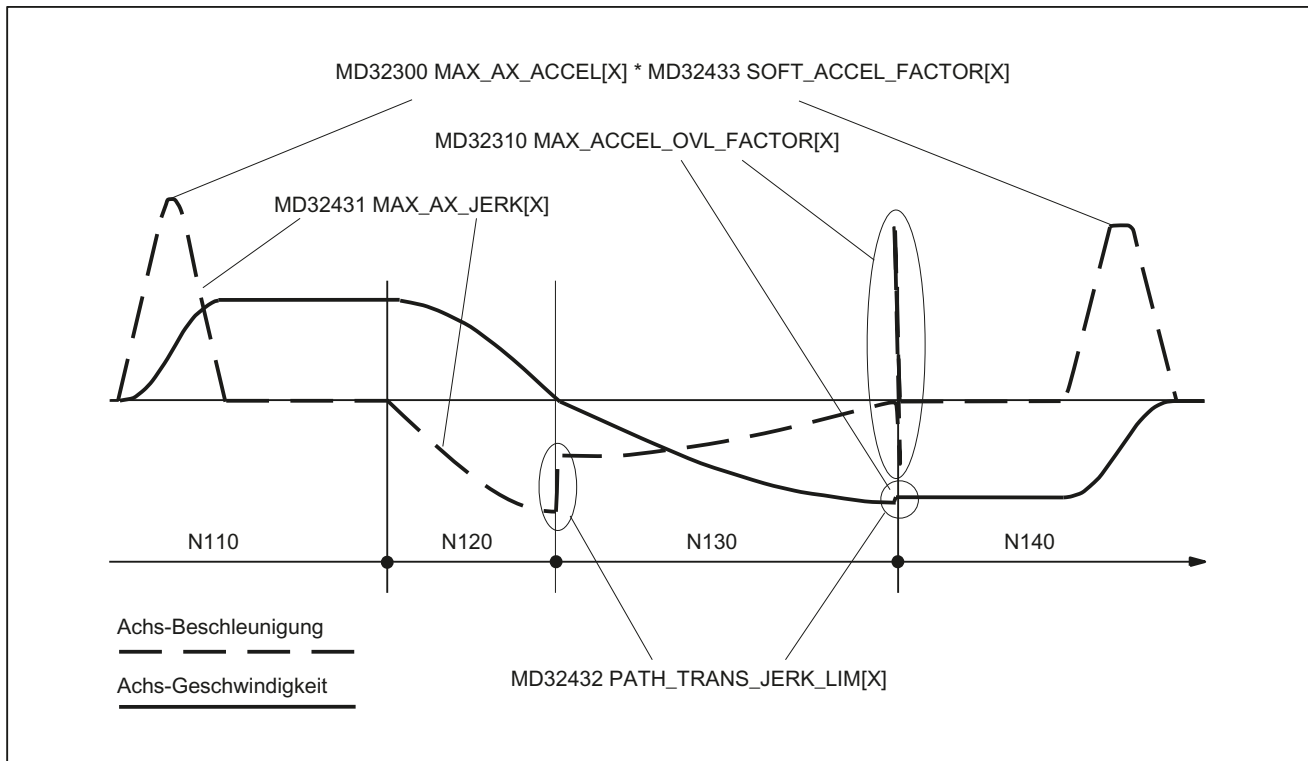


Bild 6-16 Verlauf von Geschwindigkeits- und Beschleunigungsverlauf: X-Achse

6.3.4 Geknickte Beschleunigungskennlinie

6.3.4.1 Aktivierung

Das Beispiel zeigt die Aktivierung der geknickten Beschleunigungskennlinie anhand der Maschinendaten und eines Teileprogrammausschnitts.

Maschinendaten

Maschinendatum	Wert
MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT[X]	= 0.4
MD35230 \$MA_ACCEL_REDUCTION_FACTOR[X]	= 0.85
MD35242 \$MA_ACCEL_REDUCTION_TYPE[X]	= 2
MD35240 \$MA_ACCEL_TYPE_DRIVE[X]	= TRUE
MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT[Y]	= 0.0
MD35230 \$MA_ACCEL_REDUCTION_FACTOR[Y]	= 0.6
MD35242 \$MA_ACCEL_REDUCTION_TYPE[Y]	= 1
MD35240 \$MA_ACCEL_TYPE_DRIVE[Y]	= TRUE

Maschinendatum	Wert
MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT[Z]	= 0.6
MD35230 \$MA_ACCEL_REDUCTION_FACTOR[Z]	= 0.4
MD35242 \$MA_ACCEL_REDUCTION_TYPE[Z]	= 0
MD35240 \$MA_ACCEL_TYPE_DRIVE[Z]	= FALSE

Aktivierung durch Vorgabe als kanalspezifische Grundstellung:

MC_GCODE_RESET_VALUE[20] = 3 (DRIVE)

Teileprogrammausschnitt

Programmcode	Kommentar
N10 G1 X100 Y50 Z50 F700	; Bahnbewegung (X,Y, Z) mit DRIVE
N15 Z20	; Bahnbewegung (Z) mit DRIVE
N20 BRISK	; Umschaltung auf BRISK
N25 G1 X120 Y70	; Bahnbewegung (Y, Z) mit Ersatzkennlinie
N30 Z100	; Bahnbewegung (Z) mit BRISK
N35 POS[X] = 200 FA[X] = 500	; Positionierbewegung (X) mit DRIVEA
N40 BRISKA(Z)	; Aktivieren BRISKA für Z
N40 POS[Z] = 50 FA[Z] = 200	; Positionierbewegung (Z) mit BRISKA
N45 DRIVEA(Z)	; Aktivieren DRIVEA für Z
N50 POS[Z] = 100	; Positionierbewegung (Z) mit DRIVE
N55 BRISKA(X)	; führt zu Fehlermeldung

6.4 Datenlisten

6.4.1 Maschinendaten

6.4.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
18960	POS_DYN_MODE	Art der Positionierachsdynamik

6.4.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20150	GCODE_RESET_VALUES	Löschstellung der G-Gruppen
20500	CONST_VELO_MIN_TIME	Minimale Zeit mit konstanter Geschwindigkeit

Nummer	Bezeichner: \$MC_	Beschreibung
20600	MAX_PATH_JERK	Bahnbezogener Maximalruck
20602	CURV_EFFECT_ON_PATH_ACCEL	Einfluss der Bahnkrümmung auf Bahndynamik
20610	ADD_MOVE_ACCEL_RESERVE	Beschleunigungsreserve für überlagerte Bewegungen
21158	JOG_JERK_ORI	Maximaler Ruck der Orientierungsachsen beim Verfahren in JOG
21159	JOG_JERK_ORI_ENABLE	Grundstellung der kanalspezifischen Ruckbegrenzung von Orientierungsachsen beim Verfahren in JOG
21166	JOG_ACCEL_GEO	Maximale Beschleunigung der Geometrieachsen beim Verfahren in JOG
21168	JOG_JERK_GEO	Maximaler Ruck der Geometrieachsen beim Verfahren in JOG

6.4.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
32000	MAX_AX_VELO	Maximale Achsgeschwindigkeit
32300	MAX_AX_ACCEL	Maximale Achsbeschleunigung
32301	JOG_MAX_ACCEL	Maximale Beschleunigung im JOG-Betrieb
32310	MAX_ACCEL_OVL_FACTOR	Überlastfaktor für Geschwindigkeitssprung
32320	DYN_LIMIT_RESET_MASK	Resetverhalten von Dynamikbegrenzungen
32400	AX_JERK_ENABLE	Axiale Ruckbegrenzung
32402	AX_JERK_MODE	Filtertyp für axiale Ruckbegrenzung
32410	AX_JERK_TIME	Zeitkonstante für den axialen Ruckfilter
32420	JOG_AND_POS_JERK_ENABLE	Grundstellung der axialen Ruckbegrenzung
32430	JOG_AND_POS_MAX_JERK	Axialer Ruck bei Einzelachs-bewegungen
32431	MAX_AX_JERK	Maximaler axialer Ruck am Satzübergang im Bahnsteuerbetrieb
32432	PATH_TRANS_JERK_LIM	Max. axialer Ruck einer Geo-Achse an Satzgrenze
32433	SOFT_ACCEL_FACTOR	Skalierung der Beschleunigungsbegrenzung bei SOFT
32434	G00_ACCEL_FACTOR	Skalierung der Beschleunigungsbegrenzung bei G00
32435	G00_JERK_FACTOR	Skalierung der axialen Ruckbegrenzung bei G00
32436	JOG_MAX_JERK	Maximaler axialer Ruck bei JOG-Bewegung
32437	AX_JERK_VELO	Erste Geschwindigkeitsschwelle für die geschwindigkeitsabhängige Ruckadaption
32438	AX_JERK_VEL1	Zweite Geschwindigkeitsschwelle für die geschwindigkeitsabhängige Ruckadaption
32439	MAX_AX_JERK_FACTOR	Faktor zur Einstellung des maximalen Rucks bei höheren Geschwindigkeiten (geschwindigkeitsabhängige Ruckadaption)
35220	ACCEL_REDUCTION_SPEED_POINT	Drehzahl für reduzierte Beschleunigung
35230	ACCEL_REDUCTION_FACTOR	Reduzierte Beschleunigung

Nummer	Bezeichner: \$MA_	Beschreibung
35240	ACCEL_TYPE_DRIVE	Beschleunigungskennlinie DRIVE für Achsen Ein/Aus
35242	ACCEL_REDUCTION_TYPE	Art der Beschleunigungsreduktion

6.4.2 Settingdaten

6.4.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42500	SD_MAX_PATH_ACCEL	Max. Bahnbeschleunigung
42502	IS_SD_MAX_PATH_ACCEL	Auswerten von SD 42500: Ein/Aus
42510	SD_MAX_PATH_JERK	Max. bahnbezogener Ruck
42512	IS_SD_MAX_PATH_JERK	Auswerten von SD 42510: Ein/Aus

6.4.3 Systemvariablen

Bezeichner	Beschreibung
\$AC_PATHACC	Bahnbeschleunigung für Echtzeitereignisse
\$AC_PATHJERK	Bahnruck für Echtzeitereignisse

F1: Fahren auf Festanschlag

7.1 Kurzbeschreibung

Funktion

Mit der Funktion "Fahren auf Festanschlag" können bewegliche Maschinenteile, z. B. Reitstock oder Pinole, so verfahren werden, dass mit ihnen über einen beliebigen Zeitraum ein definierbares Moment bzw. Kraft auf andere Maschinenteile ausgeübt werden kann.

Merkmale

- Programmierbarkeit über Befehle im Teileprogramm oder Synchronaktion:
 - Ein- und Ausschalten der Funktion "Fahren auf Festanschlag"
 - Einstellen des Klemmmoments
 - Einstellen des Überwachungsfensters
- Manuelle Eingriffsmöglichkeit über Settingdaten:
 - Ein- und Ausschalten der Funktion "Fahren auf Festanschlag"
 - Einstellen des Klemmmoments
 - Einstellen des Überwachungsfensters
- Voreinstellungen über Maschinendatum
 - Klemmmoment
 - Überwachungsfenster
- Erkennen des Funktionsstatus über NC/PLC-Nahtstellensignale
- Freigabe- bzw. Quittierungsmöglichkeiten über NC/PLC-Nahtstellensignale
- Lesen des Soll- und Iststatus der Funktion über Systemvariable
- "Fahren auf Festanschlag" kann mit Achsen und Spindeln durchgeführt werden.
- Gleichzeitiges "Fahren auf Festanschlag" von mehreren Achsen und parallel zum Verfahren anderer Achsen möglich.
- Mehrkanaliger Satzsuchlauf mit Berechnung aller erforderlichen Zusatzdaten möglich (SERUPRO).
- Simuliertes Verfahren von Achsen im Zusammenhang mit "Fahren auf Festanschlag" und "Momentenreduzierung" möglich.
- "Hängende" Achsen können bei FXS-Alarmen auf Festanschlag gefahren werden.

7.2 Ausführliche Beschreibung

7.2.1 Programmierung

Funktion

Fahren auf Festanschlag

Über die Befehle **FXS**, **FXST** und **FXSW** wird die Funktion "Fahren auf Festanschlag" gesteuert.

Die Aktivierung kann auch ohne Verfahrbewegung der betreffenden Achse erfolgen. Das Moment wird sofort begrenzt. Auf Anschlag wird überwacht, sobald die Achse verfahren wird.

Hinweis

Synchronaktionen

Die Funktion "Fahren auf Festanschlag" kann auch über Synchronaktionen gesteuert werden.

Literatur:

Funktionshandbuch Synchronaktionen

Fahren mit begrenztem Moment / Kraft

Über die Befehle **FOCON**, **FOCOF** und **FOC** wird das Verfahren mit begrenztem Moment / Kraft gesteuert (siehe Kapitel "Fahren mit begrenztem Moment/Kraft FOC (Seite 333)").

Syntax

```
FXS [<Achse>]=<Anforderung>
FXST [<Achse>]=<Klemmmoment>
FXSW [<Achse>]=<Fensterbreite>
```

Bedeutung

Parameter	Bedeutung
FXS:	Funktion "Fahren auf Festanschlag", Wirksamkeit: modal
<Anforderung>:	0 = Ausschalten 1 = Einschalten
FXST:	Klemmmoment einstellen
<Klemmmoment>:	Klemmmoment in % des maximalen Moments des Antriebs. SINAMICS S120: p2003
FXSW:	Überwachungsfensters einstellen
<Fensterbreite>:	Breite des Überwachungsfensters um den Festanschlag Einheit: mm, inch oder Grad
<Achse>:	Name der Kanalachse, Typ: AXIS

Änderungen der Momentenbegrenzung (FXST)

Eine Änderung des Momentengrenzwertes kann in jedem Satz vorgenommen werden. Die Änderung wird vor dem Ausführen von im Satz programmierten Verfahrbewegungen wirksam. Die Momentenbegrenzung wirkt zusätzlich zur Beschleunigungsbegrenzung (ACC).

Rampenförmige Änderung

Über das Maschinendatum kann eine Zeit eingestellt werden, innerhalb der eine Änderung des Momentengrenzwertes linear durchgeführt wird:

MD37012 \$MA_FIXED_STOP_TORQUE_RAMP_TIME (Zeitdauer bis zum Erreichen des neuen Momentengrenzwertes)

Änderungen des Überwachungsfensters (FXSW)

Eine Änderung des Überwachungsfensters kann in jedem Satz vorgenommen werden. Die Änderung wird vor dem Ausführen von im Satz programmierten Verfahrbewegungen wirksam.

Mit der Änderung des Überwachungsfensters ändert sich nicht nur die Fensterbreite, sondern auch der Bezugspunkt des Fenster auf die aktuellen Istposition der Achse.

"Fahren auf Festanschlag" bei aktivem Bahnsteuerbetrieb (G64)

Über das folgende Maschinendatum kann eingestellt werden, dass durch die Anwahl der Funktion (FXS) während eines aktiven Bahnsteuerbetriebs (G64), kein Genauhalt zum Satzwechsel (G60) ausgelöst wird:

MD37060 \$MA_FIXED_STOP_ACKN_MASK (Beachtung von PLC-Quittierungen für Fahren auf Festanschlag)

Bit	Wert	Bedeutung
0	0	Start der Verfahrbewegung ohne Quittierung durch die PLC
	1	Start der Verfahrbewegung nach Quittierung durch die PLC

Randbedingungen

- Die Verfahrbewegung zum Festanschlag kann als Bahn- oder satzbezogene oder satzübergreifende Positionierachsbewegung programmiert werden.
- Fahren auf Festanschlag kann gleichzeitig für mehrere Maschinenachsen erfolgen.
- Für eine Maschinenachse mit der auf Festanschlag gefahren wird, dürfen die Funktionen Transformation, Kopplung und Frame nicht aktiv sein:
- Im Teileprogramm muss der Verfahrweg und das Aktivieren der Funktion in einem Satz programmiert werden.
- Wird das Fahren auf Festanschlag über die Synchronaktionen aktiviert, können die Programmierung des Verfahrwegs und das Aktivieren der Funktion in getrennten Sätzen erfolgen.

7.2.2 Funktionsablauf

7.2.2.1 Anwahl

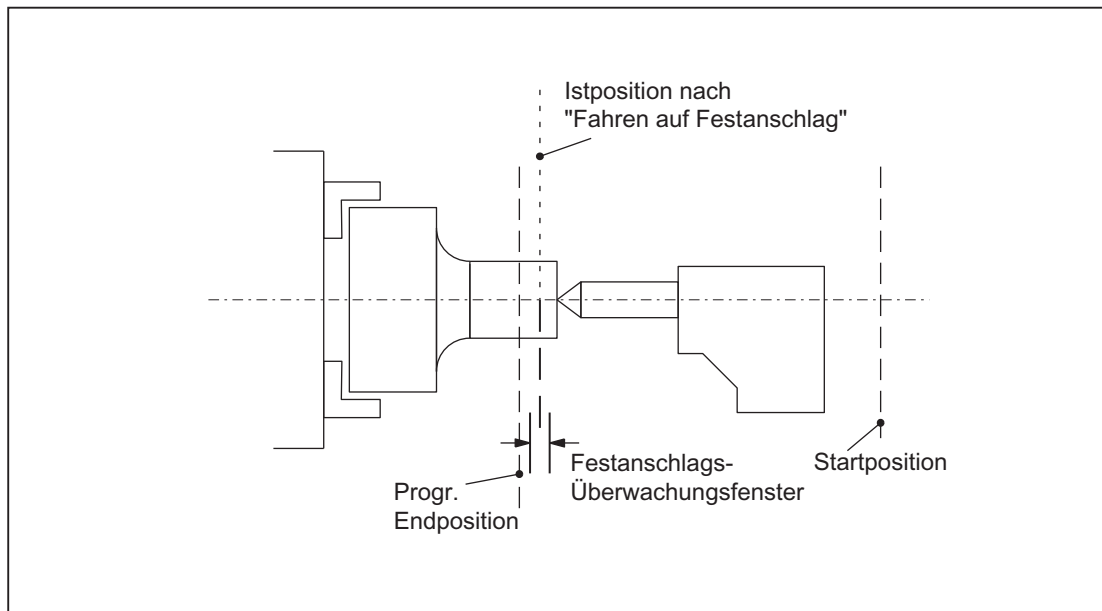


Bild 7-1 Beispiel für Fahren auf Festanschlag

Vorgang

Die NC erkennt die Funktionsanwahl "Fahren auf Festanschlag" über den Befehl `FXS[x]=1` und meldet der PLC durch das NST DB31, ... DBX62.4 ("Fahren auf Festanschlag aktivieren"), dass die Funktion angewählt wurde.

Wenn das Maschinendatum:

MD37060 \$MA_FIXED_STOP_ACKN_MASK (Beachtung von PLC-Quittierungen für Fahren auf Festanschlag)

entsprechend gesetzt ist, wird die Quittierung der PLC durch das NST DB31, ... DBX3.1 ("Fahren auf Festanschlag freigeben") abgewartet.

Anschließend wird von der Startposition aus mit der programmierten Geschwindigkeit auf die programmierte Zielposition gefahren. Der Festanschlag muss sich zwischen der Start- und Zielposition der Achse/Spindel befinden. Eine programmierte Momentenbegrenzung (Klemmmoment vorgegeben über `FXST[<Achse>]`) wirkt ab Satzbeginn, d.h. auch das Anfahren des Anschlags erfolgt mit reduziertem Moment. Dies wird in der NC durch automatische Reduzierung der Beschleunigung berücksichtigt.

Wurde in dem Satz oder seit Programmbeginn kein Moment programmiert, so gilt der Wert, der in das achsspezifische Maschinendatum:

MD37010 \$MA_FIXED_STOP_TORQUE_DEF (Voreinstellung Festanschlag-Klemmmoment) eingetragen ist.

7.2.2.2 Festanschlag wird erreicht

Erkennen des Festanschlags

Das Erkennen des Festanschlags bzw. dass die Maschinenachse den Festanschlag erreicht hat, ist über folgendes Maschinendatum einstellbar:

MD37040 \$MA_FIXED_STOP_BY_SENSOR = <Wert> (Festanschlagserkennung über Sensor)

<Wert>	Bedeutung
0	Der Festanschlag ist erreicht, wenn die Konturabweichung der Maschinenachse den eingestellten Wert im Maschinendatum überschritten hat: MD37030 \$MA_FIXED_STOP_THRESHOLD (Schwelle für Festanschlagserkennung)
1	Das Erreichen des Festanschlags wird von einem externen Sensor erkannt und über folgendes axiale NC/PLC-Nahtstellensignal der Steuerung mitgeteilt: DB31, ... DBX1.2 == 1 (Sensor Festanschlag)
2	Der Festanschlag ist erreicht, wenn eine der unter <Wert> == 0 ODER <Wert> == 1 genannten Bedingung zutrifft.

Unwirksame NC/PLC-Nahtstellensignale

Ist die Achse im Zustand "Festanschlag erreicht", sind folgende NC/PLC-Nahtstellensignale unwirksam:

- DB31, ... DBX1.3 (Achsen-/Spindelsperre)
- DB31, ... DBX2.1 (Reglerfreigabe)

Aktionen bei Erreichen des Festanschlags

Mit Erreichen des Festanschlags werden folgende Aktionen ausgeführt:

- Erhöhen des Moments im Antrieb bis zum programmierten Klemmmoment (FXST)
- Löschen des verbliebenen Restwegs
- Nachführen des Lagesollwerts
- Setzen des NC/PLC-Nahtstellensignal: DB31, ... DBX62.5 = 1 ("Festanschlag erreicht")
- Satzwechsel durchführen:
 - Satzwechsel mit Erreichen des Festanschlags durchführen: MD37060 \$MA_FIXED_STOP_ACKN_MASK, Bit 1 = 0
Der Satzwechsel wird sofort nach Erreichen des Festanschlags durchgeführt.
 - Satzwechsel erst nach Quittierung durch das PLC-Anwenderprogramm durchführen: MD37060 \$MA_FIXED_STOP_ACKN_MASK, Bit 1 = 1
Nach Erreichen des Festanschlags wird mit dem Satzwechsel solange gewartet, bis die Quittierung durch das PLC-Anwenderprogramm erfolgt:
DB31, ... DBX1.1 == 1 (Festanschlag erreicht quittieren)

Um auch nach dem Satzwechsel das Klemmmoment aufrecht zu erhalten, wird von der NC für die Maschinenachse weiter ein Sollwert ausgegeben.

- Aktivierung der Festanschlags-Überwachung bzw. des Überwachungsfensters

Überwachungsfenster

Wurde im Verfahrssatz zum Festanschlag oder seit Programmbeginn mit `FXSW` kein spezifischer Wert für das Überwachungsfenster programmiert, wirkt der im Maschinendatum eingestellte Wert:

MD37020 \$MA_FIXED_STOP_WINDOW_DEF (Voreinstellung für Festanschlags-Überwachungsfenster)

Verlässt die Achse die Position, die sie beim Erkennen des Anschlags hatte, um mehr als das angegebene Fenster, wird Alarm 20093 "Festanschlags-Überwachung hat angesprochen" angezeigt und die "Fahren auf Festanschlag" deaktiviert.

Das Überwachungsfenster sollte vom Anwender so gewählt werden, dass nur ein Wegbrechen des Anschlags zum Auslösung des Alarms führt.

ACHTUNG

"Festanschlag erreicht" und Brechen des Festanschlags

Sobald der Zustand "Festanschlag erreicht" erkannt wird, wird ein von den achsspezifischen Maschinendaten K_v -Faktor (MD32200) und Schwelle für Festanschlagserkennung (MD37030) abgeleiteter Drehzahlsollwert, an den Antrieb fest vorgegeben. Bricht der Festanschlag in diesem Zustand weg, beschleunigt die Achse bis zum Erreichen der Grenze des Überwachungsfensters. Die dabei erreichte Geschwindigkeit ist proportional zu den in den genannten Maschinendaten eingestellten Werten. Bei entsprechend großen Werten, kann dabei bis zur maximalen Motordrehzahl des Antriebs beschleunigt werden.

Übersicht

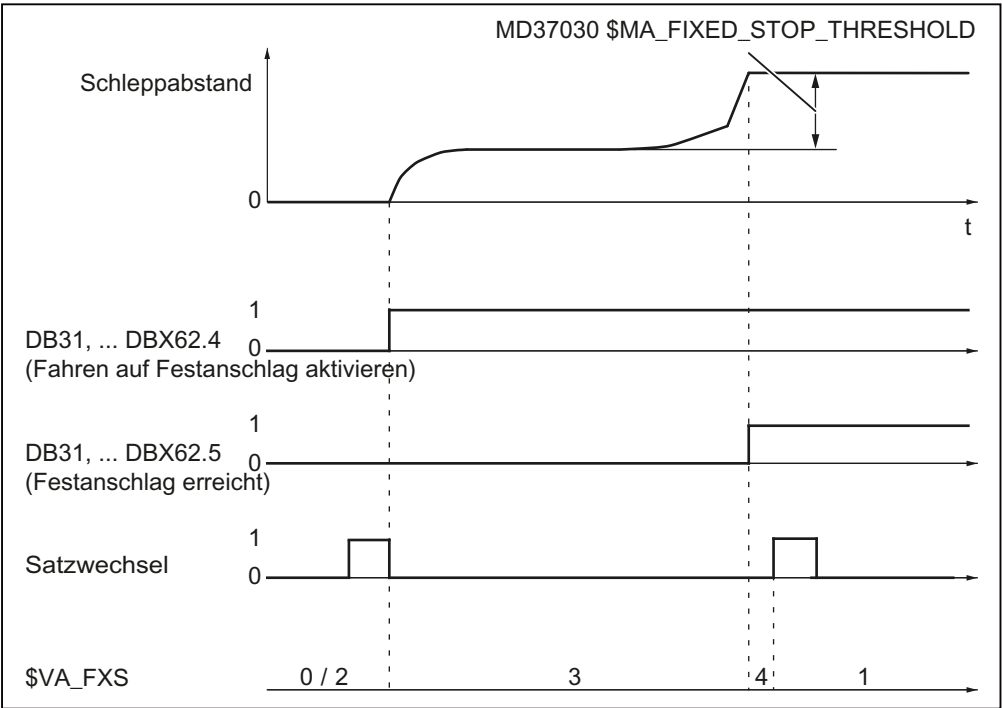


Bild 7-2 Festanschlag wird erreicht

7.2.2.3 Festanschlag wird nicht erreicht

Alarmunterdrückung

Über das Maschinendatum können Alarme für verschieden Abbruchsursachen unterdrückt werden:

MD37050 \$MA_FIXED_STOP_ALARM_MASK = <Wert>

Wert	Beschreibung: Unterdrückte Alarme
0	Alarm 20091 "Festanschlag nicht erreicht"
2	Alarm 20091 "Festanschlag nicht erreicht" Alarm 20094 "Festanschlag abgebrochen"
3	Alarm 20094 "Festanschlag abgebrochen"

Aktionen bei Störung oder Abbruch

Bei Auftreten einer Störung oder bei Abbruch, werden folgende Aktionen ausgeführt:

- Rücksetzen des NC/PLC-Nahtstellensignals: DB31, ... DBX62.4 = 0 (Fahren auf Festanschlag aktivieren)
- Abhängig von der Einstellung in Maschinendatum wird die Quittierung abgewartet:
 - MD37060 \$MA_FIXED_STOP_ACKN_MASK
 - DB31, ... DBX3.1 == 0 (Fahren auf Festanschlag freigeben)
- Momentenbegrenzung aufheben
- Satzwechsel durchführen

Übersicht

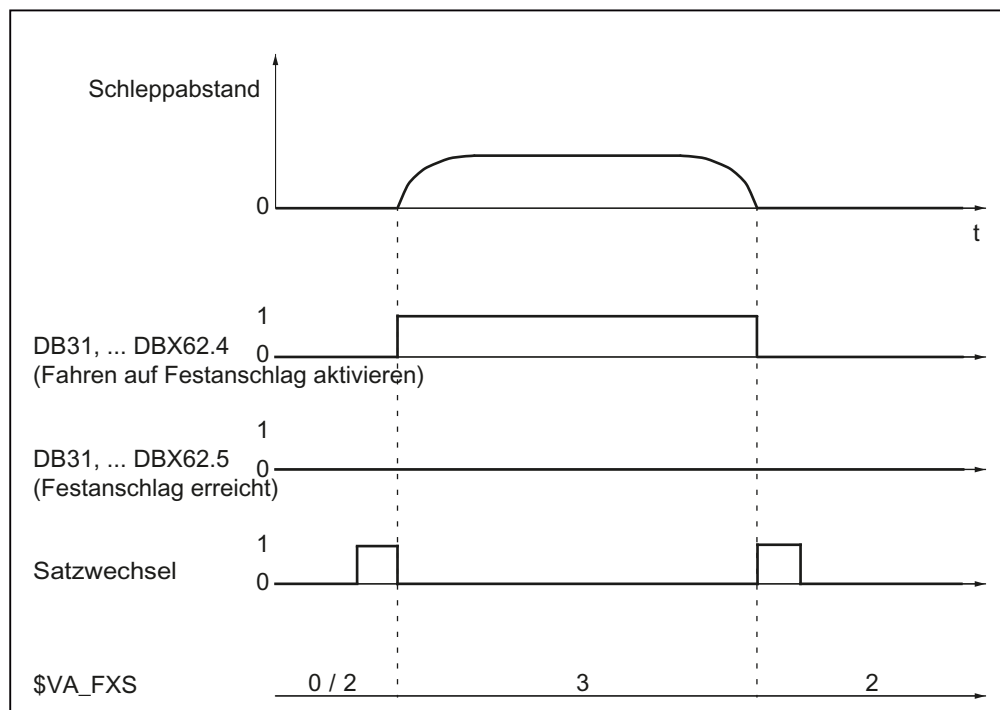


Bild 7-3 Festanschlag wird nicht erreicht

7.2.2.4 Abwahl

Die Abwahl der Funktion "Fahren auf Festanschlag" erfolgt über den Befehl `FXS [<Achse>] = 0` in einem Satz eines NC-Programms.

Aktionen bei Abwahl der Funktion

Bei Abwahl der Funktion werden folgende Aktionen ausgeführt:

- Auslösen von Vorlaufstopp (STOPRE)
- Rücksetzen der NC/PLC-Nahtstellensignale
 - DB31, ... DBX62.4 = 0 (Fahren auf Festanschlag aktivieren)
 - DB31, ... DBX62.5 = 0 (Festanschlag erreicht)
- Abhängig vom Maschinendatum wird die Quittierung durch das PLC-Anwenderprogramm abgewartet:
 - MD37060 \$MA_FIXED_STOP_ACKN_MASK
 - DB31, ... DBX3.1 == 0 (Fahren auf Festanschlag freigeben)
 - DB31, ... DBX1.1 == 0 (Festanschlag erreicht quittieren)
- Beenden des Nachführbetriebs
- Wiederaufnahme der Achse in die Lageregelung

Impulsfreigabe

Eine Wegnahme der Impulsfreigabe bzw. Impulssperre kann erfolgen über:

- Antrieb: über Klemme EP (Enable Pulses)
- NC/PLC-Nahtstellensignal: DB31, ... DBX21.7 ("Impulsfreigabe")

Über das folgende Maschinendatum kann das Verhalten am Festanschlag eingestellt werden:

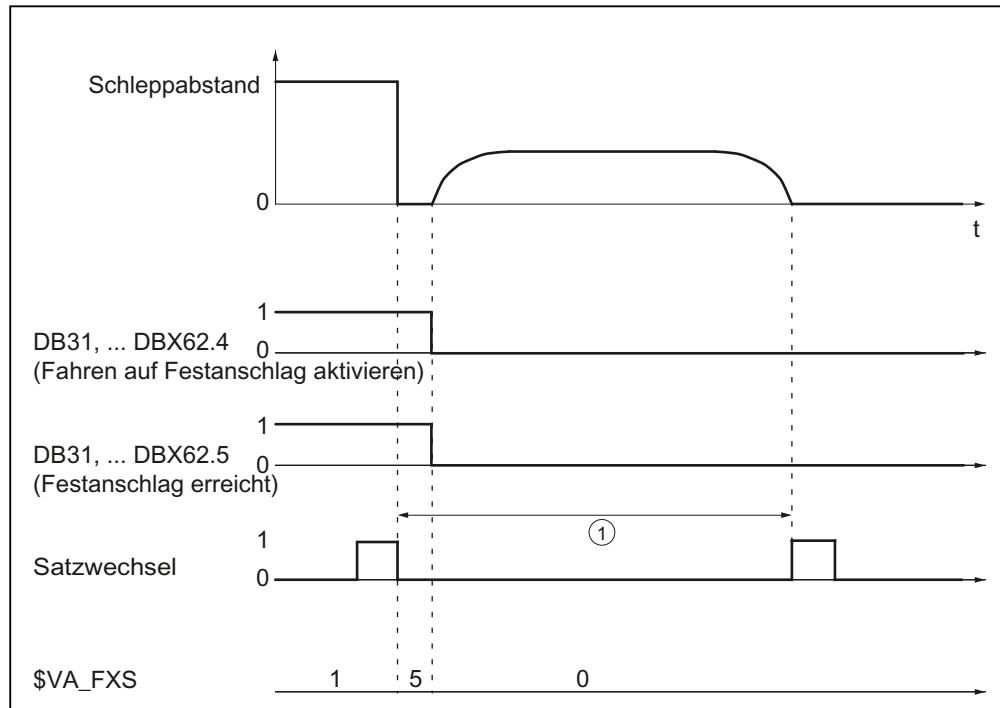
MD37002 \$MA_FIXED_STOP_CONTROL, Bit 0 und Bit 1 (Ablaufkontrolle für Fahren auf Festanschlag)

Bit	Wert	Bedeutung
0		Verhalten bei Impulssperre am Anschlag
	0	"Fahren auf Festanschlag" wird abgebrochen
	1	"Fahren auf Festanschlag" wird unterbrochen, d.h. der Antrieb wird kraftlos.
1		Verhalten bei Impulsfreigabe am Anschlag
	0	Das Moment wird sprungförmig aufgeschaltet
	1	Das Moment wird linear über die im Maschinendatum angegebene Zeit aufgeschaltet: MD37012 \$MA_FIXED_STOP_TORQUE_RAMP_TIME

Sonderfall: Löschen der Impulsfreigabe während der Abwahl

Wird während der Abwahl der Funktion im Zustand: "Warten auf PLC-Quittungen" die Impulsfreigabe gelöscht, wird die Momentenbegrenzung auf Null auf den Wert 0 gesetzt. Wird in dieser Phase die Impulsfreigabe wieder gesetzt, wird trotzdem kein Moment im Antrieb mehr aufgebaut. Sobald die Abwahl abgeschlossen ist, kann die Achse wieder normal verfahren werden.

Übersicht



① Verfahrenssatz mit Abwahl $FXS[<Achse>]=0$

Bild 7-4 Festanschlag Abwahl

7.2.3 Verhalten bei Satzsuchlauf

Funktion

Satzsuchlauf mit Berechnung

- Wenn der Zielsatz in einem Programmabschnitt liegt, in dem die Achse am Festanschlag stehen soll, wird der Festanschlag angefahren, wenn dieser nicht bereits erreicht ist.
- Wenn der Zielsatz in einem Programmabschnitt liegt, in dem die Achse nicht am Festanschlag stehen soll, wird der Festanschlag verlassen, wenn die Achse noch dort steht.
- Ist die Achse im Zustand "Festanschlag erreicht", wird die Meldung 10208 "Zur Programmfortsetzung NC-Start geben" angezeigt. Mit NC-Start kann das Programm fortgesetzt werden.
- Klemmmoment $FXST$ und Überwachungsfenster $FXSW$ haben beim Start des Zielsatzes jeweils den Wert, den sie auch bei normaler Programmbearbeitung haben.

Satzsuchlauf ohne Berechnung

Die Befehle FXS , $FXST$ und $FXSW$ werden während des Satzsuchlaufs ignoriert.

Wirksamkeit von FOCON/FOCOF

Der Zustand der modal wirkenden Momenten/Kraft-Reduzierung FOCON/FOCOF wird während des Satzsuchlaufs mitgeführt und ist im Anfahr Satz wirksam.

Satzsuchlauf mit FXS oder FOC

Der Anwender markiert in einem Programmbereich des Zielsatzes die Funktion FXS oder FOC, um alle zuletzt gültigen Zustände und Funktionen dieser Bearbeitung zu erfassen. Die NC startet selbsttätig das angewählte Programm im Modus Programmtest. Nach gefundenem Zielsatz stoppt die NC am Beginn des Zielsatzes, wählt intern Programmtest wieder ab, und zeigt anschließend die Stopp-Bedingung "Suchziel gefunden" in ihrer Satzanzeige an.



VORSICHT

SERUPRO-Anfahren berücksichtigt die Anweisung FXS nicht real

Die programmierte Endposition des FXS-Satzes wird **ohne Momentenbegrenzung** nur simuliert angefahren.

Beindet sich zwischen Programmanfang und Suchziel die Anweisung FXS, wird diese von der NC nicht real ausgeführt. Die Bewegung wird bis zum programmierten Endpunkt nur simuliert.

Der Anwender kann im Teileprogramm das Ein- und Ausschalten von FXS mitprotokollieren. Bei Bedarf kann der Benutzer ein ASUP starten, um FXS in diesem SERUPRO-ASUP ein- oder auszuschalten.

Systemvariable

Der Soll- und Istzustand der Funktion "Fahren auf Festanschlag" kann über folgende Systemvariable gelesen werden:

- \$AA_FXS (Sollzustand)
- \$VA_FXS (Istzustand)

SERUPRO: \$AA_FXS (Sollzustand)

Während SERUPRO liefert \$AA_FXS, abhängig vom Aktivierungsstatus der Funktion "Fahren auf Festanschlag", folgende Werte:

Aktivierungsstatus der Funktion "Fahren auf Festanschlag"	Systemvariable \$AA_FXS ==
"Ausgeschaltet"	0 (Achse nicht am Anschlag)
"Eingeschaltet"	3 (Anwahl "Fahren auf Festanschlag" ist aktiv)

Hinweis

Während SERUPRO liefert die Systemvariable \$AA_FXS nur die Werte 0 und 3. Dadurch kann bei Programmverzweigungen, basierend auf \$AA_FXS, der Programmablauf bei SERUPRO gegenüber der normalen Programmabarbeitung verändert werden.

SERUPRO: \$VA_FXS (Istzustand)

Während SERUPRO liefert die Variable \$VA_FXS immer den realen Zustand der Achse an der Maschine.

Beispiel

Im SERUPRO-ASUP kann über die Systemvariablen \$AA_FXS und \$VA_FXS der aktuelle Zustand der Funktion "Fahren auf Festanschlag" ermittelt und entsprechend reagiert werden:

Programmcode: FXS_SERUPRO_ASUP.MPF	Kommentar
N100 WHEN (\$AA_FXS[X]==3) AND (\$VA_FXS[X]==0) DO FXS[X]=1	; Soll=="Anwahl nicht aktiv" AND ; Ist=="Achse nicht am Festanschlag" ; => "Einschalten"
N200 WHEN (\$AA_FXS[X]==0) AND (\$VA_FXS[X]==1) DO FXS[X]=0	; Soll=="Achse nicht am Festanschlag" AND ; Ist=="Festanschlag erfolgreich angefahren" ; => "Einschalten"
N1020 REPOSA	; Wiederanfahren an die Kontur, linear mit allen Achsen

REPOS-Verschiebung anzeigen

Nachdem das Suchziel gefunden ist, wird für jede Achse der aktuelle Zustand bezüglich "Fahren auf Festanschlag" über folgende NC/PLC-Nahtstellensignale angezeigt:

- DB31, ... DBX62.4 (Fahren auf Festanschlag aktivieren)
- DB31, ... DBX62.5 (Festanschlag erreicht)

Beispiel:

Wenn die Achse am Festanschlag steht, und der Zielsatz nach der Abwahl von FXS liegt, wird die neue Zielposition über DB31, ... DBX62.5 (Festanschlag erreicht) als REPOS-Verschiebung angezeigt.

REPOS und FXS

Mit REPOS wird die Funktionalität von FXS selbsttätig wiederholt und im Folgenden mit FXS-REPOS bezeichnet. Diese Sequenz ist vergleichbar mit dem Programm FXS_SERUPRO_ASUP.MPF. Dabei wird jede Achse berücksichtigt und es wird das zuletzt vor dem Suchziel programmierte Moment verwendet.

Der Anwender kann FXS in einem SERUPRO-ASUP gesondert behandeln.

Dafür gilt:

Jede im SERUPRO-ASUP durchgeführte FXS-Aktion sorgt automatisch für

\$AA_FXS[<Achse>] = \$VA_FXS[<Achse>].

Damit wird FXS-REPOS für die Achse X inaktiv.

FXS-REPOS deaktivieren

FXS-REPOS wird deaktiviert durch:

- Eine FXS-Synchronaktion, die sich auf REPOSA bezieht
- $\$AA_FXS[X] = \$VA_FXS[X]$ im SERUPRO_ASUP

Hinweis

Ein SERUPRO-ASUP **ohne** FXS-Behandlung oder ein nicht vorhandenes SERUPRO-ASUP führt automatisch zu FXS-REPOS.

VORSICHT

Zu hohe Geschwindigkeit bei FXS-REPOS

FXS-REPOS fährt alle Bahnachsen gemeinsam auf den Zielpunkt. Damit fahren Achsen mit und ohne FXS-Behandlung gemeinsam mit dem im Zielsatz gültigen G-Befehl und Vorschub. Dies kann dazu führen, dass der Festanschlag mit Eilgang (G0) oder hoher Geschwindigkeit angefahren wird.

FOC-REPOS

FOC-REPOS verhält sich analog zu FXS-REPOS.

Ein während des Programmvorlaufs auftretender wechselnder Momentenverlauf kann mit FOC-REPOS nicht realisiert werden.

Beispiel

Achse X wird von Position 0 nach 100 verfahren. Dabei wird alle 20 Millimeter für jeweils 10 Millimeter FOC eingeschaltet. Der sich dabei ergebende Momentenverlauf wird mit satzweisem FOC erzeugt und kann durch FOC-REPOS nicht nachvollzogen werden. Durch FOC-REPOS wird die Achse X mit oder ohne FOC fahren, entsprechend der letzten Programmierung vor dem Zielsatz.

Programmierbeispiele zu FXS "Fahren auf Festanschlag" siehe Kapitel "Programmtest (Seite 494)".

7.2.4 Verhalten bei Reset und Funktionsabbruch

NC-Reset

Solange die Funktion noch nicht im Zustand "Anschlag wurde erfolgreich angefahren" ist, kann das Fahren auf den Festanschlag mit NC-Reset abgebrochen werden.


Auch wenn der Festanschlag bereits angefahren wurde, das vorgegebene Anschlagsmoment aber noch nicht vollständig erreicht, dann die Funktion mit NC-Reset noch abgebrochen werden. Dabei wird der Sollposition der Achse auf die aktuelle Istposition synchronisiert.

Sobald die Funktion im Zustand "Anschlag wurde erfolgreich angefahren" ist, bleibt die Funktion auch über NC-Reset hinaus wirksam.

Funktionsabbruch

Ein Funktionsabbruch wird durch folgende Ereignisse ausgelöst:

- Not-Halt

 VORSICHT
Gefährliche Maschinensituationen beim Fahren auf Festanschlag möglich
Es ist sicher zu stellen, dass durch das Auslösen und Rücksetzen von "Not-Halt" während Fahren auf Festanschlag aktiv ist, keine gefährlichen Maschinensituationen entstehen.
Z.B. Verhalten beim Setzen und Wegnehmen der Impulssperre: MD37002 \$MA_FIXED_STOP_CONTROL, Bit 0 (Verhalten bei Impulssperre am Anschlag)
<ul style="list-style-type: none">• Bit 0 = 0: Fahren auf Festanschlag wird abgebrochen• Bit 0 = 1: Fahren auf Festanschlag wird unterbrochen, d.h. der Antrieb wird kraftlos. Sobald die Impulssperre wieder aufgehoben wird, drückt der Antrieb wieder mit dem Anschlagsmoment.

Hinweis

NC und der Antrieb sind bei "Not-Halt" stromlos, d. h. die PLC muss reagieren.

- Funktionszustand: "Festanschlag nicht erreicht"
- Funktionszustand: "Festanschlag abgebrochen"
- Abbruch durch das PLC-Anwenderprogramm:
DB31, ... DBX62.4 = 0 ("Fahren auf Festanschlag aktivieren")
- Wegnahme der Impulsfreigabe und Maschinendatenparametrierung:
MD37002 \$MA_FIXED_STOP_CONTROL, Bit 0 = 0 (siehe oben)

7.2.5 Verhalten bezüglich anderer Funktionen

Messen mit Restweglöschen

"Fahren auf Festanschlag" (FXS) kann nicht gemeinsam in einem Satz mit "Messen mit Restweglöschen" (MEAS) programmiert werden. Außer eine Funktion wirkt auf eine Bahnachse und die andere auf eine Positionierachse oder beide wirken auf Positionierachsen.

Konturüberwachung

Während "Fahren auf Festanschlag" aktiv ist, erfolgt keine axiale Konturüberwachung.

Positionierachsen

Bei "Fahren auf Festanschlag" mit Positionierachsen POSA wird der Satzwechsel ausgeführt, auch wenn die Positionierachse bis zu diesem Zeitpunkt den Festanschlag noch nicht erreicht hat.

Hängende Achsen

Das "Fahren auf Festanschlag" mit hängenden Achsen ist auch bei Alarmmeldungen möglich.

Tritt bei einer hängenden Achse während des Fahrens auf Festanschlag ein funktionsspezifischer Alarm auf, wird das NC/PLC-Nahtstellensignal DB11, DBX6.3 (BAG betriebsbereit) nicht zurückgesetzt. Dadurch wird der entsprechende Antrieb nicht stromlos geschaltet.

Dies entspricht für die hängende Achse einem elektronischen Gewichtsausgleich und ist über das folgende Maschinendatum projektierbar:

MD37052 \$MA_FIXED_STOP_ALARM_REACTION

Literatur

Weitere Informationen zu hängenden Achsen finden sich in:

- Funktionshandbuch SINAMICS S120
- Funktionshandbuch Erweiterungsfunktionen; Kompensation (K3), Kapitel: Elektronischer Gewichtsausgleich

7.2.6 Settingdaten

Die über die funktionsspezifischen Befehle `FXS`, `FXST` und `FXSW` programmierten Werte, werden satzsynchron in folgende sofort wirksamen, achsspezifischen Settingdaten geschrieben:

Ein/Ausschalten der Funktion

SD43500 \$SA_FIXED_STOP_SWITCH (An/Abwahl Fahren auf Festanschlag)

Klemmmoment

SD43510 \$SA_FIXED_STOP_TORQUE (Klemmmoment)

Hinweis

Klemmmoment größer als 100%

Ein Wert für das Klemmmoment in SD43510 größer 100% des maximalen Motormoments ist nur kurzzeitig sinnvoll. Außerdem wird das maximale Motormoment durch den Antrieb begrenzt. Begrenzend wirken z.B. folgende Antriebsparameter:

- p1520/p1521 Drehmomentgrenze/Kraftgrenze oben / Drehmomentgrenze/Kraftgrenze unten
- p1522/p1523 Drehmomentgrenze/Kraftgrenze oben / Drehmomentgrenze/Kraftgrenze unten
- p1530/p1531 Leistungsgrenze motorisch / Leistungsgrenze generatorisch
- p0640 Stromgrenze
- p0326 Motor-Kippkraftkorrekturfaktor

Ausführliche Informationen zu den Antriebsparametern und den Funktionen finden sich in:

Literatur

- SINAMICS S120/S150 Listenhandbuch
 - SINAMICS S120 Funktionshandbuch
-

Überwachungsfenster

SD43520 \$SA_FIXED_STOP_WINDOW (Überwachungsfenster)

Vorbelegung

Über folgende Maschinendaten wird die Vorgebelegung der Settingdaten eingestellt:

- Klemmmoment:
MD37010 \$MA_FIXED_STOP_TORQUE_DEF (Vorbelegung Klemmmoment)
- Überwachungsfenster:
MD37020 \$MA_FIXED_STOP_WINDOW_DEF (Vorbelegung Überwachungsfenster)

Wirksamkeit

Die Settingdaten für Klemmmoment und Überwachungsfenster sind sofort wirksam. Dadurch kann vom Bediener oder über das PLC-Anwenderprogramm der Klemmzustand jederzeit an die Bearbeitungssituation angepasst werden.

Literatur

Weiter ausführliche Informationen zu Maschinen- und Settingdaten finden sich in:
Listenhandbuch Ausführliche Maschinendatenbeschreibung

7.2.7 Systemvariablen**Soll-/Istzustand**

Der Soll- und Istzustand der Funktion "Fahren auf Festanschlag" kann über folgende Systemvariablen gelesen werden:

- \$AA_FXS = <Wert> (Status Sollzustand "Fahren auf Festanschlag")
- \$VA_FXS = <Wert> (Status Istzustand "Fahren auf Festanschlag")

<Wert >	Beschreibung
0	Achse nicht am Festanschlag
1	Festanschlag wurde erfolgreich angefahren
2	Anfahren des Festanschlags fehlgeschlagen
3	Anwahl "Fahren auf Festanschlag" ist aktiv
4	Festanschlag wurde erkannt
5	Abwahl Fahren auf Festanschlag aktiv

Zusatzinformationen

Sind beim Fahren auf den Festanschlag Fehlern aufgetreten (\$VA_FXS == 2), werden in der folgenden Systemvariablen zusätzliche Informationen angezeigt:

- \$VA_FXS_INFO = <Wert> (Zusatzinformation bei "Fahren auf Festanschlag")

<Wert>	Beschreibung
0	Keine Zusatzinformation vorhanden
1	Keine Anfahrbewegung programmiert
2	Programmierte Endposition erreicht, Bewegung beendet
3	Abbruch durch Rest, DB21, ... DBX7.7 == 1
4	Überwachungsfenster wurde verlassen
5	Momentenreduzierung wurde vom Antrieb verweigert
6	PLC hat die Freigabe zurückgenommen, DB31, ... DBX3.1 == 0 (Fahren auf Festanschlag freigegeben)

Anwendungsbeispiel für \$AA_FXS

Damit ein Satzwechsel erfolgt, soll im Fehlerfall kein Alarm ausgelöst werden. Die Ursache wird anschließend über die Systemvariable \$VA_FXS_INFO ermittelt und spezifisch darauf reagiert.

Voraussetzung: MD37050 \$MA_FIXED_STOP_ALARM_MASK = 0 (Bei Fahren auf Festanschlag keinen Alarm auslösen)

Programmcode	Kommentar
X300 Y500 F200 FXS[X1]=1 FXST[X1]=25 FXSW[X1]=5	; Fahren auf Festanschlag
IF \$AA_FXS[X1] == 2 GOTOF FXS_ERROR	; IF Festanschlag == erreicht
R100=\$AA_IM[X1]	; THEN (Normalfall)
IF R100 ...	; Auswertung der aktuellen
...	; Istposition
GOTOF PROG_END	
FXS_ERROR:	; ELSE (Fehlerfall)
CASE(\$VA_FXS_INFO[X1]) OF 0 GOTOF LABEL_0 OF 1 GOTOF LABEL_1 ...	; Fehlerbehandlung
LABEL_0: ...	
GOTOF CASE_END	
LABEL_1: ...	
GOTOF CASE_END	
...	
CASE_END:	
...	
PROG_END: M30	; ENDIF Programmende

7.2.8 Alarme

Alarm 20091 "Festanschlag nicht erreicht"

Wird beim Fahren auf Festanschlag die Zielposition nicht erreicht, wird Alarm 20091 "Festanschlag nicht erreicht" angezeigt und ein Satzwechsel durchgeführt.

Alarm 20092 "Fahren auf Festanschlag noch aktiv"

Erfolgt für eine Achse nach Erreichen des Festanschlags eine Fahrenanforderung oder erneute Funktionsanwahl, wird der Alarm 20092 "Fahren auf Festanschlag noch aktiv" angezeigt.

Alarm 20093 "Stillstandsüberwachung am Anschlag hat ausgelöst"

Wird eine Achse nach Erreichen des Festanschlags um mehr als den im Settingdatum SD43520 FIXED_STOP_WINDOW (Festanschlags-Überwachungsfenster)

angegebenen Wert aus der Position bewegt, wird Alarm 20093 "Stillstandsüberwachung am Anschlag hat ausgelöst" angezeigt, Fahren auf Festanschlag für diese Achse abgewählt und folgende Systemvariable gesetzt:

\$AA_FXS[x] = 2

Alarm 20094 "Funktion wurde abgebrochen"

Das Fahren auf Festanschlag wird abgebrochen, wenn durch die Wegnahme der Impulsfreigabe das Klemmmoment nicht länger aufgebracht werden kann, oder das angeforderte Quittungssignal an der NC/PLC-Nahtstelle zurückgesetzt wurde:

- Quittungssignal erforderlich: MD37060 \$MA_FIXED_STOP_ACKN_MASK, Bit 0 = 1
- Quittungssignal: DB31, ... DBX3.1 == 0 (Fahren auf Festanschlag freigeben)

Freigabe der Festanschlagsalarme

Über das folgende Maschinendatum kann eingestellt werden, ob die Festanschlagsalarme

- Alarm 20091 "Festanschlag nicht erreicht",
- Alarm 20094 "Festanschlag abgebrochen"

angezeigt werden:

MD37050 \$MA_FIXED_STOP_ALARM_MASK (Freigabe der Festanschlagsalarme)

Einstellbares Funktionsverhalten bei Festanschlagsalarmen

Über das folgende Maschinendatum kann eingestellt werden, dass die Funktion auch beim Auftreten funktionspezifischer Alarme nicht abgebrochen wird:

- Alarm 20090 Fahren auf Festanschlag nicht möglich
- Alarm 20091 Festanschlag nicht erreicht
- Alarm 20092 Fahren auf Festanschlag noch aktiv
- Alarm 20093 Stillstandsüberwachung am Anschlag hat ausgelöst
- Alarm 20094 Fahren auf Festanschlag abgebrochen

MD37052 \$MA_FIXED_STOP_ALARM_REACTION (Reaktion bei Festanschlagsalarmen)

Alarmunterdrückung nach neuer Programmierung

Fahren auf Festanschlag kann für einfache Messvorgänge eingesetzt werden.

Beispielsweise kann auf Werkzeugbruch kontrolliert werden, indem die Länge des Werkzeugs durch Auffahren auf ein definiertes Hindernis gemessen wird. Dies erfordert die Unterdrückung des Anschlagsalarms. Bei anschließender "normaler" Nutzung der Funktion zum Klemmen von Werkstücken ist eine Aktivierung des Alarms durch Teileprogrammbefehle möglich.

7.2.9 Fahren mit begrenztem Moment/Kraft FOC

Funktion

Über die Funktion "Force Control" zum Fahren mit begrenztem Moment / Kraft kann das beim Verfahren maximal zulässige Moment / Kraft auf einen prozentualen Wert des maximal möglichen Moments der Achse begrenzt werden. Der Begrenzungswert kann jederzeit, auch während des Verfahrens der Achse verändert werden. Die Änderungen können im Interpolatortakt, weg-, zeitabhängig oder von beliebigen anderen Größen abgeleitet, erfolgen. Die Funktion "Force Control" kann auch in Synchronaktionen programmiert werden. Die Aktivierung der Funktion ist permanent oder satzbezogen möglich.

Programmierung

Syntax

FOCON [<Achse>]
FOCOF [<Achse>]
FOC [<Achse>]

Bedeutung

Parameter	Bedeutung
FOCON:	Moment / Kraft-Begrenzung einschalten
FOCOF:	Moment / Kraft-Begrenzung ausschalten
FOC:	Satzweise wirksame Moment/Kraft-Begrenzung einschalten
<Achse>:	Kanalachsname, Typ: AXIS

Beispiel

Programmcode	Kommentar
N10 FOCON[X]	; modale Aktivierung der Momentenbegrenzung
N20 X100 Y200 FXST[X]=15	; X fährt mit reduziertem Moment (15%)
N30 FXST[X]=75 X20	; Änderung des Moments auf 75%,
N40 FOCO[X]	; Abschalten der Momentenbegrenzung

Parametrierung

Maschinendaten

- MD37010 \$MA_FIXED_STOP_TORQUE_DEF (Voreinstellung Festanschlag-Klemmoment)
Der im Maschinendatum angegebene Wert wirkt nach Aktivierung der Funktion solange mit `FXST` kein expliziter Wert programmiert wird.
- MD36042 \$MA_FOC_STANDSTILL_DELAY_TIME (Verzögerungszeit Stillstandsüberwachung bei aktiver Momenten- / Kraftbegrenzung)

Permanente Aktivierung (FOCON/FOCOF)

Die permanente Aktivierung der Funktion "Force Control" nach POWER ON bzw. RESET ist einstellbar über das Maschinendatum:

MD37080 \$MA_FOC_ACTIVATION_MODE, Bit 0 und Bit 1 (Grundstellung der modalen Begrenzung von Kraft / Moment)

Bit	Wert	Bedeutung
0	0	Die Funktion "Force Control" ist nach POWER ON nicht aktiv.
	1	Die Funktion "Force Control" ist nach POWER ON aktiv.
1	0	Die Funktion "Force Control" ist nach RESET nicht aktiv.
	1	Die Funktion "Force Control" ist nach RESET aktiv.

Wurde vor der Aktivierung mit `FXST = <Wert>` bereits ein Begrenzungsmoment bzw. -kraft programmierte, wirkt dieses ab dem Zeitpunkt der Aktivierung.

Wurde vor der Aktivierung mit `FXST = <Wert>` kein Begrenzungsmoment bzw. -kraft programmierte, wirkt standardmäßig der im Maschinendatum angegebene Wert:

MD37010 \$MA_FIXED_STOP_TORQUE_DEF (Voreinstellung Festanschlag-Klemmoment)

Satzbezogene Aktivierung (FOC)

Mit dem Befehl `FOC` wird die Funktion für den aktuellen Satz aktiviert.

Die Aktivierung der Funktion über eine Synchronaktion wirkt bis zum Satzende des aktuellen Teileprogrammsatzes.

Priorität von FXS vor FOC

Die Aktivierung der Funktion "Fahren auf Festanschlag" `FXS` ist höherprior die Funktion "Force Control". Ist "Fahren auf Festanschlag" gleichzeitig mit "Force Control" aktiv, wird erste ausgeführt.

Die Abwahl der Funktion "Fahren auf Festanschlag" `FXS` hebt die Klemmung auf. Eine gleichzeitig permanent aktive Funktion "Force Control" bleibt weiter aktiv.

Systemvariable

Status der Funktion "Force Control"

Über die Systemvariable \$AA_FOC kann der Status der Funktion "Force Control" gelesen werden.

Eine Veränderung des Status der Funktion "Fahren auf Festanschlag" `FXS` verändert den Status der Funktion "Force Control" nicht.

\$AA_FOC	
Wert	Bedeutung
0	FOC nicht aktiv
1	FOC modal aktiv
2	FOC satzbezogen aktiv

Status der Momentenbegrenzung

Über die Systemvariablen \$VA_TORQUE_AT_LIMIT kann der aktuelle Status der Momentenbegrenzung gelesen werden.

\$VA_TORQUE_AT_LIMIT	
Wert	Bedeutung
0	aktuell wirksames Moment < Momentengrenzwert
1	aktuell wirksames Moment == Momentengrenzwert

Einschränkungen

Die Funktion "Force Control" hat folgende Einschränkungen:

- Die Veränderung der Momenten-/Kraftbegrenzung, die sich als Beschleunigungsbegrenzung darstellt, wird **nur an Satzgrenzen** in die Verfahrbewegung eingerechnet (siehe Befehl `ACC`).
- Nur FOC: Es ist **keine Überwachung** auf das Erreichen der aktiven Momentengrenze aus der NC/PLC-Nahtstelle möglich.
- Durch Nichtanpassung der Beschleunigungsbegrenzung kommt es während der Verfahrbewegung zu einer Erhöhung des Schleppabstandes.
- Durch Nichtanpassung der Beschleunigungsbegrenzung wird der Satzendpunkt eventuell später erreicht, als im Maschinendatum angegeben:
MD36040 \$MA_STANDSTILL_DELAY_TIME
Dafür wird das Maschinendatum:
MD36042 \$MA_FOC_STANDSTILL_DELAY_TIME
eingeführt, auf das in diesem Zustand überwacht wird.

Einsatzmöglichkeit bei Link- und Containerachsen

Alle Achsen, die in einem Kanal verfahren werden können, d. h. auch Linkachsen und Containerachsen, können auf Festanschlag gefahren werden.

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Mehrere Bedientafeln an mehreren NCUs, Dezentrale Systeme (B3)

Der Zustand der Maschinenachse bleibt bei einer Container-Drehung erhalten, d. h. eine geklemmte Maschinenachse bleibt am Anschlag.

Wurde mit FOCON eine modale Momentenbegrenzung aktiviert, bleibt diese auch nach einer Container-Drehung für die Maschinenachse erhalten.

7.3 Beispiele

Beispiel 1: Fahren auf Festanschlag mit statischer Synchronaktionen

Fahren auf Festanschlag (FXS) wird auf Anforderung über R-Parameter (\$R1) in einer statischen Synchronaktion ausgelöst.

Programmcode	Kommentar
N10 IDS=1 WHENEVER	; Statische Synchronaktion 1:
((\$R1==1) AND	; R1==1 (FXS für Y angefordert) UND
(\$AA_FXS[Y]==0)) DO	; Vermeidung von Mehrfachanwahl
	; \$AA_FXS[Y]==0 (Achse nicht am Anschlag): =>
\$R1=0 FXS[Y]=1	; \$R1 rücksetzen, FXS für Y einschalten
FXST[Y]=10	; Anschlagsmoment: 10%
FA[Y]=200	; Axialer Vorschub Y: 200
POS[Y]=150	; Positionierbewegung Y
N11 IDS=2 WHENEVER	; Statische Synchronaktion 2:
(\$AA_FXS[Y]==4) DO	; \$AA_FXS[Y]==4 (Anschlag erkannt): =>
FXST[Y]=30	; Anschlagsmoment: 30%
N12 IDS=3 WHENEVER	; Statische Synchronaktion 3:
(\$AA_FXS[Y]==1) DO	; \$AA_FXS[Y]==1 (Anschlag erfolgreich angefahren): =>
FXST[Y]=\$R0	; Anschlagsmoment: Wert aus R-Parameter \$R0
N13 IDS=4 WHENEVER	; Statische Synchronaktion 4:
((\$R3==1) AND	; R3==1: Abwahl von FXS für Y angefordert
(\$AA_FXS[Y]==1)) DO	; \$AA_FXS[Y]==1 (Anschlag erfolgreich angefahren): =>
FXS[Y]=0	; FXS abwählen
FA[Y]=1000 POS[Y]=0	; Positionierbewegung Y
N20 FXS[Y]=0 G0 G90 X0 Y0	; Grundstellungen einstellen: FXS abgewählt,
	; X und Y auf Ausgangsposition
N30 RELEASE(Y)	; Y für Bewegungen in Synchronaktionen freigeben

Programmcode	Kommentar
...	
N60 GET(Y)	Achse Y wieder in den Bahnverbund aufnehmen
...	

Hinweis**Vermeidung von Mehrfachanwahl von FXS**

Zur Vermeidung einer Mehrfachanwahl wird empfohlen vor der Aktivierung von FXS entweder die Systemvariable \$AA_FXS==0 oder einen anwenderspezifischen Merker abzufragen. Siehe oben, Programmbeispiel N10

Beispiel 2: Fahren auf Festanschlag mit satzbezogener Synchronaktionen

"Fahren auf Festanschlag" wird ab einer bestimmten Position der Verfahrbewegung des nachfolgenden Satzes aktiviert

Programmcode	Kommentar
N10 G0 G90 X0	; Ausgangsposition
N20 WHEN \$AA_IW[X]>17 DO FXS[X]=1	; Synchronaktion: Istposition der Achse X > 17
	; => FXS für Achse X aktivieren
N30 G1 F200 X100	; Verfahrbewegung Achse X

Hinweis

Eine satzbezogene Synchronaktion wird im nachfolgenden Hauptlaufsatz bearbeitet.

7.4 Datenlisten**7.4.1 Maschinendaten****7.4.1.1 Achs-/Spindel-spezifische Maschinendaten**

Nummer	Bezeichner: \$MA_	Beschreibung
36042	FOC_STANDSTILL_DELAY_TIME	Verzögerungszeit Stillstandsüberwachung bei aktiver Momenten-/Kraftbegrenzung
37000	FIXED_STOP_MODE	Modus Fahren auf Festanschlag
37002	FIXED_STOP_CONTROL	Ablaufkontrolle für Fahren auf Festanschlag
37010	FIXED_STOP_TORQUE_DEF	Voreinstellung Festanschlag-Klemmmoment

Nummer	Bezeichner: \$MA_	Beschreibung
37012	FIXED_STOP_TORQUE_RAMP_TIME	Zeitdauer bis zum Erreichen der geänderten Momentengrenze
37020	FIXED_STOP_WINDOW_DEF	Voreinstellung für Festanschlags-Überwachungsfenster
37030	FIXED_STOP_THRESHOLD	Schwelle für Festanschlagserkennung
37040	FIXED_STOP_BY_SENSOR	Festanschlagserkennung über Sensor
37050	FIXED_STOP_ALARM_MASK	Freigabe der Festanschlagsalarme
37052	FIXED_STOP_ALARM_REACTION	Reaktion bei Festanschlagsalarmen
37060	FIXED_STOP_ACKN_MASK	Beachtung von PLC-Quittierungen für Fahren auf Festanschlag
37070	FIXED_STOP_ANA_TORQUE	Momentengrenze beim Anfahren des Festanschlags für analoge Antriebe
37080	FOC_ACTIVATION_MODE	Grundstellung der modalen Momenten-/Kraftbegrenzung

7.4.2 Settingdaten

7.4.2.1 Achs-/Spindel-spezifische Settingdaten

Nummer	Bezeichner: \$SA_	Beschreibung
43500	FIXED_STOP_SWITCH	Anwahl Fahren auf Festanschlag
43510	FIXED_STOP_WINDOW	Festanschlagsklemmmoment
43520	FIXED_STOP_TORQUE	Festanschlags-Überwachungsfenster

7.4.3 Signale

7.4.3.1 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Festanschlag erreicht quittieren	DB31,DBX1.1	DB380x.DBX1.1
Sensor Festanschlag	DB31,DBX1.2	DB380x.DBX1.2
Achsen-/Spindelsperre	DB31,DBX1.3	DB380x.DBX1.3
Reglerfreigabe	DB31,DBX2.1	DB380x.DBX2.1
Fahren auf Festanschlag freigegeben	DB31,DBX3.1	DB380x.DBX3.1

7.4.3.2 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Fahren auf Festanschlag aktivieren	DB31,DBX62.4	DB390x.DBX2.4
Festanschlag erreicht	DB31,DBX62.5	DB390x.DBX2.5

G2: Geschwindigkeiten, Soll-/Istwertsysteme, Regelung

8

8.1 Kurzbeschreibung

Diese Funktionsbeschreibung beschreibt die Parametrierung einer Maschinenachse bezüglich:

- Istwert- bzw. Messsysteme
- Sollwertsystem
- Bediengenauigkeit
- Fahrbereiche
- Achsgeschwindigkeiten
- Regelparameter

8.2 Geschwindigkeiten, Verfahrbereiche, Genauigkeiten

8.2.1 Geschwindigkeiten

Maximale Bahn-, Achsgeschwindigkeit und Spindeldrehzahl

Die maximale Bahn-, Achsgeschwindigkeit und Spindeldrehzahl wird beeinflusst durch die Maschinenkonstruktion, Antriebsdynamikauslegung und die Grenzfrequenz der Istwerterfassung (Gebergrenzfrequenz).

Die maximale Achsgeschwindigkeit wird in dem Maschinendatum:

MD32000 \$MA_MAX_AX_VELO (Maximale Achsgeschwindigkeit)
definiert.

Die maximal zulässige Spindeldrehzahl wird über das Maschinendatum:

MD35100 \$MA_SPIND_VELO_LIMIT (Maximale Spindeldrehzahl)
vorgegeben.

Erläuterungen siehe Kapitel "S1: Spindeln (Seite 1295)".

Bei einem höheren (resultierend aus programmierten und über Vorschub-Override beeinflussten) Vorschub wird auf V_{\max} begrenzt.

Diese automatische Vorschubbegrenzung kann bei von CAD-Systemen generierten Programmen, die extrem kurze Sätze enthalten, zu einer Absenkung der Geschwindigkeit über mehrere Sätze führen.

Beispiel:

IPO-Takt = 12 ms

N10 G0 X0 Y0; [mm]

N20 G0 X100 Y100; [mm]

⇒ programmierte Weglänge im Satz = 141,42 mm

⇒ $V_{\max} = (141,42 \text{ mm} / 12 \text{ ms}) \cdot 0,9 = 10606,6 \text{ mm/s} = 636,39 \text{ m/min}$ **Minimale Bahn-, Achsgeschwindigkeit**

Für die minimale Bahn- oder Achsgeschwindigkeit gilt folgende Einschränkung:

$$V_{\min} \geq \frac{10^{-3}}{\text{Rechenfeinheit} \left[\frac{\text{Inkr.}}{\text{mm oder Grad}} \right] \cdot \text{IPO-Takt [s]}}$$

Die Rechenfeinheit wird mit dem Maschinendatum:

MD10200 \$MN_INT_INCR_PER_MM (Rechenfeinheit für Linearpositionen)

bzw.

MD10210 \$MN_INT_INCR_PER_DEG (Rechenfeinheit für Winkelpositionen)

definiert.:

Bei Unterschreitung von V_{\min} erfolgt keine Verfahrbewegung.**Beispiel:**

MD10200 \$MN_INT_INCR_PER_MM = 1000 [Inkr. / mm] ;

IPO-Takt = 12 ms;

⇒ $V_{\min} = 10^{-3} / (1000 \frac{\text{Inkr.}}{\text{mm}} \times 12 \text{ ms}) = 0,005 \text{ mm} / \text{min}$;

Der Wertebereich der Vorschübe ist abhängig von der gewählten Rechenfeinheit.

Bei der Standardvorbelegung des Maschinendatums:

MD10200 \$MN_INT_INCR_PER_MM

(Rechenfeinheit für Linearpositionen) (1000 Inkr./mm)

bzw.

MD10210 \$MN_INT_INCR_PER_DEG

(Rechenfeinheit für Winkelpositionen) (1000 Inkr./Grad)

kann folgender Wertebereich mit der angegebenen Auflösung programmiert werden:

Wertebereich für Bahnvorschub F bei Geometrieachsen:	
metrisches System:	inch-System:
$0,001 \leq F \leq 999.999,999$ [mm/min, mm/U, Grad/min, Grad/U]	$0,001 \leq F \leq 399.999,999$ [inch/min, inch/U]

Wertebereich für Vorschub bei Positionierachsen:	
metrisches System:	inch-System:
$0,001 \leq FA \leq 999.999,999$ [mm/min, mm/U, Grad/min, Grad/U]	$0,001 \leq FA \leq 399.999,999$ [inch/min, inch/U]

Wertebereich für Spindeldrehzahl S:	$0,001 \leq S \leq 999.999,999$ [U/min]
-------------------------------------	---

Wird die Rechenfeinheit um einen Faktor erhöht/erniedrigt, so ändern sich die Wertebereiche entsprechend.

8.2.2 Verfahrbereiche

Wertebereich der Verfahrbereiche

Der Wertebereich der Verfahrbereiche ist abhängig von der gewählten Rechenfeinheit.

Bei der Standardbelegung des Maschinendatums:

MD10200 \$MN_INT_INCR_PER_MM

(Rechenfeinheit für Linearpositionen) (1000 Inkr./mm)

bzw.

MD10210 \$MN_INT_INCR_PER_DEG

(Rechenfeinheit für Winkelpositionen) (1000 Inkr./Grad)

kann folgender Wertebereich mit der angegebenen Auflösung programmiert werden:

Tabelle 8-1 Verfahrbereiche der Achsen

	G71 [mm, Grad]	G70 [inch, Grad]
	Bereich	Bereich
Linearachsen X, Y, Z, ...	± 999.999,999	± 399.999,999
Rundachsen A, B, C, ...	± 999.999,999	± 999.999,999
Interpolationsparameter I, J, K	± 999.999,999	± 399.999,999

Rundachsen tragen immer die Maßeinheit Grad.

Wird die Rechenfeinheit um den Faktor 10 erhöht/erniedrigt, so ändern sich die Wertebereiche entsprechend.

Der Verfahrbereich kann durch Software-Endschalter und Arbeitsbereiche begrenzt werden (siehe Kapitel "A3: Achsüberwachungen (Seite 83)").

Besonderheiten bei großem Verfahrbereich für Linear- und Rundachsen siehe Kapitel "R1: Referenzieren (Seite 1243)".

Der Verfahrbereich für Rundachsen kann über Maschinendaten beschränkt werden.

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Rundachsen (R2)

8.2.3 Positioniergenauigkeit der Steuerung

Istwertauflösung und Rechenfeinheit

Die Positioniergenauigkeit der Steuerung ist abhängig von der Istwertauflösung (= Geberinkremente / (mm oder Grad)) und der Rechenfeinheit (= interne Inkremente / (mm oder Grad)).

Die gröbere Auflösung der beiden bestimmt die Positioniergenauigkeit der Steuerung.

Die Wahl der Eingabefeinheit, des Interpolator- und Lageregeltaktes haben keinen Einfluss auf diese Genauigkeit.

Neben der Begrenzung durch MD32000 begrenzt die Steuerung situationsbedingt die maximale Bahngeschwindigkeit nach folgender Formel:

$$\frac{\text{Interne Inkremente / mm}}{\text{Geberinkremente / mm}} = \frac{1}{\text{ENC_RESOL [n]} * \text{ENC_PULSE_MULT[n]} * \frac{\text{DRIVE_ENC_RATIO_NUMERA [n]}}{\text{DRIVE_ENC_RATIO_DENOM [n]}} * \frac{\text{DRIVE_AX_RATIO_DENOM [n]}}{\text{DRIVE_AX_RATIO_NUMERA [n]}} * \text{LEADSCREW_PITCH} * \text{INT_INCR_PER_MM}}$$

8.2.4 Eingabe-/Anzeigeinheit, Rechenfeinheit

Feinheiten: Unterschiede

Bei den Feinheiten, d. h. der Auflösung von Linear- und Winkelpositionen, Geschwindigkeiten, Beschleunigungen und Ruck, ist zu unterscheiden zwischen:

- Eingabefeinheit
Eingabe von Daten über die Bedientafelfront oder über Teileprogramme.
- Anzeigeinheit
Anzeige von Daten über die Bedientafelfront.
- Rechenfeinheit
Interne Darstellung der über Bedientafelfront oder Teileprogramm eingegebenen Daten.

Die Eingabe- und Anzeigefeinheit wird durch die verwendete Bedientafelfront vorgegeben, wobei die Anzeigefeinheit für Positionswerte mit dem Maschinendatum:

MD9004 \$MM_DISPLAY_RESOLUTION (Anzeigefeinheit)

veränderbar ist.

Über das Maschinendatum:

MD9011 \$MM_DISPLAY_RESOLUTION_INCH (Anzeigefeinheit INCH-Maßsystem)

kann die Anzeigefeinheit für Positionswerte bei Inch-Einstellung projiziert werden.

Damit ist es möglich, bei Inch-Einstellung bis zu sechs Nachkommastellen anzeigen zu können.

Für die Programmierung in Teileprogrammen gelten die in der Programmieranleitung aufgeführten Eingabefeinheiten.

Die gewünschte Rechenfeinheit wird mit den Maschinendaten:

MD10200 \$MN_INT_INCR_PER_MM (Rechenfeinheit für Linearpositionen)

und

MD10210 \$MN_INT_INCR_PER_DEG (Rechenfeinheit für Winkelpositionen)

festgelegt.

Sie ist unabhängig von der Eingabe-/Anzeigefeinheit, sollte aber mindestens die gleiche Auflösung haben.

Mit der Rechenfeinheit wird die Anzahl der max. wirksamen Nachkommastellen für Positionswerte, Geschwindigkeiten usw. im Teileprogramm, sowie die Anzahl der Nachkommastellen für Werkzeugkorrekturen, Nullpunktverschiebungen etc. (und somit auch für die max. erreichbare Genauigkeit) festgelegt.

Die Genauigkeit der Eingabe von Winkel- und Linear-Positionen wird auf die Rechenfeinheit begrenzt, indem das Produkt des programmierten Wertes mit der Rechenfeinheit auf eine ganze Zahl gerundet wird.

Um die ausgeführte Rundung leicht nachvollziehbar zu halten, ist es sinnvoll, für die Rechenfeinheit 10er-Potenzen zu verwenden.

Beispiel für Rundung:

Rechenfeinheit : 1000 Inkremente / mm

Programmierter Weg : 97,3786 mm

Wirksamer Wert = 97,379 mm

Beispiel für Programmierung im $1/10$ -µm-Bereich:

Alle Linearachsen einer Maschine sollen im Wertebereich 0,1 ... 1000 µm programmiert und verfahren werden.

⇒ Um auf 0,1 µm genau zu positionieren, muss die Rechenfeinheit auf $\geq 10^4$ Inkr./mm gesetzt werden.

⇒ MD10200 \$MN_INT_INCR_PER_MM = 10000 [Inkr./mm]:

⇒ Beispiel für zugehöriges Teileprogramm:

Programmcode	Kommentar
N20 G0 X 1.0000 Y 1.0000	; Achsen fahren auf die Position X=1.0000 mm, Y=1.0000 mm
N25 G0 X 5.0002 Y 2.0003	; Achsen fahren auf die Position =5.0002 mm, Y=2.0003 mm

8.2.5 Normierung physikalischer Größen der Maschinen- und Settingdaten

Ein-/Ausgabeeinheiten

Maschinen- und Settingdaten mit einer physikalische Einheit, werden je nach Grundsystem (metrisch/inch) standardmäßig in folgenden Ein-/Ausgabeeinheiten interpretiert:

Physikalische Größe	Ein- /Ausgabeeinheiten für Standardgrundsystem	
	Metrisch	Inch
Linear-Position	1 mm	1 inch
Winkel-Position	1 Grad	1 Grad
Linear-Geschwindigkeit	1 mm/min	1 inch/min
Winkel-Geschwindigkeit	1 Umdr./min	1 Umdr./min
Linear-Beschleunigung	1 m/s ²	1 inch/s ²
Winkel-Beschleunigung	1 Umdr./s ²	1 Umdr./s ²
Linear-Ruck	1 m/s ³	1 inch/s ³
Winkel-Ruck	1 Umdr./s ³	1 Umdr./s ³
Zeit	1 s	1 s
Lageregler-Kreisverstärkung	1/s	1/s
Umdrehungsvorschub	1 mm/Umdr.	inch/Umdr.
Kompensationswert Linear-Position	1 mm	1 inch
Kompensationswert Winkel-Position	1 Grad	1 Grad

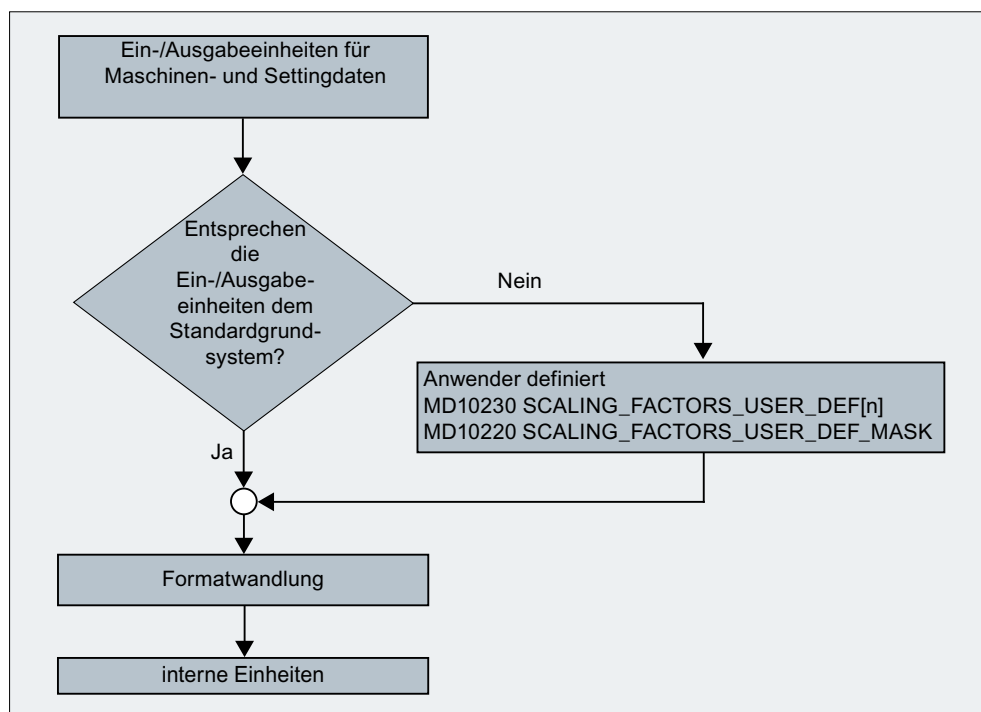
Steuerungsintern werden, unabhängig von dem gewählten Grundsystem, folgende Einheiten verwendet:

Physikalische Größe	Einheit
Linear-Position	1 mm
Winkel-Position	1 Grad
Linear-Geschwindigkeit	1 mm/s
Winkel-Geschwindigkeit	1 Grad/s
Linear-Beschleunigung	1 mm/s ²
Winkel-Beschleunigung	1 Grad/s ²
Linear-Ruck	1 mm/s ³
Winkel-Ruck	1 Grad/s ³
Zeit	1 s

Physikalische Größe	Einheit
Lageregler-Kreisverstärkung	1/s
Umdrehungsvorschub	1 mm/Grad
Kompensationswert Linear-Position	1 mm
Kompensationswert Winkel-Position	1 Grad

Der Anwender hat die Möglichkeit, andere Ein-/Ausgabeeinheiten für Maschinen- und Settingdaten zu definieren. Dazu muss über folgende Maschinendaten eine Anpassung zwischen den neu gewählten Ein-/Ausgabeeinheiten und den internen Einheiten erfolgen:

- MD10220 \$MN_SCALING_USER_DEF_MASK
- MD10230 \$MN_SCALING_FACTORS_USER_DEF[n]



Dabei gilt:

Gewählte Ein-/Ausgabeeinheit = (MD10230 \$MN_SCALING_FACTORS_USER_DEF[n]) * interne Einheit

In das Maschinendatum MD10230 \$MN_SCALING_FACTORS_USER_DEF[n] ist somit jeweils die gewählte Ein-/Ausgabeeinheit, ausgedrückt in den internen Einheiten 1 mm, 1 Grad und 1 s, einzugeben.

Beispiel 1:

Die Maschinendaten-Ein-/Ausgabe von Lineargeschwindigkeiten soll statt in mm/min (Grundstellung) in m/min erfolgen.

(Die interne Einheit ist mm/s)

⇒ Der Normierungsfaktor für Lineargeschwindigkeiten soll von der Standardeinstellung abweichen. Dazu muss im Maschinendatum:

MD10220 \$MN_SCALING_USER_DEF_MASK

das Bit Nummer 2 gesetzt werden.

⇒ MD10220 \$MN_SCALING_USER_DEF_MASK = 'H4'; (Bit-Nr. 2 als Hex-Wert)

⇒ Der Normierungsfaktor für Lineargeschwindigkeiten soll von der Standardeinstellung abweichen. Dazu muss im Maschinendatum:

MD10220 \$MN_SCALING_USER_DEF_MASK

das Bit Nummer 2 gesetzt werden.

⇒ MD10220 \$MN_SCALING_USER_DEF_MASK = 'H4'; (Bit-Nr. 2 als Hex-Wert)

⇒ Der Normierungsfaktor errechnet sich nach folgender Formel:

$$\text{MD10230 SCALING_FACTORS_USER_DEF}[n] = \frac{\text{Gewählte Ein-/Ausgabeeinheit}}{\text{Interne Einheit}}$$

$$\text{MD10230 SCALING_FACTORS_USER_DEF}[n] = \frac{1 \frac{\text{m}}{\text{min}}}{1 \frac{\text{mm}}{\text{s}}} = \frac{\frac{1000 \text{ mm}}{60 \text{ s}}}{1 \frac{\text{mm}}{\text{s}}} = \frac{1000}{60} = 16,667;$$

⇒ MD10230 SCALING_FACTORS_USER_DEF[n] = 16,667

Der Index n spezifiziert in der Liste der "Normierungsfaktoren der physikalischen Größen" die "Linear-Geschwindigkeit".

Beispiel 2:

Zusätzlich zu der Änderung von Beispiel 1 soll die Maschinendaten-Ein-/Ausgabe von Linear-Beschleunigungen statt in m/s² (Grundstellung) in ft/s² erfolgen.

(Die interne Einheit ist mm/s²)

⇒ MD10220 SCALING_USER_DEF_MASK = H14 ; (Bit-Nr. 4 und Bit-Nr. 2 von Beispiel 1 als Hex-Wert)

$$\Rightarrow \text{MD10220 SCALING_FACTORS_USER_DEF}[4] = \frac{1 \frac{\text{ft}}{\text{s}^2}}{1 \frac{\text{mm}}{\text{s}^2}} = \frac{12 \cdot 25,4 \frac{\text{mm}}{\text{s}^2}}{1 \frac{\text{mm}}{\text{s}^2}} = 304,8 ;$$

⇒ MD10220 SCALING_FACTORS_USER_DEF[4] = 304,8

Der Index 4 spezifiziert in der Liste der "Normierungsfaktoren der physikalischen Größen" die "Linear-Beschleunigung".

8.3 Maßsystem inch/metrisch

8.3.1 Funktion

8.3.1.1 Parametriertes und Programmiertes Maßsystem

Eine SINUMERIK-Steuerung kann sowohl mit metrischen Maßangaben als auch mit Inch-Maßangaben arbeiten.

Parametriertes Maßsystem (Grundsystem)

Die Grundstellung des Maßsystems (Grundsystem) wird eingestellt im Maschinendatum MD10240 \$MN_SCALING_SYSTEM_IS_METRIC (siehe "Inbetriebnahme (Seite 353)").

Entsprechend des Grundsystems werden alle längenbehafteten Daten als metrische oder Inch-Maßangaben interpretiert:

Metrisches Maßsystem:	mm, mm/min, m/s ² , m/s ³ , mm/Umdr.
Inch-Maßsystem:	inch, inch/min, inch/s ² , inch/s ³ , inch/Umdr.

Das Grundsystem legt auch die Interpretation des programmierten F-Werts für Linearachsen fest:

Vorschubtyp	Metrisches Maßsystem	Inch-Maßsystem
G94	mm/min	inch/min
G95	mm/Umdr.	inch/Umdr.

Programmiertes Maßsystem

Mit den Befehlen der G-Gruppe 13 (Maßsystem inch/metrisch) kann innerhalb des Teileprogramms zwischen dem metrischen Maßsystem und dem Inch-Maßsystem umgeschaltet werden (siehe "Programmierung (Seite 357)").

Das programmierte Maßsystem und das Grundsystem können zu jeder Zeit gleich oder ungleich sein. Mit der Umschaltung des Maßsystems in einem Teileprogrammabschnitt kann z. B. bei metrischem Grundsystem an einem Werkstück ein Zoll-Gewinde bearbeitet werden.

Die Löschestellung der G-Gruppe ist festgelegt im Maschinendatum MD20150 \$MC_GCODE_RESET_VALUES (siehe "Inbetriebnahme (Seite 353)").

HMI-Anzeige

Auf der Bedienoberfläche werden längenbehaftete Daten wie folgt angezeigt:

	Anzeige	
	im Grundsystem	im programmierten Maßsystem
Maschinendaten	x	
Daten im Maschinenkoordinatensystem	x	
Werkzeugdaten	x	
Nullpunktverschiebungen	x	
Daten im Werkstückkoordinatensystem		x

NC/PLC-Nahtstelle

Bei NC/PLC-Nahtstellensignalen, die maßabhängige Informationen enthalten, z. B. Vorschub für Bahn- und Positionierachsen, erfolgt der Datenaustausch mit der PLC im Grundsystem.

Einlesen von externen Teileprogrammen

Werden Teileprogramme inklusive Datensätze (Nullpunktverschiebungen, Werkzeugkorrekturen, ...), die in einem vom Grundsystem abweichenden Maßsystem programmiert wurden, von Extern eingelesen, muss vorher das Grundsystem geändert werden.

8.3.1.2 Erweiterte Maßsystem-Funktionalität

Ab SW 5 ist eine erweiterte Maßsystem-Funktionalität verfügbar, die die Umschaltung des Maßsystems erheblich vereinfacht.

Neue Funktionen sind:

- Maßsystem-Umschaltung über die Bedienoberfläche (Seite 351)
- Neue G-Befehle G700/G710 (siehe "Programmierung (Seite 357)")
- Datensicherung mit Maßsystem-Kennung INCH/METRIC (siehe "Inbetriebnahme (Seite 353)")
- Automatische Datenumrechnungen beim Maßsystem-Wechsel, z. B. für:
 - Nullpunktverschiebungen
 - Kompensationsdaten (EEC, QEC)
 - Werkzeugkorrekturen
 - ...

Aus Kompatibilitätsgründen muss die erweiterte Maßsystem-Funktionalität über ein Maschinendatum aktiviert werden (siehe "Inbetriebnahme (Seite 353)").

8.3.1.3 Maßsystem-Umschaltung über die Bedienoberfläche

Voraussetzung

Erweiterte Maßsystem-Funktionalität ist aktiv (MD10260
\$MN_CONVERT_SCALING_SYSTEM = 1; siehe "Inbetriebnahme (Seite 353)")

Funktion

Die Maßsystem-Umschaltung der Steuerung erfolgt über den entsprechenden Softkey der HMI-Bedienoberfläche im Bedienbereich "Maschine".

Die Maßsystem-Umschaltung wird nur unter folgenden Bedingungen durchgeführt:

- MD20110 \$MC_RESET_MODE_MASK, Bit 0, ist in jedem Kanal gesetzt.
- Alle Kanäle sind im Reset-Zustand.
- Achsen werden aktuell nicht über JOG, DRF oder die PLC verfahren.
- Konstante Scheibenumfangsgeschwindigkeit (SUG) ist nicht aktiv.

Für die Dauer der Maßsystem-Umschaltung werden Aktionen wie Teileprogrammstart oder BA-Wechsel gesperrt.

Die eigentliche Umschaltung des Maßsystems wird über das Schreiben aller notwendigen Maschinendaten und anschließendem Wirksamsetzen vorgenommen.

Die folgenden Maschinendaten werden für alle projektierten Kanäle automatisch und konsistent umgeschaltet:

- MD10240 \$MN_SCALING_SYSTEM_IS_METRIC
- MD20150 \$MC_GCODE_RESET_VALUES

Löschstellung

Bei der Umschaltung des Maßsystems über die Bedienoberfläche wird die Löschstellung der G-Gruppe 13 automatisch an das aktuelle Maßsystem angepasst:

MD20150 \$MC_GCODE_RESET_VALUES[12] = <Aktuelles Maßsystem>

NCU-Link

Hinweis

Sind mehrere NCUs durch NCU-Link verbunden, wirkt sich die Umschaltung einheitlich auf alle verbundenen NCUs aus. Sind die Voraussetzungen für eine Umschaltung auf einer der verbundenen NCUs nicht erfüllt, findet die Umschaltung auf keiner der NCUs statt. Es wird davon ausgegangen, dass bei vorhandenem NCU-Link NCU-übergreifende Interpolationen stattfinden sollen, die nur bei einheitlichem Maßsystem korrekte Ergebnisse liefern können.

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kapitel "B3: Dezentrale Systeme"

Systemdaten

Bei der Umschaltung des Maßsystems werden aus Sicht des Bedieners alle längenbehafteten Angaben in das neue Maßsystem automatisch umgerechnet.

Dazu zählen:

- Positionen
- Vorschübe
- Beschleunigungen
- Ruck
- Werkzeugkorrekturen
- Programmierbare, einstellbare und externe Nullpunktverschiebungen, DRF-Verschiebungen
- Kompensationswerte
- Schutzbereiche
- Maschinendaten
- JOG und Handradbewertungen

Nach der Umschaltung sind alle o. g. Daten in physikalischen Größen verfügbar.

Daten, für die keine eindeutigen physikalischen Einheiten definiert sind, werden **keiner** automatischen Umrechnung unterzogen.

Dazu zählen:

- R-Parameter
- GUDs (**G**lobal **U**ser **D**ata)
- LUDs (**L**ocal **U**ser **D**ata)
- PUDs (**P**rogram **g**lobal **U**ser **D**ata)
- Analoge Ein-/Ausgänge
- Datenaustausch über den FC21

Hier ist der Anwender gefordert, das aktuell gültige Maßsystem (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) zu berücksichtigen.

NC/PLC-Nahtstelle

Die aktuelle Maßsystem-Einstellung kann an der NC/PLC-Nahtstelle über das folgende Signal gelesen werden:

DB10 DBX107.7 (Inch-Maßsystem)

Die Anzahl an Maßsystem-Umschaltungen seit dem letzten Steuerungshochlauf kann über das folgende Signalbyte ausgelesen werden:

DB10 DBB71 (Änderungszähler Maßsystem inch/metrisch)

Startwert nach Steuerungshochlauf: 1

8.3.2 Inbetriebnahme

NC-spezifische Maschinendaten

Grundsystem

Das von der Steuerung zu verwendende Grundsystem für die Skalierung längenabhängiger physikalischer Größen bei der Daten-Ein-/Ausgabe wird festgelegt über das Maschinendatum:

MD10240 \$MN_SCALING_SYSTEM_IS_METRIC

Wert	Bedeutung
0 (= FALSE)	Längenbehaftete Daten werden als Inch -Maßangaben interpretiert.
1 (= TRUE)	Längenbehaftete Daten werden als metrische Maßangaben interpretiert.

Nach Änderung dieses Maschinendatums ist ein Hochlauf notwendig, da sonst zugehörige Maschinendaten, die physikalische Einheiten besitzen, falsch normiert werden.

Folgendes Vorgehen ist zu beachten:

- MD-Änderung durch manuelle Eingabe:
 - 1. Hochlauf durchführen.
 - 2. Maschinendaten mit physikalischen Einheiten: Wert eingeben.
- MD-Änderung erfolgt über Maschinendatendatei:
 - 1. Hochlauf durchführen.
 - 2. Maschinendatendatei noch einmal laden, damit die neuen physikalischen Einheiten berücksichtigt werden.

Bei Änderung von MD10240 wird Alarm 4070 "Normierendes Maschinendatum geändert" gemeldet.

Hinweis

Bei einer Umstellung des Maßsystems an einer Steuerung müssen alle längenbehafteten Daten konsistent und vollständig in das andere Maßsystem umgerechnet werden.

Umrechnungsfaktor (NC-Spezifisch)

Der Faktor für die Umrechnung vom metrischen in das Inch-Maßsystem wird eingestellt im Maschinendatum:

MD10250 \$MN_SCALING_VALUE_INCH (Umrechnungsfaktor für Inch)

Standardwert: 25,4

Der Umrechnungsfaktor wird bei Anwahl des nicht-metrischen Grundsystems (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC = 0) wirksam.

Folgende Daten werden mit dem Umrechnungsfaktor multipliziert:

- Längenbehaftete Daten bei der Ein-/Ausgabe (z. B. beim Upload von Maschinendaten, Nullpunktverschiebungen)
- Programmierte F-Werte für Linearachsen
- Programmierte Geometrie einer Achse (Position, Polynomkoeffizienten, Radius bei Kreisprogrammierung, ...), wenn das mit G70/G71 programmierte Maßsystem vom Grundsystem (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) abweicht.

Hinweis

Durch Änderung des Umrechnungsfaktors kann die Steuerung an kundenspezifische Maßsysteme angepasst werden.

Erweiterte Maßsystem-Funktionalität

Die erweiterte Maßsystem-Funktionalität kann über das folgende Kompatibilitätsmaschinendatum aktiviert werden:

MD10260 \$MN_CONVERT_SCALING_SYSTEM

Wert	Bedeutung
0 (= FALSE)	Erweiterte Maßsystem-Funktionalität nicht aktiv (Standardeinstellung; Kompatibilität zu älteren Softwareständen)
1 (= TRUE)	Erweiterte Maßsystem-Funktionalität aktiv

Bei Änderung von MD10240 wird Alarm 4070 "Normierendes Maschinendatum geändert" gemeldet.

Datensicherung

Einzelne Datensätze, die aus der Steuerung gelesen werden und die über maßsystemrelevante Daten verfügen, erhalten beim Lesevorgang eine Kennung entsprechend der aktuellen Maßsystemeinstellung:

- MD10260 \$MN_CONVERT_SCALING_SYSTEM
- MD10240 \$MN_SCALING_SYSTEM_IS_METRIC

Über die Kennung wird festgehalten, in welchem Maßsystem die Daten ausgelesen wurden. Dadurch wird sichergestellt, dass keine Datensätze mit einem zur aktuellen Einstellung abweichenden Maßsystem in die Steuerung eingelesen werden.

Da die Kennung auch in Teileprogrammen ausgewertet wird, können auch diese auf die o. g. Art und Weise gegen Fehlbedienung "abgesichert" werden. Man kann damit verhindern, dass Teileprogramme, die z. B. nur metrische Angaben enthalten, in einem Inch-Maßsystem ablaufen können.

Archive und Maschinendatensätze sind bei folgender Einstellung abwärtskompatibel:

MD11220 \$MN_INI_FILE_MODE = 2

Hinweis

Die INCH/METRIC-Kennung wird nur beim gesetzten Kompatibilitätsmaschinendatum generiert:

MD10260 \$MN_CONVERT_SCALING_SYSTEM = TRUE

Hinweis**Runden von Maschinendaten**

Um Rundungsproblemen vorzubeugen, werden alle längenbehafteten Maschinendaten beim Schreiben im Inch-Maßsystem (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC=0 und MD10260 \$MN_CONVERT_SCALING_SYSTEM=1) auf 1 µm gerundet.

Dadurch werden Genauigkeitsverluste, die durch die Wandlung nach ASCII beim Auslesen einer Datensicherung im Inch-System entstehen, beim Einlesen wieder korrigiert.

Eingabe- und Rechenfeinheit

Die Eingabe- und Rechenfeinheit wird als interne Anzahl von Inkrementen pro Millimeter festgelegt.

MD10200 \$MN_INT_INCR_PER_MM

Die Genauigkeit der Eingabe von Linearpositionen wird auf die Rechenfeinheit begrenzt. Dabei wird das Produkt aus programmiertem Positionswert und Rechenfeinheit auf die nächste Ganzzahl gerundet. Um die Rundung leicht nachvollziehbar zu halten, wird empfohlen, für die Rechenfeinheit 10er-Potenzen zu verwenden.

Beispiel:

1 Inch = 25,4 mm \Rightarrow 0,0001 Inch = 0,00254 mm = 2,54 µm = 2540 nm

Um die letzten 40 nm noch programmieren und darstellen zu können, muss für die Eingabe- und Rechenfeinheit ein Wert von 100000 parametrisiert werden.

Erst mit dieser für beide Maßsysteme gleichen Einstellung können Maßsystemumschaltungen ohne einen nennenswerten Genauigkeitsverlust realisiert werden. Einmal so eingestellt muss das MD10200 nicht mehr bei jeder Maßsystemumschaltung verändert werden.

Maßsystem der Positionstabellen

Die Projektierung des Maßsystems für Positionsangaben der Teilungsachstabellen und der Schaltpunkte für Softwaresnocken erfolgt über das Maschinendatum:

MD10270 \$MN_POS_TAB_SCALING_SYSTEM

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kapitel "N3: Softwaresnocken, Wegschaltsignale" und "T1: Teilungsachsen"

Anwenderwerkzeugdaten

Für anwenderdefinierte Werkzeug- und Werkzeugschneidendaten können die physikalischen Einheiten über folgende Maschinendaten eingestellt werden:

- MD10290 \$MN_CC_TDA_PARAM_UNIT
- MD10292 \$MN_CC_TOA_PARAM_UNIT

Hinweis

Beim Umschalten des Maßsystems werden alle längenbehafteten Werkzeugdaten in das neue Maßsystem umgerechnet.

Kanal-spezifische Maschinendaten

Löschstellung

Die Löschstellung der G-Gruppe 13 (Maßsystem inch/metrisch) wird für jeden Kanal eingestellt im Maschinendatum:

MD20150 \$MC_GCODE_RESET_VALUES[12] (Löschstellung der G-Gruppe 13)

Achs-/Spindel-spezifische Maschinendaten

Umrechnungsfaktor (achsspezifisch)

Um reine Positionierachsen von G70/G71 unabhängig zu machen, kann der Faktor für die Umrechnung vom metrischen in das Inch-Maßsystem für jede Achse individuell eingestellt werden:

MD31200 \$MA_SCALING_FACTOR_G70_G71 (Faktor für die Umrechnung bei aktivem G70/G71)

Standardwert: 25,4

Hinweis

Der Umrechnungsfaktor sollte für alle drei Geometrieachsen gleich sein.

JOG und Handradbewertung

Das folgende Maschinendatum besteht aus zwei Werten, die achsspezifische Inkrementbewertungen für jedes der beiden Maßsysteme beinhaltet:

MD31090 \$MA_JOG_INCR_WEIGHT

In Abhängigkeit von der aktuellen Einstellung in MD10240 \$MN_SCALING_SYSTEM_IS_METRIC wählt die Steuerung automatisch den passenden Wert aus.

Bereits zum Zeitpunkt der Inbetriebnahme legt der Anwender beide Inkrementbewertungen fest.

Beispiel: Inkrementbewertungen für die 1. Achse

- Metrisch:
MD31090 \$MA_JOG_INCR_WEIGHT[0 ; AX1] = 0,001 mm
- Inch:
MD31090 \$MA_JOG_INCR_WEIGHT[1 ; AX1] = 0,00254 mm \pm 0,0001 inch

Damit muss nicht mehr bei jeder Inch/Metrisch-Umschaltung MD31090 geschrieben werden.

Ein inkrementelles Verfahren über JOG akkumuliert bei wechselnden Maßsystemen keine Wegreste, da alle internen Positionen sich immer auf mm beziehen.

Maßsystem für Durchhangkompensation

Die Projektierung des Maßsystems für Durchhangkompensation erfolgt über das Maschinendatum:

MD32711 \$MA_CEC_SCALING_SYSTEM_METRIC

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kapitel "K3: Kompensationen"

8.3.3 Programmierung

8.3.3.1 Maßsystem umschalten (G70/G71/G700/G710)

Mit den Befehlen der G-Gruppe 13 (Maßsystem inch/metrisch) kann innerhalb des Teileprogramms zwischen dem metrischen Maßsystem und dem Inch-Maßsystem umgeschaltet werden.

Syntax

```
G70/G700
...
G71/G710
```

Bedeutung

G70:	<p>Einschalten des Inch-Maßsystems</p> <p>Längenbehaftete geometrische Daten werden im Inch-Maßsystem gelesen und geschrieben.</p> <p>Längenbehaftete technologische Daten (z. B. Vorschübe, Werkzeugkorrekturen, einstellbare Nullpunktverschiebungen, Maschinendaten und Systemvariablen) werden im parametrisierten Grundsystem gelesen und geschrieben.</p>
G71:	<p>Einschalten des metrischen Maßsystems</p> <p>Längenbehaftete geometrische Daten werden im metrischen Maßsystem gelesen und geschrieben.</p> <p>Längenbehaftete technologische Daten werden im parametrisierten Grundsystem gelesen und geschrieben.</p>

8.3 Maßsystem inch/metrisch

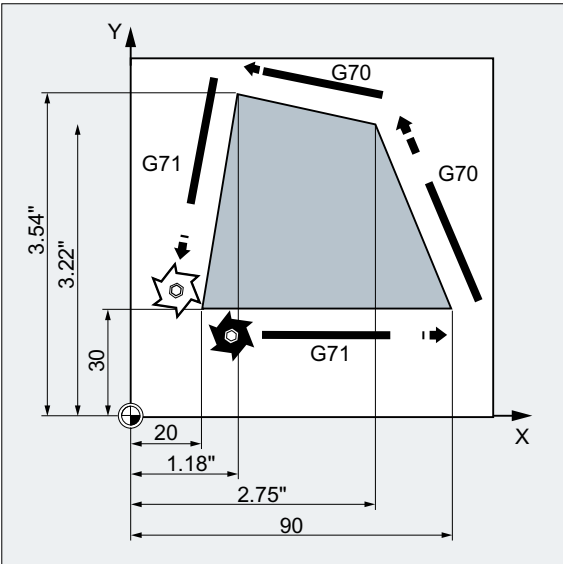
G700:	Einschalten des Inch-Maßsystems Alle längenbehafteten geometrischen und technologischen Daten werden im Inch-Maßsystem gelesen und geschrieben.
G710:	Einschalten des metrischen Maßsystems Alle längenbehafteten geometrischen und technologischen Daten werden im metrischen Maßsystem gelesen und geschrieben.

Hinweis
G700/G710

Die Befehle G700 und G710 sind nur verfügbar, wenn die erweiterte Maßsystem-Funktionalität aktiv ist (MD10260 \$MN_CONVERT_SCALING_SYSTEM = 1).

Beispiel

Das Grundsystem ist metrisch (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC = 1). Die Werkstückzeichnung enthält jedoch Maßangaben in inch. Es wird deshalb innerhalb des Teileprogramms auf das Inch-Maßsystem umgeschaltet. Nach den inch-Maßangaben wird wieder zum metrischen Maßsystem gewechselt.



Programmcode	Kommentar
N10 G0 G90 X20 Y30 Z2 S2000 M3 T1	; X=20 mm, Y=30 mm, Z=2 mm, F=Eilgang mm/min
N20 G1 Z-5 F500	; Z=-5 mm, F=500 mm/min
N30 X90	; X=90 mm
N40 G70 X2.75 Y3.22	; Programmiertes Maßsystem: inch ; X=2.75 inch, Y=3.22 inch, F=500 mm/min
N50 X1.18 Y3.54	; X=1.18 inch, Y=3.54 inch, F=500 mm/min
N60 G71 X20 Y30	; Programmiertes Maßsystem: metrisch

Programmcode	Kommentar
N70 G0 Z2	; X=20 mm, Y=30 mm, F=500 mm/min
N80 M30	; Z=2 mm, F=Eilgang mm/min
	; Programmende

Weitere Informationen

Lesen und Schreiben von Daten bei G70/G71 und G700/G710

Datenbereich	G70 / G71		G700 / G710	
	Lesen	Schreiben	Lesen	Schreiben
Anzeigen, Nachkommastellen (WKS)	P	P	P	P
Anzeigen, Nachkommastellen (MKS)	G	G	G	G
Vorschübe	G	G	P	P
Positionsangaben X, Y, Z	P	P	P	P
Interpolationsparameter I, J, K	P	P	P	P
Kreisradius (CR)	P	P	P	P
Polarradius (RP)	P	P	P	P
Gewindesteigung	P	P	P	P
Programmierbarer FRAME	P	P	P	P
Einstellbare FRAMES	G	G	P	P
Basisframes	G	G	P	P
Externe Nullpunktverschiebungen	G	G	P	P
axiale Preset-Verschiebung	G	G	P	P
Arbeitsfeldbegrenzungen (G25/G26)	G	G	P	P
Schutzbereiche	P	P	P	P
Werkzeugkorrekturen	G	G	P	P
Längenbehaftete Maschinendaten	G	G	P	P
Längenbehaftete Settingdaten	G	G	P	P
Längenbehaftete Systemvariablen	G	G	P	P
GUD's	G	G	G	G
LUD's	G	G	G	G
PUD's	G	G	G	G
R-Parameter	G	G	G	G
Siemenszyklen	P	P	P	P
Inkrementbewertung JOG/Handrad	G	G	G	G
P: Schreiben/Lesen erfolgt im programmierten Maßsystem				
G: Schreiben/Lesen erfolgt im parametrisierten Grundsystem				

Synchronaktionen

Hinweis

Lesen von Positionsdaten in Synchronaktionen

Ohne explizite Programmierung des Maßsystems in der Synchronaktion (Bedingungsteil und/oder Aktionsteil bzw. Technologiefunktion) werden **längenbehaftete Positionsdaten** in der Synchronaktion immer im parametrisierten **Grundsystem** gelesen.

Literatur: Funktionshandbuch Synchronaktionen

8.4 Soll-/Istwertsystem

8.4.1 Allgemeine Informationen

Regelkreis

Für jede geregelte Achse/Spindel ist ein Regelkreis mit folgendem Aufbau konfigurierbar:

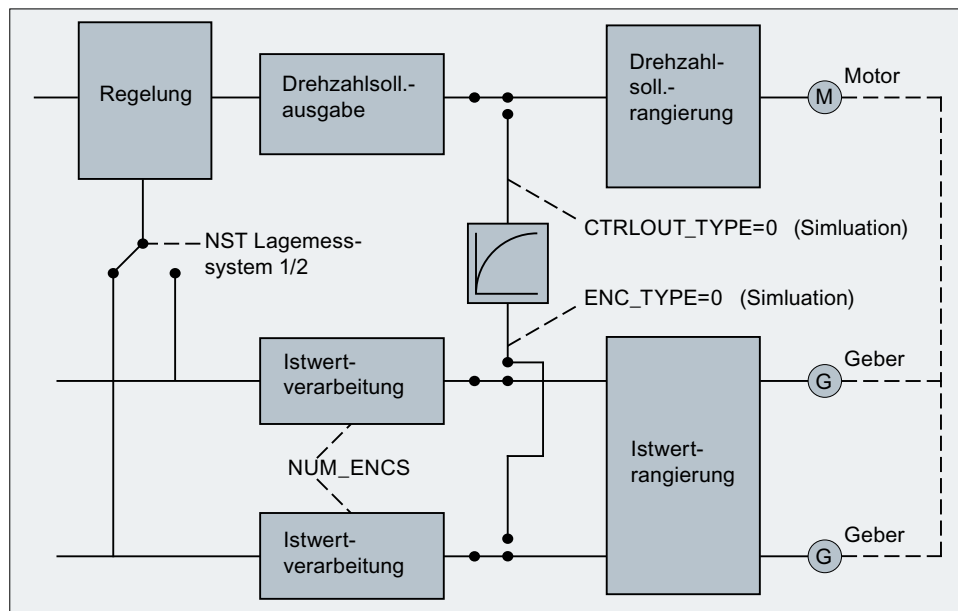


Bild 8-1 Prinzipschaltung eines Regelkreises

Sollwertausgabe

Je Achse/Spindel kann ein Sollwerttelegramm ausgegeben werden. Die Sollwertausgabe an den Steller erfolgt bei SINUMERIK 840D sl.

Istwerterfassung

Je Achse/Spindel können max. zwei Messsysteme angeschlossen werden, z. B. ein direktes Messsystem für den Bearbeitungsprozess mit hoher Anforderung an die Genauigkeit und ein indirektes Messsystem für schnelle Positionieraufgaben.

Die Anzahl der verwendeten Geber wird eingetragen in das Maschinendatum:

MD30200 \$MA_NUM_ENCS (Anzahl der Geber)

Bei zwei vorhandenen Istwertzweigen erfolgt die Istwerterfassung über beide Zweige.

Für die Lageregelung, Absolutwertberechnung und Anzeige wird immer das aktive Messsystem verwendet. Werden von der PLC-Nahtstelle beide gleichzeitig aktiviert, so wird steuerungsintern das Lagemesssystem 1 ausgewählt.

Das Referenzpunktfahren wird mit dem angewählten Messsystem durchgeführt.

Jedes Lagemesssystem muss getrennt referenziert werden.

Erläuterungen zu Geberüberwachungen siehe Kapitel "A3: Achsüberwachungen (Seite 83)".

Erläuterungen zu Kompensationsfunktionen für die Istwerterfassung siehe:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kompensationen (K3)

Umschalten der Messsysteme

Über die folgenden NC/PLC-Nahtstellensignale kann zwischen den beiden Messsystemen umgeschaltet werden:

DB31, ... DBX1.5 (Lagemesssystem 1)

DB31, ... DBX1.6 (Lagemesssystem 2)

Weitere Informationen siehe Kapitel "A2: Diverse NC/PLC-Nahtstellensignale und Funktionen (Seite 39)".

Die Umschaltung zwischen den Messsystemen kann jederzeit geschehen, sie ist nicht vom Stillstand der Achsen abhängig. Die Umschaltung erfolgt nur, wenn eine zulässige Abweichung zwischen den Istwerten der beiden Messsysteme nicht überschritten wird.

Die zugehörige Toleranz wird eingetragen in das Maschinendatum:

MD36500 \$MA_ENC_CHANGE_TOL (Maximale Toleranz bei Lageistwertumschaltung)

Bei der Umschaltung wird die aktuelle Abweichung zwischen Lagemesssystem 1 und 2 sofort verfahren.

Überwachung

Die zulässige Abweichung zwischen den Istwerten der beiden Messsysteme ist einzutragen in das Maschinendatum:

MD36510 \$MA_ENC_DIFF_TOL

Diese Differenz darf beim zyklischen Vergleich der beiden verwendeten Messsysteme nicht überschritten werden, ansonsten wird der Alarm 25105 "Messsysteme laufen auseinander" generiert.

8.4 Soll-/Istwertsystem

Nicht aktiv ist die zugehörige Überwachung bei MD36510 = 0, wenn keine 2 Messsysteme in der Achse aktiv/vorhanden sind bzw. wenn die Achse nicht referenziert ist (zumindest akt. Regelungs-Messsystem).

Arten der Istwerterfassung

Der verwendete Gebertyp muss festgelegt werden über das Maschinendatum:

MD30240 \$MA_ENC_TYPE (Art der Istwerterfassung (Lageistwert))

Simulationsachsen

Zu Testzwecken kann der Drehzahlregelkreis einer Achse simuliert werden.

Die Achse "fährt" dadurch mit Schleppfehler, ähnlich wie eine echte Achse.

Eine Simulationsachse wird definiert, indem die beiden folgenden Maschinendaten auf "0" gesetzt werden:

MD30130 \$MA_CTRL_OUT_TYPE[n] (Ausgabeart des Sollwertes)

MD30240 \$MA_ENC_TYPE[n] (Art der Istwerterfassung)

Nach Laden der Standardmaschinendaten sind die Achsen auf Simulation gesetzt.

Mit Referenzpunktfahren können Soll- und Istwert auf den Referenzpunktwert gesetzt werden.

Über das Maschinendatum:

MD30350 \$MA_SIMU_AX_VDI_OUTPUT (Ausgabe der Achssignale bei Simulationsachsen)

kann festgelegt werden, ob während der Simulation die achsspezifischen Nahtstellensignale an die PLC ausgegeben werden.

Istwertkorrektur

Sollen Istwertkorrekturen, die von der NC auf dem für die Lageregelung ausgewählten Geber vorgenommen werden, nicht den Istwert eines weiteren in der gleichen Achse definierten Gebers beeinflussen, so ist dieser mit dem folgenden Maschinendatum "independent" (unabhängig) zu erklären:

MD30242 \$MA_ENC_IS_INDEPENDENT

Zu den Istwertkorrekturen zählt man:

- Modulo-Behandlung
- Referenzpunktfahren
- Messsystemabgleich
- PRESET

8.4.2 Sollwert- und Geberzuordnung

Sollwert-Rangierung

Für die Sollwert-Zuordnung einer Maschinenachse sind folgende Maschinendaten relevant.

MD30100		\$MA_CTRLOUT_SEGMENT_NR
Sollwertzuordnung: Bussegment		
System	Wert	Bedeutung
840D sl	5	PROFIBUS-DP / PROFINET (Voreinstellung)

MD30110		\$MA_CTRLOUT_MODULE_NR
Sollwertzuordnung: Antriebsnummer / Baugruppennummer		
System	Wert	Bedeutung
840D sl	x	Über die Antriebsnummer wird die logische E/A-Adresse des Antriebs aus MD13050 \$MN_DRIVE_LOGIC_ADDRESS[n] zugeordnet. Die Antriebsnummer (x) ergibt sich aus dem Index (n) von MD13050: $x = n + 1$ Hinweis: Wird der Antrieb simuliert (MD30130 \$MA_CTRLOUT_TYPE = 0), hat das Maschinendatum keine Bedeutung.

MD30120		\$MA_CTRLOUT_NR
Sollwertzuordnung: Sollwertausgang auf Antriebsmodul / Baugruppe		
System	Wert	Bedeutung
840D sl	1	Modularer Antrieb am PROFIBUS / PROFINET mit Profil PROFIdrive (Voreinstellung)

MD30130		\$MA_CTRLOUT_TYPE
Ausgabeart des Sollwerts		
System	Wert	Bedeutung
840D sl	0	Simulation (Betrieb ohne Antrieb)
	1	Sollwertausgang aktiv

Geberzuordnung

Für die Zuordnung der im PROFIdrive-Telegramm übertragenen Geber-Informationen des Antriebs zu den Gebereingängen der Maschinenachse sind folgende Maschinendaten relevant:

MD30210		\$MA_ENC_SEGMENT_NR[n]
Istwertzuordnung Bussegment		
System	Wert	Bedeutung
840D sl	5	PROFIBUS-DP / PROFINET

8.4 Soll-/Istwertsystem

MD30220		\$MA_ENC_MODULE_NR[n]
Istwertzuordnung: Antriebsnummer / Messkreisnummer		
System	Wert	Bedeutung
840D sl	x	Über die Antriebsnummer wird die logische E/A-Adresse des Antriebs aus MD13050 \$MN_DRIVE_LOGIC_ADDRESS[n] zugeordnet. Die Antriebsnummer (x) ergibt sich aus dem Index (n) von MD13050: $x = n + 1$

MD30230		\$MA_ENC_INPUT_NR[n]
Istwertzuordnung: Eingang auf Antriebsmodul / Messkreiskarte		
System	Wert	Bedeutung
840D sl	x	Nummer der Geber-Schnittstelle innerhalb des PROFIdrive-Telegramms Beispiele PROFIdrive-Telegramm 103 $x = 1 \rightarrow$ 1. Geber-Schnittstelle (G1_ZSW, G1_XIST1, G1_XIST2) $x = 2 \rightarrow$ 2. Geber-Schnittstelle (G2_ZSW, G2_XIST1, G2_XIST2) PROFIdrive-Telegramm 118 $x = 1 \rightarrow$ 1. Geber-Schnittstelle (G2_ZSW, G2_XIST1, G2_XIST2) $x = 2 \rightarrow$ 2. Geber-Schnittstelle (G3_ZSW, G3_XIST1, G3_XIST2) Hinweis: Bei SINAMICS S120: - Geber 1 (G1_...): Motorgeber - Geber 2 (G2_...): Direktes Messsystem - Geber 3 (G3_...): Zusätzliches Messsystem

MD30240		\$MA_ENC_TYPE[n]
Geber-Typ der Istwerterfassung (Lageistwert)		
System	Wert	Bedeutung
840D sl	0	Simulation (Betrieb ohne Geber)
	1	Inkrementalgeber
	4	Absolutgeber
Hinweis: Korrespondiert mit PROFIdrive-Parameter p979		

MD30242		\$MA_ENC_IS_INDEPENDENT[n, Achse]
Geber ist unabhängig		
System	Wert	Bedeutung
840D sl	0	Der Geber ist nicht unabhängig.
	1	<p>Der Geber ist unabhängig.</p> <p>Sollen Istwertkorrekturen, die auf dem für die Lageregelung ausgewählten Geber vorgenommen werden, nicht den Istwert des zweiten in der gleichen Achse definierten Gebers beeinflussen, so ist dieser als unabhängig (independent) zu erklären.</p> <p>Istwertkorrekturen sind:</p> <ul style="list-style-type: none"> • - Modulo-Behandlung, • - Referenzpunktfahren, • - Messsystemabgleich, • - PRESET <p>Beispiel: eine Achse, 2 Geber, 2. Geber ist unabhängig MD30200 \$MA_NUM_ENCS[AX1] = 2 MD30242 \$MA_ENC_IS_INDEPENDENT[0, AX1] = 0 MD30242 \$MA_ENC_IS_INDEPENDENT[1, AX1] = 1</p> <p>Auswahl Lagemesssystem 1 / 2: DB31.DBX1.5 / 1.6</p> <p>Ist Geber 1 zur Lageregelung ausgewählt, werden die Istwertkorrekturen nur auf diesem Geber durchgeführt, da Geber 2 unabhängig ist.</p> <p>Ist Geber 2 zur Lageregelung ausgewählt, werden die Istwertkorrekturen auf beiden Gebern durchgeführt, da Geber 1 nicht unabhängig ist.</p> <p>Das Maschinendatum hat also nur Auswirkung auf die passiven Geber einer Maschinenachse.</p>
	2	<p>Der passive Geber ist abhängig.</p> <p>Der Geberistwert wird durch den aktiven Geber verändert. In der Kombination mit MD35102 \$MA_REFP_SYNC_ENCS = 1 wird der passive Geber beim Referenzpunktfahren auf den aktiven Geber abgeglichen NICHT aber referenziert.</p> <p>Im Referenziermodus MD34200 \$MA_ENC_REFP_MODE = 3 (abstandscodierte Referenzmarken) wird der passive Geber mit der nächsten Verfahrbewegung nach dem Überfahren der Nullmarkendistanz automatisch referenziert. Dieses geschieht unabhängig von der aktuellen Betriebsarteneinstellung.</p>
	3	<p>Der Geber ist unabhängig.</p> <p>Bei Modulo-Rundachsen werden Modulo-Istwertkorrekturen auch im passiven Geber durchgeführt.</p>

Hinweis**Maschinendaten-Index [n]**

Der Maschinendaten-Index [n] für die Geberzuordnung hat folgende Bedeutung:

- n = 0: Erster der Maschinenachse zugeordneter Geber
- n = 1: Zweiter der Maschinenachse zugeordneter Geber

Die Zuordnung erfolgt über die Maschinendaten:

- MD30220 \$MA_ENC_MODULE_NR[n]
- MD30230 \$MA_ENC_INPUT_NR[n]

Literatur

Inbetriebnahmehandbuch Inbetriebnahme CNC: NC, PLC, Antrieb;
Kapitel: "Kommunikation zwischen NC und Antrieb" > "Antriebe: Achse zuordnen"

8.4.3 Anpassungen der Motor/Last-Verhältnisse**Getriebearten**

Zur Anpassung der mechanischen Verhältnisse existieren folgende Getriebearten:

Getriebeart	Aktivierung	Anpassung	Anbauort
Motor-/Last-Getriebe	Parametersatz	fest projektiert	Getriebekasten
Geber-Mess-Getriebe	Power On	geberabhängig	geberseitig
Last-Vorsatz-Getriebe	Warmstart	lastabhängig	werkzeugseitig

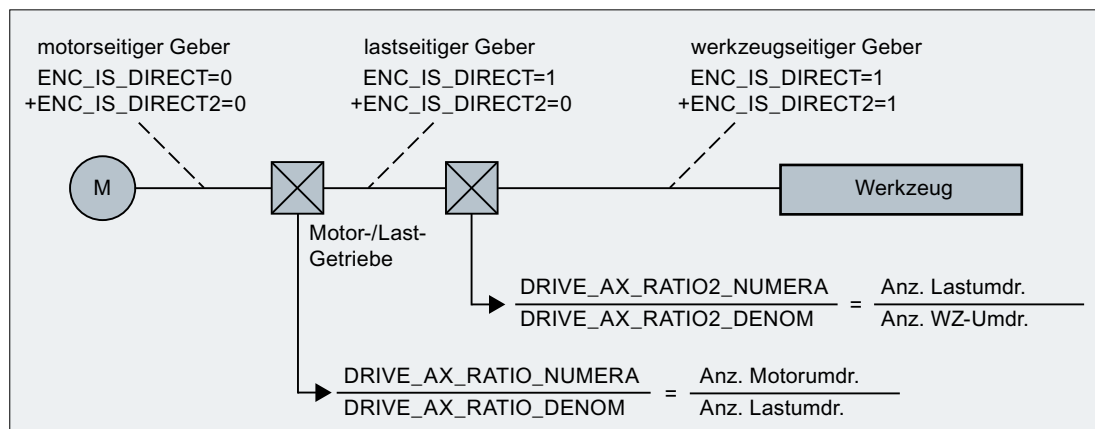
Örtliche Lage der Getriebe / Geber

Bild 8-2 Getriebearten und Geberanbauorte

Motor-/Last-Getriebe

Das von SINUMERIK unterstützte Motor-/Last-Getriebe wird über folgende Maschinendaten projiziert:

- MD31060 \$MA_DRIVE_AX_RATIO_NUMERA (Zähler Lastgetriebe)
- MD31050 \$MA_DRIVE_AX_RATIO_DENOM (Nenner Lastgetriebe)

Die Getriebeübersetzung ergibt sich aus dem Verhältnis Zähler zu Nenner der beiden Maschinendaten. Über die dazugehörigen Parametersätze wird der Lageregler standardmäßig automatisch von der Steuerung auf die jeweiligen Übersetzungsverhältnisse synchronisiert.

Da ein Getriebestufenwechsel nicht immer automatisch erfolgen muss und es außerdem mehrere Möglichkeiten gibt, eine neue Getriebestufe anzufordern, wird nicht immer die Lageregelung über die Parametersätze einbezogen.

Hinweis

Weitere Informationen zu den Parametersätzen beim Getriebestufenwechsel siehe Kapitel "S1: Spindeln (Seite 1295)".

Vorsatz-Getriebe

Von der Steuerung werden zusätzlich projektierbare Last-Vorsatz-Getriebe unterstützt:

- MD31066 \$MA_DRIVE_AX_RATIO2_NUMERA (Zähler Vorsatzgetriebe)
- MD31064 \$MA_DRIVE_AX_RATIO2_DENOM (Nenner Vorsatzgetriebe)

Angetriebene Werkzeuge bringen meistens ihr "eigenes" Vorsatz-Getriebe mit. Solche variablen Mechaniken können durch das multiplikativ zum Motor-/Last-Getriebe wirkende Vorsatz-Getriebe parametrisiert werden.



VORSICHT

Unterschiedliche Getriebeübersetzungen beim Umschalten

Im Unterschied zum Motor-/Last-Getriebe gibt es beim Vorsatz-Getriebe keinen Parametersatz und damit auch keine Möglichkeit, die zeitsynchrone Umschaltung zum Teileprogramm oder zur PLC (NC/PLC-Nahtstelle) steuern zu können. Eine Teile-Bearbeitung während des Getriebewechsels ist deshalb auszuschließen. Es bleibt Aufgabe des Anwenders, die Synchronisation der jeweils geänderten Maschinendaten mit der entsprechenden mechanischen Umschaltung abzustimmen und zu aktivieren. Bei einer Umschaltung während einer Bewegung sind aufgrund der sich sprunghaft ändernden Normierungsfaktoren Ausgleichsvorgänge **nicht** auszuschließen. Diese werden nicht auf Einhaltung der maximalen Beschleunigung überwacht.

Geber direkt am Werkzeug

Für das Vorsatz-Getriebe wird eine weitere Anschlussvariante für einen "werkzeugseitigen Geber" durch Projektierung des folgenden Maschinendatum möglich:

MD31044 \$MA_ENC_IS_DIRECT2

Geber nicht direkt am Werkzeug

Bei einer Getriebeumschaltung des Vorsatz-Getriebes im lagegeregelten Betrieb gilt: Die umzuschaltende Getriebe-Übersetzung geht in diesem Fall auch in eine Umnormierung der Geber-Informationen ein.

In diesem Fall gilt für Achsen/Spindeln im Positionierbetrieb:

- Eine Getriebeumschaltung ist **nur bei Stillstand sprungfrei** möglich. Hierfür wird die werkzeugseitige Position softwareseitig vor und nach einer Getriebeumschaltung bei einer Übersetzungs-Änderung gleichgesetzt, da sich die mechanische Position während eines Getriebestufenwechsels nur wenig oder gar nicht ändert.

Empfehlung:

Zur Vermeidung von Alarm 21612 "Reglerfreigabe während der Bewegung zurückgesetzt" ist die Umschaltung "nur bei Stillstand" vorzuziehen. Es bleibt weiterhin zulässig und auch sinnvoll, die Achse bzw. Spindel vor oder während einer Getriebe-Umschaltung zusätzlich in den Drehzahl-Steuerbetrieb oder in den Nachführbetrieb zu schalten.

Randbedingungen

Ist der für die Lageregelung genutzte **Geber direkt am Werkzeug** angeschlossen, so wirkt ein Getriebestufenwechsel des Motor-/Last-Getriebes oder des Vorsatz-Getriebes ausschließlich auf die physikalischen Größen an der Drehzahlschnittstelle zwischen NC und Antrieb. Die steuerungsinternen Parametersätze werden dabei nicht umgeschaltet.

Referenzpunkt und Maschinenbezug

VORSICHT

Verlust des Maschinenbezugs

Die Steuerung kann nicht alle Situationen erkennen, die zum Verlust des Maschinenbezugs führen. Es liegt daher in der alleinigen Verantwortung des Inbetriebnehmers bzw. Anwenders, in solchen Fällen ein explizites Referenzieren bzw. Nullmarken-Synchronisieren anzustoßen.

Bei Getriebeumschaltungen kann mit Auswirkung auf die Geber-Normierung keine Aussage über Referenzpunkt- oder Maschinenpositions-Bezug getroffen werden. Teilweise nimmt die Steuerung in solchen Fällen den Status "Achse referenziert/synchronisiert" weg.

Ist der Maschinenbezug verlorengegangen, muss durch einen Justage- oder Referenzvorgang der verloren gegangene Maschinenbezug wieder hergestellt werden.

Siehe auch

R1: Referenzieren (Seite 1243)

8.4.4 Drehzahlsollwertausgabe

Regelsinn und Verfahrrichtung der Vorschubachsen

Vor Beginn der Arbeit ist die Verfahrrichtung der Vorschubachse zu klären.

Regelsinn

Vor der Inbetriebnahme der Lageregelung müssen der Drehzahlregler und der Stromregler des Antriebs in Betrieb genommen und optimiert werden.

Verfahrrichtung

Mit dem Maschinendatum:

MD32100 \$MA_AX_MOTION_DIR (Verfahrrichtung)

kann die Bewegungsrichtung der Achse umgekehrt werden,
ohne Auswirkung auf den Regelsinn der Lageregelung.

Drehzahlsollwertanpassung

SINUMERIK 840D sl

Beim Drehzahlsollwertabgleich wird der NC zur Parametrierung der axialen Regelung und Überwachung mitgeteilt, welchem Drehzahlsollwert welche Motordrehzahl im Antrieb entspricht. Dieser Abgleich erfolgt automatisch.

Für PROFIBUS-DP Antriebe ist alternativ auch der manuelle Drehzahlsollwertabgleich möglich.

- **Manueller Abgleich**

In das Maschinendatum:

MD32250 \$MA_RATED_OUTVAL

wird ein Wert ungleich Null eingetragen.

Hinweis

Geschwindigkeitsabgleich und Maximaler Drehzahlsollwert

Ein Geschwindigkeitsabgleich ist bei einer SINUMERIK 840D sl aufgrund des automatischen Drehzahlsollwertabgleichs nicht erforderlich!

Maximaler Drehzahlsollwert

Bei SINUMERIK 840D sl ist der maximale Drehzahlsollwert in Prozent festgelegt. 100% bedeutet maximaler Drehzahlsollwert bzw. Maximaldrehzahl bei PROFIdrive-Antrieben (herstellerspezifischer Einstellparameter im Antrieb, z.B. p1082 bei SINAMICS).

Die Ausgabe der Spindeldrehzahl ist bei der SINUMERIK 840D sl in der NC realisiert.

In der Steuerung sind Daten für 5 Getriebestufen realisiert.

Die Getriebestufen sind durch eine Minimal- und Maximaldrehzahl für die Getriebestufe und eine Minimaldrehzahl und eine Maximaldrehzahl für den automatischen Getriebestufenwechsel definiert. Die Ausgabe einer neuen Soll-Getriebestufe erfolgt nur,

wenn der neu programmierte Drehzahlsollwert nicht in der aktuellen Getriebestufe gefahren werden kann.

Mit dem Maschinendatum:

MD36210 \$MA_CTRLOUT_LIMIT[n] (Maximaler Drehzahlsollwert)

wird der Drehzahlsollwert prozentual begrenzt.

Werte bis 200% sind möglich.

Bei Überschreiten der Grenze wird ein Alarm ausgegeben.

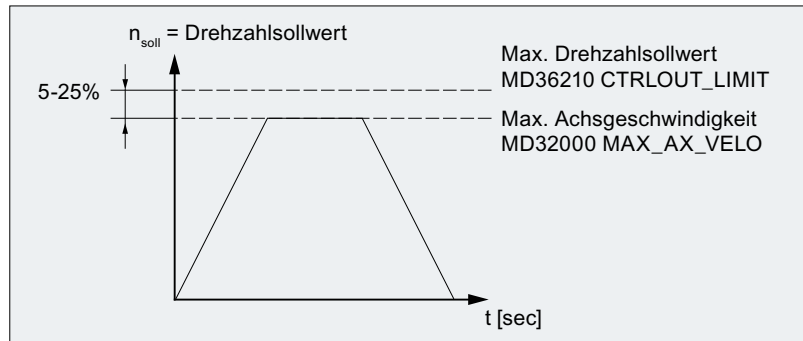


Bild 8-3 Maximaler Drehzahlsollwert

Die Achsen sollten wegen Regelungsvorgängen jedoch nicht erst bei 100% Drehzahlsollwert ihre Maximalgeschwindigkeit (MD32000 \$MA_MAX_AX_VELO) erreichen, sondern bereits bei 80% bis 95%.

Bei Achsen, deren maximale Geschwindigkeit bei ca. 80% des Drehzahlsollwertbereiches erreicht wird, kann der Standardwert (default 80%) des Maschinendatums:

MD32000 \$MA_MAX_AX_VELO (Maximale Achsgeschwindigkeit)

übernommen werden.

8.4.5 Maschinendaten des Istwertsystems

Achsspezifische Maschinendaten

Zur Parametrierung des Istwertsystems sind folgende achsspezifische Maschinendaten einzustellen:

Geber- und Parametersatz-unabhängige Maschinendaten: \$MA_	Bedeutung
MD30200 NUM_ENCS	Anzahl der Geber
MD30300 IS_ROT_AX	Rundachse / Spindel
MD30310 ROT_IS_MODULO	Modulo-Wandlung für Rundachse / Spindel
MD30320 DISPLAY_IS_MODULO	Modulo 360 Grad Anzeige bei Rundachse oder Spindel
MD30330 MODULO_RANGE	Größe des Modulo-Bereichs

Geber- und Parametersatz-unabhängige Maschinendaten: \$MA_	Bedeutung
MD30340 MODULO_RANGE_START	Startposition des Modulo-Bereichs
MD31030 \$MA_LEADSCREW_PITCH	Steigung der Kugelrollspindel
MD31064 \$MA_DRIVE_AX_RATIO2_DENOM	Nenner Vorsatz-Gertriebe
MD31066 \$MA_DRIVE_AX_RATIO2_NUMERA	Zähler Vorsatz-Gertriebe
MD32000 \$MA_MAX_AX_VELO	Maximale Achsgeschwindigkeit

Geber-abhängige Maschinendaten: \$MA_	Bedeutung
MD30210 ENC_SEGMENT_NR[n]	Istwertzuordnung: Nummer des Bus-Segments
MD30220 ENC_MODULE_NR[n]	Istwertzuordnung: Antriebsnummer/Messkreisnummer
MD30230 ENC_INPUT_NR[n]	Istwertzuordnung: Eingang auf Antriebsmodul/Messkreiskarte
MD30240 ENC_TYPE[n]	Geber-Typ der Istwerterfassung (Lageistwert)
MD30242 ENC_IS_INDEPENDENT[n]	Geber ist unabhängig
MD30244 ENC_MEAS_TYPE[n]	Encoder-Mess-Type
MD30250 ACT_POS_ABS[n]	Interne Geberposition
MD30260 ABS_INC_RATIO[n]	Absolutgeber: Verhältnis Absolutauflösung zu Inkrementalauflösung
MD30270 ENC_ABS_BUFFERING[n]	Absolutgeber: Verfahrbereichserweiterung
MD34090 \$MA_REFP_MOVE_DIST_CORR[n]	Referenzpunktverschiebung
MD34320 \$MA_ENC_INVERS[n]	Längenmesssystem ist gegensinnig
n: Geber-Index, mit n = 0, 1, ... (1. Geber, 2. Geber, ...)	

Hinweis

Die "Maschinendaten wirksam setzen", können entweder im Teileprogramm mit dem Befehl `NEWCONF` oder über die Bedienoberfläche per Softkey aktiviert werden.

8.4.6 Istwertauflösung

8.4.6.1 Maschinendaten der Istwertauflösung

Abhängig davon wo das Messsystem (Geber) angebaut ist, sind folgende Messsystemarten zu berücksichtigen:

- Lastseitiger Geber: Direktes Messsystem (DM)
- Motorseitiger Geber: Indirektes Messsystem (IM)

Parametrierung der Istwertauflösung abhängig vom Achstyp (Linear-/Rundachse)

Anhand der folgenden Maschinendaten wird die Istwertauflösung von der Steuerung errechnet.

Maschinendaten zur Berechnung der Istwertauflösung	Linearachse	Linearachse		Rundachse	
	Linearmaßstab / direktes Messsystem	Indirektes Messsystem	Direktes Messsystem: Maschine / Werkzeug	Indirektes Messsystem	Direktes Messsystem: Maschine / Werkzeug
MD30300 \$MA_IS_ROT_AX	0	0	0	1	1
MD31000 \$MA_ENC_IS_LINEAR[n]	1	0	0	0	0
MD31010 \$MA_ENC_GRID_POINT_DIST[n]	Teilung	-	-	-	-
MD34320 \$MA_ENC_INVERS[n]	¹⁾	-	-	-	-
MD31020 \$MA_ENC_RESOL[n]	-	Striche/ Umdr.	Striche/ Umdr.	Striche/ Umdr.	Striche/ Umdr.
MD31025 \$MA_ENC_PULSE_MULT[n]	Gebervervielfachung				
MD31030 \$MA_LEADSCREW_PITCH	-	mm/Umdr.	mm/Umdr.	-	-
MD31040 \$MA_ENC_IS_DIRECT[n]	- / 1	0	1	0	1
MD31044 \$MA_ENC_IS_DIRECT2[n]	- / 1	0	1	0	1
MD31050 \$MA_DRIVE_AX_RATIO_DENOM[n]	-	Lastumdr.	- / 1	Lastumdr.	²⁾
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA[n]	-	Motorumdr. falls Zustellgetriebe vorh.	- / 1	Motorumdr.	²⁾
MD31070 \$MA_DRIVE_ENC_RATIO_DENOM[n]	-	Geberumdr.	Geberumdr.	Geberumdr.	Geberumdr.
MD31080 \$MA_DRIVE_ENC_RATIO_NUMERA[n]	-	Motorseitiger Geber ³⁾	Motorumdr.	Motorumdr.	Lastumdr.

- Nicht relevant

- 1) Bei abstandscodierten Messsystemen
- 2) Diese Maschinendaten werden nicht für die Geberanpassung (Wegbewertung) benötigt. Sie müssen jedoch für die Sollwertberechnung richtig eingegeben werden! Es stellt sich sonst nicht der gewünschte K_v -Faktor ein. In Maschinendatum MD31050 \$MA_DRIVE_AX_RATIO_DENOM werden die Lastumdrehungen, in Maschinendatum MD31060 \$MA_DRIVE_AX_RATIO_NUMERA die Motorumdrehungen eingetragen.
- 3) Der motorseitige Geber ist ein Einbaugeber und hat damit **kein** Messgetriebe. Das Übersetzungsverhältnis ist immer 1:1.

Maschinendaten der Istwertauflösung

Die Istwertauflösung ergibt sich aus dem Aufbau der Maschine, ob Getriebe vorhanden sind und deren Übersetzung, der Steigung der Kugelrollspindel bei Linearachsen und der Auflösung des verwendeten Gebers. An der Steuerung sind dazu folgende Maschinendaten einzustellen:

Geber- und Parametersatz-unabhängige Maschinendaten: \$MA_	Bedeutung
MD30300 IS_ROT_AX	Achse ist Rundachse / Spindel
MD31010 ENC_GRID_POINT_DIST	Teilungsperiode des Linearmaßstabs
MD31030 LEADSCREW_PITCH	Steigung der Kugelrollspindel
MD31064 DRIVE_AX_RATIO2_DENOM	Nenner Vorsatzgetriebe
MD31066 DRIVE_AX_RATIO2_NUMERA	Zähler Vorsatzgetriebe

Geber-abhängige Maschinendaten: \$MA_	Bedeutung
MD31000 ENC_IS_LINEAR[n]	Messsystem ist ein Linearmaßstab
MD31020 ENC_RESOL[n]	Geberstriche pro Umdrehung
MD31025 ENC_PULSE_MULT[n]	Geber-Vervielfachung (Hochauflösung)
MD31040 ENC_IS_DIRECT[n]	Direktes oder indirektes Messsystem
MD31044 ENC_IS_DIRECT2[n]	Geber am Vorsatzgetriebe
MD31070 DRIVE_ENC_RATIO_DENOM[n]	Nenner Messgetriebe
MD31080 DRIVE_ENC_RATIO_NUMERA[n]	Zähler Messgetriebe
n: Geber-Index, mit n = 0, 1, ... (1. Geber, 2. Geber, ...)	

Parametersatz-abhängige Maschinendaten: \$MA_	Bedeutung
MD31050 \$MA_DRIVE_AX_RATIO_DENOM[m]	Nenner Lastgetriebe
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA[m]	Zähler Lastgetriebe
m: Parametersatz-Index, mit m = 0, 1, ... (1. Parametersatz, 2. Parametersatz, ...)	

Parametrierung der Rechenfeinheit

Das Verhältnis von steuerungsinterner Rechenfeinheit zur Istwertauflösung ist ein Maß dafür, wie exakt die von der Steuerung berechneten Werte an der Maschine umgesetzt werden können.

Rechenfeinheit: Linearachsen

$\frac{\text{Rechenfeinheit}}{\text{Istwertauflösung}} = \frac{\text{Interne Inkremente / (mm)}}{\text{Geberinkremente / (mm)}}$
--

Rechenfeinheit: Rundachsen

$$\frac{\text{Rechenfeinheit}}{\text{Istwertauflösung}} = \frac{\text{Interne Inkremente / (Grad)}}{\text{Geberinkremente / (Grad)}}$$

Maschinendaten

Allgemeine Maschinendaten: \$MN_	Bedeutung
MD10200 INT_INCR_PER_MM	Rechenfeinheit für Linearpositionen
MD10210 INT_INCR_PER_DEG	Rechenfeinheit für Winkelpositionen

Einstellempfehlung

Die oben genannten Komponenten und Einstellungen, die für die Istwertauflösung verantwortlich sind, sollte so gewählt werden, dass die Istwertauflösung größer ist als die parametrisierte Rechenfeinheit.

$$\frac{\text{Rechenfeinheit}}{\text{Istwertauflösung}} \leq 1$$

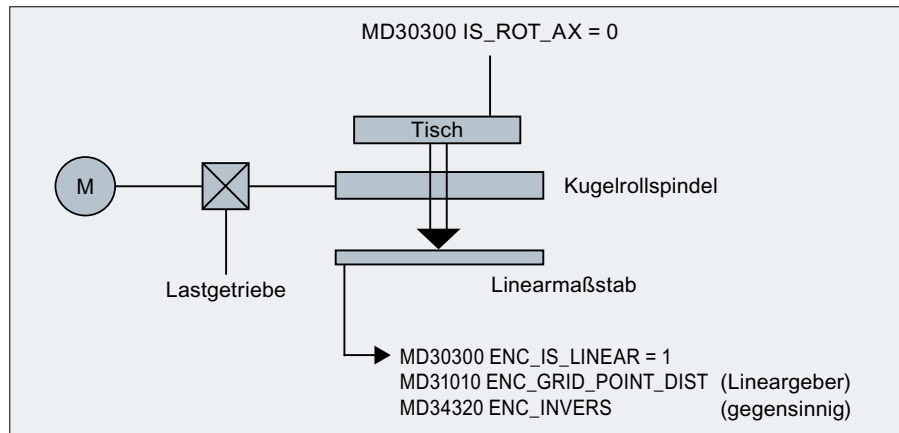
8.4.6.2 Beispiel: Linearachse mit Linearmaßstab

Bild 8-4 Linearachse mit Linearmaßstab

Das Verhältnis von internen Inkrementen zu Geberinkrementen pro mm berechnet sich zu:

$$\frac{\text{Interne Inkremente / mm}}{\text{Geberinkremente / mm}} = \frac{\text{ENC_GRID_POINT_DIST [n]} * \text{INT_INCR_PER_MM}}{\text{ENC_PULSE_MULT[n]}}$$

8.4.6.3 Beispiel: Linearachse mit rotatorischem Geber am Motor

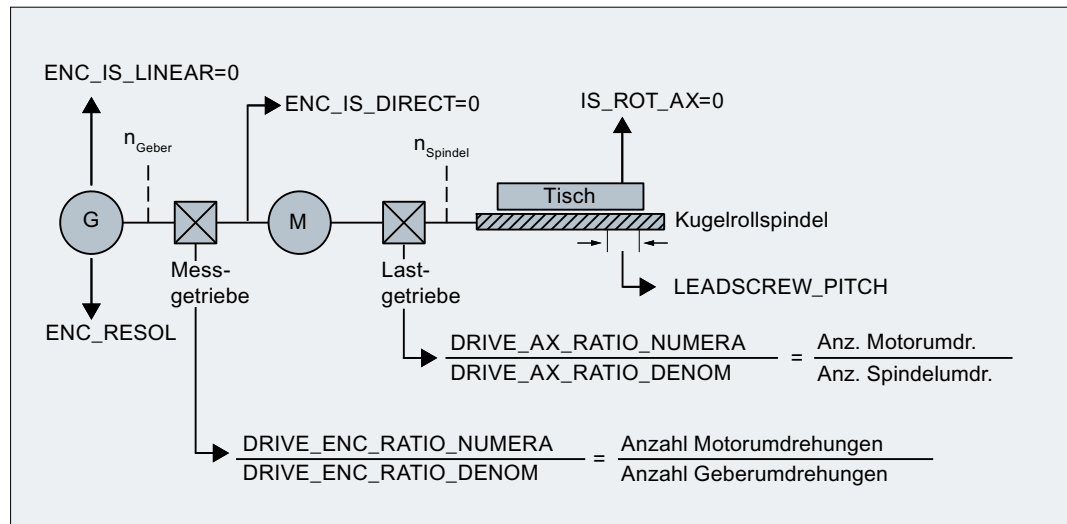


Bild 8-5 Linearachse mit rotatorischem Geber am Motor

Das Verhältnis von internen Inkrementen zu Geberinkrementen pro mm berechnet sich zu:

$$\frac{\text{Interne Inkremente / mm}}{\text{Geberinkremente / mm}} = \frac{1}{\text{ENC_RESOL [n]} * \text{ENC_PULSE_MULT[n]}}$$

$$* \frac{\text{DRIVE_ENC_RATIO_NUMERA [n]}}{\text{DRIVE_ENC_RATIO_DENOM [n]}}$$

$$* \frac{\text{DRIVE_AX_RATIO_DENOM [n]}}{\text{DRIVE_AX_RATIO_NUMERA [n]}}$$

$$* \text{LEADSCREW_PITCH}$$

$$* \text{INT_INCR_PER_MM}$$

Beispiel

Annahmen:

- Rotatorischer Geber am Motor: 2048 Impulse / Umdrehung
- Interne Impulsvervielfachung: 2048
- Getriebe, Motor / Kugelrollspindel: 5:1
- Steigung der Kugelrollspindel: 10 mm / Umdrehung
- Rechenfeinheit: 10000 Inkremente pro mm

Maschinendatum	Wert
MD30300 \$MA_IS_ROT_AX	0
MD31000 \$MA_ENC_IS_LINEAR[0]	0
MD31040 \$MA_ENC_IS_DIRECT[0]	0
MD31020 \$MA_ENC_RESOL[0]	2048
MD31025 \$MA_ENC_PULSE_MULT	2048
MD31030 \$MA_LEADSCREW_PITCH	10
MD31080 \$MA_DRIVE_ENC_RATIO_NUMERA[0]	1
MD31070 \$MA_DRIVE_ENC_RATIO_DENOM[0]	1
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA[0]	5
MD31050 \$MA_DRIVE_AX_RATIO_DENOM[0]	1
MD10210 \$MN_INT_INCR_PER_DEG	10000

$$\rightarrow \frac{\text{Interne Inkremente / mm}}{\text{Geberinkremente / mm}} = \frac{1}{2048 * 2048} * \frac{1}{1} * \frac{1}{5} * 10 \text{ mm}$$

$$* 10000 \text{ Inkr./mm} = 0,004768$$

Ein Geberinkrement entspricht 0,004768 internen Inkrementen bzw. 209,731543 Geberinkremente entsprechen einem internen Inkrement.

8.4.6.4

Beispiel: Linearachse mit rotatorischem Geber an der Maschine

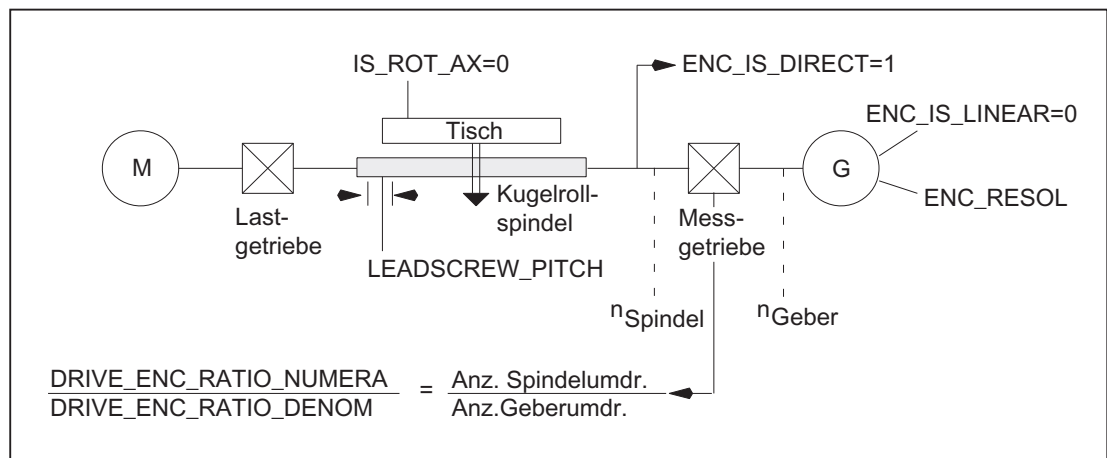


Bild 8-6 Linearachse mit rotatorischem Geber an der Maschine

Das Verhältnis von internen Inkrementen zu Geberinkrementen pro mm berechnet sich zu:

$$\frac{\text{Interne Inkremente / mm}}{\text{Geberinkremente / mm}} = \frac{1}{\text{ENC_RESOL [n]} * \text{ENC_PULSE_MULT[n]}}$$

$$* \frac{\text{DRIVE_ENC_RATIO_NUMERA [n]}}{\text{DRIVE_ENC_RATIO_DENOM [n]}}$$

$$* \text{LEADSCREW_PITCH}$$

$$* \text{INT_INCR_PER_MM}$$

8.4.6.5 Beispiel: Rundachse mit rotatorischem Geber am Motor

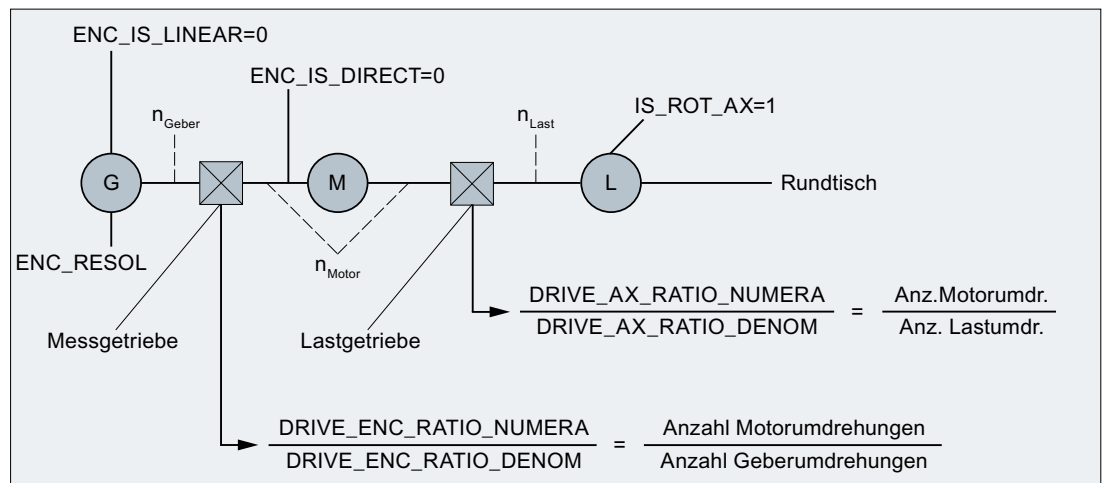


Bild 8-7 Rundachse mit rotatorischem Geber am Motor

Das Verhältnis von internen Inkrementen zu Geberinkrementen pro Grad berechnet sich zu:

$$\frac{\text{Interne Inkremente / Grad}}{\text{Geberinkremente / Grad}} = \frac{360 \text{ Grad}}{\text{ENC_RESOL [n]} * \text{ENC_PULSE_MULT[n]}}$$

$$* \frac{\text{DRIVE_ENC_RATIO_NUMERA [n]}}{\text{DRIVE_ENC_RATIO_DENOM [n]}}$$

$$* \frac{\text{DRIVE_AX_RATIO_DENOM [n]}}{\text{DRIVE_AX_RATIO_NUMERA [n]}}$$

$$* \text{INT_INCR_PER_DEG}$$

Beispiel

Annahmen:

- Rotatorischer Geber am Motor: 2048 Impulse / Umdrehung
- Interne Impulsvervielfachung: 2048
- Getriebe, Motor / Rundachse: 5:1
- Rechenfeinheit: 1000 Inkremente pro Grad

Maschinendatum	Wert
MD30300 \$MA_IS_ROT_AX	1
MD31000 \$MA_ENC_IS_LINEAR[0]	0
MD31040 \$MA_ENC_IS_DIRECT[0]	0
MD31020 \$MA_ENC_RESOL[0]	2048
MD31025 \$MA_ENC_PULSE_MULT	2048
MD31080 \$MA_DRIVE_ENC_RATIO_NUMERA[0]	1
MD31070 \$MA_DRIVE_ENC_RATIO_DENOM[0]	1
MD31060 \$MA_DRIVE_AX_RATIO_NUMERA[0]	5
MD31050 \$MA_DRIVE_AX_RATIO_DENOM[0]	1
MD10210 \$MN_INT_INCR_PER_DEG	1000

$$\Rightarrow \frac{\text{Interne Inkremente / Grad}}{\text{Geberinkremente / Grad}} = \frac{360 \text{ Grad}}{2048 * 2048} * \frac{1}{1} * \frac{1}{5}$$

$$* 1000 \text{ Inkr./Grad} = 0,017166$$

Ein Geberinkrement entspricht 0,017166 internen Inkrementen bzw. 58,254689 Geberinkremente entsprechen einem internen Inkrement.

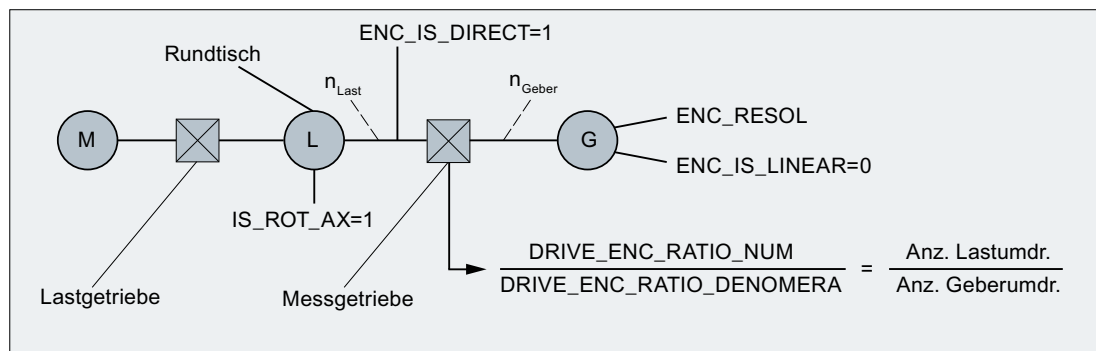
8.4.6.6**Beispiel: Rundachse mit rotatorischem Geber an der Maschine**

Bild 8-8 Rundachse mit rotatorischem Geber an der Maschine

Das Verhältnis von internen Inkrementen zu Geberinkrementen pro Grad berechnet sich zu:

$$\frac{\text{Interne Inkremente / Grad}}{\text{Geberinkremente / Grad}} = \frac{360 \text{ Grad}}{\text{ENC_RESOL [n]} * \text{ENC_PULSE_MULT[n]}}$$

$$* \frac{\text{DRIVE_ENC_RATIO_NUMERA [n]}}{\text{DRIVE_ENC_RATIO_DENOM [n]}}$$

$$* \text{INT_INCR_PER_DEG}$$

8.4.6.7

Beispiel: Vorsatz-Getriebe mit Geber am Werkzeug

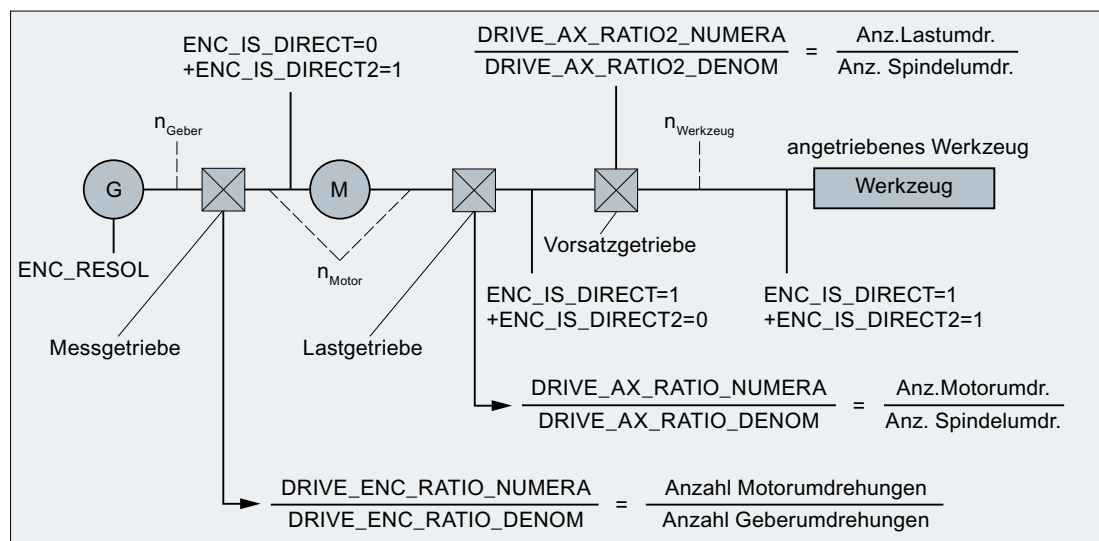


Bild 8-9 Vorsatz-Getriebe mit Geber direkt am angetriebenen Werkzeug

Das Verhältnis von internen Inkrementen zu Geberinkrementen pro Grad berechnet sich zu:

$$\frac{\text{Interne Inkremente / Grad}}{\text{Geberinkremente / Grad}} = \frac{360 \text{ Grad}}{\text{ENC_RESOL [n]} * \text{ENC_PULSE_MULT[n]}}$$

$$* \frac{\text{DRIVE_ENC_RATIO_NUMERA [n]}}{\text{DRIVE_ENC_RATIO_DENOM [n]}}$$

$$* \text{INT_INCR_PER_DEG}$$

8.5 Regelung

8.5.1 Allgemeine Informationen

Lageregelung einer Achse/Spindel

Die Regelung einer Achse besteht aus dem Strom- und Drehzahlregelkreis des Antriebs und einem übergeordneten Lageregelkreis in der NC.

Die Lageregelung einer Achse/Spindel ist im Prinzip wie folgt aufgebaut:

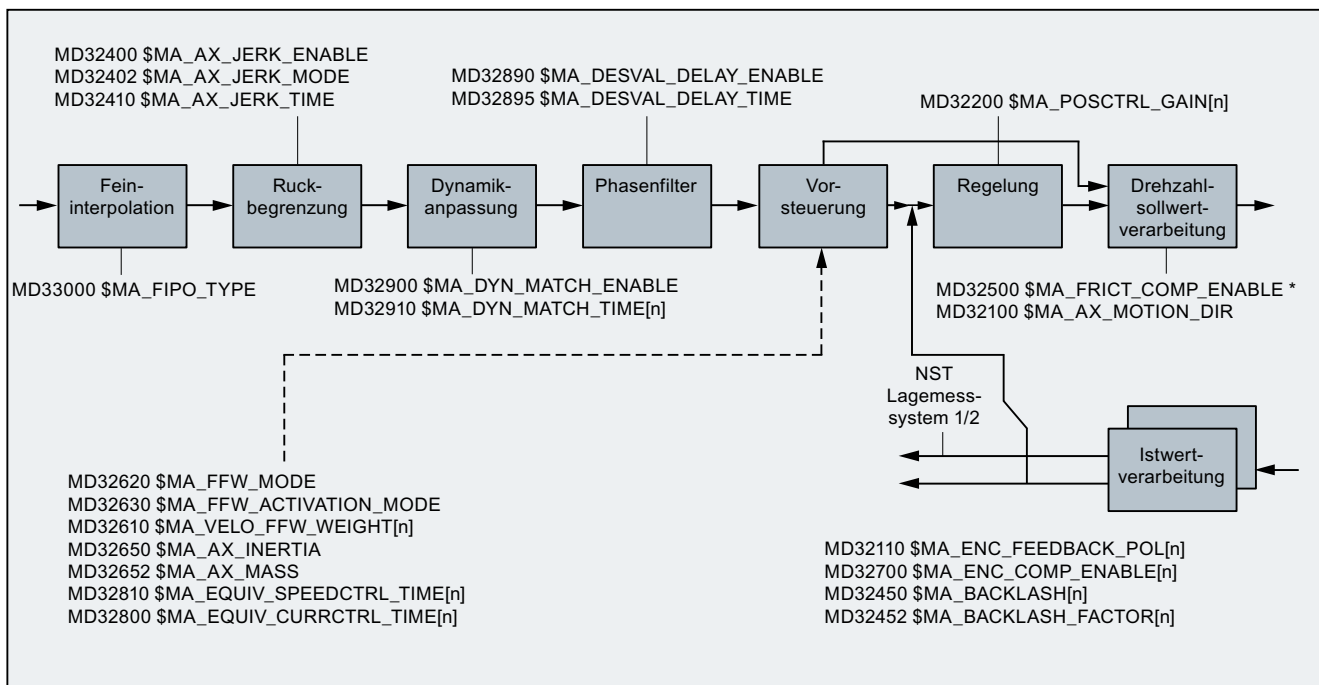


Bild 8-10 Prinzipielle Darstellung der Sollwert-Verarbeitung und Regelung

Informationen zur Ruckbegrenzung siehe Kapitel "B2: Beschleunigung (Seite 265)".

Beschreibung der Vorsteuerung, Umkehrlose-, Reibkompensation und Spindelsteigungsfehlerkompensation siehe:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kompensationen (K3)

Feininterpolation

Mit dem Feininterpolator (FIPO) kann die Konturgüte durch die Verringerung des Treppeneffekts beim Drehzahlsollwert weiter erhöht werden. Es können 3 Arten der Feininterpolation eingestellt werden:

MD33000 \$MA_FIPO_TYPE = <FIPO-Mode>

<FIPO-Mode>	Bedeutung
1	Differenzielle Feininterpolation mit Mittelwertbildung (Glättung) über einen IPO-Takt
2	Kubische Feininterpolation
3	Für die Verwendung mit der Vorsteuerung optimierte kubische Feininterpolation für höchste Konturgenauigkeit

K_v-Faktor

Damit im Bahnsteuerbetrieb nur geringe Konturabweichungen auftreten, ist ein hoher K_v-Faktor erforderlich:

MD32200 \$MA_POSCTRL_GAIN[n]

Ein zu hoher K_v-Faktor führt jedoch zu Instabilität, Überschwingungen und evtl. zu unzulässig hohen Maschinenbelastungen.

Der maximal zulässige K_v-Faktor ist abhängig von:

- Auslegung und Dynamik des Antriebs
(Anregelzeit, Beschleunigungs- und Bremsvermögen)
- Güte der Maschine
(Elastizität, Schwingungsdämpfung)
- Lageregeltakt bzw. Drehzahlregeltakt bei aktivem DSC

Der K_v-Faktor ist definiert als:

$$K_v = \frac{\text{Geschwindigkeit}}{\text{Schleppabstand}} \quad ; \quad \frac{[\text{m/min}]}{[\text{mm}]}$$

Dynamikanpassung

Achsen, die miteinander interpolieren aber unterschiedliche K_v-Faktoren aufweisen, können durch die Dynamikanpassung auf gleichen Schleppabstand eingestellt werden. Dadurch lässt sich eine optimale Konturgenauigkeit ohne Verlust von Regelgüte durch Verringerung der K_v-Faktoren auf die dynamisch schwächste Achse erreichen.

Die Funktion wird aktiviert über:

MD32900 \$MA_DYN_MATCH_ENABLE = 1 (Dynamikanpassung)

Die Dynamikanpassung erfolgt durch Vorgabe einer neuen Ersatzzeitkonstanten. Diese berechnet sich aus der Differenz der Ersatzzeitkonstanten der dynamisch schwächsten Achse und der anzupassenden Achse:

MD32910 \$MA_DYN_MATCH_TIME[n] = <Differenz der Ersatzzeitkonstanten>

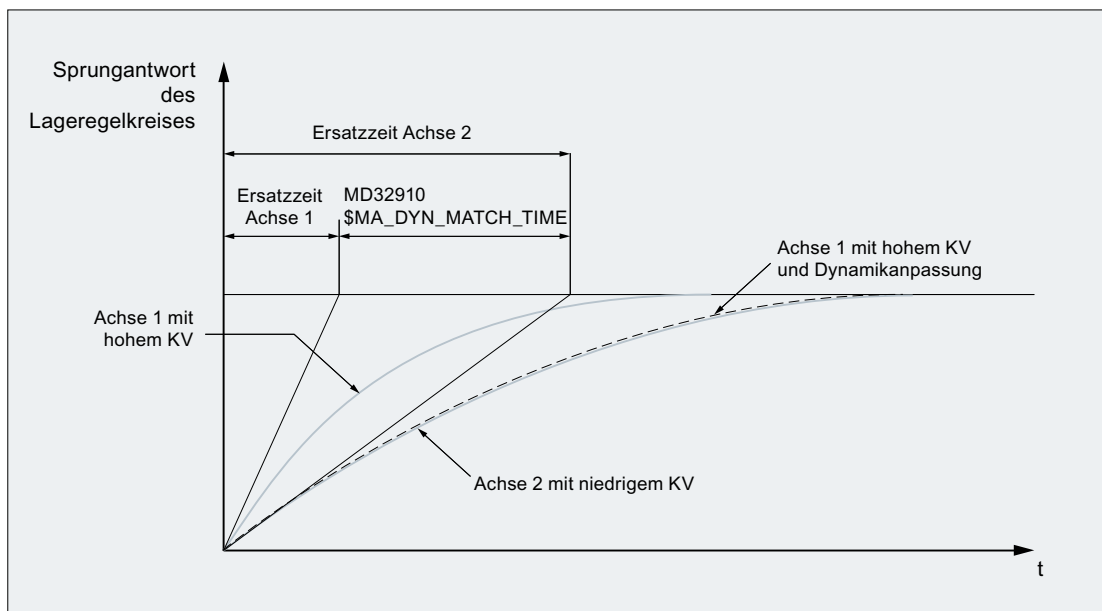


Bild 8-11 Dynamikanpassung

Beispiel für eine Dynamikanpassung von 3 Achsen ohne Drehzahlvorsteuerung

Die Ersatzzeitkonstante des Lageregelkreises betrage:	
Achse 1:	30 ms
Achse 2:	20 ms
Achse 3:	24 ms

Mit einer Ersatzzeitkonstante von 30 ms ist die Achse 1 die dynamisch schwächste Achse.

Daraus ergeben sich für die Achsen folgende neuen Ersatzzeitkonstanten:

Achse 1:	MD32910 \$MA_DYN_MATCH_TIME = 0 ms
Achse 2:	MD32910 \$MA_DYN_MATCH_TIME = 30 ms - 20 ms = 10 ms
Achse 3:	MD32910 \$MA_DYN_MATCH_TIME = 30 ms - 24 ms = 6 ms

Näherungsformeln für die Ersatzzeitkonstante des Lageregelkreises einer Achse

Die Ersatzzeitkonstante T_{Ersatz} des Lageregelkreises einer Achse berechnet sich abhängig von der Art der Vorsteuerung näherungsweise zu:

- Ohne Vorsteuerung:

$$T_{\text{Ersatz}} \approx \frac{1}{\text{MD32200 POSCTRL_GAIN [1/s]}}$$

- Mit Drehzahlvorsteuerung:

$$T_{\text{Ersatz}} \approx \text{MD32810 EQUIV_SPEEDCTRL_TIME}$$

- Bei kombinierter Momenten-/Drehzahl-Vorsteuerung

$$T_{\text{Ersatz}} \approx \text{MD32800 EQUIV_CURRCTRL_TIME}$$

Hinweis

Wird eine Dynamikanpassung bei einer Geometrieachse vorgenommen, müssen alle anderen Geometrieachsen auf die gleiche Dynamik eingestellt werden.

Literatur:

Inbetriebnahmehandbuch CNC: NC, PLC, Antriebe

8.5.2 Parametersätze des Lagereglers

Zur schnellen Anpassung der Lageregelung an veränderte Eigenschaften der Maschine während des Betriebs, z. B. bei Getriebeumschaltung der Spindel, oder zur Dynamikanpassung an eine andere Achse, z. B. beim Gewindebohren, stehen pro Maschinenachse sechs Parametersätze zur Verfügung.

Maschinendaten

Ein Parametersatz umfasst folgende Maschinendaten:

Nummer	Bezeichner \$MA_	Bedeutung
31050	DRIVE_AX_RATIO_DENOM	Nenner Lastgetriebe
31060	DRIVE_AX_RATIO_NUMERA	Zähler Lastgetriebe
32200	POSCTRL_GAIN	K_V -Faktor
32452	BACKLASH_FACTOR	Losekompensation
32610	VELO_FFW_WEIGHT	Vorsteuerfaktor
36012	STOP_LIMIT_FACTOR	Faktor Genauhalt grob/fein und Stillstand
32800	EQUIV_CURRCTRL_TIME	Ersatzzeitkonst. Stromregelkreis für Vorsteuerung
32810	EQUIV_SPEEDCTRL_TIME	Ersatzzeitkonst. Drehzahlregelkreis für Vorsteuerung
32910	DYN_MATCH_TIME	Zeitkonstante der Dynamikanpassung
36200	AX_VELO_LIMIT	Schwellwert für Geschwindigkeitsüberwachung

Gewindebohren / -schneiden

Für das Gewindebohren / -schneiden gilt bezüglich der Parametersätze von Achsen:

- Bei Maschinenachsen, die **nicht** am Gewindebohren oder -schneiden beteiligt sind, ist immer Parametersatz 1 (Index = 0) aktiv. Die anderen Parametersätze müssen nicht berücksichtigt werden.
- Bei Maschinenachsen, die am Gewindebohren oder -schneiden beteiligt sind, wird die gleiche Parametersatznummer wie die der aktuellen Getriebestufe der Spindel aktiv. Alle Parametersätze entsprechen den Getriebestufen und müssen deshalb parametrieren werden.

Der aktuelle Parametersatz wird auf der Bedienoberfläche angezeigt unter:

Bedienbereichsumschaltung" > "Diagnose" > "Service Achse"

Parametersätze beim Getriebestufenwechsel

Jeder Getriebestufe einer Spindel ist ein eigener Parametersatz zugeordnet. Die Anwahl der Getriebestufe erfolgt über folgendes NC/PLC-Nahtstellensignal:

DB31, ... DBX16.0 - 16.2 = <Istgetriebestufe>

Istgetriebestufe	DB31, ... DBX16.0 - 16.2	Parametersatz	
1. Getriebestufe	000	2	(Index=1)
1. Getriebestufe	001	2	(Index=1)
2. Getriebestufe	010	3	(Index=2)
3. Getriebestufe	011	4	(Index=3)
4. Getriebestufe	100	5	(Index=4)
5. Getriebestufe	101 110 111	6	(Index=5)

Weitere Informationen zu Getriebestufen für Spindeln siehe Kapitel "S1: Spindeln (Seite 1295)".

8.6 Optimierung der Regelung

8.6.1 Lageregler Lagesollwertfilter: Symmetrierfilter

Funktion

Bei aktiver Vorsteuerung wird der Lagesollwert vor Erreichen des eigentlichen Reglers über ein so genanntes Symmetrierfilter geschickt. Dadurch ist es möglich, den Drehzahlsollwert zu 100 % vorzusteuern, ohne dass beim Positionieren Überschwinger entstehen.

Aktivierung

Die Auswahl der Vorsteuerungsvariante und damit auch die Aktivierung des Filters erfolgt achsspezifisch über das Maschinendatum:

MD32620 \$MA_FFW_MODE (Vorsteuerungsart)

Wert	Bedeutung
3	Drehzahl-Vorsteuerung
4	Kombinierte Momenten-/Drehzahl-Vorsteuerung

Aktivierung/Deaktivierung per Teileprogramm

Die Aktivierung und Deaktivierung der Vorsteuerung für alle Achsen kann per Teileprogramm durch die Anweisungen `FFWON` und `FFWOF` vorgenommen werden.

Soll die Vorsteuerung bei einzelnen Achsen nicht durch `FFWON/FFWOF` beeinflusst werden, muss für diese Achsen die Einstellung im folgenden Maschinendatum angepasst werden:

MD32630 \$MA_FFW_ACTIVATION_MODE (Vorsteuerung aktivieren von Programm)

Parametrierung

Empfohlene Einstellung bei Neuinbetriebnahme

Bei einer Neuinbetriebnahme oder wenn zuvor Standardwerte geladen sind (Schalterstellung 1 am Inbetriebnahmeschalter und POWER ON), gelten folgende Maschinendaten-Vorbelegungen:

- MD32620 \$MA_FFW_MODE = 3
- MD32610 \$MA_VELO_FFW_WEIGHT (Vorsteuerfaktor für Drehzahl-Vorsteuerung) = 1

Für die Drehzahlvorsteuerung ist dann nur noch die Symmetrierzeit im folgenden Maschinendatum anzupassen:

MD32810 \$MA_EQUIV_SPEEDCTRL_TIME (Ersatzzeitkonstante Drehzahlregelkreis für Vorsteuerung)

Einstellung der Ersatzzeitkonstante des Drehzahlregelkreises (MD32810)

Es empfiehlt sich, die Achse in der Betriebsart "AUTOMATIK" mit einem Teileprogramm hin- und herfahren zu lassen und das Einfahren in die Zielposition, d. h. den Lageistwert des aktiven Messsystems, mit dem Servo-Trace zu beobachten.

Startwert für die Einstellung ist die Zeitkonstante des Drehzahlregelkreises. Diese kann aus dem Führungsfrequenzgang des Drehzahlregelkreises abgelesen werden. In dem häufig vorkommenden Fall PI-Regler mit Drehzahlsollwertglättung kann die Ersatzzeit näherungsweise aus den Antriebsmaschinendaten p1414, p1415, p1416 und p1421 abgelesen werden.

Man trägt nun diesen Startwert (z. B. 1,5 ms) ein:

MD32810 \$MA_EQUIV_SPEEDCTRL_TIME = 0.0015

Dann fährt man die Achse hin und her und beobachtet den Verlauf des Lageistwerts an der Zielposition in starker Vergrößerung.

Für den manuellen Feinabgleich gelten folgende Regeln:

Beobachtung	Maßnahme	Auswirkung
Überschwinger	MD32810 vergrößern	Der Lageregler reagiert langsamer. ⇒ Die Neigung zum Überspringen nimmt ab. Problem: Der Konturfehler (Abweichung von der programmierten Kontur) an Krümmungen nimmt zu.
Zu langsame Annäherung	MD32810 verkleinern	Der Lageregler reagiert schneller. ⇒ Der Konturfehler an Krümmungen nimmt ab. Problem: Die Neigung zum Überspringen nimmt zu.

Hinweis

Die Auswirkungen sind ähnlich wie beim Verändern der Lagereglerverstärkung (MD32200 \$MA_POSCTRL_GAIN). Sie können an der Bedienoberfläche im Bedienbereich "Diagnose" > "Service Achse" anhand des errechneten K_V -Werts beobachtet werden.

Feinabstimmung MD32810

Man versucht also, MD32810 so klein wie möglich auszulegen, wobei die Überschwinger beim Positionieren die Grenze setzen. Dazu verstellt man beim Feinabgleich den Anfangswert nur sehr wenig, typisch um 0,25 ms nach oben oder unten.

Bei einem Anfangswert von z. B. 1,5 ms liegt das manuell gefundene Optimum normalerweise im Bereich 1,25 ms bis 1,75 ms.

An Achsen mit direkten Messsystemen (Lastgeber) und ausgeprägter Elastizität kann es vorkommen, dass man kleine Überschwinger von einigen Mikrometern akzeptieren muss. Diese lassen sich mit Hilfe der Lagesollwertfilter für Dynamik-Anpassung (MD32910 \$MA_DYN_MATCH_TIME) und für Ruck (MD32410 \$MA_AX_JERK_TIME) reduzieren, wodurch die Achse wieder etwas langsamer wird.

Gleiche Achsdaten eines Interpolationsverbundes

Alle Achsen eines Interpolationsverbundes sollten in den folgenden Daten **gleich eingestellt sein**:

- MD32200 \$MA_POSCTRL_GAIN (angepasst über MD32910)
- MD32620 \$MA_FFW_MODE
- MD32610 \$MA_VELO_FFW_WEIGHT
- MD32810 \$MA_EQUIV_SPEEDCTRL_TIME (oder MD32800 \$MA_EQUIV_CURRCTRL_TIME) (abhängig von Mechanik und Antrieb)
- MD32400 \$MA_AX_JERK_ENABLE
- MD32402 \$MA_AX_JERK_MODE
- MD32410 \$MA_AX_JERK_TIME

Zur Kontrolle dient die K_V -Anzeige der Bedienoberfläche im Bedienbereich "Diagnose" > "Service Achse".

Ungleiche Achsdaten eines Interpolationsverbundes

Sind gleiche Werte in den o. g. Daten nicht möglich, so kann mit Hilfe des folgenden Maschinendatums eine Anpassung bewirkt werden:

MD32910 \$MA_DYN_MATCH_TIME (Zeitkonstante der Dynamikanpassung)

Dadurch ist es möglich, den gleichen K_V -Wert anzuzeigen.

Normalerweise sind unterschiedliche K_V -Anzeige-Werte ein Indiz für folgende Fälle:

- In einer oder in mehreren Achsen passen die Getriebefaktoren nicht.
- Die Einstelldaten der Vorsteuerung stimmen nicht.

Einstellung der Ersatzzeitkonstante des Stromregelkreises (MD32800)

Die Aktivierung des Filters für Drehmoment-Vorsteuerung erfolgt mit:

MD32620 \$MA_FFW_MODE = 4

Für die Einstellung der Zeitkonstante des Stromregelkreises MD32800

\$MA_EQUIV_CURRCTRL_TIME gelten dieselben Regeln und Empfehlungen wie bei der Drehzahlvorsteuerung.

Einschränkung auf steife Maschinen

Dieser Aufwand lohnt sich erfahrungsgemäß nur an sehr steifen Maschinen und setzt entsprechende Erfahrung voraus. Oft werden durch die Aufschaltung des Drehmoments die Elastizitäten der Maschine so stark angeregt, dass die entstehenden Schwingungen den Gewinn an Konturtreue neutralisieren.

Es lohnt sich in diesem Fall, alternativ den Einsatz der Funktion "Dynamische Steifigkeits-Regelung (DSC; Dynamic Stiffness Control)" zu erproben:

MD32640 \$MA_STIFFNESS_CONTROL_ENABLE = 1

Hinweis

Für die Anwendung von DSC gelten folgende Bedingungen:

- In der NC darf keine Istwertinvertierung (MD32110 \$MA_ENC_FEEDBACK_POL = -1) parametrierbar sein. Eine Istwertinvertierung im Antrieb (SINAMICS-Parameter p0410) ist zulässig.
- Die Funktion muss für alle Achsen, die in einem interpolatorischen Zusammenhang stehen (Bahnbewegung oder Kopplung), aktiviert werden (MD32640 = 1).
- Ist eine der Achsen als simulierte Achse parametrierbar (MD30130 \$MA_CTRL_OUT_TYPE = 0, MD30240 \$MA_ENC_TYPE = 0), kann DSC nicht angewendet werden. In diesem Fall muss die Funktion für alle Achsen, die in einem interpolatorischen Zusammenhang stehen, deaktiviert werden (MD32640 = 0).

**VORSICHT****Ausnahmefehlerzustand**

Ein Ausnahmefehlerzustand kann auftreten, wenn p1192 für ein nicht vorhandenes Messsystem gesetzt ist und DSC aktiviert ist!

Bei Servicefällen, bei denen ein zusätzliches Messsystem deaktiviert werden muss und DSC aktiv ist, muss sichergestellt werden, dass p1192 ebenfalls angepasst wird.

Steuerungsverhalten bei POWER ON, RESET, REPOS etc.

Bei POWER ON und RESET sowie bei "Maschinendaten wirksam setzen" werden die Einstelldaten der Vorsteuerung neu eingelesen (siehe hierzu die entsprechenden Angaben der jeweiligen Maschinendaten).

Betriebsartenwechsel, Satzsuchlauf und REPOS haben **keine** Auswirkung auf die Vorsteuerung.

8.6.2 Lageregler Lagesollwertfilter: Ruckfilter

Funktion

In einigen Anwendungen, z. B. beim Fräsen von Freiformflächen, kann es vorteilhaft sein, die Lagesollwertverläufe mit Hilfe des Ruckfilters zu glätten. Ziel dabei ist, Anregungen von Maschinenschwingungen zu minimieren und dadurch die Oberflächengüte zu optimieren. Die Filterwirkung der Lagesollwerte muss möglichst ausgeprägt sein, ohne aber die Konturgenauigkeit unzulässig zu beeinträchtigen. Außerdem muss das Filter ein möglichst "symmetrisches" Glättungsverhalten haben, d. h. fährt man dieselbe Kontur vorwärts und rückwärts ab, sollte der vom Filter verrundete Verlauf in beiden Richtungen möglichst ähnlich sein.

Anhand der Anzeige des effektiven K_V -Faktors im Bild "Service Achse" kann die Wirkung des Filters beobachtet werden. Durch die Filterwirkung werden die Lagesollwerte geringfügig verschliffen und deshalb die Bahngenauigkeit etwas reduziert, weshalb mit wachsender Filterzeit ein kleinerer effektiver K_V -Faktor angezeigt wird.

Hinweis

Der Ruckfilter erzeugt zu jeder Amplitudeneinstellung eine abhängige Phasenverschiebung. Erst der zusätzliche Einsatz des Phasenfilters (siehe "Lageregler Lagesollwertfilter: Phasenfilter (Seite 390)") ermöglicht eine transparente Einstellung der Achsdynamik.

Aktivierung

Die Funktion des achsspezifischen Sollwert-Ruckfilters muss über folgendes Maschinendatum aktiviert werden:

MD32400 \$MA_AX_JERK_ENABLE[<Achse>] = "TRUE"

Parametrierung

Filtertyp für achsspezifische Ruckbegrenzung

Für die achsspezifische Ruckbegrenzung wird i. d. R. der Filtertyp "Gleitende Mittelwertbildung" verwendet:

MD32402 \$MA_AX_JERK_MODE[<Achse>] = 2

Hinweis

Wenn vorher kein Filtertyp MD32402 \$MA_AX_JERK_MODE = 2 aktiviert war, muss einmalig "Power On" ausgelöst werden. Andernfalls reicht "Maschinendatum wirksam setzen" oder "Reset" an der Maschinensteuertafel.

Zeitkonstante für das achsspezifische Ruckfilter

Die Zeitkonstante des achsspezifischen Ruckfilters in Sekunden [s] wird eingestellt im Maschinendatum:

MD32410 \$MA_AX_JERK_TIME[<Achse>]

Feineinstellung

Die Feineinstellung des Ruckfilters wird wie folgt vorgenommen:

1. Fahrverhalten der Achse beurteilen (z. B. anhand von Positioniervorgängen bei Servo-Trace).
2. Filterzeit in MD32410 \$MA_AX_JERK_TIME ändern.
3. Geänderte Zeit mittels "Maschinendatum wirksam setzen" oder "Reset" an der Maschinensteuertafel aktivieren.

Deaktivierung

Ausschalten des Ruckfilters:

1. Filterberechnung sperren:
MD32400 \$MA_AX_JERK_ENABLE = 0
2. Sperre mittels "Maschinendatum wirksam setzen" oder "Reset" an der Maschinensteuertafel aktivieren.

Randbedingungen**Filterzeiten**

Der Ruckfilter ist nur dann wirksam, wenn die Zeitkonstante (MD32410) größer ist als ein Lageregeltakt.

Filterwirkung

- Die Anzeige des berechneten K_v -Faktors im Bild "Service Achse" zeigt kleinere Werte an als anhand der Filterwirkung angemessen wäre.
- Die Bahngenaugkeit ist besser, als der angezeigte K_v erwarten lässt. Daher kann sich bei Umstellung von MD32400 = 1 nach MD32400 = 2 der angezeigte K_v bei gleicher Filterzeit verkleinern, obwohl die Bahngenaugkeit besser wird.

Miteinander interpolierende Achsen

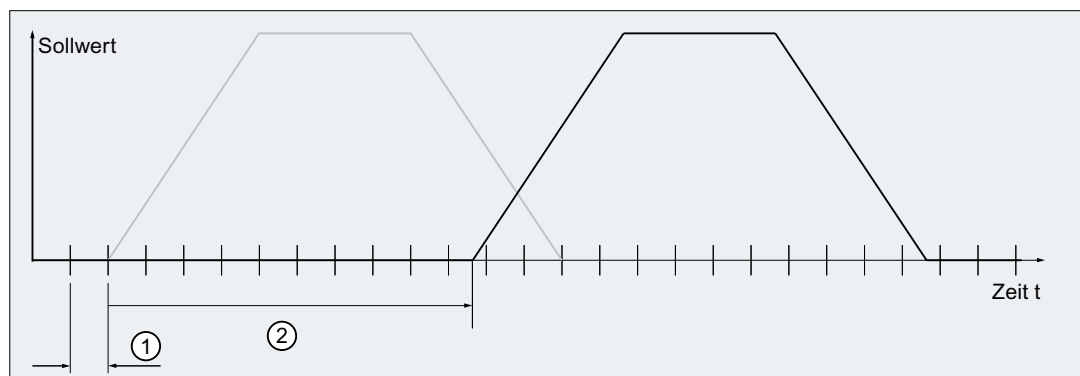
- Miteinander interpolierende Achsen müssen gleich eingestellt sein.
- Wenn man für solche Achsen jeweils ein Optimum gefunden hat, muss unter diesen Einstellungen diejenige mit der größten Filterzeit für alle Achsen des Interpolationsverbundes eingetragen werden.

Literatur

Weitere Informationen zur Ruckbegrenzung auf Interpolatorebene siehe Kapitel "Ruckbegrenzung bei Einzelachsinterpolation (SOFTA) (achsspezifisch) (Seite 284)" und "Achs-/Spindel-spezifische Maschinendaten (Seite 313)".

8.6.3 Lageregler Lagesollwertfilter: Phasenfilter**Funktion**

Das achsspezifische Sollwert-Phasenfilter (Totzeit/Verzögerung) realisiert einen reinen Phasenschieber, mit dem der Sollwert-Phasengang beeinflusst werden kann. Zusammen mit dem achsspezifischen Sollwert-Ruckfilter ($\text{MD32402_\$MA_AX_JERK_MODE[<Achse>]} = 2$; siehe Kapitel "Lageregler Lagesollwertfilter: Ruckfilter (Seite 388)") ist es somit möglich, Amplituden- und Phasengang unabhängig voneinander an die dynamisch schwächste von mehreren Achsen anzupassen, die zusammen eine programmierte Bahn verfahren sollen.



① Lageregler-Takt: 2 ms

② Zeitkonstante des achsspezifischen Sollwert-Phasenfilters: 19,2 ms

Bild 8-12 Wirkung des achsspezifischen Sollwert-Phasentfilters

Parametrierung: Zeitkonstante

Die Zeitkonstante für das achsspezifische Sollwert-Phasenfilter ist einstellbar im Bereich:

(0 bis 64) * Lageregler-Takt ($\text{MD10061 \$MN_POSCTRL_CYCLE_TIME}$)

Einzugeben ist der Wert der Zeitkonstanten in Sekunden [s] in das Maschinendatum:

$\text{MD32895 \$MA_DESVAL_DELAY_TIME[<Achse>]}$

Beispiel: Lageregler-Takt: 2 ms \Rightarrow einstellbare Zeitkonstante: 0,0 bis 0,128 s

Hinweis

Die Zeitkonstante des Sollwert-Phasenfilters verzögert das Ansprechverhalten der Achse bei z. B. Gewindebohren, Rückzugsbewegungen oder Genauhalt/Satzwechsel. Es wird deshalb empfohlen, die Zeitkonstante so klein wie möglich einzustellen.

Begrenzung auf den Maximalwert

Eine größere Zeitkonstante als der erlaubte Maximalwert ($64 \cdot \text{Lageregler-Takt}$) wird intern auf den Maximalwert begrenzt. Eine Meldung/Alarm wird nicht angezeigt.

Parametrierung: Aktivierung

Die Funktion des achsspezifischen Sollwert-Phasenfilters muss über folgendes Maschinendatum aktiviert werden:

MD32890 \$MA_DESVAL_DELAY_ENABLE[<Achse>] = "TRUE"

Beispiele

Annahme: Lageregler-Takt = 2 ms

1. MD32890 \$MA_DESVAL_DELAY_ENABLE[<Achse>] = "FALSE"
MD32895 \$MA_DESVAL_DELAY_TIME[<Achse>] = <Zeitkonstante>
 - Sollwert-Phasenfilter: nicht aktiv
 - Zeitkonstante: ohne Bedeutung
2. MD32890 \$MA_DESVAL_DELAY_ENABLE[<Achse>] = TRUE
MD32895 \$MA_DESVAL_DELAY_TIME[<Achse>] = 0,002
 - Sollwert-Phasenfilter: aktiv
 - Zeitkonstante: 2 ms \Rightarrow Die Ausgabe der Sollwerte wird um einen Lageregler-Takt verzögert.
3. MD32890 \$MA_DESVAL_DELAY_ENABLE[<Achse>] = TRUE
MD32895 \$MA_DESVAL_DELAY_TIME[<Achse>] = 0,256
 - Sollwert-Phasenfilter: aktiv
 - eingestellte Zeitkonstante: 256 ms;
maximal mögliche Zeitkonstante: $64 \cdot 2 \text{ ms} = 128 \text{ ms}$
 \Rightarrow intern wirksame Zeitkonstante: 128 ms

Randbedingungen

SINUMERIK Safety Integrated

Das Sollwert-Phasenfilter verzögert die Ausgabe von achsspezifischen Sollwerten z. B. bei Rückzugsbewegungen (ESR bei Stop E). Das Sollwert-Phasenfilter hat aber keinen Einfluss auf die Abschaltvorgänge, z. B. Aktivierungszeitpunkt von SBH.

8.6.4 Lageregler: Lagedifferenz-Aufschaltung

Voraussetzungen

- Die Funktion ist nur bei Achsen mit zwei Gebern nutzbar:
MD30200 \$MA_NUM_ENCS = 2
Einer der Geber muss als indirektes Messsystem, der andere als direktes Messsystem projektiert sein:
 - Direktes Messsystem:
MD31040 \$MA_ENC_IS_DIRECT[1]=1
Geber für Lageistwerterfassung ist direkt an der Maschine angebracht (Lastgeber).
 - Indirektes Messsystem:
MD31040 \$MA_ENC_IS_DIRECT[0]=0
Geber für Lageistwerterfassung ist am Motor angebracht (Motorgeber).
- Als Standard-Telegrammtyp für PROFIdrive muss sowohl im Antrieb als auch in der NC (MD13060 \$MN_DRIVE_TELEGRAM_TYPE) Telegrammtyp 136 oder 138 projektiert sein.

Funktion

Bei aktiver Lagedifferenz-Aufschaltung wird die Differenzlage zwischem dem direkten und indirekten Messsystem einer Achse ermittelt und entsprechend der Einstellung des Wichtungsfaktors als Zusatz-Stromsollwert zur Vorsteuerung im Lageregler-Takt aufgeschaltet. Die dadurch bewirkte Schwingungsdämpfung verbessert das Stabilitäts- und Positionierverhalten der Achse.

Anwendung

Die Funktion wird bei Achsen mit verstärkter Schwingungsneigung eingesetzt.

Wirksamkeit

Die Funktion ist nur wirksam bei Achsen mit niedriger Eigenfrequenz (bis ca. 20 Hz).

Aktivierung/Parametrierung

Die Funktion wird aktiviert durch die Angabe des Wichtungsfaktors:

MD32950 \$MA_POSCTRL_DAMPING (Dämpfung des Drehzahlregelkreises) = <Wert>

Wertebereich: -100% ... +100%

Ein Eingabewert "100%" bedeutet: Es wird ein Zusatz-Moment entsprechend SINAMICS-Parameter p2003 aufgeschaltet, wenn die ermittelte Lagedifferenz zwischen den beiden Messsystemen folgenden Wert erreicht:

- bei Linearmotoren: 1 mm
- bei Linearachse mit rotatorischem Motor: MD31030 \$MA_LEADSCREW_PITCH (Steigung der Kugelrollspindel)
- bei Rundachse/Spindel: 360 Grad

Standardeinstellung ist 0. Die Lagedifferenz-Aufschaltung ist in diesem Fall inaktiv.

Hinweis

Der Wichtungsfaktor MD32950 \$MA_POSCTRL_DAMPING ist z. B. anhand von Sprungantworten einstellbar.

Wenn die Regelung an die Stabilitätsgrenze kommt (Schwingungsneigung verstärkt sich), dann ist der Parameter entweder zu groß eingestellt oder der Wert hat das falsche Vorzeichen.

8.6.5 Lageregelung mit PI-Regler

Funktion

Im Standardfall ist der Kern des Lagereglers ein P-Regler. Für besondere Einsätze (wie Elektronisches Getriebe) ist die Zuschaltung eines Integral-Anteils möglich. Der dann vorliegende PI-Regler regelt den Fehler zwischen Soll- und Istposition bei entsprechender Einstellung der zugehörigen Maschinendaten zu Null in endlicher, einstellbarer Zeit aus.



VORSICHT

Überschwingen der Istposition bei aktiviertem PI-Regler

Der Anwender muss in diesem Fall prüfen, ob dies für die jeweilige Nutzung zulässig bzw. tolerierbar ist. Für die Nutzung der Funktion sind regeltechnische Fachkenntnisse und Messungen mit Servo-Trace unumgänglich. Bei falschen Einstellungen der entsprechenden Maschinendaten besteht durch das mögliche Auftreten von Instabilitäten Maschinenschadengefahr.

Vorgehensweise

1. Optimieren Sie den Lageregelkreis zunächst als P-Regler mit den in den vorangehenden Unterkapiteln beschriebenen Möglichkeiten.
2. Vergrößern Sie die Toleranzen folgender Maschinendaten für die Dauer der Messungen zur Feststellung der Qualität der Lageregelung mit PI-Lageregler:
 - MD36020 \$MA_POSITIONING_TIME
 - MD36030 \$MA_STANDSTILL_POS_TOL
 - MD36040 \$MA_STANDSTILL_DELAY_TIME
 - MD36400 \$MA_CONTOUR_TOL

8.6 Optimierung der Regelung

3. Aktivieren Sie den Lageregelkreis als PI-Regler durch Einstellung folgender Maschinendaten:
MD32220 \$MA_POSCTRL_INTEGR_ENABLE ; Wert 1 setzen
MD32210 \$MA_POSCTRL_INTEGR_TIME ; Nachstellzeit [s]
Wirkungsweise der Nachstellzeit:
 - $T_n \rightarrow 0$:
Der Regelfehler wird schnell ausgeregelt, der Regelkreis kann jedoch instabil werden.
 - $T_n \rightarrow \infty$:
Wirksamkeit des Integralanteils geht gegen 0. Verhalten des Reglers wie reiner P-Regler.
4. Suchen Sie zwischen diesen beiden Extremfällen den für die Anwendung richtigen Kompromiss für T_n .

Hinweis

T_n darf nicht zu nahe der Stabilitätsgrenze gewählt werden, da bei Auftreten einer Instabilität Maschinenschadengefahr besteht.

Es gilt daher die Empfehlung, T_n nicht kleiner als 1 s einzustellen.

Zeichnen Sie mit dem Servo-Trace das Einfahren eines Automatikprogramms zum Hin- und Herfahren in eine Zielposition auf.

5. Lassen Sie im Servo-Trace darstellen:
 - Schleppfehler
 - Istgeschwindigkeit
 - Istposition
 - Sollposition
6. Setzen Sie die Toleranzwerte in den folgenden Maschinendaten wieder auf die notwendigen Werte zurück, wenn für T_n der optimale Wert gefunden ist:
 - MD36020 \$MA_POSITIONING_TIME
 - MD36030 \$MA_STANDSTILL_POS_TOL
 - MD36040 \$MA_STANDSTILL_DELAY_TIME
 - MD36400 \$MA_CONTOUR_TOL

Randbedingungen

DSC

Wird die Integrator-Funktion verwendet, muss DSC (Dynamic Stiffness Control) ausgeschaltet werden.

Beispiel

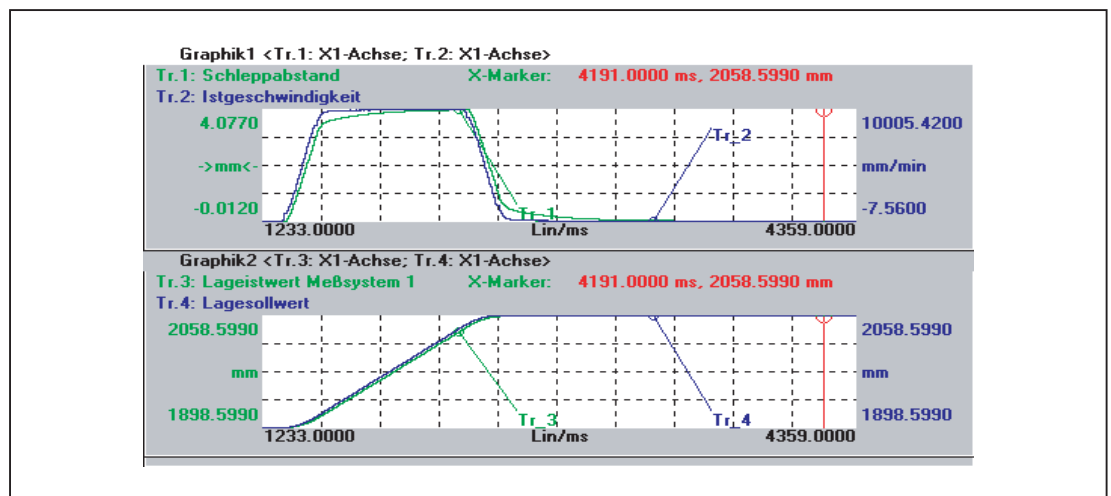
Einstellerggebnis nach mehreren Iterationsschritten für K_R und T_n

Maschinendaten-Einstellungen:

- MD32220 \$MA_POSCTRL_INTEGR_ENABLE = 1
- MD32210 \$MA_POSCTRL_INTEGR_TIME = 0.003
- MD32200 \$MA_POSCTRL_GAIN[1] = 5.0

Parametersatzanwahl 0

Es wurden Schleppabstand, Istgeschwindigkeit, Lageistwert und Lagesollwert jeweils per Servo-Trace aufgezeichnet. Beim Fahren im JOG-Betrieb wurde schließlich der im folgenden Bild dargestellte Verlauf der einzelnen Daten aufgezeichnet.



8.7 Datenlisten

8.7.1 Maschinendaten

8.7.1.1 Anzeige-Maschinendaten

Nummer	Bezeichner: \$MM_	Beschreibung
9004	DISPLAY_RESOLUTION	Anzeigefeinheit
9010	SPIND_DISPLAY_RESOLUTION	Anzeigefeinheit für Spindeln
9011	DISPLAY_RESOLUTION_INCH	Anzeigefeinheit INCH-Maßsystem

8.7.1.2 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10000	AXCONF_MACHAX_NAME_TAB	Maschinenachsname
10050	SYSCLOCK_CYCLE_TIME	Systemgrundtakt
10070	IPO_SYSCLOCK_TIME_RATIO	Faktor für Interpolatortakt
10060	POSCTRL_SYSCLOCK_TIME_RATIO	Faktor für Lageregeltakt
10200	INT_INCR_PER_MM	Rechenfeinheit für Linearpositionen
10210	INT_INCR_PER_DEG	Rechenfeinheit für Winkelpositionen
10220	SCALING_USER_DEF_MASK	Aktivierung der Normierungsfaktoren
10230	SCALING_FACTORS_USER_DEF	Normierungsfaktoren der physikalischen Größen
10240	SCALING_SYSTEM_IS_METRIC	Grundsystem metrisch
10250	SCALING_VALUE_INCH	Umrechnungsfaktor für Umschaltung auf Inch-System
10260	CONVERT_SCALING_SYSTEM	Grundsystem Umschaltung aktiv
10270	POS_TAB_SCALING_SYSTEM	Maßsystem der Positionstabellen
10290	CC_TDA_PARAM_UNIT	Physikalische Einheiten der Werkzeugdaten für CC
10292	CC_TOA_PARAM_UNIT	Physikalische Einheiten der Schneidendaten für CC
13050	DRIVE_LOGIC_ADDRESS	Logische Antriebsadressen
13060	DRIVE_TELEGRAM_TYPE	Standard-Telegramm-Typ für PROFIBUS-DP
13070	DRIVE_FUNCTION_MASK	Benutzte DP-Funktion
13080	DRIVE_TYPE_DP	Antriebsart PROFIBUS-DP

8.7.1.3 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20150	GCODE_RESET_VALUES	Löschstellung der G-Gruppen

8.7.1.4 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30110	CTRLOUT_MODULE_NR	Sollwertzuordnung: Antriebsnummer
30120	CTRLOUT_NR	Sollwertzuordnung: Sollwertausgang auf Antriebsmodul
30130	CTRLOUT_TYPE	Ausgabeart des Sollwertes
30200	NUM_ENCS	Anzahl der Geber
30220	ENC_MODULE_NR	Istwertzuordnung: Antriebsmodulnr.
30230	ENC_INPUT_NR	Istwertzuordnung: Eingang auf Antriebsmodul
30240	ENC_TYPE	Art der Istwerterfassung (Lageistwert)
30242	ENC_IS_INDEPENDENT	Geber ist unabhängig
30300	IS_ROT_AX	Rundachse

Nummer	Bezeichner: \$MA_	Beschreibung
31000	ENC_IS_LINEAR	Direktes Messsystem (Linearmaßstab)
31010	ENC_GRID_POINT_DIST	Teilungsperiode bei Linearmaßstäben
31020	ENC_RESOL	Geberstriche pro Umdrehung
31030	LEADSCREW_PITCH	Steigung der Kugelrollspindel
31040	ENC_IS_DIRECT	Geber ist direkt an der Maschine angebracht
31044	ENC_IS_DIRECT2	Geber am Vorsatz-Getriebe
31050	DRIVE_AX_RATIO_DENOM	Nenner Lastgetriebe
31060	DRIVE_AX_RATIO_NUMERA	Zähler Lastgetriebe
31064	DRIVE_AX_RATIO2_DENOM	Nenner Vorsatz-Getriebe
31066	DRIVE_AX_RATIO2_NUMERA	Zähler Vorsatz-Getriebe
31070	DRIVE_ENC_RATIO_DENOM	Nenner Messgetriebe
31080	DRIVE_ENC_RATIO_NUMERA	Zähler Messgetriebe
31090	JOG_INCR_WEIGHT	Bewertung des Inkrements bei INC/Handrad
31200	SCALING_FACTOR_G70_G71	Faktor für die Umrechnung der Werte bei aktivem G70/ G71
32000	MAX_AX_VELO	Maximale Achsgeschwindigkeit
32100	AX_MOTION_DIR	Verfahrriichtung
32110	ENC_FEEDBACK_POL	Vorzeichen Istwert (Regelsinn)
32200	POSCTRL_GAIN	K _v -Faktor
32210	POSCTRL_INTEGR_TIME	Nachstellzeit Lageregler
32220	POSCTRL_INTEGR_ENABLE	Aktivierung Integralanteil Lageregler
32250	RATED_OUTVAL	Nenn-Ausgangsspannung
32260	RATED_VELO	Nenn-Motordrehzahl
32450	BACKLASH	Umkehrlose
32500	FRICT_COMP_ENABLE	Reibkompensation aktiv
32610	VELO_FFW_WEIGHT	Vorsteuerfaktor für Drehzahlvorsteuerung
32620	FFW_MODE	Vorsteuerungsart
32630	FFW_ACTIVATION_MODE	Vorsteuerung aktivieren von Programm
32650	AX_INERTIA	Trägheitsmoment für Drehmomentvorsteuerung
32652	AX_MASS	Achsmasse für Drehmomentvorsteuerung
32711	CEC_SCALING_SYSTEM_METRIC	Maßsystem der Durchhangkompensation
32800	EQUIV_CURRCTRL_TIME	Ersatzzeitkonstante Stromregelkreis für Vorsteuerung
32810	EQUIV_SPEEDCTRL_TIME	Ersatzzeitkonstante Drehzahlregelkreis für Vorsteuerung
32890	DESVAL_DELAY_ENABLE	Achsspezifisches Sollwert-Phasenfilter
32895	DESVAL_DELAY_TIME	Zeitkonstante für das achsspezifische Sollwert-Phasenfilter
32900	DYN_MATCH_ENABLE	Dynamik-Anpassung
32910	DYN_MATCH_TIME[n]	Zeitkonstante der Dynamikanpassung
32930	POSCTRL_OUT_FILTER_ENABLE	Aktivierung des Tiefpassfilters am Legereglerausgang
32950	POSCTRL_DAMPING	Dämpfung des Drehzahlregelkreises
33000	FIPO_TYPE	Feininterpolortyp

8.7 Datenlisten

Nummer	Bezeichner: \$MA_	Beschreibung
34320	ENC_INVERS[n]	Längen-Messsystem ist gegensinnig
35100	SPIND_VELO_LIMIT	Maximale Spindeldrehzahl
36200	AX_VELO_LIMIT[n]	Schwellwert für Geschwindigkeitsüberwachung
36210	CTRL_OUT_LIMIT[n]	Maximaler Drehzahlsollwert
36400	AX_JERK_ENABLE	Achsspezifische Ruckbegrenzung
36410	AX_JERK_TIME	Zeitkonstante für das achsspezifische Ruckfilter
36500	ENC_CHANGE_TOL	Max. Toleranz bei Lageistwertumschaltung
36510	ENC_DIFF_TOL	Toleranz Messsystem-Gleichlauf
36700	ENC_COMP_ENABLE[n]	Interpolatorische Kompensation

8.7.2 Signale

8.7.2.1 Signale von NC

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Änderungszähler Maßsystem inch/metrisch	DB10.DBB71	DB2700.DBB0015
Inch-Maßsystem	DB10.DBX107.7	DB2700.DBX1.7

8.7.2.2 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Lagemesssystem 1	DB31,DBX1.5	DB38xx.DBX1.5
Lagemesssystem 2	DB31,DBX1.6	DB38xx.DBX1.6

H2: Hilfsfunktionsausgaben an PLC

9.1 Kurzbeschreibung

9.1.1 Funktion

Hilfsfunktionen bieten die Möglichkeit, Systemfunktionen des NC und PLC-Anwenderfunktionen zu aktivieren. Hilfsfunktionen können programmiert werden in:

- Teileprogrammen
- Synchronaktionen
- Anwenderzyklen

Ausführliche Informationen zur Verwendung von Hilfsfunktionsausgaben in Synchronaktionen siehe:

Literatur:

Funktionshandbuch Synchronaktionen

Vordefinierte Hilfsfunktionen

Vordefinierte Hilfsfunktionen aktivieren Systemfunktionen. Zusätzlich wird die Hilfsfunktion an die NC/PLC-Nahtstelle ausgegeben.

Vordefiniert sind folgende Hilfsfunktionen:

Typ	Funktion	Beispiel	Bedeutung
M	Zusatzfunktion	M30	Programmende
S	Spindelfunktion	S100	Spindeldrehzahl 100 (z. B. 1/min)
T	Werkzeugnummer	T3	Werkzeugnummer 3
D, DL	Werkzeugkorrektur	D1	Werkzeugschneidennummer 1
F	Vorschub	F1000	Vorschub 1000 (z. B. mm/min)

Anwenderdefinierte Hilfsfunktionen

Anwenderdefinierte Hilfsfunktionen sind entweder erweiterte vordefinierte Hilfsfunktionen oder anwenderspezifische Hilfsfunktionen.

Erweiterung von vordefinierten Hilfsfunktionen

Die Erweiterung von vordefinierten Hilfsfunktionen bezieht sich auf den Parameter "Adresserweiterung". Über die Adresserweiterung wird die Nummer der Spindel definiert, auf die sich die Hilfsfunktion bezieht. Vordefiniert ist z. B. die Spindelfunktion M3 (Spindel rechts) für die Masterspindel eines Kanals. Ist einem Kanal eine 2. Spindel zugeordnet, ist eine

entsprechende anwenderdefinierte Hilfsfunktionen zu definieren, welche die vordefinierte Hilfsfunktion erweitert.

Typ	Funktion	Beispiel	Bedeutung
M	Zusatzfunktion	M2=3	2. Spindel: Spindel rechts
S	Spindelfunktion	S2=100	2. Spindel: Spindeldrehzahl = 100 (z. B. 1/min)
T	Werkzeugnummer	T2=3	

Anwenderspezifische Hilfsfunktionen

Über anwenderspezifische Hilfsfunktionen werden keine Systemfunktionen aktiviert. Anwenderspezifische Hilfsfunktionen werden lediglich an die NC/PLC-Nahtstelle ausgegeben. Die Funktionalität der Hilfsfunktion ist vom Maschinenhersteller/Anwender im PLC-Anwenderprogramm zu realisieren.

Typ	Funktion	Beispiel	Bedeutung
H ¹⁾	Hilfsfunktion	H2=5	Anwenderspezifische Funktion

¹⁾ Empfehlung

9.1.2 Definition einer Hilfsfunktion

Eine Hilfsfunktion ist über folgende Parameter definiert:

- **Typ, Adresserweiterung und Wert**
Die 3 Parameter werden an die NC/PLC-Nahtstelle ausgegeben.
- **Ausgabeverhalten**
Über das Hilfsfunktionsspezifische Ausgabeverhalten wird festgelegt, wie lange eine Hilfsfunktion an die NC/PLC-Nahtstelle ausgegeben wird und wann die Ausgabe, bezogen auf die im gleichen Teileprogrammsatz programmierte Verfahrenbewegung, erfolgt.
- **Gruppenzuordnung**
Eine Hilfsfunktion kann einer bestimmten Hilfsfunktionsgruppe zugeordnet werden. Für jede Hilfsfunktionsgruppe kann ein eigenes Ausgabeverhalten festgelegt werden. Dieses wird wirksam, wenn kein Hilfsfunktionsspezifisches Ausgabeverhalten festgelegt wurde. Darüber hinaus beeinflusst die Gruppenzugehörigkeit die Ausgabe einer Hilfsfunktion nach Satzsuchlauf.

Ausführliche Informationen zur Hilfsfunktionsausgabe an die NC/PLC-Nahtstelle siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)".

9.1.3 Übersicht der Hilfsfunktionen

M-Funktionen

M (Zusatzfunktion)					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich ¹⁰⁾	Typ	Bedeutung	Anzahl ⁸⁾

M (Zusatzfunktion)					
0 (implizit)	---	0 ... 99	INT	Funktion	5
Wertebereich	Bedeutung	Wertebereich ¹¹⁾	Typ	Bedeutung	Anzahl ⁸⁾
1 ... 20	Spindelnummer	1 ... 99	INT	Funktion	5
Wertebereich	Bedeutung	Wertebereich ¹²⁾	Typ	Bedeutung	Anzahl ⁸⁾
0 ... 99	beliebig	100 ... 2147483647	INT	Funktion	5

⁸⁾ Siehe "Bedeutung der Fußnoten" am Ende der Übersicht

¹⁰⁾ Für den Wertebereich 0 ... 99 ist die Adresserweiterung 0. Zwingend ohne Adresserweiterung: M0, M1, M2, M17, M30

¹¹⁾ M3, M4, M5, M19, M70: Die Adresserweiterung ist die Spindel-Nummer, z. B. M2=5 ⇒ Spindel-Halt (M5) für Spindel 2.
Ohne Adresserweiterung wirkt die M-Funktion auf die Masterspindel.

¹²⁾ Anwenderspezifische M-Funktionen.

Verwendung

Maschinenfunktionen synchron zum Teileprogramm steuern.

Weitere Informationen

- Folgende M-Funktionen haben vordefinierte Bedeutung: M0, M1, M2, M17, M30, M3, M4, M5, M6, M19, M70, M40, M41, M42, M43, M44, M45.
- Den M-Funktionen (M0 - M99) ist ein dynamisches NC/PLC-Nahtstellensignal zur Anzeige der Gültigkeit zugeordnet. Zusätzlich können 64 weitere Signale für eigene M-Funktionen zugeordnet werden (siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)").
- Für Unterprogramme ist über Maschinendatum einstellbar, ob eine Ausgabe der M-Funktion für das Teileprogrammende M17, M2 oder M30 an die PLC erfolgen soll: MD20800 \$MC_SPF_END_TO_VDI (Unterprogrammende an PLC)
- Für die vordefinierten M-Funktionen M40 ... M45 kann die Ausgabespezifikation nur eingeschränkt umprojektiert werden.

- Die vordefinierten Hilfsfunktionen M0, M1, M17, M30, M6, M4, M5 lassen sich nicht umprojektieren.
- M-Funktionsspezifische Maschinendaten:
 - MD10800 \$MN_EXTERN_CHAN_SYNC_M_NO_MIN
 - MD10802 \$MN_EXTERN_CHAN_SYNC_M_NO_MAX
 - MD10804 \$MN_EXTERN_M_NO_SET_INT
 - MD10806 \$MN_EXTERN_M_NO_DISABLE_INT
 - MD10814 \$MN_EXTERN_M_NO_MAC_CYCLE
 - MD10815 \$MN_EXTERN_M_NO_MAC_CYCLE_NAME
 - MD20094 \$MC_SPIND_RIGID_TAPPING_M_NR
 - MD20095 \$MC_EXTERN_RIGID_TAPPING_M_NR
 - MD20096 \$MC_T_M_ADDRESS_EXT_IS_SPINO
 - MD22200 \$MC_AUXFU_M_SYNC_TYPE
 - MD22530 \$MC_TOCARR_CHANGE_M_CODE
 - MD22532 \$MC_GEOAX_CHANGE_M_CODE
 - MD22534 \$MC_TRAFO_CHANGE_M_CODE
 - MD22560 \$MC_TOOL_CHANGE_M_CODE

S-Funktionen

S (Spindelfunktion)					
Adresserweiterung ¹⁰⁾		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
0 ... 20	Spindelnummer ⁵⁾	0 ... $\pm 3.4028 \exp 38$ ³⁾	REAL	Spindeldrehzahl	3

³⁾ ⁵⁾ ⁸⁾ Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

¹⁰⁾ Ohne Angabe einer Adresserweiterung wird die Master-Spindel des Kanals adressiert.

Verwendung

Spindeldrehzahl.

Weitere Informationen

- Die S-Funktionen sind standardmäßig der 3. Hilfsfunktionsgruppe zugeordnet.
- Ohne Adresserweiterung beziehen sich die S-Funktionen auf die Masterspindel des Kanals.
- S-Funktionsspezifische Maschinendaten:
MD22210 \$MC_AUXFU_S_SYNC_TYPE (Ausgabezeitpunkt der S-Funktionen)

H-Funktionen

Die Funktionalität einer H-Funktion ist im PLC-Anwenderprogramm zu realisieren.

H (Hilfsfunktion) ¹⁰⁾					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
0 ... 99	beliebig	- 2147483648 ... + 2147483647	INT	beliebig	3
		0 ... $\pm 3.4028 \exp 38$ ^{2) 3) 4)}	REAL		

^{2) 3) 4) 8)} Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

Verwendung

Anwenderspezifische Hilfsfunktionen.

Weitere Informationen

H-Funktionsspezifische Maschinendaten:

- MD22110 \$MC_AUXFU_H_TYPE_INT (Typ von H-Hilfsfunktionen ist Integer)
- MD22230 \$MC_AUXFU_H_SYNC_TYPE (Ausgabezeitpunkt der H-Funktionen)

T-Funktionen

Werkzeugnamen werden nicht an die NC/PLC-Nahtstelle ausgegeben.

T (Werkzeugnummer) ^{1) 5) 6)}					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
1 ... 12	Spindelnummer (bei aktiver Werkzeug-verwaltung)	0 ... 32000 (auch symbolische Werkzeugnamen bei aktiver Werkzeugverwaltung)	INT	Anwahl des Werkzeugs	1

^{1) 5) 6) 8)} Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

Verwendung

Werkzeuganwahl.

Weitere Informationen

- Identifikation der Werkzeuge, wahlweise über Werkzeugnummer oder Platznummer (siehe Kapitel "W1: Werkzeugkorrektur (Seite 1475)").

Literatur

Funktionshandbuch Werkzeugverwaltung

- Mit T0 wird das aktuelle Werkzeug aus der Werkzeughalterung entfernt und kein neues eingewechselt (Grundeinstellung).
- T-Funktionsspezifische Maschinendaten:
MD22220 \$MC_AUXFU_T_SYNC_TYPE (Ausgabezeitpunkt der T-Funktionen)

D-Funktionen

Die Abwahl der Werkzeugkorrektur erfolgt mit D0. Vorbesetzung ist D1.

D (Werkzeugkorrektur)					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
---	---	0 ... 9	INT	Anwahl der Werkzeugkorrektur	1

⁸⁾ Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

Verwendung

Anwahl der Werkzeugkorrektur.

Weitere Informationen

- Grundstellung: D1
- Die Default-Werkzeugschneide nach Werkzeugwechsel ist parametrierbar über:
MD20270 \$MC_CUTTING_EDGE_DEFAULT (Grundstellung der Werkzeugschneide ohne Programmierung)
- Abwahl der Werkzeugkorrektur: D0
- D-funktionsspezifische Maschinendaten:
MD22250 \$MC_AUXFU_D_SYNC_TYPE (Ausgabezeitpunkt der D-Funktionen)

DL-Funktionen

Die mit DL angewählte Werkzeugsummenkorrektur bezieht sich auf aktive D-Nummer.

DL (Werkzeugsummenkorrektur)					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
---	---	0 ... 6	INT	Anwahl der Werkzeugsummenkorrektur	1

⁸⁾ Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

Verwendung

Anwahl der Werkzeugsummenkorrektur bezüglich einer aktiven Werkzeugkorrektur.

Weitere Informationen

- Grundstellung: DL = 0
- DL-Werte können nicht über Synchronaktionen an die PLC ausgegeben werden.
- Voreinstellung der Werkzeugsummenkorrektur ohne aktive DL-Funktion:
MD20272 \$MC_SUMCORR_DEFAULT (Grundstellung Summenkorrektur ohne Programm)

- Abwahl der Werkzeugsummenkorrektur: DL = 0
- DL-Funktionsspezifische Maschinendaten:
MD22252 \$MC_AUXFU_DL_SYNC_TYPE (Ausgabezeitpunkt DL-Funktionen)

F-Funktionen

F (Bahnvorschub)					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
---	---	0.001 ... 999 999.999	REAL	Bahnvorschub	6

⁸⁾ Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

Verwendung

Bahngeschwindigkeit.

Weitere Informationen

F-funktionsspezifische Maschinendaten:

- MD22240 \$MC_AUXFU_F_SYNC_TYPE (Ausgabezeitpunkt der F-Funktionen)

FA-Funktionen

FA (achsspezifischer Vorschub)					
Adresserweiterung		Wert			
Wertebereich	Bedeutung	Wertebereich	Typ	Bedeutung	Anzahl ⁸⁾
1 - 31	Achsennummer	0.001 ... 999 999.999	REAL	Achsvorschub	6

⁸⁾ Siehe "Bedeutung der Fußnoten" am Ende der Übersicht.

Verwendung

Achsspezifische Geschwindigkeit.

Weitere Informationen

F-funktionsspezifische Maschinendaten:

- MD22240 \$MC_AUXFU_F_SYNC_TYPE (Ausgabezeitpunkt der F-Funktionen)

Bedeutung der Fußnoten

- ¹⁾ Bei aktiver Werkzeugverwaltung wird weder ein T-Änderungssignal noch ein T-Wort an die kanalspezifische NC/PLC-Nahtstelle ausgegeben.
- ²⁾ Der Typ für die Werte kann durch MD22110 \$MC_AUXFU_H_TYPE_INT vom Anwender gewählt werden.

- 3) Aufgrund der begrenzten Anzeigemöglichkeiten auf den Bildschirmen der Bediengeräte sind die angezeigten Werte des Typs REAL begrenzt auf:
–999 999 999.9999 bis 999 999 999.9999
Die NC rechnet intern aber mit der vollen Genauigkeit.
- 4) Die REAL–Werte werden gerundet an die PLC ausgegeben, bei der Einstellung des Maschinendatums:
MD22110 \$MC_AUXFU_H_TYPE_INT = 1 (Typ von H-Hilfsfunktionen ist Integer)
Das PLC-Anwenderprogramm muss den übergebenen Wert entsprechend der Maschinendaten-Einstellung interpretieren.
- 5) Bei aktiver Werkzeugverwaltung kann die Bedeutung der Adresserweiterung parametrierbar werden. Adresserweiterung = 0 bedeutet, der Wert ist durch den der Masterspindelnummer zu ersetzen; ist somit gleichbedeutend mit Nichtprogrammierung der Adresserweiterung. Bei Satzsuchlauf aufgesammelte Hilfsfunktionen M19 "Spindel positionieren" werden nicht an die PLC ausgegeben.
- 6) M6: Wertebereich der Adresserweiterung:
- ohne Werkzeugverwaltung: 0 ... 99
- mit Werkzeugverwaltung: 0 ... maximale Spindelnummer
0: Ersetzen durch Wert der Masterspindelnummer bzw. des Masterwerkzeughalters
- 7) Mit aktiver Werkzeugverwaltung kann die Hilfsfunktion M6 "Werkzeugwechsel" nur einmal in einem Teileprogrammsatz programmiert werden, unabhängig davon, welche Adresserweiterungen programmiert werden.
- 8) Maximale Anzahl von Hilfsfunktionen pro Teileprogrammsatz.

9.2 Vordefinierte Hilfsfunktionen

Funktion

Jeder vordefinierten Hilfsfunktion ist nicht änderbar eine Systemfunktion zugeordnet. Ist in einem Teileprogramm / Zyklus eine vordefinierte Hilfsfunktion programmiert, wird diese über die NC/PLC-Nahtstelle an die PLC ausgegeben und im NC die entsprechende Systemfunktion ausgeführt.

Definition einer vordefinierten Hilfsfunktion

Die Parameter der vordefinierten Hilfsfunktionen sind in Maschinendaten hinterlegt und können teilweise geändert werden. Alle Maschinendaten die zu einer Hilfsfunktion gehören, haben den gleichen Index <n>.

- MD22040 \$MC_AUXFU_PREDEF_GROUP[<n>] (Gruppenzuordnung von vordefinierten Hilfsfunktionen)
- MD22050 \$MC_AUXFU_PREDEF_TYPE[<n>] (Typ von vordefinierten Hilfsfunktionen)
- MD22060 \$MC_AUXFU_PREDEF_EXTENSION[<n>] (Adresserweiterung für vordefinierte Hilfsfunktionen)

- MD22070 \$MC_AUXFU_PREDEF_VALUE[<n>] (Wert von vordefinierten Hilfsfunktionen)
- MD22080 \$MC_AUXFU_PREDEF_SPEC[<n>] (Ausgabeverhalten von vordefinierten Hilfsfunktionen)

9.2.1 Übersicht: vordefinierten Hilfsfunktionen

Bedeutung der in den nachfolgenden Tabellen aufgeführten Parameter:

Parameter	Bedeutung
Index <n>	Maschinendatenindex der Parameter einer Hilfsfunktion
Typ	MD22050 \$MC_AUXFU_PREDEF_TYPE[<n>]
Adresserweiterung	MD22060 \$MC_AUXFU_PREDEF_EXTENSION[<n>]
Wert	MD22070 \$MC_AUXFU_PREDEF_VALUE[<n>]
Gruppe	MD22040 \$MC_AUXFU_PREDEF_GROUP[<n>]

Vordefinierte Hilfsfunktionen

Allgemeine Hilfsfunktionen, Teil 1					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Halt	0	M	0	0	1
bedingter Halt	1	M	0	1	1
Unterprogramm Ende	2	M	0	2	1
	3	M	0	17	1
	4	M	0	30	1
Werkzeugwechsel	5	M	(0)	6 ¹⁾	(1)

Spindel-spezifische Hilfsfunktionen, Spindel 1					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	6	M	1	3	(2)
Spindel links	7	M	1	4	(2)
Spindel halt	8	M	1	5	(2)
Spindel positionieren	9	M	1	19	(2)
Achsbetrieb	10	M	1	70 ²⁾	(2)
automatische Getriebestufe	11	M	1	40	(4)
Getriebestufe 1	12	M	1	41	(4)
Getriebestufe 2	13	M	1	42	(4)
Getriebestufe 3	14	M	1	43	(4)
Getriebestufe 4	15	M	1	44	(4)
Getriebestufe 5	16	M	1	45	(4)
Spindel-Drehzahl	17	S	1	-1	(3)

Allgemeine Hilfsfunktionen, Teil 2					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Vorschub	18	F	0	-1	(1)
Schneidenanwahl	19	D	0	-1	(1)
DL	20	L	0	-1	(1)
Werkzeuganwahl	21	T	(0)	-1	(1)
Halt (assoziiert)	22	M	0	-1 ³⁾	1
bedingter Halt (assoziiert)	23	M	0	-1 ⁴⁾	1
Unterprogramm Ende	24	M	0	-1 ⁵⁾	1
Nibbeln	25	M	0	20 ⁶⁾	(10)
Nibbeln	26	M	0	23 ⁶⁾	(10)
Nibbeln	27	M	0	22 ⁶⁾	(11)
Nibbeln	28	M	0	25 ⁶⁾	(11)
Nibbeln	29	M	0	26 ⁶⁾	(12)
Nibbeln	30	M	0	122 ⁶⁾	(11)
Nibbeln	31	M	0	125 ⁶⁾	(11)
Nibbeln	32	M	0	27 ⁶⁾	(12)

Spindel-spezifische Hilfsfunktionen, Spindel 2					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	33	M	2	3	(72)
Spindel links	34	M	2	4	(72)
Spindel halt	35	M	2	5	(72)
Spindel positionieren	36	M	2	19	(72)
Achsbetrieb	37	M	2	70 ²⁾	(72)
automatische Getriebestufe	38	M	2	40	(74)
Getriebestufe 1	39	M	2	41	(74)
Getriebestufe 2	40	M	2	42	(74)
Getriebestufe 3	41	M	2	43	(74)
Getriebestufe 4	42	M	2	44	(74)
Getriebestufe 5	43	M	2	45	(74)
Spindel-Drehzahl	44	S	2	-1	(73)

Spindel-spezifische Hilfsfunktionen, Spindel 3					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	45	M	3	3	(75)
Spindel links	46	M	3	4	(75)
Spindel halt	47	M	3	5	(75)
Spindel positionieren	48	M	3	19	(75)
Achsbetrieb	49	M	3	70 ²⁾	(75)

Spindel-spezifische Hilfsfunktionen, Spindel 3					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
automatische Getriebestufe	50	M	3	40	(77)
Getriebestufe 1	51	M	3	41	(77)
Getriebestufe 2	52	M	3	42	(77)
Getriebestufe 3	53	M	3	43	(77)
Getriebestufe 4	54	M	3	44	(77)
Getriebestufe 5	55	M	3	45	(77)
Spindel-Drehzahl	56	S	3	-1	(76)

Spindel-spezifische Hilfsfunktionen, Spindel 4					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	57	M	4	3	(78)
Spindel links	58	M	4	4	(78)
Spindel halt	59	M	4	5	(78)
Spindel positionieren	60	M	4	19	(78)
Achsbetrieb	61	M	4	70 ²⁾	(78)
automatische Getriebestufe	62	M	4	40	(80)
Getriebestufe 1	63	M	4	41	(80)
Getriebestufe 2	64	M	4	42	(80)
Getriebestufe 3	65	M	4	43	(80)
Getriebestufe 4	66	M	4	44	(80)
Getriebestufe 5	67	M	4	45	(80)
Spindel-Drehzahl	68	S	4	-1	(79)

Spindel-spezifische Hilfsfunktionen, Spindel 5					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	69	M	5	3	(81)
Spindel links	70	M	5	4	(81)
Spindel halt	71	M	5	5	(81)
Spindel positionieren	72	M	5	19	(81)
Achsbetrieb	73	M	5	70 ²⁾	(81)
automatische Getriebestufe	74	M	5	40	(83)
Getriebestufe 1	75	M	5	41	(83)
Getriebestufe 2	76	M	5	42	(83)
Getriebestufe 3	77	M	5	43	(83)
Getriebestufe 4	78	M	5	44	(83)
Getriebestufe 5	79	M	5	45	(83)
Spindel-Drehzahl	80	S	5	-1	(82)

Spindel-spezifische Hilfsfunktionen, Spindel 6					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	81	M	6	3	(84)
Spindel links	82	M	6	4	(84)
Spindel halt	83	M	6	5	(84)
Spindel positionieren	84	M	6	19	(84)
Achsbetrieb	85	M	6	70 ²⁾	(84)
automatische Getriebestufe	86	M	6	40	(86)
Getriebestufe 1	87	M	6	41	(86)
Getriebestufe 2	88	M	6	42	(86)
Getriebestufe 3	89	M	6	43	(86)
Getriebestufe 4	90	M	6	44	(86)
Getriebestufe 5	91	M	6	45	(86)
Spindel-Drehzahl	92	S	6	-1	(85)

Spindel-spezifische Hilfsfunktionen, Spindel 7					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	93	M	7	3	(87)
Spindel links	94	M	7	4	(87)
Spindel halt	95	M	7	5	(87)
Spindel positionieren	96	M	7	19	(87)
Achsbetrieb	97	M	7	70 ²⁾	(87)
automatische Getriebestufe	98	M	7	40	(89)
Getriebestufe 1	99	M	7	41	(89)
Getriebestufe 2	100	M	7	42	(89)
Getriebestufe 3	101	M	7	43	(89)
Getriebestufe 4	102	M	7	44	(89)
Getriebestufe 5	103	M	7	45	(89)
Spindel-Drehzahl	104	S	7	-1	(88)

Spindel-spezifische Hilfsfunktionen, Spindel 8					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	105	M	8	3	(90)
Spindel links	106	M	8	4	(90)
Spindel halt	107	M	8	5	(90)
Spindel positionieren	108	M	8	19	(90)
Achsbetrieb	109	M	8	70 ²⁾	(90)
automatische Getriebestufe	110	M	8	40	(92)
Getriebestufe 1	111	M	8	41	(92)
Getriebestufe 2	112	M	8	42	(92)

Spindel-spezifische Hilfsfunktionen, Spindel 8					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Getriebestufe 3	113	M	8	43	(92)
Getriebestufe 4	114	M	8	44	(92)
Getriebestufe 5	115	M	8	45	(92)
Spindel-Drehzahl	116	S	8	-1	(91)

Spindel-spezifische Hilfsfunktionen, Spindel 9					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	117	M	9	3	(93)
Spindel links	118	M	9	4	(93)
Spindel halt	119	M	9	5	(93)
Spindel positionieren	120	M	9	19	(93)
Achsbetrieb	121	M	9	70 ²⁾	(93)
automatische Getriebestufe	122	M	9	40	(95)
Getriebestufe 1	123	M	9	41	(95)
Getriebestufe 2	124	M	9	42	(95)
Getriebestufe 3	125	M	9	43	(95)
Getriebestufe 4	126	M	9	44	(95)
Getriebestufe 5	127	M	9	45	(95)
Spindel-Drehzahl	128	S	9	-1	(94)

Spindel-spezifische Hilfsfunktionen, Spindel 10					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	129	M	10	3	(96)
Spindel links	130	M	10	4	(96)
Spindel halt	131	M	10	5	(96)
Spindel positionieren	132	M	10	19	(96)
Achsbetrieb	133	M	10	70 ²⁾	(96)
automatische Getriebestufe	134	M	10	40	(98)
Getriebestufe 1	135	M	10	41	(98)
Getriebestufe 2	136	M	10	42	(98)
Getriebestufe 3	137	M	10	43	(98)
Getriebestufe 4	138	M	10	44	(98)
Getriebestufe 5	139	M	10	45	(98)
Spindel-Drehzahl	140	S	10	-1	(97)

Spindel-spezifische Hilfsfunktionen, Spindel 11					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	141	M	11	3	(99)
Spindel links	142	M	11	4	(99)
Spindel halt	143	M	11	5	(99)
Spindel positionieren	144	M	11	19	(99)
Achsbetrieb	145	M	11	70 ²⁾	(99)
automatische Getriebestufe	146	M	11	40	(101)
Getriebestufe 1	147	M	11	41	(101)
Getriebestufe 2	148	M	11	42	(101)
Getriebestufe 3	149	M	11	43	(101)
Getriebestufe 4	150	M	11	44	(101)
Getriebestufe 5	151	M	11	45	(101)
Spindel-Drehzahl	152	S	11	-1	(100)

Spindel-spezifische Hilfsfunktionen, Spindel 12					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	153	M	11	3	(102)
Spindel links	154	M	12	4	(102)
Spindel halt	155	M	12	5	(102)
Spindel positionieren	156	M	12	19	(102)
Achsbetrieb	157	M	12	70 ²⁾	(102)
automatische Getriebestufe	158	M	12	40	(104)
Getriebestufe 1	159	M	12	41	(104)
Getriebestufe 2	160	M	12	42	(104)
Getriebestufe 3	161	M	12	43	(104)
Getriebestufe 4	162	M	12	44	(104)
Getriebestufe 5	163	M	12	45	(104)
Spindel-Drehzahl	164	S	12	-1	(103)

Spindel-spezifische Hilfsfunktionen, Spindel 13					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	165	M	13	3	(105)
Spindel links	166	M	13	4	(105)
Spindel halt	167	M	13	5	(105)
Spindel positionieren	168	M	13	19	(105)
Achsbetrieb	169	M	13	70 ²⁾	(105)
automatische Getriebestufe	170	M	13	40	(107)
Getriebestufe 1	171	M	13	41	(107)
Getriebestufe 2	172	M	13	42	(107)

Spindel-spezifische Hilfsfunktionen, Spindel 13					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Getriebestufe 3	173	M	13	43	(107)
Getriebestufe 4	174	M	13	44	(107)
Getriebestufe 5	175	M	13	45	(107)
Spindel-Drehzahl	176	S	13	-1	(106)

Spindel-spezifische Hilfsfunktionen, Spindel 14					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	177	M	14	3	(108)
Spindel links	178	M	14	4	(108)
Spindel halt	179	M	14	5	(108)
Spindel positionieren	180	M	14	19	(108)
Achsbetrieb	181	M	14	70 ²⁾	(108)
automatische Getriebestufe	182	M	14	40	(110)
Getriebestufe 1	183	M	14	41	(110)
Getriebestufe 2	184	M	14	42	(110)
Getriebestufe 3	185	M	14	43	(110)
Getriebestufe 4	186	M	14	44	(110)
Getriebestufe 5	187	M	14	45	(110)
Spindel-Drehzahl	188	S	14	-1	(109)

Spindel-spezifische Hilfsfunktionen, Spindel 15					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	189	M	15	3	(111)
Spindel links	190	M	15	4	(111)
Spindel halt	191	M	15	5	(111)
Spindel positionieren	192	M	15	19	(111)
Achsbetrieb	193	M	15	70 ²⁾	(111)
automatische Getriebestufe	194	M	15	40	(113)
Getriebestufe 1	195	M	15	41	(113)
Getriebestufe 2	196	M	15	42	(113)
Getriebestufe 3	197	M	15	43	(113)
Getriebestufe 4	198	M	15	44	(113)
Getriebestufe 5	199	M	15	45	(113)
Spindel-Drehzahl	200	S	15	-1	(112)

Spindel-spezifische Hilfsfunktionen, Spindel 16					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	201	M	16	3	(114)
Spindel links	202	M	16	4	(114)
Spindel halt	203	M	16	5	(114)
Spindel positionieren	204	M	16	19	(114)
Achsbetrieb	205	M	16	70 ²⁾	(114)
automatische Getriebestufe	206	M	16	40	(116)
Getriebestufe 1	207	M	16	41	(116)
Getriebestufe 2	208	M	16	42	(116)
Getriebestufe 3	209	M	16	43	(116)
Getriebestufe 4	210	M	16	44	(116)
Getriebestufe 5	211	M	16	45	(116)
Spindel-Drehzahl	212	S	16	-1	(115)

Spindel-spezifische Hilfsfunktionen, Spindel 17					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	213	M	17	3	(117)
Spindel links	214	M	17	4	(117)
Spindel halt	215	M	17	5	(117)
Spindel positionieren	216	M	17	19	(117)
Achsbetrieb	217	M	17	70 ²⁾	(117)
automatische Getriebestufe	218	M	17	40	(119)
Getriebestufe 1	219	M	17	41	(119)
Getriebestufe 2	220	M	17	42	(119)
Getriebestufe 3	221	M	17	43	(119)
Getriebestufe 4	222	M	17	44	(119)
Getriebestufe 5	223	M	17	45	(119)
Spindel-Drehzahl	224	S	17	-1	(118)

Spindel-spezifische Hilfsfunktionen, Spindel 18					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	225	M	18	3	(120)
Spindel links	226	M	18	4	(120)
Spindel halt	227	M	18	5	(120)
Spindel positionieren	228	M	18	19	(120)
Achsbetrieb	229	M	18	70 ²⁾	(120)
automatische Getriebestufe	230	M	18	40	(122)
Getriebestufe 1	231	M	18	41	(122)
Getriebestufe 2	232	M	18	42	(122)

Spindel-spezifische Hilfsfunktionen, Spindel 18					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Getriebestufe 3	233	M	18	43	(122)
Getriebestufe 4	234	M	18	44	(122)
Getriebestufe 5	235	M	18	45	(122)
Spindel-Drehzahl	236	S	18	-1	(121)

Spindel-spezifische Hilfsfunktionen, Spindel 19					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	237	M	19	3	(123)
Spindel links	238	M	19	4	(123)
Spindel halt	239	M	19	5	(123)
Spindel positionieren	240	M	19	19	(123)
Achsbetrieb	241	M	19	70 ²⁾	(123)
automatische Getriebestufe	242	M	19	40	(125)
Getriebestufe 1	243	M	19	41	(125)
Getriebestufe 2	244	M	19	42	(125)
Getriebestufe 3	245	M	19	43	(125)
Getriebestufe 4	246	M	19	44	(125)
Getriebestufe 5	247	M	19	45	(125)
Spindel-Drehzahl	248	S	19	-1	(124)

Spindel-spezifische Hilfsfunktionen, Spindel 20					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Spindel rechts	249	M	20	3	(126)
Spindel links	250	M	20	4	(126)
Spindel halt	251	M	20	5	(126)
Spindel positionieren	252	M	20	19	(126)
Achsbetrieb	253	M	20	70 ²⁾	(126)
automatische Getriebestufe	254	M	20	40	(128)
Getriebestufe 1	255	M	20	41	(128)
Getriebestufe 2	256	M	20	42	(128)
Getriebestufe 3	257	M	20	43	(128)
Getriebestufe 4	258	M	20	44	(128)
Getriebestufe 5	259	M	20	45	(128)
Spindel-Drehzahl	260	S	20	-1	(127)

Werkzeughalter-spezifische Hilfsfunktionen, T-Hilfsfunktionen					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Werkzeugwahl	261	T	1	-1	129
Werkzeugwahl	262	T	2	-1	130
Werkzeugwahl	263	T	3	-1	131
Werkzeugwahl	264	T	4	-1	132
Werkzeugwahl	265	T	5	-1	133
Werkzeugwahl	266	T	6	-1	134
Werkzeugwahl	267	T	7	-1	135
Werkzeugwahl	268	T	8	-1	136
Werkzeugwahl	269	T	9	-1	137
Werkzeugwahl	270	T	10	-1	138
Werkzeugwahl	271	T	11	-1	139
Werkzeugwahl	272	T	12	-1	140
Werkzeugwahl	273	T	13	-1	141
Werkzeugwahl	274	T	14	-1	142
Werkzeugwahl	275	T	15	-1	143
Werkzeugwahl	276	T	16	-1	144
Werkzeugwahl	277	T	17	-1	145
Werkzeugwahl	278	T	18	-1	146
Werkzeugwahl	279	T	19	-1	147
Werkzeugwahl	280	T	20	-1	148

Werkzeughalter-spezifische Hilfsfunktionen, M6-Hilfsfunktionen					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Werkzeugwechsel	281	M	1	6 ¹⁾	149
Werkzeugwechsel	282	M	2	6 ¹⁾	150
Werkzeugwechsel	283	M	3	6 ¹⁾	151
Werkzeugwechsel	284	M	4	6 ¹⁾	152
Werkzeugwechsel	285	M	5	6 ¹⁾	153
Werkzeugwechsel	286	M	6	6 ¹⁾	154
Werkzeugwechsel	287	M	7	6 ¹⁾	155
Werkzeugwechsel	288	M	8	6 ¹⁾	156
Werkzeugwechsel	289	M	9	6 ¹⁾	157
Werkzeugwechsel	290	M	10	6 ¹⁾	158
Werkzeugwechsel	291	M	11	6 ¹⁾	159
Werkzeugwechsel	292	M	12	6 ¹⁾	160
Werkzeugwechsel	293	M	13	6 ¹⁾	161
Werkzeugwechsel	294	M	14	6 ¹⁾	162
Werkzeugwechsel	295	M	15	6 ¹⁾	163
Werkzeugwechsel	296	M	16	6 ¹⁾	164

Werkzeughalter-spezifische Hilfsfunktionen, M6-Hilfsfunktionen					
Systemfunktion	Index <n>	Typ	Adresserw.	Wert	Gruppe
Werkzeugwechsel	297	M	17	6 ¹⁾	165
Werkzeugwechsel	298	M	18	6 ¹⁾	166
Werkzeugwechsel	299	M	19	6 ¹⁾	167
Werkzeugwechsel	300	M	20	6 ¹⁾	168

Legende:

() Der Wert kann geändert werden.

¹⁾ Der Wert ist abhängig von Maschinendatum:

MD22560 \$MC_TOOL_CHANGE_M_MODE (M-Funktion für Werkzeugwechsel)

²⁾ Der Wert lässt sich über folgende Maschinendaten mit einem anderen Wert vorbesetzen:

MD20095 \$MC_EXTERN_RIGID_TAPPING_M_NR (M-Funktion für das Umschalten in den gesteuerten Achsbetrieb (Ext. mode))

MD20094 \$MC_SPIND_RIGID_TAPPING_M_NR (M-Funktion für das Umschalten in den gesteuerten Achsbetrieb)

Hinweis

An die PLC wird immer der Wert 70 ausgegeben.

³⁾ Der Wert wird eingestellt über Maschinendatum:

MD22254 \$MC_AUXFU_ASSOC_M0_VALUE (Zusätzliche M-Fkt. für Programm-Halt)

⁴⁾ Der Wert wird eingestellt über Maschinendatum:

MD22256 \$MC_AUXFU_ASSOC_M1_VALUE (Zusätzliche M-Fkt. für bedingten Halt)

⁵⁾ Der Wert wird eingestellt über Maschinendatum:

MD10714 \$MN_M_NO_FCT_EOP (M-Funktion für Spindel aktiv nach Reset)

⁶⁾ Der Wert wird eingestellt über Maschinendatum:

MD26008 \$MC_NIBBLE_PUNCH_CODE (Festlegung der M-Funktionen)

9.2.2 Übersicht: Ausgabeverhalten

Bedeutung der in der nachfolgenden Tabelle aufgeführten Parameter:

Parameter	Bedeutung
Index <n>	Maschinendatenindex der Parameter einer Hilfsfunktion
Ausgabeverhalten	MD22080 \$MC_AUXFU_PREDEF_SPEC[<n>], Bit 0 ... 18 Bit 19 ... 31: reserviert

Ausgabeverhalten der vordefinierten Hilfsfunktionen

Systemfunktion	Index <n>																			
	Ausgabeverhalten, Bit																			
18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Halt	0	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
bedingter Halt	1	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
Unterprogramm Ende	2	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
	3	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
	4	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
Werkzeugwechsel	5	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	(1)
Spindel rechts	6	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	(1)
Spindel links	7	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	(1)
Spindel halt	8	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	(1)
Spindel positionieren	9	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	(1)
Achsbetrieb	10	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	(1)
automatische Getriebe- stufe	11	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	0	0	1	(0)	(0)	(0)	(0)	(1)
Getriebestufe 1	12	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	0	0	1	(0)	(0)	(0)	(0)	(1)
Getriebestufe 2	13	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	0	0	1	(0)	(0)	(0)	(0)	(1)
Getriebestufe 3	14	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	0	0	1	(0)	(0)	(0)	(0)	(1)
Getriebestufe 4	15	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	0	0	1	(0)	(0)	(0)	(0)	(1)
Getriebestufe 5	16	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	0	0	1	(0)	(0)	(0)	(0)	(1)
Spindel-Drehzahl	17	0	0	0	(0)	0	0	0	(0)	(0)	(0)	0	(0)	(1)	(0)	(0)	(0)	0	(0)	(1)
Vorschub	18	0	0	0	(0)	0	0	0	(0)	(0)	(0)	0	(0)	(1)	(0)	0	(1)	0	(0)	(1)
Schneidenanwahl	19	0	0	0	(0)	0	0	0	(0)	(0)	(0)	0	(0)	(0)	(1)	0	(0)	0	(0)	(1)
DL	20	0	0	0	(0)	0	0	0	(0)	(0)	(0)	0	(0)	(0)	(1)	0	(0)	0	(0)	(1)
Werkzeuganwahl	21	0	0	0	(0)	0	0	0	(0)	(0)	(0)	0	(0)	(0)	(1)	0	(0)	0	(0)	(1)
Halt (assoziiert)	22	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
bedingter Halt (assozii- ert)	23	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
Unterprogramm Ende	24	0	0	0	(0)	0	0	0	0	(0)	0	0	1	0	0	0	0	0	(0)	(1)
Nibbeln	25	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	26	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	27	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	28	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	29	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	30	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	31	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)
Nibbeln	32	0	0	0	(0)	0	0	0	(0)	(0)	(0)	(0)	(0)	(1)	(0)	0	(0)	(0)	(0)	(1)

() Der Wert kann geändert werden.

Bedeutung der Bits

Bit	Bedeutung
0	<p>Quittierung "normal" nach einen OB1-Takt</p> <p>Eine Hilfsfunktion mit normaler Quittierung wird zu Beginn des OB1-Zyklus in die NC/PLC-Nahtstelle ausgegeben. Über das hilfsfunktionsspezifische Änderungssignal wird dem PLC-Anwenderprogramm angezeigt, dass die Hilfsfunktion gültig ist.</p> <p>Die Quittierung der Hilfsfunktion erfolgt, sobald der Organisationsbaustein OB1 einmal komplett durchlaufen wurde. Dies entspricht einem vollständigen PLC-Anwenderzyklus.</p> <p>Die Ausgabe einer Hilfsfunktion mit normaler Quittierung erfolgt synchron zum Teileprogrammsatz, in dem sie programmiert ist. Ist die Abarbeitung des Teileprogrammsatzes, z. B. Bahn- und/oder Positionierachsbewegungen, vor der Quittierung der Hilfsfunktion beendet, wird der Satzwechsel so lange verzögert, bis die Quittierung durch die PLC erfolgt.</p> <p>Bei Bahnsteuerbetrieb kann eine konstante Bahngeschwindigkeit im Zusammenhang mit Hilfsfunktion mit normaler Quittierung nur dann eingehalten werden, wenn die Hilfsfunktionsausgabe während der Verfahrbewegung erfolgt und vor dem Erreichen des Satzendes durch die PLC quittiert wurde.</p>
1	<p>Quittierung "quick" mit OB40</p> <p>Eine Hilfsfunktion mit schneller Quittierung wird vor dem nächsten OB1-Zyklus in die NC/PLC-Nahtstelle ausgegeben. Über das Hilfsfunktionsspezifische Änderungssignal wird dem PLC-Anwenderprogramm angezeigt, dass die Hilfsfunktion gültig ist.</p> <p>Die Quittierung der Hilfsfunktion erfolgt sofort durch das PLC-Grundprogramm im nächsten OB40-Takt. Die Quittierung der Hilfsfunktion ist daher keine Bestätigung für die Ausführung der entsprechenden PLC-Anwenderfunktion. Die Bearbeitung der Hilfsfunktion erfolgt weiterhin im OB1-Zyklus. Die nächste Ausgabe von Hilfsfunktionen an die PLC kann daher erst nach dem vollständigen Durchlauf dieses OB1-Zyklus erfolgen. Dies macht sich bei Bahnsteuerbetrieb vor allem dann bemerkbar (Absenken der Bahngeschwindigkeit), wenn in mehreren aufeinander folgenden Teileprogrammsätzen Hilfsfunktionen schneller Quittierung ausgegeben werden.</p> <p>Bei Hilfsfunktionen mit schneller Quittung ist nicht gewährleistet, dass die Reaktion im PLC-Anwenderprogramm satzsynchron erfolgt.</p> <p>Hinweis Die Parametrierung des Ausgabeverhaltens von Hilfsfunktionen als "Schnelle Hilfsfunktionen" ist nur in Verbindung mit anwenderdefinierten Hilfsfunktionen möglich.</p>
2	<p>Keine vordefinierte Hilfsfunktion</p> <p>Mit dieser Einstellung wird eine vordefinierte Hilfsfunktion wie eine anwenderdefinierte Hilfsfunktion behandelt. Die Hilfsfunktion löst dann nicht mehr die entsprechende Systemfunktion aus, sondern wird nur noch an die PLC ausgegeben.</p> <p>Beispiel: Umprojektierung der Hilfsfunktion "Spindel positionieren" (Index 9) zu einer anwenderdefinierten Hilfsfunktion mit normaler Quittierung und Ausgabe vor der Verfahrbewegung. MD22080 \$MC_AUXFU_PREDEF_SPEC [9] = 'H25' (100101B)</p>
3	<p>Keine Ausgabe an die PLC</p> <p>Die Hilfsfunktion wird nicht an die PLC ausgegeben.</p>
4	<p>Spindelreaktion nach der Quittung durch die PLC</p> <p>Die zugehörige Systemfunktion wird erst nach der Quittierung durch die PLC ausgeführt.</p>
5	<p>Ausgabe vor der Bewegung</p> <p>Die Ausgabe der Hilfsfunktion an die PLC erfolgt vor den im Teileprogrammsatz programmierten Verfahrbewegungen (Bahn- und/oder satzbezogene Positionierachsbewegungen).</p>
6	<p>Ausgabe während der Bewegung</p> <p>Die Ausgabe der Hilfsfunktion an die PLC erfolgt während der im Teileprogrammsatz programmierten Verfahrbewegungen (Bahn- und/oder satzbezogene Positionierachsbewegungen).</p>

Bit	Bedeutung
7	Ausgabe am Satzende Die Ausgabe der Hilfsfunktion an die PLC erfolgt nach Abschluss der im Teileprogrammsatz programmierten Verfahrbewegungen (Bahn- und/oder satzbezogene Positionierachsbewegungen).
8	Keine Ausgabe nach Satzsuchlauf Type 1, 2, 4 Satzsuchlauf Typ 1, 2, 4: Die während des Satzsuchlaufs aufgesammelte Hilfsfunktion wird nicht ausgegeben.
9	Aufsammlung während Satzsuchlauf mit Programmtest (Type 5, SERUPRO) Die Hilfsfunktion wird bei Satzsuchlauf mit Programmtest gruppenspezifisch in folgenden Systemvariablen aufgesammelt: <ul style="list-style-type: none"> • \$AC_AUXFU_M_VALUE[<n>] • \$AC_AUXFU_M_EXT[<n>] • \$AC_AUXFU_M_STATE[<n>]
10	Keine Ausgabe während Satzsuchlauf mit Programmtest (Type 5, SERUPRO) Die Hilfsfunktion wird bei Satzsuchlauf mit Programmtest nicht an die PLC ausgegeben.
11	Kanalübergreifende Hilfsfunktion (SERUPRO) Die Hilfsfunktion wird bei Satzsuchlauf mit Programmtest (SERUPRO) kanalübergreifend in der globalen Liste der Hilfsfunktionen aufgesammelt. Hinweis Pro Hilfsfunktionsgruppe wird immer nur die letzte Hilfsfunktion der Gruppe aufgesammelte.
12	Ausgabe erfolgte über Synchronaktion (nur lesbar) Das Bit wird gesetzt wenn die Hilfsfunktion über eine Synchronaktion an die PLC ausgegeben wurde.
13	Implizite Hilfsfunktion(nur lesbar) Das Bit wird gesetzt wenn die Hilfsfunktion implizit an die PLC ausgegeben wurde.
14	Aktives M01(nur lesbar) Das Bit wird gesetzt wenn die Hilfsfunktion bei aktivem M01 an die PLC ausgegeben wurde.
15	Keine Ausgabe während Einfahr-Testlauf Die Hilfsfunktion wird während des Einfahr-Testlaufs nicht an die PLC ausgegeben.
16	Nibbeln aus
17	Nibbeln ein
18	Nibbeln

Hinweis

Bei Hilfsfunktionen, für die kein Ausgabeverhalten festgelegt ist, wird folgendes Standard-Ausgabeverhalten wirksam:

- Bit 0 = 1: Ausgabedauer einen OB1-Takt
- Bit 7 = 1: Ausgabe am Satzende

9.2.3 Parametrierung

9.2.3.1 Gruppenzuordnung

Über die Gruppenzuordnung einer Hilfsfunktion wird die Behandlung der Hilfsfunktion bei Satzsuchlauf festgelegt. Die 168 zur Verfügung stehenden Hilfsfunktionsgruppen sind in vordefinierte und anwenderdefinierbare Gruppen aufgeteilt:

vordefinierte Gruppen:	1 ... 4	10 ... 12	72 ... 168
anwenderdefinierbare Gruppen:	5 ... 9	13 ... 71	

Jede vordefinierte Hilfsfunktion ist standardmäßig einer Hilfsfunktionsgruppe zugeordnet. Diese Zuordnung kann für die meisten vordefinierte Hilfsfunktion über folgendes Maschinendatum geändert werden:

MD22040 \$MC_AUXFU_PREDEF_GROUP[<n>] (Gruppenzuordnung von vordefinierten Hilfsfunktionen)

Soll eine Hilfsfunktion keiner Gruppe zugeordnet sein, ist im Maschinendatum der Wert "0" einzutragen.

Für die vordefinierten Hilfsfunktionen mit folgenden Indizes <n> kann die Gruppenzuordnung nicht geändert werden: 0, 1, 2, 3, 4, 22, 23, 24

Hinweis

1. Hilfsfunktionsgruppe und Satzsuchlauf

Hilfsfunktionen der 1. Hilfsfunktionsgruppe werden bei Satzsuchlauf nur aufgesammelt, aber nicht ausgegeben.

9.2.3.2 Typ, Adresserweiterung und Wert

Über die Parameter Typ, Adresserweiterung und Wert erfolgt die Programmierung einer Hilfsfunktion (siehe Kapitel "Programmierung einer Hilfsfunktion (Seite 434)").

Typ

Über den "Typ" wird der Bezeichner einer Hilfsfunktion festgelegt, z. B.:

"M"	für Zusatzfunktion
"S"	für Spindelfunktion
"F"	für Vorschub

Die Einstellung erfolgt über das Maschinendatum:

MD22050 \$MC_AUXFU_PREDEF_TYPE[<n>] (Typ von vordefinierten Hilfsfunktionen)

Hinweis

Bei vordefinierten Hilfsfunktionen kann der "Typ" nicht verändert werden.

Adresserweiterung

Die "Adresserweiterung" einer Hilfsfunktion dient zur Adressierung unterschiedlicher Komponenten des gleichen Typs. Bei vordefinierten Hilfsfunktionen entspricht der Wert der "Adresserweiterung" der Spindelnummer, auf die sich die Hilfsfunktion bezieht.

Die Einstellung erfolgt über das Maschinendatum:

MD22060 \$MC_AUXFU_PREDEF_EXTENSION[<n>] (Adresserweiterung für vordefinierte Hilfsfunktionen)

Zusammenfassen von Hilfsfunktionen

Soll eine Hilfsfunktion für alle Spindeln eines Kanals der gleichen Hilfsfunktionsgruppe zugeordnet werden, ist für den Parameter "Adresserweiterung" ein Wert von "-1" einzugeben.

Beispiel:

Für alle Spindeln des Kanals wird die Hilfsfunktion M3 (Maschinendatenindex = 6) der 2. Hilfsfunktionsgruppe zugeordnet.

```
MD22040 $MC_AUXFU_PREDEF_GROUP[ 6 ]      = 2
MD22050 $MC_AUXFU_PREDEF_TYPE[ 6 ]        = "M"
MD22060 $MC_AUXFU_PREDEF_EXTENSION[ 6 ]    = -1
MD22070 $MC_AUXFU_PREDEF_VALUE[ 6 ]        = 3
```

Wert

Die Parameter "Wert" und "Typ" definieren die Bedeutung einer Hilfsfunktion. D. h. die Systemfunktion, die aufgrund dieser Hilfsfunktion aktiviert wird.

Der "Wert" einer Hilfsfunktion ist festgelegt im Maschinendatum:

MD22070 \$MC_AUXFU_PREDEF_VALUE[<n>] (Wert von vordefinierten Hilfsfunktionen)

Hinweis

Bei vordefinierter Hilfsfunktion kann der "Wert" nicht verändert werden. Für einige vordefinierte Hilfsfunktionen kann der "Wert" über zusätzliche Maschinendaten umprojektiert werden (siehe Kapitel "Assoziierte Hilfsfunktionen (Seite 430)").

9.2.3.3 Ausgabeverhalten

Über den Parameter "Ausgabeverhalten" wird festgelegt, wann eine vordefinierte Hilfsfunktion an die NC/PLC-Nahtstelle ausgegeben und wann sie von der PLC quittiert wird.

Die Einstellung erfolgt über das Maschinendatum:

MD22080 \$MC_AUXFU_PREDEF_SPEC[<n>] (Ausgabeverhalten von vordefinierten Hilfsfunktionen)

Ausgabeverhalten bezüglich der Bewegung

Ausgabe vor der Bewegung

- Die Verfahrbewegungen (Bahn- und/oder satzbezogene Positionierachsbewegungen) des vorausgehenden Teileprogrammsatzes werden mit Genauhalt beendet.
- Die Ausgabe der Hilfsfunktionen erfolgt mit Beginn des aktuellen Teileprogrammsatzes.
- Die Verfahrbewegungen des aktuellen Teileprogrammsatzes (Bahn- und/oder Positionierachsbewegungen) werden erst nach Quittierung der Hilfsfunktionen durch die PLC gestartet:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung): nach einem OB1-Zyklus
 - Ausgabedauer einen OB40-Zyklus (Schnelle Quittierung): nach einem OB40-Zyklus

Ausgabe während der Bewegung

- Die Ausgabe der Hilfsfunktionen erfolgt mit dem Start der Verfahrbewegungen (Bahn- und/oder Positionierachsbewegungen).
- Die Bahngeschwindigkeit des aktuellen Teileprogrammsatzes wird so reduziert, dass die Zeit bis zum Satzende größer als die Zeit zum Quittieren der Hilfsfunktionen durch die PLC ist:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung): ein OB1-Zyklus
 - Ausgabedauer einen OB40-Zyklus (Schnelle Quittierung): ein OB40-Zyklus

Ausgabe nach der Bewegung

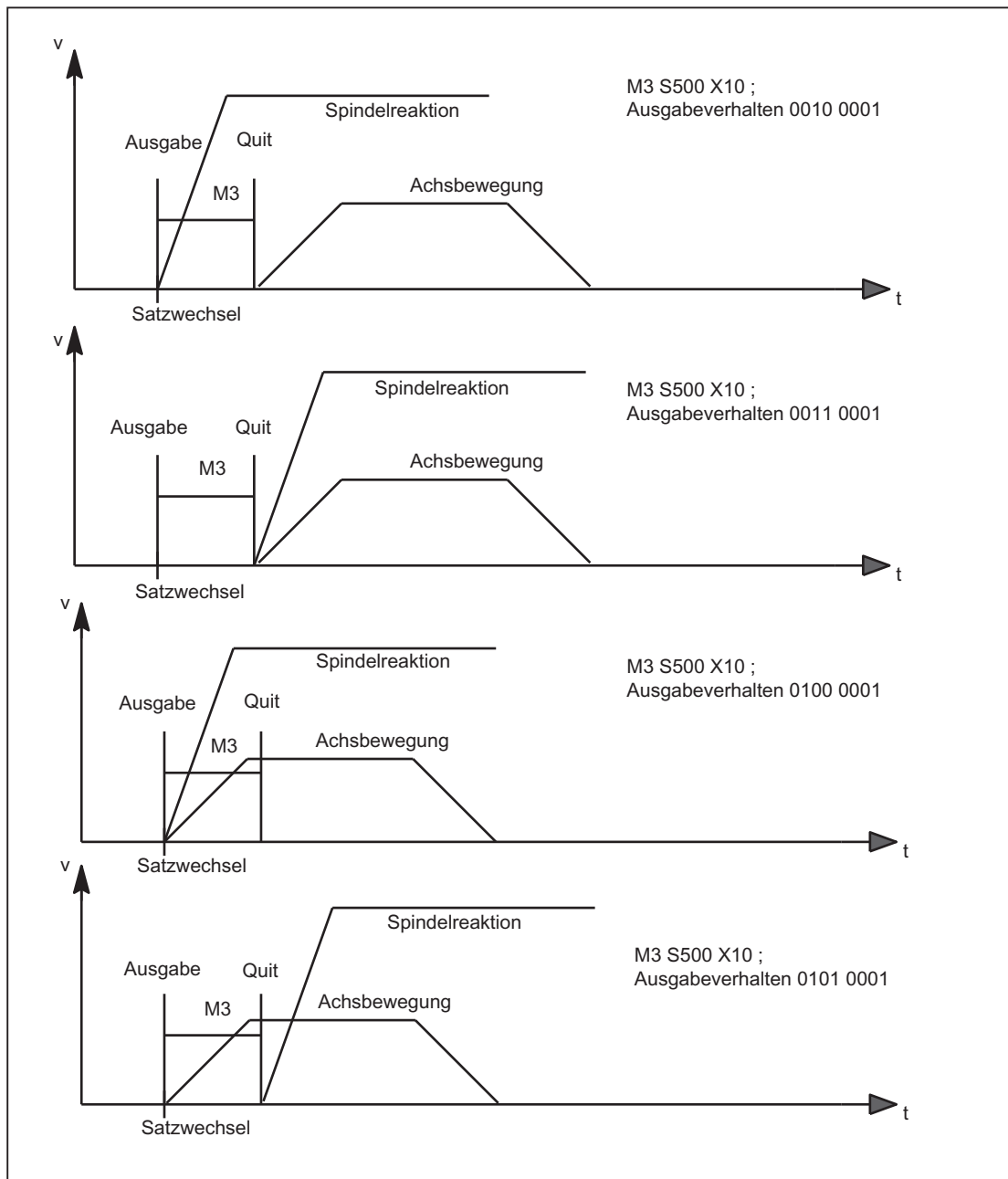
- Die Verfahrbewegungen (Bahn- und/oder satzbezogene Positionierachsbewegungen) des aktuellen Teileprogrammsatzes werden mit Genauhalt beendet.
- Die Ausgabe der Hilfsfunktionen erfolgt nach dem Beenden der Verfahrbewegungen.
- Der Satzwechsel erfolgt nach Quittierung der Hilfsfunktionen durch die PLC:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung): nach einem OB1-Zyklus
 - Ausgabedauer einen OB40-Zyklus (Schnelle Quittierung): nach einem OB40-Zyklus

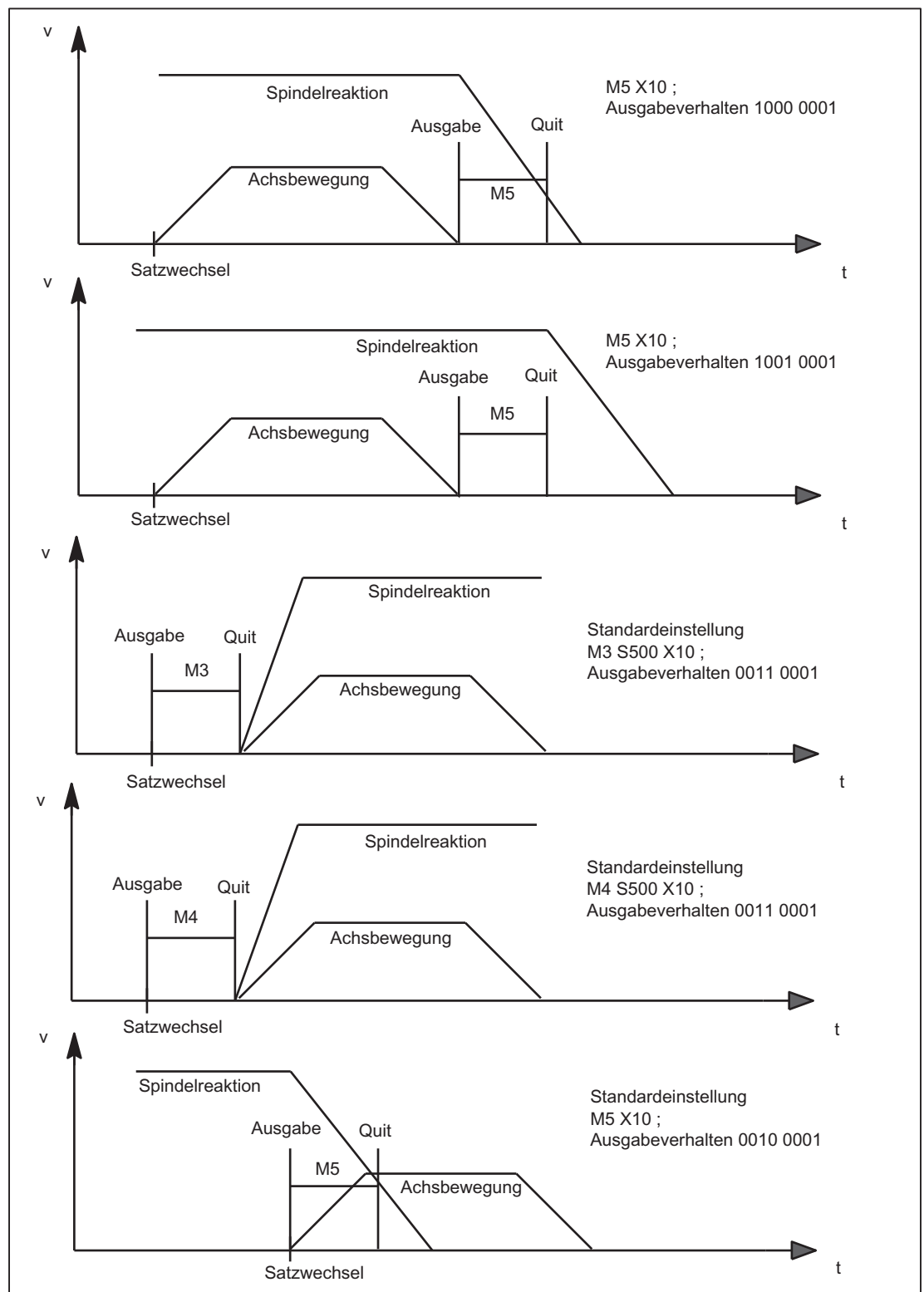
Beispiele für unterschiedliches Ausgabeverhalten

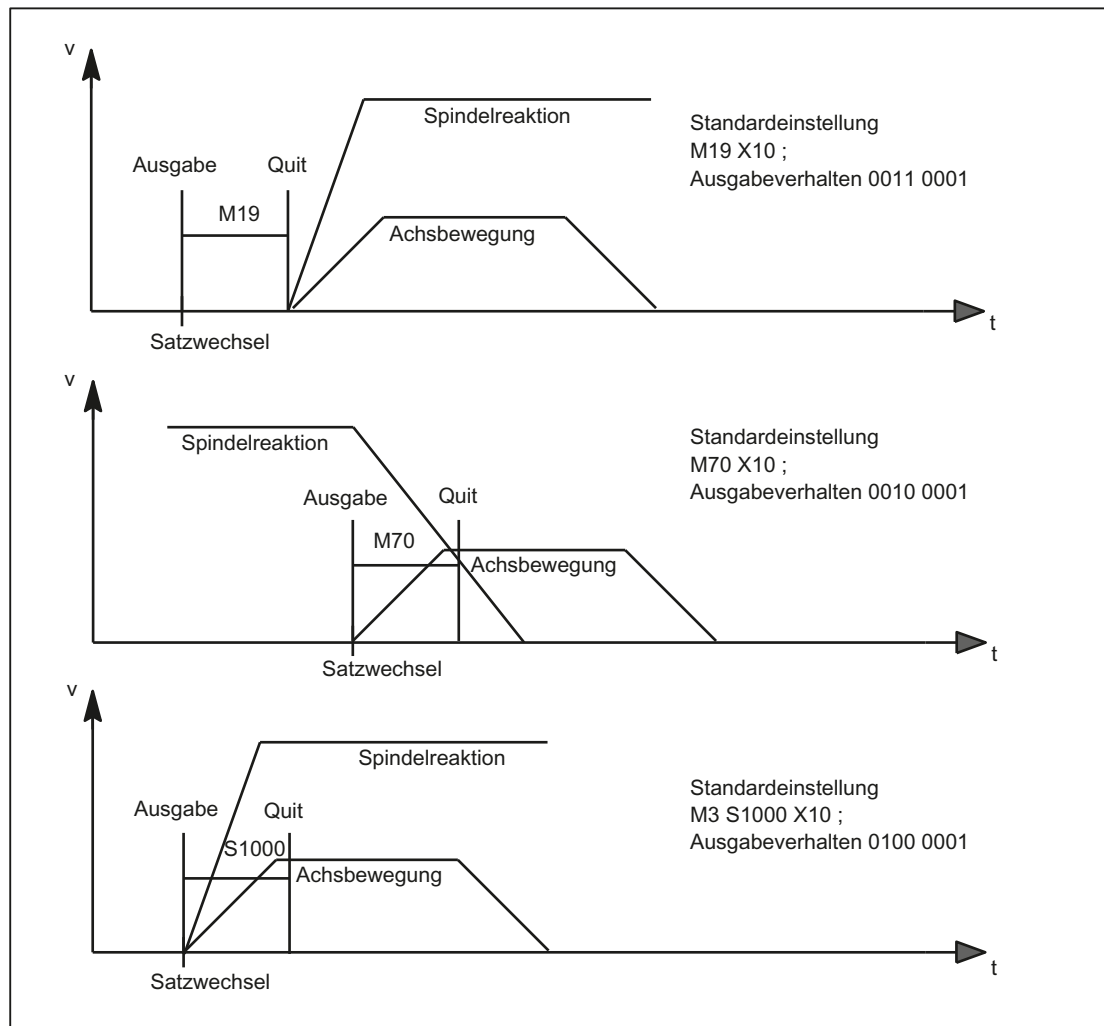
Die folgenden Abbildungen veranschaulichen das unterschiedliche Verhalten bezüglich:

- Ausgabe und Quittierung der Hilfsfunktion
- Spindelreaktion (Drehzahländerung)
- Verfahrbewegung (Geschwindigkeitsänderung)

Die in den Abbildungen unter "Ausgabeverhalten" angegebenen binären Werte beziehen sich auf das parametrisierte Ausgabeverhalten (MD22080).







9.3 Anwenderdefinierte Hilfsfunktionen

Die Verwendung von anwenderdefinierten Hilfsfunktionen lässt sich in zwei Bereiche unterteilen:

- Erweiterung von vordefinierten Hilfsfunktionen
- Anwenderspezifische Hilfsfunktionen

Erweiterung von vordefinierten Hilfsfunktionen

Da die Maschinendaten der vordefinierten Hilfsfunktionen nur einmal vorhanden sind, kann darüber immer nur eine Spindel des Kanals adressiert werden. Zur Adressierung weiterer Spindeln müssen anwenderdefinierte Hilfsfunktionen zur Erweiterung der vordefinierten Hilfsfunktionen parametrisiert werden.

Die Erweiterung von vordefinierten Hilfsfunktionen bezieht sich ausschließlich auf den Parameter "Adresserweiterung". In den Parameter "Adresserweiterung" wird die Nummer der Spindel eingetragen, auf die sich die Hilfsfunktion bezieht.

Für folgende Systemfunktionen können die entsprechenden vordefinierten Hilfsfunktionen erweitert werden:

Systemfunktion	Typ		
		Adresserweiterung ¹⁾	
			Wert
Werkzeugwechsel	M	1	6
Spindel rechts	M	1	3
Spindel links	M	1	4
Spindel halt	M	1	5
Spindel positionieren	M	1	19
Achsbetrieb	M	1	70
automatische Getriebestufe	M	1	40
Getriebestufe 1	M	1	41
Getriebestufe 2	M	1	42
Getriebestufe 3	M	1	43
Getriebestufe 4	M	1	44
Getriebestufe 5	M	1	45
Spindel-Drehzahl	S	1	-1
Werkzeuganwahl	T	1	-1

¹⁾ Adresserweiterung = 1 ist der in den Maschinendaten der vordefinierten Hilfsfunktionen verwendete Standardwert

Beispiel:

Erweiterung der vordefinierten Hilfsfunktion für die Systemfunktion "Spindel rechts" für die 2. und 3. Spindel des Kanals.

Hilfsfunktion "Spindel rechts" für die 2. Spindel des Kanals:

MD22010 \$MC_AUXFU_ASSIGN_TYPE [n] = "M"
MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [n] = 2
MD22030 \$MC_AUXFU_ASSIGN_VALUE [n] = 3

Hilfsfunktion "Spindel rechts" für die 3. Spindel des Kanals:

MD22010 \$MC_AUXFU_ASSIGN_TYPE [m] = "M"
MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [m] = 3
MD22030 \$MC_AUXFU_ASSIGN_VALUE [m] = 3

Anwenderspezifische Hilfsfunktionen

Anwenderspezifische Hilfsfunktionen haben folgende Eigenschaften:

- Über anwenderspezifische Hilfsfunktionen werden ausschließlich Anwenderfunktionen aktiviert.
- Über anwenderspezifische Hilfsfunktionen können keine Systemfunktionen aktiviert werden.
- Eine anwenderspezifische Hilfsfunktion wird entsprechend dem parametrisierten Ausgabeverhalten an die PLC ausgegeben.
- Die Funktionalität einer anwenderspezifischen Hilfsfunktion wird durch den Maschinenhersteller/Anwender im PLC-Anwenderprogramm realisiert.

9.3.1 Parametrierung

9.3.1.1 Maximale Anzahl anwenderdefinierter Hilfsfunktionen

Die maximale Anzahl von anwenderdefinierten Hilfsfunktionen pro Kanal ist parametrierbar über das Maschinendatum:

MD11100 \$MN_AUXFU_MAXNUM_GROUP_ASSIGN (Maximale Anzahl von anwenderdefinierten Hilfsfunktionen)

9.3.1.2 Gruppenzuordnung

Über die Gruppenzuordnung einer Hilfsfunktion wird die Behandlung der Hilfsfunktion bei Satzsuchlauf festgelegt. Die 168 zur Verfügung stehenden Hilfsfunktionsgruppen sind in vordefinierte und anwenderdefinierbare Gruppen aufgeteilt:

vordefinierte Gruppen:	1 ... 4	10 ... 12	72 ... 168
anwenderdefinierbare Gruppen:	5 ... 9	13 ... 71	

Jede anwenderdefinierte Hilfsfunktion ist standardmäßig der 1. Hilfsfunktionsgruppe zugeordnet. Die Zuordnung kann über das folgende Maschinendatum geändert werden:

MD22000 \$MC_AUXFU_ASSIGN_GROUP[<n>] (Gruppenzuordnung von anwenderdefinierten Hilfsfunktionen)

Soll eine Hilfsfunktion keiner Gruppe zugeordnet sein, ist im Maschinendatum der Wert "0" einzutragen.

Hinweis

1. Hilfsfunktionsgruppe und Satzsuchlauf

Hilfsfunktionen der 1. Hilfsfunktionsgruppe werden bei Satzsuchlauf nur aufgesammelt, aber nicht ausgegeben.

9.3.1.3 Typ, Adresserweiterung und Wert

Über die Parameter Typ, Adresserweiterung und Wert erfolgt die Programmierung einer Hilfsfunktion (siehe Kapitel "Programmierung einer Hilfsfunktion (Seite 434)").

Typ

Über den "Typ" wird der Bezeichner einer Hilfsfunktion festgelegt.

Bezeichner für anwenderdefinierte Hilfsfunktionen sind:

Typ	Bezeichner	Bedeutung
"H"	Hilfsfunktion	Anwenderspezifische Hilfsfunktionen
"M"	Zusatzfunktion	Erweiterung von vordefinierten Hilfsfunktionen
"S"	Spindelfunktion	
"T"	Werkzeugnummer	

Die Einstellung erfolgt über das Maschinendatum:

MD22010 \$MC_AUXFU_ASSIGN_TYPE[<n>] (Typ von anwenderdefinierten Hilfsfunktionen)

Adresserweiterung

MD22020 \$MC_AUXFU_ASSIGN_EXTENSION[<n>] (Adresserweiterung für anwenderdefinierte Hilfsfunktionen)

Bei anwenderspezifischen Hilfsfunktionen ist die Funktionalität der Adresserweiterung nicht festgelegt. Sie dient allgemein der Unterscheidung von Hilfsfunktionen mit dem gleichen "Wert".

Zusammenfassen von Hilfsfunktionen

Sollen alle Hilfsfunktionen vom gleichen Typ und Wert der gleichen Hilfsfunktionsgruppe zugeordnet werden, ist für den Parameter "Adresserweiterung" ein Wert von "-1" einzugeben.

Beispiel:

Alle anwenderspezifischen Hilfsfunktionen mit Wert = 8 werden der 10. Hilfsfunktionsgruppe zugeordnet.

MD22000 \$MC_AUXFU_ASSIGN_GROUP [1]	= 10
MD22010 \$MC_AUXFU_ASSIGN_TYPE [1]	= "H"
MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [1]	= -1
MD22030 \$MC_AUXFU_ASSIGN_VALUE [1]	= 8

Wert

MD22030 \$MC_AUXFU_ASSIGN_VALUE[<n>] (Wert von anwenderdefinierten Hilfsfunktionen)

Bei anwenderspezifischen Hilfsfunktionen ist die Funktionalität des Parameters "Wert" nicht festgelegt. Allgemein wird über den Wert die entsprechende PLC-Anwenderfunktion aktiviert.

Zusammenfassen von Hilfsfunktionen

Sollen alle Hilfsfunktionen vom gleichen Typ und Adresserweiterung der gleichen Hilfsfunktionsgruppe zugeordnet werden, ist für den Parameter "Wert" ein Wert von "-1" einzugeben.

Beispiel:

Alle anwenderspezifischen Hilfsfunktionen mit Adresserweiterung = 2 werden der 11. Hilfsfunktionsgruppe zugeordnet.

MD22000 \$MC_AUXFU_ASSIGN_GROUP [2]	= 11
MD22010 \$MC_AUXFU_ASSIGN_TYPE [2]	= "H"
MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [2]	= 2
MD22030 \$MC_AUXFU_ASSIGN_VALUE [2]	= -1

9.3.1.4 Ausgabeverhalten

Das "Ausgabeverhalten" von anwenderdefinierten Hilfsfunktionen ist parametrierbar über das Maschinendatum:

MD22035 \$MC_AUXFU_ASSIGN_SPEC[<n>] (Ausgabeverhalten von anwenderdefinierten Hilfsfunktionen)

Zur Beschreibung der einzelnen Ausgabeparameter siehe Kapitel "Ausgabeverhalten (Seite 422)" der vordefinierten Hilfsfunktionen. Die dort gemachten Aussagen gelten sinngemäß auch für das Ausgabeverhalten von anwenderdefinierten Hilfsfunktionen.

9.4 Assoziierte Hilfsfunktionen

Funktion

Assoziierte Hilfsfunktionen sind anwenderdefinierte Hilfsfunktionen, die die gleiche Wirkung wie die entsprechenden vordefinierten Hilfsfunktionen haben. Für folgende vordefinierten Hilfsfunktionen können anwenderdefinierte Hilfsfunktionen assoziiert werden:

- M0 (Halt)
- M1 (bedingter Halt)

Voraussetzungen

Voraussetzung für die Assoziierung einer anwenderdefinierten Hilfsfunktion zu einer der oben genannten vordefinierten Hilfsfunktionen ist die Parametrierung einer anwenderdefinierten Hilfsfunktion. Als Parameter "Typ" der anwenderdefinierten Hilfsfunktion ist dabei nur "M" erlaubt.

Parametrierung

Die Assoziierung einer anwenderdefinierten Hilfsfunktion zu einer der oben genannten vordefinierten Hilfsfunktionen erfolgt in den Maschinendaten:

MD22254 \$MC_AUXFU_ASSOC_M0_VALUE (Zusätzliche M-Funktion für Programm-Halt)

MD22256 \$MC_AUXFU_ASSOC_M1_VALUE (Zusätzliche M-Funktion für bedingten Halt)

Gruppenzuordnung

Als Gruppenzuordnung einer assoziierten anwenderdefinierten Hilfsfunktion wird fest die Gruppenzuordnung der entsprechenden vordefinierten Hilfsfunktion übernommen.

Anwendung

Assoziierte Hilfsfunktionen dürfen angewendet werden in:

- Hauptprogramm
- Unterprogramm
- Zyklus

Hinweis

Assoziierte Hilfsfunktionen dürfen nicht in Synchronaktionen angewendet werden.

NC/PLC-Nahtstellensignale

Bei einer assoziierten anwenderdefinierten Hilfsfunktion werden an die NC/PLC-Nahtstelle dieselben Signale ausgegeben wie bei der entsprechenden vordefinierten Hilfsfunktion. Zur Unterscheidung, welche Hilfsfunktion tatsächlich programmiert wurde, wird aber als Wert der Hilfsfunktion der Wert der anwenderdefinierten Hilfsfunktionen (Parameter "Wert") ausgegeben. Im PLC-Anwenderprogramm besteht damit die Möglichkeit, zwischen vordefinierter und anwenderdefinierter Hilfsfunktion zu unterscheiden.

Hinweis

Eine Änderung der Maschinendaten MD22254 und/oder MD22256 erfordert u. U. eine entsprechende Anpassung des PLC-Anwenderprogramms.

Spezifische NC/PLC-Nahtstellensignale

Folgende spezifische NC/PLC-Nahtstellensignale stehen zur Verfügung:

- DB21, ... DBX24.4 (NC assoziiertes M01 aktivieren)
- DB21, ... DBX30.5 (Assoziiertes M01 aktivieren)
- DB21, ... DBX318.5 (Assoziiertes M01 aktiv)

Randbedingungen

Folgenden Randbedingungen sind zu beachten:

- Eine anwenderdefinierte Hilfsfunktion darf nicht mehrfach assoziiert werden.
- Vordefinierte Hilfsfunktionen (z. B. M3, M4, M5 etc.) dürfen nicht assoziiert werden.

Beispiel

Assoziieren der anwenderdefinierten Hilfsfunktion M123 mit M0:

MD22254 \$MC_AUXFU_ASSOC_M0_VALUE = 123

Die anwenderdefinierte Hilfsfunktion M123 hat damit die gleiche Funktionalität wie M0.

9.5 Typ-spezifisches Ausgabeverhalten

Funktion

Das Ausgabeverhalten der Hilfsfunktionen bezüglich einer im Teileprogrammsatz programmierten Verfahrbewegung kann Typ-spezifisch festgelegt werden.

Parametrierung

Die Parametrierung des Typ-spezifischen Ausgabeverhaltens erfolgt über die Maschinendaten:

MD22200 \$MC_AUXFU_M_SYNC_TYPE (Ausgabezeitpunkt M-Funktionen)

MD22210 \$MC_AUXFU_S_SYNC_TYPE (Ausgabezeitpunkt S-Funktionen)

MD22220 \$MC_AUXFU_T_SYNC_TYPE (Ausgabezeitpunkt T-Funktionen)

MD22230 \$MC_AUXFU_H_SYNC_TYPE (Ausgabezeitpunkt H-Funktionen)

MD22240 \$MC_AUXFU_F_SYNC_TYPE (Ausgabezeitpunkt F-Funktionen)

MD22250 \$MC_AUXFU_D_SYNC_TYPE (Ausgabezeitpunkt D-Funktionen)

MD22252 \$MC_AUXFU_DL_SYNC_TYPE (Ausgabezeitpunkt DL-Funktionen)

Folgende Ausgabeverhalten können parametriert werden:

MD \$MC_AUXFU_XX_SYNC_TYPE = <Wert>

Wert	Ausgabeverhalten
0	Ausgabe vor der Bewegung
1	Ausgabe während der Bewegung
2	Ausgabe am Satzende
3	Keine Ausgabe an PLC
4	Ausgabe entsprechend des mit MD22080 festgelegten Ausgabeverhaltens

Zur Beschreibung der verschiedenen Ausgabeverhalten siehe Kapitel "Ausgabeverhalten (Seite 422)".

Hinweis

Die für den jeweiligen Hilfsfunktionstyp einstellbaren Ausgabeverhalten sind dem Listenhandbuch "Ausführliche Maschinendaten-Beschreibung" zu entnehmen.

Beispiel

Ausgabe von Hilfsfunktionen mit unterschiedlichem Ausgabeverhalten in einem Teileprogrammsatz mit Verfahrbewegung.

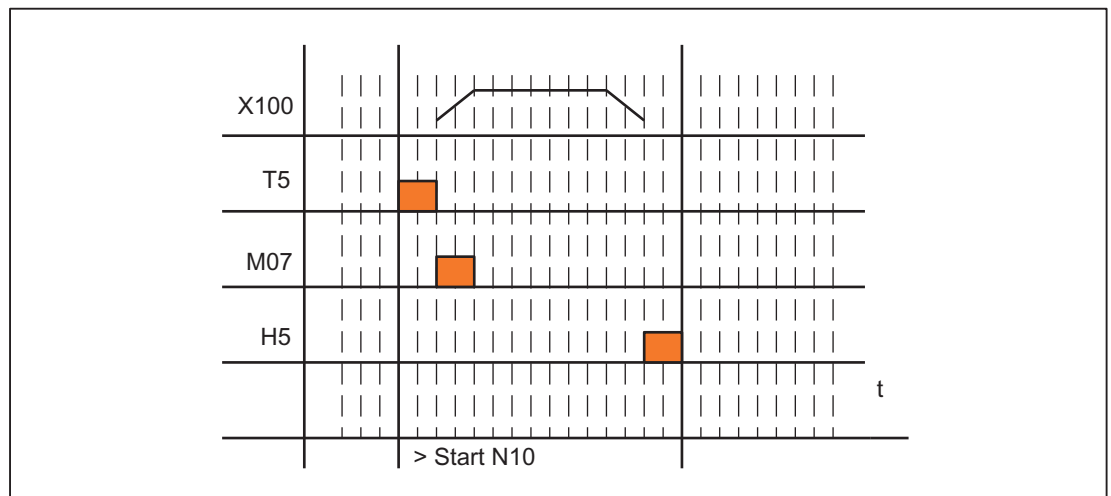
Parametriertes Ausgabeverhalten:

MD22200 \$MC_AUXFU_M_SYNC_TYPE = 1	⇒ M-Funktion: Ausgabe während der Bewegung
MD22220 \$MC_AUXFU_T_SYNC_TYPE = 0	⇒ T-Funktion: Ausgabe vor der Bewegung
MD22230 \$MC_AUXFU_H_SYNC_TYPE = 2	⇒ H-Funktion: Ausgabe am Satzende

Teileprogrammsatz:

Programmcode
...
N10 G01 X100 M07 H5 T5
...

Zeitlicher Ablauf der Hilfsfunktionsausgabe:



9.6 Prioritäten des parametrierten Ausgabeverhaltens

Die Prioritäten bezüglich des parametrierten Ausgabeverhaltens einer Hilfsfunktion müssen für folgende Kriterien getrennt beachtet werden:

- Ausgabedauer (normale / schnelle Quittierung)
- Ausgabe bezüglich der Bewegung (vor / während / nach der Bewegung)

Allgemein gilt, dass das parametrierte Ausgabeverhalten mit der niedrigeren Priorität immer dann wirksam wird, wenn kein höher priorisiertes Ausgabeverhalten parametriert wurde.

Ausgabedauer

Für die Ausgabedauer gelten folgende Prioritäten:

Priorität	Ausgabeverhalten	Festlegung über:
Höchste	Hilfsfunktionsspezifisch	Teileprogrammanweisung: QU(...) (siehe Kapitel "Programmierbare Ausgabedauer (Seite 436)")
↓	Hilfsfunktionsspezifisch	MD22035 \$MC_AUXFU_ASSIGN_SYNC[<n>] MD22080 \$MC_AUXFU_PREDEF_SYNC[<n>]
↓	Gruppenspezifisch	MD11110 \$MC_AUXFU_GROUP_SPEC[<n>]
Niedrigste	Keine Festlegung	Standard-Ausgabeverhalten: Ausgabedauer einen OB1-Takt

Ausgabe bezüglich der Bewegung

Für die Ausgabe bezüglich der Bewegung gelten folgende Prioritäten:

Priorität	Ausgabeverhalten	Festlegung über:
Höchste	Hilfsfunktionsspezifisch	MD22035 \$MC_AUXFU_ASSIGN_SYNC[<n>] MD22080 \$MC_AUXFU_PREDEF_SYNC[<n>]
↓	Gruppenspezifisch	MD11110 \$MC_AUXFU_GROUP_SPEC[<n>]
↓	Typspezifisch	MD22200 \$MC_AUXFU_M_SYNC_TYPE MD22210 \$MC_AUXFU_S_SYNC_TYPE MD22220 \$MC_AUXFU_T_SYNC_TYPE MD22230 \$MC_AUXFU_H_SYNC_TYPE MD22240 \$MC_AUXFU_F_SYNC_TYPE MD22250 \$MC_AUXFU_D_SYNC_TYPE MD22252 \$MC_AUXFU_DL_SYNC_TYPE
Niedrigste	Keine Festlegung	Standard-Ausgabeverhalten: Ausgabe am Satzende

Hinweis

Teileprogrammsätze ohne Bahnbewegung

In einem Teileprogrammsatz ohne Bahnbewegung (also auch bei Positionierachsen und Spindeln) werden die Hilfsfunktionen sofort in einem Block ausgegeben.

9.7 Programmierung einer Hilfsfunktion

Syntax

Die Programmierung einer Hilfsfunktion erfolgt in einem Teileprogrammsatz mit folgender Syntax:

```
<Typ> [<Adresserweiterung>=] <Wert>
```

Hinweis

Wird keine Adresserweiterung programmiert, wird implizit die Adresserweiterung = 0 gesetzt.

Vordefinierte Hilfsfunktionen mit der Adresserweiterung = 0 beziehen sich immer auf die Masterspindel des Kanals.

Symbolische Adressierung

Die Werte für die Parameter "Adresserweiterung" und "Wert" können auch symbolisch angegeben werden. Der symbolische Name für die Adresserweiterung muss dann in eckigen Klammern angegeben werden.

Beispiel:

Symbolische Programmierung der Hilfsfunktion M3 (Spindel rechts) für die 1. Spindel:

Programmcode	Kommentar
DEF SPINDEL_NR=1	; 1.Spindel im Kanal
DEF DREHRICHTUNG=3	; Drehrichtung rechts
N100 M[SPINDEL_NR]=DREHRICHTUNG	; entsprechend: M1=3

Hinweis

Bei der Verwendung von symbolischen Namen zur Programmierung einer Hilfsfunktion wird bei der Ausgabe der Hilfsfunktion an die PLC nicht der symbolische Name übergeben, sondern der entsprechende Zahlenwert.

Beispiele

Beispiel 1: Programmierung vordefinierter Hilfsfunktionen

Programmcode	Kommentar
N10 M3	; "Spindel rechts" für die Master-Spindel des Kanals.
N20 M0=3	; "Spindel rechts" für die Master-Spindel des Kanals.
N30 M1=3	; "Spindel rechts" für die 1.Spindel des Kanals.
N40 M2=3	; "Spindel rechts" für die 2.Spindel des Kanals.

Beispiel 2: Programmierbeispiele von Hilfsfunktionen mit den entsprechenden Werten zur Ausgabe an die PLC

Programmcode	Kommentar
DEF Kühlmittel=12	; Ausgabe an die PLC: - - -
DEF Schmiermittel=130	; Ausgabe an die PLC: - - -
H[Kühlmittel]=Schmiermittel	; Ausgabe an die PLC: H12=130
H=Kühlmittel	; Ausgabe an die PLC: H0=12
H5	; Ausgabe an die PLC: H0=5

Programmcode	Kommentar
H=5.379	; Ausgabe an die PLC: H0=5.379
H17=3.5	; Ausgabe an die PLC: H17=3.5
H[Kühlmittel]=13.8	; Ausgabe an die PLC: H12=13.8
H='HFF13'	; Ausgabe an die PLC: H0=65299
H='B1110'	; Ausgabe an die PLC: H0=14
H5.3=21	; Fehler

9.8 Programmierbare Ausgabedauer

Funktion

Anwenderspezifischen Hilfsfunktionen, für die das Ausgabeverhalten "Ausgabedauer einen OB1-Zyklus (Langsame Quittierung)" parametrisiert wurde, können für einzelne Ausgaben über die Teileprogrammanweisung QU (Quick) zu Hilfsfunktionen mit schneller Quittierung definiert werden.

Syntax

Die Definition einer Hilfsfunktion mit schneller Quittierung erfolgt in einem Teileprogrammsatz mit folgender Syntax:

<Typ> [<Adresserweiterung>]=QU (<Wert>)

Beispiel

Unterschiedliches Verhalten bei der Ausgabe der Hilfsfunktionen M100 und M200 in einem Teileprogramm. Das Ausgabeverhalten der Hilfsfunktionen ist folgendermaßen parametrisiert:

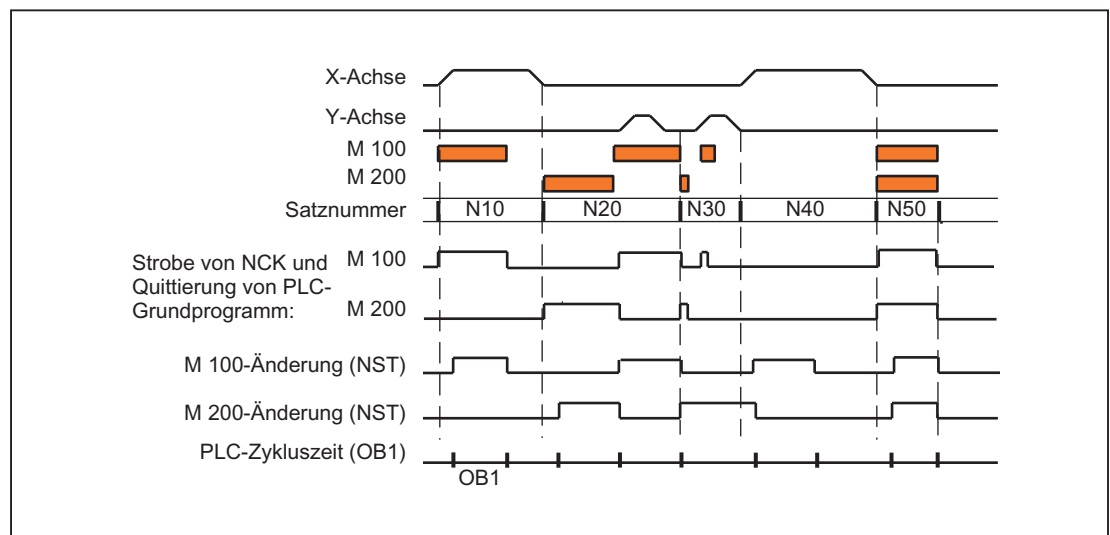
- M100
 - Ausgabedauer einen OB1-Zyklus (Langsame Quittierung)
 - Ausgabe während der Bewegung
- M200
 - Ausgabedauer einen OB1-Zyklus (Langsame Quittierung)
 - Ausgabe vor der Bewegung

Programmcode	Kommentar
N10 G94 G01 X50 M100	; Ausgabe von M100: während der Bewegung
	; Quittierung: langsam
N20 Y5 M100 M200	; Ausgabe von M200: vor der Bewegung
	; Ausgabe von M100: während der Bewegung
	; Quittierung: langsam
N30 Y0 M=QU(100) M=QU(200)	; Ausgabe von M200: vor der Bewegung
	; Ausgabe von M100: während der Bewegung
	; Quittierung: schnell

Programmcode	Kommentar
N40 X0	
N50 M100 M200	; Ausgabe von M200: sofort 1) ; Ausgabe von M100: sofort 1) ; Quittierung: langsam
M17	

¹⁾ Ohne Verfahrbewegung erfolgt die Ausgabe von Hilfsfunktionen an die PLC immer sofort.

Das folgende Bild zeigt den zeitlichen Ablauf des Teileprogramms. Besonders zu beachten ist der zeitliche Unterschied bei der Abarbeitung der Teileprogrammsätze N20 und N30.



9.9 Hilfsfunktionsausgabe an die PLC

Funktion

Bei der Ausgabe einer Hilfsfunktion an die PLC werden folgende Signale und Werte an die NC/PLC-Nahtstelle übergeben:

- Änderungssignale
- Parameter "Adresserweiterung"
- Parameter "Wert"

Datenbereiche in der NC/PLC-Nahtstelle

Die Änderungssignale und Werte der Hilfsfunktionen liegen in der NC/PLC-Nahtstelle in folgenden Datenbereichen:

- Änderungssignale bei Hilfsfunktionsübergabe von NC-Kanal:
DB21, ... DBB58 - DBB67
- Übergebene M-/ und S-Funktionen:
DB21, ... DBB68 - DBB112
- Übergebene T-/ D-/ DL-Funktionen:
DB21, ... DBB116 - DBB136
- Übergebene H-/ F-Funktionen:
DB21, ... DBB140 - DBB190
- Dekodierte M-Signale (M0 - M99):
DB21, ... DBB194 - DBB206 (dynamische M-Funktionen)

Für Informationen der Zugriffsverfahren zur NC/PLC-Nahtstelle siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)".

Die Ausführliche Beschreibung der o. g. Datenbereiche in der NC/PLC-Nahtstelle findet sich in:

Literatur:

Listenhandbuch Listen, Buch 2; PLC-Anwendernahtstellen,
Kapitel: Kanalspezifische Signale (DB 21 – DB 30)"

9.10 Hilfsfunktionen ohne Satzwechselverzögerung

Funktion

Auch bei Hilfsfunktionen mit einem parametrisierten und/oder programmierten Ausgabeverhalten:

- "Ausgabedauer einen OB40-Zyklus (Schnelle Quittierung)"
- "Ausgabe vor der Bewegung" oder "Ausgabe während der Bewegung"

kann es bei Bahnsteuerbetrieb (kurze Verfahrswege und hohen Geschwindigkeiten) zu Geschwindigkeitseinbrüchen kommen, weil zum Satzende hin auf die Quittierung der Hilfsfunktion durch die PLC gewartet werden muss. Um derartige Geschwindigkeitseinbrüche zu vermeiden, kann der Satzwechsel unabhängig bezüglich der Quittierung derartiger Hilfsfunktionen gemacht werden.

Parametrierung

Das Unterdrücken der Satzwechselverzögerung bei schnellen Hilfsfunktionen wird eingestellt über das Maschinendatum:

MD22100 \$MC_AUXFU_QUICK_BLOCKCHANGE (Satzwechselverzögerung bei schnellen Hilfsfunktionen)

Wert	Bedeutung
0	Bei der schnellen Hilfsfunktionsausgabe an die PLC wird der Satzwechsel bis zur Quittierung durch die PLC (OB40) verzögert.
1	Bei der schnellen Hilfsfunktionsausgabe an die PLC wird der Satzwechsel nicht verzögert.

Randbedingungen

Die Synchronität von Hilfsfunktionen, die ohne Satzwechselverzögerung ausgegeben werden, ist bezüglich des Teileprogrammsatzes, in dem sie programmiert sind, nicht mehr gewährleistet. Im "Worst-Case"-Szenario erfolgt die Quittierung einen OB40-Takt und die Bearbeitung der Hilfsfunktion einen OB1-Takt nach dem Satzwechsel zum nächsten Teileprogrammsatz.

9.11 M-Funktion mit implizitem Vorlaufstopp

Funktion

Soll im Zusammenhang mit einer Hilfsfunktion ein Vorlaufstopp ausgelöst werden, kann dieser explizit über den Teileprogrammbefehl `STOPRE` programmiert werden. Soll bei Programmierung der M-Funktion immer ein Vorlaufstopp ausgelöst werden, kann dieses Verhalten M-Funktionsspezifisch über folgendes Maschinendatum parametrieren:

MD10713 \$MN_M_NO_FCT_STOPRE[<n>] (M-Funktion mit Vorlaufstopp)

Beispiel

Die anwenderdefinierte M-Funktion M88 soll einen Vorlaufstopp auslösen.

Parametrierung:

MD10713 \$MN_M_NO_FCT_STOPRE [0] = 88

Anwendung:

Teileprogramm (Auszug)

Programmcode	Kommentar
...	
N100 G0 X10 M88	; Verfahrbewegung und impliziter Vorlaufstopp durch M88.
N110 Y=R1	; N110 wird erst nach Abschluss der Verfahrbewegung und Quittierung der M-Funktion interpretiert.
...	

Randbedingungen

Wird in einem Teileprogramm ein Unterprogramm durch eine der beiden folgenden Möglichkeiten indirekt über eine M-Funktion aufgerufen, erfolgt dabei kein Vorlaufstopp:

- MD10715 \$MN_M_NO_FCT_CYCLE (Durch Unterprogramm zu ersetzende M-Funktion)
- M98 (ISO-Dialekt-T / ISO-Dialekt-M)

9.12 Verhalten bei Überspeichern

Überspeichern

Über die SINUMERIK-Bedienoberfläche können vor dem Start folgender Funktionen:

- NC-START eines Teileprogramms
- NC-START zur Wiederaufnahme eines unterbrochenen Teileprogramms

die Hilfsfunktionen, die mit dem Start ausgegeben werden, durch die Funktion "Überspeichern" verändert werden.

Mögliche Anwendungsfälle sind:

- Ergänzen von Hilfsfunktionen nach Satzsuchlauf
- Herstellen des Ausgangszustandes zum Einfahren eines Teileprogramms

Überspeicherbare Hilfsfunktionstypen

Folgende Hilfsfunktionstypen können überspeichert werden:

- M (Zusatzfunktion)
- S (Spindeldrehzahl)
- T (Werkzeugnummer)
- H (Hilfsfunktion)
- D (Werkzeugkorrekturnummer)
- DL (Summenkorrektur)
- F (Vorschub)

Gültigkeitsdauer

Eine überspeicherte Hilfsfunktion, z. B. M3 (Spindel rechts), ist so lange gültig, bis sie durch eine andere Hilfsfunktion der gleichen Hilfsfunktionsgruppe, durch erneute Überspeicherung oder durch Programmierung in einem Teileprogrammsatz überschrieben wird.

9.13 Verhalten bei Satzsuchlauf

9.13.1 Hilfsfunktionsausgabe bei Satzsuchlauf Typ 1,2 und 4

Ausgabeverhalten

Bei Satzsuchlauf Typ 1, 2 und 4 werden die Hilfsfunktionen gruppenspezifisch aufgesammelt. Die jeweils letzte Hilfsfunktion einer Hilfsfunktionsgruppe wird nach NC-START in einem eigenen Teileprogrammsatz vor dem eigentlichen Wiedereinstiegssatz mit folgendem Ausgabeverhalten ausgegeben:

- Ausgabedauer einen OB1-Zyklus (normale Quittierung)
- Ausgabe vor der Bewegung

Ausgabesteuerung

Ob eine Hilfsfunktion nach Satzsuchlauf an die PLC ausgegeben wird oder nicht, lässt sich projektieren über das Bit 8 der Maschinendaten:

- MD22080 \$MC_AUXFU_PREDEF_SPEC[<n>]
(Ausgabeverhalten von vordefinierten Hilfsfunktionen)
mit <n> = Systemfunktionsindex (0 ... 32)
- MD22035 \$MC_AUXFU_ASSIGN_SPEC[<n>]
(Ausgabeverhalten von anwenderdefinierten Hilfsfunktionen)
mit <n> = Hilfsfunktionsindex (0 ... 254)
- MD11110 \$MN_AUXFU_GROUP_SPEC[<n>]
(Ausgabeverhalten der Hilfsfunktionen einer Gruppe)
mit <n> = Gruppenindex (0 ... 63)

Bit	Wert	Bedeutung
10	0	Ausgabe während Satzsuchlauf Typ 1,2,4
	1	Keine Ausgabe während Satzsuchlauf Typ 1,2,4

Dieses Verhalten hat keinen Einfluss auf die Anzeige oder auf die Variablen \$AC_AUXFU_STATE[<n>], \$AC_AUXFU_VALUE[<n>] und \$AC_AUXFU_EXT[<n>]. Die Hilfsfunktionen gelten nach Satzsuchlauf immer noch als aufgesammelt, obwohl sie nicht an die PLC ausgegeben werden.

Eine nach Satzsuchlauf nicht ausgegebene Hilfsfunktion überschreibt beim Aufsammeln auch eine Hilfsfunktion, deren Bit 8 nicht gesetzt ist.

Der Anwender kann nach Satzsuchlauf die aufgesammelten Hilfsfunktionen abfragen und unter Umständen diese selber nochmals über das Teileprogramm oder über Synchronaktionen ausgeben.

Hinweis

Folgende Hilfsfunktionen werden nicht aufgesammelt:

- Hilfsfunktionen, die keiner Hilfsfunktionsgruppe zugeordnet sind.
 - Hilfsfunktionen, die der 1. Hilfsfunktionsgruppe zugeordnet sind.
-

Überspeichern von Hilfsfunktionen

Nach erfolgtem Satzsuchlauf werden mit dem nächsten NC-START die aufgesammelten Hilfsfunktionen ausgegeben. Besteht die Anforderung, zusätzliche Hilfsfunktionen auszugeben, können diese über die Funktion "Überspeichern" hinzugefügt werden (siehe Kapitel "Verhalten bei Überspeichern (Seite 440)").

Verhalten bezüglich M19 (Spindel positionieren)

Nach Satzvorlauf wird immer die letzte mit M19 programmierte Spindelpositionierung durchgeführt, auch wenn vom Teileprogrammsatz mit M19 bis zum Zielsatz noch andere spindelspezifische Hilfsfunktionen programmiert sind. Das Setzen der erforderlichen Spindelfreigaben muss im PLC-Anwenderprogramm daher abgeleitet werden von den Nahtstellensignalen der Fahrbefehle:

DB31, ... DBX64.6 / 64.7 (Fahrbefehl minus / plus)

Die spindelspezifischen Hilfsfunktionen M3, M4, M5 eignen sich in diesem Fall nicht dazu, da sie unter Umständen erst nach der Spindelpositionierung an die PLC ausgegeben werden.

Für ausführliche Informationen zum Satzsuchlauf siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)".

9.13.2 Zuordnung einer Hilfsfunktion zu mehreren Gruppen

Funktion

Über die Gruppenzuordnung (MD22000 \$MC_AUXFU_ASSIGN_GROUP) können anwenderdefinierte Hilfsfunktionen auch mehreren Gruppen zugeordnet werden. Beim Satzsuchlauf werden diese Hilfsfunktionen bezüglich aller projizierten Gruppen aufgesammelt.

Hinweis

Vordefinierte Hilfsfunktionen können nur einer Gruppe zugeordnet werden.

Beispiel

In der DIN sind folgende M-Befehle zur Kühlmittelausgabe vorgesehen:

- M7: Kühlmittel 2 EIN
- M8: Kühlmittel 1 EIN
- M9: Kühlmittel 1 und 2 AUS

Damit beide Kühlmittel auch gemeinsam aktiv sein können:

- Müssen M7 und M8 in zwei getrennten Gruppen aufgesammelt werden (z. B. Gruppe 5 und 6)
- Muss M9 diesen beiden Gruppen zugeordnet sein, z. B.:
 - Gruppe 5: M7, M9
 - Gruppe 6: M8, M9

Parametrierung:

MD11100 \$MN_AUXFU_MAXNUM_GROUP_ASSIGN = 4

MD22000 \$MC_AUXFU_ASSIGN_GROUP [0] = 5

MD22000 \$MC_AUXFU_ASSIGN_GROUP [1] = 5

MD22000 \$MC_AUXFU_ASSIGN_GROUP [2] = 6

MD22000 \$MC_AUXFU_ASSIGN_GROUP [3] = 6

MD22010 \$MC_AUXFU_ASSIGN_TYPE [0] = M

MD22010 \$MC_AUXFU_ASSIGN_TYPE [1] = M

MD22010 \$MC_AUXFU_ASSIGN_TYPE [2] = M

MD22010 \$MC_AUXFU_ASSIGN_TYPE [3] = M

MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [0] = 0

MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [1] = 0

MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [2] = 0

MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [3] = 0

MD22030 \$MC_AUXFU_ASSIGN_VALUE [0] = 7

MD22030 \$MC_AUXFU_ASSIGN_VALUE [1] = 9

MD22030 \$MC_AUXFU_ASSIGN_VALUE [2] = 8

MD22030 \$MC_AUXFU_ASSIGN_VALUE [3] = 9

MD22035 \$MC_AUXFU_ASSIGN_SPEC [0] = 'H121'

MD22035 \$MC_AUXFU_ASSIGN_SPEC [1] = 'H121'

MD22035 \$MC_AUXFU_ASSIGN_SPEC [2] = 'H121'

MD22035 \$MC_AUXFU_ASSIGN_SPEC [3] = 'H121'

Teileprogramm (Ausschnitt):**Programmcode**

```

...
N10 ... M8
N20 ... M9
N30 ... M7
...
```

Beim Satzsuchlauf wird die Hilfsfunktion M9 bezüglich der Gruppen 5 und 6 aufgesammelt.

Abfrage der aufgesammelten M-Hilfsfunktionen:

M-Funktion der 5. Gruppe: \$AC_AUXFU_M_VALUE [4] = 7

M-Funktion der 6. Gruppe: \$AC_AUXFU_M_VALUE [5] = 9

9.13.3 Zeitstempel der aktiven M-Hilfsfunktion

Bei der Ausgabe aufgesamelter Hilfsfunktionen nach Satzsuchlauf muss die Reihenfolge beim Aufsammeln beachtet werden. Es wird deshalb zu jeder Gruppe ein Zeitstempel mitgeführt, der über die folgende Systemvariable gruppenspezifisch abgefragt werden kann:

\$AC_AUXFU_M_TICK[<n>] (Zeitstempel der aktiven M-Hilfsfunktion)

9.13.4 Ermittlung der Ausgabe-Reihenfolge**Funktion**

Um die Ermittlung der Ausgabe-Reihenfolge für M-Hilfsfunktionen für den Programmierer einfach zu machen, wird folgende vordefinierte Prozedur zur Verfügung gestellt:

```
AUXFUMSEQ(VAR INT _NUM_IN, VAR INT _M_IN[], VAR INT _EXT_IN[], VAR
INT _NUM_OUT, VAR INT _M_OUT[], VAR INT _EXT_OUT[])
```

Eingangsparameter:

VAR INT _NUM_IN:	Anzahl relevanter M-Befehle
VAR INT _M_IN[]:	Feld von relevanten M-Codes
VAR INT _EXT_IN[]:	Feld von relevanten M-Adresserweiterungen

Ausgangsparameter:

VAR INT _NUM_OUT:	Anzahl ermittelter M-Codes
VAR INT _M_OUT[]:	Feld von ermittelten M-Codes
VAR INT _EXT_OUT[]:	Feld von ermittelten M-Adresserweiterungen

Die Funktion ermittelt für die vorgegebenen M-Codes die Ausgabe-Reihenfolge der gruppenspezifisch aufgesammelten M-Hilfsfunktionen. Die Reihenfolge ergibt sich aus den

Aufsammlzeitpunkten \$AC_AUXFU_M_TICK[<n>] (siehe Kapitel "Zeitstempel der aktiven M-Hilfsfunktion (Seite 444)").

Ein bestimmter M-Code wird immer nur einmal berücksichtigt, auch wenn er mehreren Gruppen angehört. Ist die Anzahl der relevanten M-Befehle kleiner oder gleich 0, so werden alle aufgesammelten M-Codes ausgegeben. Die Anzahl der relevanten M-Befehle ist auf 64 begrenzt.

Beispiel

M-Befehle zur Kühlmittelausgabe:

- M7: Kühlmittel 2 EIN
- M8: Kühlmittel 1 EIN
- M9: Kühlmittel 1 und 2 AUS

Gruppenzuordnung:

- Gruppe 5: M7, M9
- Gruppe 6: M8, M9

Teileprogramm (Ausschnitt):

Programmcode
...
N10 ... M8
N20 ... M9
N30 ... M7
...

Bei Satzsuchlauf werden die Hilfsfunktionen gruppenspezifisch aufgesammelt. Die jeweils letzte Hilfsfunktion einer Hilfsfunktionsgruppe wird nach Satzsuchlauf an die PLC ausgegeben:

- Gruppe 5: M7
- Gruppe 6: M9

Werden diese in der Reihenfolge M7 → M9 ausgegeben, so ist anschließend kein Kühlmittel aktiv. Im Programmablauf wäre jedoch Kühlmittel 2 aktiv. Die korrekte Ausgabe-Reihenfolge für die M-Hilfsfunktionen wird deshalb mit einem ASUP ermittelt, dass die vordefinierte Prozedur AUXFUMSEQ (...) enthält:

Programmcode
DEF INT _I, _M_IN[3], _EXT_IN[3], _NUM_OUT, _M_OUT[2], _EXT_OUT[2]
_M_IN[0]=7 _EXT_IN[0]=0
_M_IN[1]=8 _EXT_IN[1]=0
_M_IN[2]=9 _EXT_IN[2]=0
AUXFUMSEQ(3, _M_IN, _EXT_IN, _NUM_OUT, _M_OUT, _EXT_OUT)
FOR _I = 0 TO _NUM_OUT-1
M[_EXT_OUT[_I]]=_M_OUT[_I]
ENDFOR

9.13.5 Ausgabeunterdrückung von Spindel-spezifischen Hilfsfunktionen

Funktion

In Verbindung mit bestimmten Situationen, z. B. einem Werkzeugwechsel, kann es erforderlich sein, die bei Satzsuchlauf aufgesammelten spindelspezifischen Hilfsfunktionen nicht in den Aktionssätzen, sondern erst zu einem späteren Zeitpunkt, z. B. nach einem Werkzeugwechsel, auszugeben. Die automatische Ausgabe der spindelspezifischen Hilfsfunktionen nach Satzsuchlauf kann dazu unterdrückt werden. Die Ausgabe kann dann zu einem späteren Zeitpunkt manuell durch Überspeichern oder durch ein ASUP erfolgen.

Parametrierung

Die Unterdrückung der automatischen Ausgabe der spindelspezifischen Hilfsfunktionen nach Satzsuchlauf wird eingestellt über das Maschinendatum:

MD11450 \$MN_SEARCH_RUN_MODE (Verhalten nach Satzsuchlauf)

Bit	Wert	Bedeutung
2	0	Die Ausgabe der spindelspezifischen Hilfsfunktionen erfolgt in den Aktionssätzen
	1	Die Ausgabe der Hilfsfunktionen in den Aktionssätzen wird unterdrückt.

Systemvariablen

Die spindelspezifischen Hilfsfunktionen werden bei Satzsuchlauf, unabhängig von der oben genannten Parametrierung, immer in den folgenden Systemvariablen gespeichert:

Systemvariable	Beschreibung	
\$P_SEARCH_S [<n>]	Aufgesammelte Spindeldrehzahl	
	Wertebereich:	0 ... Smax
\$P_SEARCH_SDIR [<n>]	Aufgesammelte Spindeldrehrichtung	
	Wertebereich:	3, 4, 5, -5, -19, 70
\$P_SEARCH_SGEAR [<n>]	Aufgesammelte Spindelgetriebestufe-M-Funktion	
	Wertebereich:	40 ... 45
\$P_SEARCH_SPOS [<n>]	Aufgesammelte Spindelposition	
	Wertebereich:	0 ... MD30330 \$MA_MODULO_RANGE (Größe des Modulbereichs)
	bzw.	
	Aufgesammelter Fahrweg	
	Wertebereich:	-100.000.000 ... 100.000.000
\$P_SEARCH_SPOSMODE [<n>]	Aufgesammelter Positionsanfahrmodus	
	Wertebereich:	0 ... 5

Zur späteren Ausgabe der spindelspezifischen Hilfsfunktionen können die Systemvariablen z. B. in einem ASUP gelesen und nach Ausgabe der Aktionssätze ausgegeben werden:

DB21, ... DBX32.6 = 1 (Letzter Aktionssatz aktiv)

Hinweis

Die Inhalte der Systemvariablen \$P_S, \$P_DIR und \$P_SGEAR können nach Satzsuchlauf durch Synchronisationsvorgänge verloren gehen.

Für weiterführende Informationen zu ASUP, Satzsuchlauf und Aktionssätzen siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)".

Beispiel

Satzsuchlauf auf Kontur mit Unterdrückung der Ausgabe der spindelspezifischen Hilfsfunktionen und Start eines ASUP nach der Ausgabe der Aktionssätze.

Parametrierung: MD11450 \$MN_SEARCH_RUN_MODE, Bit 2 = 1

Nach dem Satzsuchlauf auf N55 wird das ASUP gestartet.

Teileprogramm:

Programmcode	Kommentar
N05 M3 S200	; Spindel 1
N10 G4 F3	
N15 SPOS=111	; Spindel 1 wird im ASUP auf 111 Grad positioniert
N20 M2=4 S2=300	; Spindel 2
N25 G4 F3	
N30 SPOS[2]=IC(77)	; Spindel 2 verfährt inkrementell um 77 Grad
N55 X10 G0	; Zielsatz
N60 G4 F10	
N99 M30	

ASUP:

Programmcode	Kommentar
PROC ASUP_SAVE	
MSG ("Ausgabe der Spindelfunktionen")	
DEF INT SNR=1	
AUSG_SPI:	
M[SNR]=\$P_SEARCH_SGEAR[SNR]	; Getriebestufe ausgeben.
S[SNR]=\$P_SEARCH_S[SNR]	; Drehzahl ausgeben (bei M40 wird passende Getriebestufe ermittelt).
M[SNR]=\$P_SEARCH_SDIR[SNR]	; Drehrichtung, Positionierung oder Achsbetrieb ausgeben.
SNR=SNR+1	; Nächste Spindel.
REPEAT AUSG_SPI P=\$P_NUM_SPINDLES-1	; Für alle Spindeln.
MSG ("")	
REPOSA	
RET	

Erläuterungen zum Beispiel

Ist die Anzahl der Spindeln bekannt, können zur Reduzierung der Programmbearbeitungszeit gleichartige Ausgaben in einem Teileprogrammsatz geschrieben werden.

Die Ausgabe von \$P_SEARCH_SDIR sollte in einem separaten Teileprogrammsatz erfolgen, da die Spindelpositionierung bzw. die Umschaltung in den Achsbetrieb zusammen mit dem Getriebestufenwechsel zu einer Alarmmeldung führen kann.

Wird das gestartete ASUP mit REPOSA abgeschlossen, bleibt Spindel 1 auf der Position 111 Grad stehen, während Spindel 2 auf die Position 77 Grad repositioniert wird.

Ist ein anderes Verhalten erforderlich, muss die Programmsequenz von z. B. "N05 M3 S..." und "N30 SPOS[2]=IC(...)" für Satzsuchlauf besonders behandelt werden.

Ob Satzsuchlauf aktiv ist, kann im ASUP über die Systemvariablen \$P_SEARCH ermittelt werden:

```
$P_SEARCH==1 ; Satzsuchlauf aktiv
```

Bei einer inkrementellen Positionierung nach Drehzahlsteuerbetrieb ist zwar der zu verfahrenende Weg definiert, jedoch ergibt sich die erreichte Endposition in einigen Fällen erst während des Positioniervorganges. Dies ist z. B. beim Positionsabgleich während des Überfahrens der Nullmarke oder beim Einschalten der Lageregelung der Fall. Aus diesem Grund wird als REPOS-Position (REPOSA im ASUP) der programmierte Weg ab der Position Null angenommen.

Randbedingungen

Aufgesammelte S-Werte

Die Bedeutung eines S-Werts im Teileprogramm ist abhängig vom aktuell aktiven Vorschubtyp:

G93, G94, G95, G97, G971: S-Wert wird als Drehzahl interpretiert

G96, G961: S-Wert wird als konstante Schnittgeschwindigkeit interpretiert

Wird die Vorschubart (z. B. für einen Werkzeugwechsel) vor der Ausgabe der Systemvariablen \$P_SEARCH_S geändert, muss, um zu vermeiden, dass ein falscher Vorschubtyp zugrunde gelegt wird, die Vorschubart wieder auf die ursprüngliche Einstellung aus dem Zielsatz des Teileprogramms restauriert werden.

Aufgesammelte Drehrichtung

Bei der Ausgabe der Drehrichtung wird die Systemvariable \$P_SEARCH_SDIR zum Zeitpunkt des Satzsuchlauf-Starts mit dem Wert "-5" vorbesetzt. Dieser Wert ist bei der Ausgabe wirkungslos.

Damit wird sichergestellt, dass bei Satzsuchlauf über Bereiche, in denen Spindeln nicht mit einer Drehrichtung, Positionierung oder Achsbetrieb programmiert werden, der letzte Spindelbetriebsmode erhalten bleibt.

Die Programmierung von M19, SPOS und SPOSA wird als "M-19" (interner M19) in der Systemvariablen \$P_SEARCH_SDIR alternativ zu M3, M4, M5 und M70 aufgesammelt.

Bei der Ausgabe von "M-19" werden die Positionierdaten intern aus der Systemvariablen \$P_SEARCH_SPOS und \$P_SEACH_SPOSMODE gelesen. Beide Systemvariable sind auch beschreibbar, um z. B. Korrekturen vornehmen zu können.

Hinweis

Die Werte "-5" und "19" bleiben dem Anwender aufgrund der oben genannten Zuweisungen (z. B. M[<n>] = \$P_SEARCH_SDIR[<n>]) prinzipiell verborgen und müssen nur bei spezieller Auswertung der Systemvariablen im ASUP beachtet werden.

9.13.6 Hilfsfunktionsausgabe bei Satzsuchlauf Typ 5 (SERUPRO)

Ausgabeverhalten

Beim Satzsuchlauf Typ 5 (SERUPRO) kann eine Hilfsfunktion während des Satzsuchlaufs an die PLC ausgegeben und/oder in den folgenden Systemvariablen gruppenspezifisch aufgesammelt werden:

- \$AC_AUXFU_PREDEF_INDEX[<n>] (Index einer vordefinierten Hilfsfunktion)
- \$AC_AUXFU_TYPE[<n>] (Typ der Hilfsfunktion)
- \$AC_AUXFU_STATE[<n>] (Ausgabestatus der Hilfsfunktion)
- \$AC_AUXFU_EXT[<n>] (Adresserweiterung der Hilfsfunktion)
- \$AC_AUXFU_VALUE[<n>] (Wert der Hilfsfunktion)

Zur Beschreibung der Systemvariablen siehe Kapitel "Abfrage von Systemvariablen (Seite 462)".

Ausgabesteuerung

Ob eine Hilfsfunktion während Satzsuchlauf Typ 5 (SERUPRO) an die PLC ausgegeben und/oder in den genannten Systemvariablen gruppenspezifisch aufgesammelt wird, lässt sich projektieren über Bit 9 und 10 der Maschinendaten:

- MD22080 \$MC_AUXFU_PREDEF_SPEC[<n>]
(Ausgabeverhalten von vordefinierten Hilfsfunktionen)
mit <n> = Systemfunktionsindex (0 ... 32)
- MD22035 \$MC_AUXFU_ASSIGN_SPEC[<n>]
(Ausgabeverhalten von anwenderdefinierten Hilfsfunktionen)
mit <n> = Hilfsfunktionsindex (0 ... 254)
- MD11110 \$MN_AUXFU_GROUP_SPEC[<n>]
(Ausgabeverhalten der Hilfsfunktionen einer Gruppe)
mit <n> = Gruppenindex (0 ... 63)

Bit	Wert	Bedeutung
9	0	Keine Aufsammlung während Satzsuchlauf Typ 5 (SERUPRO)
	1	Aufsammlung während Satzsuchlauf Typ 5 (SERUPRO)

Bit	Wert	Bedeutung
10	0	Ausgabe während Satzsuchlauf Typ 5 (SERUPRO)
	1	Keine Ausgabe während Satzsuchlauf Typ 5 (SERUPRO)

Ausgabebezüher

Der Anwender kann die aufgesammelten Hilfsfunktionen kanalweise im Satzsuchlauf-ASUP an die PLC ausgeben. Zum Zwecke der serialisierten Ausgabe über mehrere Kanäle werden die drei Ausgabebezüher bei jeder Ausgabe einer Hilfsfunktion über alle Kanäle verändert:

Systemvariable	Bedeutung	
\$AC_AUXFU_TICK[<n>,<m>]	Ausgabebezüher der aktiven Hilfsfunktion	
	Index	Bedeutung
	<n>	Gruppenindex (0 ... 63)
	<m>	Ausgabebezüher (0 ... 2)
	Wert	Bedeutung
	0	Ausgabesequenzbezüher (Alle Ausgaben innerhalb eines IPO-Takts)
	1	Paket-Zähler innerhalb einer Ausgabesequenz im IPO-Takt
	2	Hilfsfunktionsbezüher innerhalb eines Pakets

Erläuterung

- Ein Hilfsfunktionspaket besteht aus max. 10 Hilfsfunktionen.
- Pro Kanal können während SERUPRO zwei Pakete pro IPO-Takt abgearbeitet werden, da Synchronaktionen in diesem Takt bearbeitet werden.
- Innerhalb eines IPO-Takts kann über alle Kanäle eine Ausgabesequenz von bis zu max. 20 Paketen (2 Pakete pro Kanal * 10 Kanäle) abgearbeitet werden.

Über die Kodierung wird angezeigt, wie viele Hilfsfunktionspakete und wieviele darin enthaltene Hilfsfunktionen im gleichen IPO-Takt abgearbeitet wurden:

- Alle Hilfsfunktionen, die in einem IPO-Takt aufgesammelt wurden, haben den gleichen Sequenzbezüher.
- Alle Hilfsfunktionen, die in einem Paket (Satz oder Synchronaktion) aufgesammelt wurden, haben den gleichen Paketbezüher.

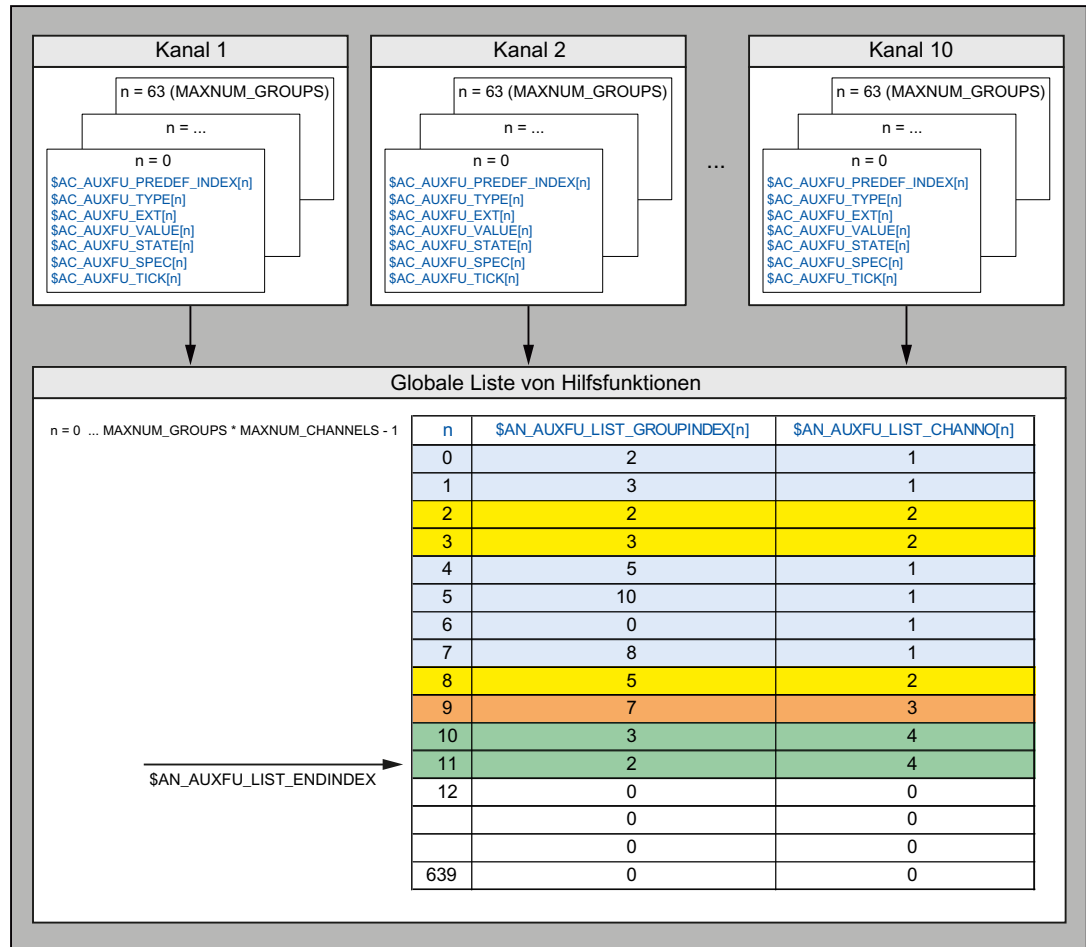
Der Hilfsfunktionsbezüher wird bei jeder aufgesammelten Hilfsfunktion inkrementiert.

Globale Liste von Hilfsfunktionen

Am Ende von SERUPRO werden die in den einzelnen Kanälen gruppenspezifisch aufgesammelten Hilfsfunktionen entsprechend ihrem Zählerstand in eine kanalübergreifende, globale Liste mit Kanalnummer und Gruppenindex eingetragen.

Systemvariable ¹⁾	Bedeutung
\$AC_AUXFU_TICK[<n>,<m>]	Zählerstand
\$AN_AUXFU_LIST_CHANNO[<n>] ²⁾	Kanalnummer

Systemvariable ¹⁾	Bedeutung
\$AN_AUXFU_LIST_GROUPINDEX[<n>] ²⁾	Gruppenindex
1) Wertebereich Index <n>: 0 ... MAXNUM_GROUPS * MAXNUM_CHANNELS - 1	
2) Die Systemvariablen sind schreib- und lesbar.	



Die globale Liste wird aufgebaut, nachdem das Suchziel gefunden wurde. Sie soll als Systemvorschlag für im nachfolgenden SERUPRO-Ende-ASUP auszugebende Hilfsfunktionen dienen. Soll eine Hilfsfunktion nicht ausgegeben werden, so ist der entsprechende Gruppenindex auf "0" zu setzen.

Verhalten bezüglich Spindel-Hilfsfunktionen

Nach Satzsuchlaufstart sammeln alle Kanäle die Hilfsfunktionen gruppenspezifisch in den Kanal-Variablen auf. Zur weitgehenden Restaurierung des Spindelzustandes im SERUPRO-Zielsatz durch aufgesammelte Hilfsfunktionen ist es notwendig, dass die letzte aktive Hilfsfunktion einer jeden Spindel-Hilfsfunktionsgruppe den Zustand der Spindel im Zielsatz repräsentiert. Bei Spindelzustandsübergängen werden dabei obsolete Hilfsfunktionen gelöscht oder, wenn notwendig, implizite Hilfsfunktionen eingetragen.

Alle Spindel-Hilfsfunktionen der globalen Hilfsfunktionsliste müssen im Zielsatz den erreichten Spindelzuständen entsprechen, damit die Hilfsfunktionen bei Ausgabe der Liste

abarbeitungsfähig sind und keine Alarmer oder ungewollte Spindelzustände angefordert werden, die eine Fortsetzung des Teileprogrammes verhindern können.

Betroffen davon sind die Hilfsfunktionsgruppen einer jeden im System projektierten Spindel, wobei die Spindelnummer der Adresserweiterung einer Hilfsfunktion entspricht.

Gruppe a: M3, M4, M5, M19, M70

Gruppe b: M40, M41, M42, M43, M44, M45

Gruppe c: S

Löschen obsoleter Hilfsfunktionen

Bei folgenden Funktionen werden für die betroffene Spindel die Hilfsfunktionen der Gruppe a gelöscht:

- Für die Folgespindel bei Einschalten einer generischen Kopplung wie COUPON, TRAILON, EGON, ...

Erzeugen von impliziten Hilfsfunktionen der Gruppe a

Bei folgenden Funktionen werden für die betroffene Spindel Hilfsfunktionen der Gruppe a implizit erzeugt:

- Für die Folgespindel beim Ausschalten der Synchronspindelkopplung
 - COUPOF erzeugt im Hauptlauf je nach Kopplungssituation M3, M4 und S oder M5.
 - COUPOF(S<n>, S<m>, POS) und COUPOFS(S<n>, S<m>, POS, POS) erzeugen M3, M4 und S.
 - COUPOFS erzeugt im Hauptlauf M5.
 - COUPOFS(S<n>, S<m>, POS) erzeugt im Hauptlauf M19.
Das implizite M19 (im ASUP "SPOS[<Adresserweiterung>] = IC(0)") schaltet den Positionierbetrieb ein ohne eine Verfahrbewegung.
- Bei einer Verfahrbewegung als Achse oder beim Übergang in den Achsbetrieb durch Anwahl einer Transformation, bei der die Spindel als Achse eingeht, wird M70 erzeugt.
- Bei SPCOF wird M5 erzeugt.

Hinweis

Die Hilfsfunktionen zur Spindelprogrammierung müssen in Zusammenhang mit den Funktionen "Achstausch" und "Achscontainer-Drehung" immer passend zum tatsächlichen Zustand (Motor) bei Tausch/Drehung mitgegeben werden. Dabei unterscheiden sich die Mechanismen von Achstausch und Achscontainer.

Beispiel für Achscontainer-Drehung:

Ein Achscontainer enthält 4 Spindeln. Jede Spindel ist jeweils einem Kanal (1 - 4) zugewiesen. Im Kanal 1 wird immer M3 S1000 programmiert und dann eine Achscontainer-Drehung ausgeführt. Die anderen Kanäle führen keine Spindelprogrammierung aus. Nach der 3. Achscontainer-Drehung und der 4. Spindelprogrammierung M3 drehen alle 4 Spindeln mit 1000 U/min rechts herum. Liegt das SERUPRO-Ende nun in diesem Bereich, so wird erwartet, dass jedes ASUP eines Kanals für die dort befindliche Spindel ein M3 S1000 enthält.

Beim Achstausch hingegen müssen die aufgesammelten Hilfsfunktionen nur dem Kanal zugeordnet werden, in dem sich die Spindel aktuell befindet.

Kanalübergreifende Hilfsfunktion

Eine Hilfsfunktion kann bei Satzsuchlauf Typ 5 (SERUPRO) auch kanalübergreifend in der globalen Hilfsfunktionsliste aufgesammelt werden. Es wird dann nur die letzte aufgesammelte Hilfsfunktion dieser Gruppe (höchster Zählerstand) in die globale Liste eingetragen.

Die entsprechende Projektierung erfolgt über folgende Maschinendaten:

- MD22080 \$MC_AUXFU_PREDEF_SPEC[<n>], Bit 11
(Ausgabeverhalten von vordefinierten Hilfsfunktionen)
mit <n> = Systemfunktionsindex
- MD22035 \$MC_AUXFU_ASSIGN_SPEC[<n>], Bit 11
(Ausgabeverhalten von anwenderdefinierten Hilfsfunktionen)
mit <n> = Hilfsfunktionsindex
- MD11110 \$MN_AUXFU_GROUP_SPEC[<n>], Bit 11
(Ausgabeverhalten der Hilfsfunktionen einer Gruppe)
mit <n> = Gruppenindex

Bit	Wert	Bedeutung
11	0	Aufsammlung kanalspezifisch
	1	Aufsammlung kanalübergreifend

Die Spindel-Hilfsfunktionen werden vorab je nach Spindelzustand am Ende des Satzsuchlaufs herausgefiltert. Die Kanaldaten werden entsprechend aktualisiert. Die globale Hilfsfunktionsliste kann in den SERUPRO-Ende-ASUPs der Reihe nach abgearbeitet und die sortierten Hilfsfunktionen kanalsynchron ausgegeben werden.

Abfrage der letzten aufgesammelten Hilfsfunktion

Der Index der letzten aufgesammelten Hilfsfunktion in der globalen Liste kann mit der Systemvariablen \$AN_AUXFU_LIST_ENDINDEX abgefragt werden.

9.13.7 SERUPRO-Ende-ASUP

Funktion

Nach Abschluss des Satzsuchlaufs mit Programmtest (SERUPRO) müssen vor dem Start der nachfolgenden Bearbeitung die während des Satzsuchlaufs aufgesammelten Hilfsfunktionen ausgegeben werden. Während des Satzsuchlaufs werden dazu die Hilfsfunktionen in einer globalen Liste aufgesammelt. Aus dieser Liste generieren kanalspezifisch die SERUPRO-Ende-ASUPs die entsprechenden Teileprogrammsätze. Damit wird sicher gestellt, dass die aufgesammelten Hilfsfunktionen sowohl kanalspezifisch als auch kanalübergreifend in zeitlich korrekter Reihenfolge ausgegeben werden. Ein vollständiges funktionsfähiges SERUPRO-Ende-ASUP ist Bestandteil der NC-Software.

Das SERUPRO-Ende-ASUP kann vom Anwender/Maschinenhersteller verändert werden. Die nachfolgend beschriebenen Funktionen unterstützen die Bearbeitung der globalen Liste der Hilfsfunktionen und die Generierung der für die synchronisierte Hilfsfunktionsausgabe erforderlichen Teileprogrammsätze.

Funktion AUXFUSYNC(...)

Funktion:

Die Funktion `AUXFUSYNC` generiert aus der globalen Liste der Hilfsfunktionen pro Aufruf einen vollständigen Teileprogrammsatz als String. Der Teileprogrammsatz enthält entweder Hilfsfunktionen oder Befehle zur Synchronisation der Hilfsfunktionsausgaben (`WAITM`, `G4`, etc.).

Die Funktion löst Vorlaufstopp aus.

Syntax:

```
PROC AUXFUSYNC (VAR INT <NUM>, VAR INT <GROUPINDEX>[10], VAR
STRING[400] <ASSEMBLED>)
```

Parameter:

<NUM>: Enthält Informationen über den im Parameter `<ASSEMBLED>` gelieferten Teileprogrammsatz bzw. den darin enthaltenen Hilfsfunktionen.

Wertebereich: -1, 0, 1 ... 10

Wert Bedeutung

≥1	Anzahl der im Teileprogrammsatz enthaltenen Hilfsfunktionen
0	Teileprogrammsatz ohne Hilfsfunktionen, z.B. <code>WAITM</code> , <code>G4</code>
-1	Ende-Kennung. Die globale Liste der Hilfsfunktionen ist für den aktuellen Kanal vollständig bearbeitet.

<GROUPINDEX>: Enthält die Indizes der im Teileprogrammsatz enthaltenen Hilfsfunktionsgruppen. Mit Index = Nummer der Hilfsfunktionsgruppe - 1

<ASSEMBLED>: Enthält den vollständigen Teileprogrammsatz für das kanalspezifische SERUPRO-Ende-ASUP als String.

Weitere Informationen:

Wurden Hilfsfunktionen über eine Synchronaktion aufgesammelt, werden zwei NC-Sätze erzeugt. Ein NC-Satz zur Ausgabe der Hilfsfunktionen. Ein ausführbarer NC-Satz über den der NC-Satz zur Ausgabe der Hilfsfunktionen zum Hauptlauf transportiert wird:

1. Ausgabe der Hilfsfunktionen über Synchronaktion, z. B.: `WHEN TRUE DO M100 M102`
2. Ausführbaren NC-Satz, z. B.: `G4 F0.001`

Funktion AUXFUDEL(...)

Funktion:

Die Funktion AUXFUDEL löscht kanalspezifisch für den aufrufenden Kanal die angegebene Hilfsfunktion aus der globalen Liste der Hilfsfunktionen. Das Löschen erfolgt durch Setzen des entsprechenden Gruppenindex ...GROUPINDEX[n] auf 0.

Der Aufruf der Funktion muss vor dem Aufruf von AUXFUSYNC erfolgen.

Die Funktion löst Vorlaufstopp aus.

Syntax:

```
PROC AUXFUDEL (CHAR <TYPE>, INT <EXTENSION>, REAL <VALUE>, INT
<GROUP>)
```

Parameter:

<TYPE>:	Typ der zu löschenden Hilfsfunktion
<EXTENSION>:	Adresserweiterung der zu löschenden Hilfsfunktion
<VALUE>:	Wert der zu löschenden Hilfsfunktion
<GROUP>:	Nummer der Hilfsfunktionsgruppe

Funktion AUXFUDELG(...)

Funktion:

Die Funktion AUXFUDELG löscht kanalspezifisch für den aufrufenden Kanal alle Hilfsfunktionen der angegebenen Hilfsfunktionsgruppe aus der globalen Liste der Hilfsfunktionen. Das Löschen erfolgt durch Setzen des entsprechenden Gruppenindex ...GROUPINDEX[n] auf 0.

Der Aufruf der Funktion muss vor dem Aufruf von AUXFUSYNC erfolgen.

Die Funktion löst Vorlaufstopp aus.

Syntax:

```
PROC AUXFUDELG (INT <GROUP>)
```

Parameter:

<GROUP>:	Nummer der Hilfsfunktionsgruppe
----------	---------------------------------

Mehrkanaliger Satzsuchlauf

ACHTUNG

Mehrkanaliger Satzsuchlauf und AUXFUDEL / AUXFUDELG

Werden bei einem mehrkanaligen Satzsuchlauf in den SERUPRO-Ende-ASUPs Hilfsfunktionen mit AUXFUDEL / AUXFUDELG aus der globalen Liste der Hilfsfunktionen gelöscht, muss vor dem Aufruf der Funktion AUXFUSYNC, eine Synchronisation der beteiligten Kanäle erfolgen. Durch die Synchronisation ist dann sicher gestellt, dass vor dem Aufruf der Funktion AUXFUSYNC alle Löschaufträge bearbeitet wurden und eine konsistente Liste vorliegt.

Beispiele

Zwei Beispiele für den Aufbau eines anwenderspezifischen SERUPRO-Ende ASUPs.

Beispiel 1: Löschen von Hilfsfunktionen und Erzeugen der Hilfsfunktionsausgabe mit AUXFUSYNC(...)

Programmcode	Kommentar
N10 DEF STRING[400] ASSEMBLED=""	
N20 DEF STRING[31] FILENAME="/_N_CST_DIR/_N_AUXFU_SPF"	
N30 DEF INT GROUPINDEX[10]	
N40 DEF INT NUM	
N60 DEF INT ERROR	
N90	
N140 AUXFUDEL("M",2,3,5)	; M2=3 (5. Hilfsfunktionsgruppe) löschen
N150	
N170 AUXFUDELG(6)	; die aufgesammelte Hilfsfunktion der ; 6. Gruppe löschen.
N180	
N190 IF ISFILE(FILENAME)	
N210 DELETE(ERROR,FILENAME)	; Datei FILENAME löschen
N220 IF (ERROR<>0)	; Fehlerauswertung
N230 SETAL(61000+ERROR)	
N240 ENDIF	
N250 ENDIF	
; VORSICHT!	
; Werden bei einem mehrkanaligen Satzsuchlauf Hilfsfunktionen mit AUXFUDEL/AUXFUDELG	
; aus der globalen Liste der Hilfsfunktionen gelöscht, muss vor der Schleife zum	
; Erzeugen des Unterprogramms FILENAME mit AUXFUSYNC eine Synchronisation der Kanäle	
; erfolgen. Durch die Synchronisation wird sicher gestellt, dass alle Löschaufträge	
; in allen Kanälen bearbeitet wurden und eine konsistente Liste vorliegt.	
; Beispiel: WAITM(99,1,2,3)	
N270 LOOP	
N300 AUXFUSYNC(NUM,GROUPINDEX,ASSEMBLED)	; Teileprogrammsatz erzeugen lassen

Programmcode	Kommentar
N310	
N320 IF (NUM== -1)	; Alle Hilfsfunktionen des Kanals
	; sind abgearbeitet.
N340 GOTO LABEL1	
N350 ENDIF	
N380 WRITE (ERROR, FILENAME, ASSEMBLED)	; Teileprogrammsatz in Datei FILENAME schreiben.
N390 IF (ERROR <> 0)	; Fehlerauswertung
N400 SETAL (61000+ERROR)	
N410 ENDIF	
N430 ENDLOOP	
N440	
N450 LABEL1:	
N460	
N480 CALL FILENAME	; Erzeugtes Unterprogramm abarbeiten.
N490	
N510 DELETE (ERROR, FILENAME)	; Datei nach Ausführung wieder löschen.
N520 IF (ERROR <> 0)	
N530 SETAL (61000+ERROR)	
N540 ENDIF	
N550	
N560 M17	

Beispiel 2: Löschen von Hilfsfunktionen und Erzeugen der Hilfsfunktionsausgabe ohne AUXFUSYNC(...)

Programmcode	Kommentar
N0610 DEF STRING[400] ASSEMBLED=""	
N0620 DEF STRING[31] FILENAME="/_N_CST_DIR/_N_AUXFU_SPF"	
N0630 DEF INT GROUPINDEX[10]	
N0640 DEF INT NUM	
N0650 DEF INT LAUF	
N0660 DEF INT ERROR	
N0670 DEF BOOL ISQUICK	
N0680 DEF BOOL ISSYNACT	
N0690 DEF BOOL ISIMPL	
...	
N0760 AUXFUDEL("M", 2, 3, 5)	; M2=3 (5. Hilfsfunktionsgruppe) löschen
N0770	
N0790 AUXFUDELG(6)	; die aufgesammelte Hilfsfunktion der
	; 6. Gruppe löschen.
N0800	
N0810 IF ISFILE(FILENAME)	
N0830 DELETE (ERROR, FILENAME)	; Datei ist bereits vorhanden und muss
	; gelöscht werden.

Programmcode	Kommentar
N0840 IF (ERROR<>0)	
N0850 SETAL(61000+ERROR)	
N0860 ENDIF	
N0870 ENDIF	
N0880	
; VORSICHT!	
; Werden bei einem mehrkanaligen Satzsuchlauf Hilfsfunktionen mit AUXFUDEL/AUXFUDELG	
; aus der globalen Liste der Hilfsfunktionen gelöscht, muss vor der Schleife zum	
; Erzeugen des Unterprogramms FILENAME mit AUXFUSYNC eine Synchronisation der Kanäle	
; erfolgen. Durch die Synchronisation wird sicher gestellt, dass alle Löschaufträge	
; in allen Kanälen bearbeitet wurden und eine konsistente Liste vorliegt.	
; Beispiel: WAITM(99,1,2,3)	
N0890 LOOP	
N0920 AUXFUSYNC(NUM,GROUPINDEX,ASSEMBLED)	; Prozedur zur Erzeugung von
	; Hilfsfunktionssätzen aus der globalen
	; Hilfsfunktionsliste.
N0930	
N0940 IF (NUM== -1)	; Alle Hilfsfunktionen des Kanals sind
	; abgearbeitet.
N0960 GOTO LABEL1	
N0970 ENDIF	
N0980	
N1000 IF (NUM>0)	; Wenn Hilfsfunktionen zur Ausgabe kommen
	; wird der Satz erzeugt.
N1010	
N1020 ASSEMBLED=""	
N1030	
N1050 FOR LAUF=0 TO NUM-1	; Aufgesammelte Hilfsfunktionen für einen
	; Satz.
N1060	
N1080 IF GROUPINDEX[LAUF]<>0	; Aus der globalen Liste gelöschte
	; Hilfsfunktionen haben den Gruppenindex 0.
N1090	
N1100 ISQUICK=\$AC_AUXFU_SPEC[GROUPINDEX[LAUF]] BAND'H2'	
N1110	
N1120 ISSYNACT=\$AC_AUXFU_SPEC[GROUPINDEX[LAUF]] BAND'H1000'	
N1130	
N1140 ISIMPL=\$AC_AUXFU_SPEC[GROUPINDEX[LAUF]] BAND'H2000'	
N1150	
N1180 IF ISSYNACT	; Satz für die M-Hilfsfunktionsausgabe
	; zusammenbauen
N1190 ASSEMBLED= ASSEMBLED << "WHEN TRUE DO "	
N1200 ENDIF	
N1210 ; Implizit erzeugtes M19 wird auf SPOS[SPI(<Spindelnr.>)] = IC(0) abgebildet.	

Programmcode	Kommentar
N1230	IF (ISIMPL AND (\$AC_AUXFU_VALUE[GROUPINDEX[LAUF]]==19))
N1240	ASSEMBLED= ASSEMBLED << "SPOS[SPI(" << \$AC_AUXFU_EXT[GROUPINDEX[LAUF]] << ")=IC(0) "
N1260	ELSE
N1270	ASSEMBLED= ASSEMBLED << "M[" << \$AC_AUXFU_EXT[GROUPINDEX[LAUF]] << "]="
N1280	
N1290	IF ISQUICK
N1300	ASSEMBLED= ASSEMBLED << "QU("
N1310	ENDIF
N1320	
N1330	ASSEMBLED= ASSEMBLED << \$AC_AUXFU_VALUE[GROUPINDEX[LAUF]]
N1340	
N1350	IF ISQUICK
N1360	ASSEMBLED= ASSEMBLED << ")"
N1370	ENDIF
N1380	ENDIF
N1400	ENDIF
N1420	ENDFOR
N1430	
N1450	WRITE(ERROR,FILENAME,ASSEMBLED) ; Hilfsfunktionssatz in Datei schreiben.
N1460	
N1470	IF ISSYNACT
N1480	ASSEMBLED="G4 F0.001"
N1490	WRITE(ERROR,FILENAME,ASSEMBLED)
N1500	ENDIF
N1510	
N1520	ELSE
N1540	WRITE(ERROR,FILENAME,ASSEMBLED) ; Hilfsfunktionssatz in Datei schreiben.
N1550	ENDIF
N1560	
N1570	ENDLOOP
N1580	
N1590	LABEL1:
N1600	
N1620	CALL FILENAME ; Erzeugtes Unterprogramm abarbeiten.
N1630	
N1650	DELETE(ERROR,FILENAME) ; Datei nach Ausführung wieder löschen.
N1660	IF (ERROR<>0)
N1670	SETAL(61000+ERROR)
N1680	ENDIF
N1690	
N1700	M17

9.14 Implizit ausgegebene Hilfsfunktionen

Funktion

Implizit ausgegebene Hilfsfunktionen sind Hilfsfunktionen, die nicht explizit programmiert wurden und zusätzlich von anderen Systemfunktionen (z. B. Transformationsanwahl, Werkzeuganwahl, etc.) ausgegeben werden. Diese impliziten Hilfsfunktionen führen zu keiner Systemfunktion, sondern die M-Codes werden entsprechend ihres parametrisierten Ausgabeverhaltens aufgesammelt und/oder an die PLC ausgegeben.

Parametrierung

Die M-Codes für implizit auszugebende Hilfsfunktionen werden festgelegt mit den Maschinendaten:

- MD22530 \$MC_TOCARR_CHANGE_M_CODE (M-Code bei Werkzeugträgerwechsel)
Der Betrag dieses Maschinendatums gibt die Nummer des M-Codes an, der bei Aktivierung eines Werkzeugträgers an der NC/PLC-Nahtstelle ausgegeben wird.
Ist der Wert positiv, wird immer der unveränderte M-Code ausgegeben.
Ist der Wert negativ, wird die Nummer des Werkzeugträgers zum Betrag des Maschinendatums addiert und diese Nummer ausgegeben.
- MD22532 \$MC_GEOAX_CHANGE_M_CODE (M-Code bei Umschaltung der Geometrieachsen)
Nummer des M-Codes, der bei einer Umschaltung der Geometrieachsen an der NC/PLC-Nahtstelle ausgegeben wird.
- MD22534 \$MC_TRAFO_CHANGE_M_CODE (M-Code bei Transformationswechsel)
Nummer des M-Codes, der bei einer Transformationsumschaltung der Geometrieachsen an der NC/PLC-Nahtstelle ausgegeben wird.

Hinweis

Hat die Nummer des auszugebenden M-Codes oder der Betrag von MD22530 / MD22532 / MD22534 selbst einen der Werte 0 bis 6, 17 oder 30, wird kein M-Code ausgegeben. Es wird nicht überwacht, ob ein derart erzeugter M-Code zu Konflikten mit anderen Funktionen führt.

Ausgabeverhalten

Bei implizit ausgegebenen Hilfsfunktionen ist das Bit 13 im Maschinendatum MD22080 bzw. MD22035 (Ausgabeverhalten von vordefinierten bzw. anwenderdefinierten Hilfsfunktionen) gesetzt.

Über die Systemvariable \$AC_AUXFU_SPEC[<n>] kann dieses Bit abgefragt werden.

Implizit ausgegebene Hilfsfunktion M19

Um eine Durchgängigkeit von M19 und SPOS bzw. SPOSA bezüglich des Verhaltens an der NC/PLC-Nahtstelle zu erreichen, kann bei SPOS und SPOSA die Hilfsfunktion M19 an die NC/PLC-Nahtstelle ausgegeben werden (siehe Kapitel "Allgemeine Funktionalität (Seite 1301)").

Die Implizit ausgegebene Hilfsfunktion M19 wird beim Satzsuchlauf aufgesammelt.

9.15 Informationsmöglichkeiten

Informationen zu Hilfsfunktionen (z. B. über den Ausgabestatus) sind möglich über:

- die gruppenspezifische modale M-Hilfsfunktionsanzeige an der Bedienoberfläche.
- die Abfrage von Systemvariablen in Teileprogrammen und Synchronaktionen.

9.15.1 Gruppenspezifische modale M-Hilfsfunktionsanzeige

Funktion

An der Bedienoberfläche kann der Ausgabe- und Quittungsstatus von M-Hilfsfunktionen gruppenspezifisch angezeigt werden.

Voraussetzungen

Vorraussetzung für die Realisierung einer funktionsorientierten Quittierung und Anzeige von M-Hilfsfunktionen ist, dass die Verwaltung der Hilfsfunktionen in der PLC, also im Anwenderprogramm selbst, erfolgen muss. Deshalb obliegt es den PLC-Programmierer, die Quittierung dieser Hilfsfunktionen selbst vorzunehmen. Er muss wissen, welche Hilfsfunktionen in welcher Gruppe sind, die er quittieren muss.

Standard

M-Hilfsfunktionen, die nicht über die PLC verwaltet werden, werden vom NC an die PLC ausgegeben und als "übergeben" gekennzeichnet. Eine funktionale Quittierung gibt es für diese Hilfsfunktionen nicht. Alle nach einem Satzsuchlauf aufgesammelten M-Hilfsfunktionen werden zusätzlich zur Anzeige gebracht, damit der Bediener weiß, welche Hilfsfunktionen nach dem Start nach Satzsuchlauf ausgegeben werden.

PLC-Aktivitäten

Bei Hilfsfunktionsgruppen, die von der PLC selbst verwaltet werden, muss das PLC-Anwenderprogramm alle Hilfsfunktionen dieser Gruppe bei **Übernahme** und **Funktionsende** quittieren. Der PLC-Programmierer muss alle Hilfsfunktionen dieser Gruppen kennen.

Sonstiges

Es werden nur die M-Hilfsfunktionen **gruppenspezifisch** angezeigt. Die satzweise Anzeige bleibt zusätzlich erhalten. Es können bis zu 15 Gruppen angezeigt werden, wobei je Gruppe immer **nur die letzte M-Funktion einer Gruppe**, die entweder aufgesammelt oder an die PLC ausgegeben wurde, angezeigt wird. Die M-Funktionen werden ihrem Status entsprechend in verschiedener Anzeigeart dargestellt:

Status	Anzeigeart
Hilfsfunktion ist aufgesammelt	Invertiert mit gelber Schrift
Hilfsfunktion ist von NC an PLC ausgegeben	Invertiert

Status	Anzeigeart
Hilfsfunktion ist von NC an PLC übergeben worden und Transportquittierung ist erfolgt	Schwarze Schrift auf grauem Hintergrund
Hilfsfunktion wird von PLC verwaltet und ist von PLC direkt übernommen worden.	Schwarze Schrift auf grauem Hintergrund
Hilfsfunktion wird von PLC verwaltet und die Funktionsquittierung ist erfolgt.	Schwarze Schrift auf grauem Hintergrund

Anzeigeaktualisierung

Die Anzeige wird so organisiert, dass zuerst immer die aufgesammelten Hilfsfunktionen vor den PLC-verwalteten und vor den NC-verwalteten ausgegeben werden. Eine aufgesammelte Hilfsfunktion bleibt solange als aufgesammelt markiert, bis sie vom NC an die PLC ausgegeben wurde. PLC-verwaltete Hilfsfunktionen bleiben solange bestehen, bis diese von einer anderen Hilfsfunktion abgelöst werden. Bei Reset werden nur die aufgesammelten und die NC-verwalteten gelöscht.

9.15.2 Abfrage von Systemvariablen

Funktion

Im Teileprogramm und über Synchronaktionen können Hilfsfunktionen gruppenspezifisch über Systemvariablen abgefragt werden:

\$AC_AUXFU_... [<n>] = <Wert>

Systemvariable	Bedeutung
\$AC_AUXFU_PREDEF_INDEX[<n>]	<Wert>: Index der zuletzt für eine Hilfsfunktionsgruppe aufgesammelten (Satzsuchlauf) oder ausgegebenen vordefinierten Hilfsfunktion
	Typ: INT
	Ist für die spezifizierte Gruppe noch keine Hilfsfunktion ausgegeben worden oder ist die Hilfsfunktion eine anwenderdefinierte Hilfsfunktion, so liefert die Variable den Wert "-1".
	<n>: Gruppenindex (0 ... 63)
\$AC_AUXFU_TYPE[<n>]	Hinweis: Über diese Variable lässt sich eine vordefinierte Hilfsfunktion eindeutig identifizieren.
	<Wert>: Typ der zuletzt für eine Hilfsfunktionsgruppe aufgesammelten (Satzsuchlauf) oder ausgegebenen Hilfsfunktion
	Typ: CHAR
	<n>: Gruppenindex (0 ... 63)

Systemvariable	Bedeutung	
\$AC_AUXFU_EXT[<n> bzw. M-Funktionsspezifisch: \$AC_AUXFU_M_EXT[<n>]	<Wert>:	Adresserweiterung der zuletzt für eine Hilfsfunktionsgruppe aufgesammelten (Satzsuchlauf) oder ausgegebenen Hilfsfunktion
		Typ: INT
	<n>:	Gruppenindex (0 ... 63)
\$AC_AUXFU_VALUE[<n> bzw. M-Funktionsspezifisch: \$AC_AUXFU_M_VALUE[<n>]	<Wert>:	Wert der zuletzt für eine Hilfsfunktionsgruppe aufgesammelten (Satzsuchlauf) oder ausgegebenen Hilfsfunktion
		Typ: REAL
	<n>:	Gruppenindex (0 ... 63)
\$AC_AUXFU_SPEC[<n>]	Wert:	Bit-kodiertes Ausgabeverhalten entsprechend MD22080/MD22035 (oder QU-Programmierung) der zuletzt für eine Hilfsfunktionsgruppe aufgesammelten (Satzsuchlauf) oder ausgegebenen Hilfsfunktion
		Typ: INT
	<n>:	Gruppenindex (0 ... 63)
	Hinweis: Über diese Variable lässt sich ermitteln, ob die Hilfsfunktionsausgabe mit schneller Quittierung erfolgen soll.	
\$AC_AUXFU_STATE[<n> bzw. M-Funktionsspezifisch: \$AC_AUXFU_M_STATE[<n>]	<Wert>:	Ausgabestatus der zuletzt für eine Hilfsfunktionsgruppe aufgesammelten (Satzsuchlauf) oder ausgegebenen Hilfsfunktion
		Typ: INT
		Wertebereich: 0 ... 5
		0: Hilfsfunktion ist nicht vorhanden
		1: M-Hilfsfunktion wurde per Satzsuchlauf aufgesammelt
		2: M-Hilfsfunktion wurde an die PLC ausgegeben
		3: M-Hilfsfunktion wurde an die PLC ausgegeben und die Transportquittierung ist erfolgt
		4: M-Hilfsfunktion wird von der PLC verwaltet und ist von der PLC übernommen worden
		5: M-Hilfsfunktion wird von der PLC verwaltet und die Funktionsquittierung ist erfolgt
	<n>:	Gruppenindex (0 ... 63)

Beispiel

Alle M-Hilfsfunktionen der 1. Gruppe sollen der Reihe ihrer Ausgabe nach abgespeichert werden:

```
id=1 every $AC_AUXFU_M_STATE[0]==2 do $AC_FIFO[0,0]=
$AC_AUXFU_M_VALUE[0]
```

Literatur

Weitere Informationen zu den Systemvariablen siehe:

Listenhandbuch Systemvariablen

9.16 Randbedingungen

9.16.1 Allgemeine Randbedingungen

Spindeltausch

Da die Parametrierung der Hilfsfunktionen kanalspezifisch erfolgt, müssen bei Verwendung der Funktion: "Spindeltausch" die Spindel-spezifischen Hilfsfunktionen in allen Kanälen in denen die Spindel verwendet wird, gleich parametriert werden.

Werkzeugverwaltung

Bei aktiver Werkzeugverwaltung gelten folgende Randbedingungen:

- T- und M<k>-Funktionen werden nicht an die PLC ausgegeben.
Hinweis
k ist der parametrierte Wert der Hilfsfunktion für den Werkzeugwechsel (Default: 6):
MD22560 \$MC_TOOL_CHANGE_M_CODE (Hilfsfunktion für Werkzeugwechsel)
- Wird keine Adresserweiterung programmiert, bezieht sich die Hilfsfunktion auf die Master-Spindel bzw. des Master-Werkzeughalters des Kanals.
Definition der Master-Spindel:
 - MD20090 \$MC_SPIND_DEF_MASTER_SPIND
 - Teileprogrammanweisung: SETMSDefinition des Master-Werkzeughalters
 - MD20124 \$MC_TOOL_MANAGEMENT_TOOLHOLDER
 - Teileprogrammanweisung: SETMTH

Maximale Anzahl von Hilfsfunktionen pro Teileprogrammsatz

In einem Teileprogrammsatz dürfen maximal 10 Hilfsfunktionen programmiert werden.

DL (Summenkorrektur)

Für die DL-Funktion gelten folgende Einschränkungen:

- Pro Teileprogrammsatz kann nur eine DL-Funktion programmiert werden.
- Bei Verwendung von DL-Funktionen in Synchronaktionen wird der Parameter: "Wert" nicht an die PLC ausgegeben.

9.16.2 Ausgabeverhalten

Gewindeschneiden

Während aktivem Gewindeschneiden G33, G34 und G35 wirkt für die Spindel-spezifischen Hilfsfunktionen:

- M3 (Spindel rechts)
- M4 (Spindel links)

immer mit folgendes Ausgabeverhalten:

- Ausgabedauer einen OB40-Takt (Schnelle Quittierung)
- Ausgabe während der Bewegung

Die Spindel-spezifische Hilfsfunktion M5 (Spindel halt) wird immer am Satzende ausgegeben. Der Teileprogrammsatz, der M5 beinhaltet, wird immer d. h. auch bei aktivem Bahnsteuerbetrieb, mit Genauhalt beendet.

Synchronaktionen

Bei der Hilfsfunktionsausgabe aus Synchronaktionen wird das parametrierte Ausgabeverhalten bis auf folgende Parameter ignoriert:

- Bit0: Ausgabedauer einen OB1-Zyklus (Normale Quittierung)
- Bit1: Ausgabedauer einen OB40-Takt (Schnelle Quittierung)

Hilfsfunktionen: M17 bzw. M2 / M30 (Unterprogrammende)

Allein in einem Teileprogrammsatz

Steht eine der Hilfsfunktionen M17, M2 oder M30 allein in einem Teileprogrammsatz und es ist noch eine Achse in Bewegung, erfolgt die Ausgabe der Hilfsfunktion an die PLC erst, nachdem die Achse steht.

Überlagerung des parametrierten Ausgabeverhaltens

Das parametrierte Ausgabeverhalten der Hilfsfunktionen M17 bzw. M2/M30 wird durch das im folgenden Maschinendatum festgelegte Ausgabeverhalten überlagert:

MD20800 \$MC_SPF_END_TO_VDI, Bit 0 (Unterprogrammende / Halt an PLC)

Bit	Wert	Bedeutung
0	0	Die Hilfsfunktionen M17 bzw. M2/M30 (Unterprogrammende) werden nicht an die PLC ausgegeben. Bahnsteuerbetrieb wird am Unterprogrammende nicht unterbrochen.
	1	Die Hilfsfunktionen M17 bzw. M2/M30 (Unterprogrammende) werden an die PLC ausgegeben.

Hilfsfunktion: M1 (Bedingter Halt)

Überlagerung des parametrierten Ausgabeverhaltens

Das parametrierte Ausgabeverhalten der Hilfsfunktion M1 wird durch das im folgenden Maschinendatum festgelegte Ausgabeverhalten überlagert:

MD20800 \$MC_SPF_END_TO_VDI, Bit 1 (Unterprogrammende / Halt an PLC)

Bit	Wert	Bedeutung
1	0	Die Hilfsfunktionen M01 (Bedingter Halt) wird immer an PLC ausgegeben. Eine schnelle Quittierung ist unwirksam, da M01 fest der 1. Hilfsfunktionsgruppe zugeordnet ist und damit immer am Satzende ausgegeben wird.
	1	Die Hilfsfunktionen M01 (Bedingter Halt) wird nur dann an PLC ausgegeben, wenn die Funktion: "Programmierter Halt" über die HMI-Bedienoberfläche aktiviert ist. Bei schneller Quittierung wird die M1 während der Bewegung an die PLC ausgegeben. Solange die Funktion nicht aktiv ist, wird dadurch ein Bahnsteuerbetrieb nicht unterbrochen.

Teileprogrammsätze ohne Verfahrbewegung

In einem Teileprogrammsatz ohne Verfahrbewegung werden alle Hilfsfunktionen, unabhängig von ihrem jeweiligen parametrisierten Ausgabeverhalten, sofort in einem Block ausgegeben.

Spindelspezifische Hilfsfunktionsausgabe nur als Information für das PLC-Anwenderprogramm

In bestimmten Steuerungssituationen, z.B. zum Abschluss eines Satzsuchlaufs, werden die aufgesammelten spindelspezifischen Hilfsfunktionen (z.B. M3, M4, M5, M19, M40...M45, M70) nur zur Information für das PLC-Anwenderprogramm an die NC/PLC-Nahtstelle ausgegeben. Die Steuerung erzeugt dazu einen Teileprogrammsatz (Aktionssatz) in welchem die aufgesammelten Hilfsfunktionen mit **negativer** Adresserweiterung eingetragen sind. Die entsprechenden Systemfunktionen werden dann **nicht** ausgeführt.

Beispiel: M(-2) = 41 Anforderung Getriebestufenwechsel für die 2. Spindel

9.17 Beispiele

9.17.1 Erweiterung von vordefinierten Hilfsfunktionen

Aufgabe

Parametrierung der Hilfsfunktionen M3, M4 und M5 für die 2. Spindel des Kanals.

Parametrierung: M3

Anforderungen:

- Maschinendaten-Index: 0 (erste anwenderdefinierte Hilfsfunktion)
- Hilfsfunktionsgruppe: 5
- Typ und Wert: M3 (Spindel rechts)

- Adresserweiterung: 2 entsprechend der 2. Spindel des Kanals
- Ausgabeverhalten:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung)
 - Ausgabe vor der Bewegung

Parametrierung:

MD22000 \$MC_AUXFU_ASSIGN_GROUP [0] = 5
 MD22010 \$MC_AUXFU_ASSIGN_TYPE [0] = "M"
 MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [0] = 2
 MD22030 \$MC_AUXFU_ASSIGN_VALUE [0] = 3
 MD22035 \$MC_AUXFU_ASSIGN_SPEC [0] = 'H21'

Parametrierung: M4**Anforderungen:**

- Maschinendaten-Index: 1 (zweite anwenderdefinierte Hilfsfunktion)
- Hilfsfunktionsgruppe: 5
- Typ und Wert: M4 (Spindel links)
- Adresserweiterung: 2 entsprechend der 2. Spindel des Kanals
- Ausgabeverhalten:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung)
 - Spindelreaktion nach der Quittierung
 - Ausgabe während der Bewegung

Parametrierung:

MD22000 \$MC_AUXFU_ASSIGN_GROUP [1] = 5
 MD22010 \$MC_AUXFU_ASSIGN_TYPE [1] = "M"
 MD22020 \$MC_AUXFU_ASSIGN_EXTENSION [1] = 2
 MD22030 \$MC_AUXFU_ASSIGN_VALUE [1] = 4
 MD22035 \$MC_AUXFU_ASSIGN_SPEC [1] = 'H51'

Parametrierung: M5**Anforderungen:**

- Maschinendaten-Index: 2 (dritte anwenderdefinierte Hilfsfunktion)
- Hilfsfunktionsgruppe: 5
- Typ und Wert: M5 (Spindel halt)

9.17 Beispiele

- Adresserweiterung: 2 entsprechend der 2. Spindel des Kanals
- Ausgabeverhalten:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung)
 - Spindelreaktion nach der Quittierung
 - Ausgabe am Satzende

Parametrierung:

```
MD22000 $MC_AUXFU_ASSIGN_GROUP [ 2 ]      = 5
MD22010 $MC_AUXFU_ASSIGN_TYPE [ 2 ]        = "M"
MD22020 $MC_AUXFU_ASSIGN_EXTENSION [ 2 ]    = 2
MD22030 $MC_AUXFU_ASSIGN_VALUE [ 2 ]        = 5
MD22035 $MC_AUXFU_ASSIGN_SPEC [ 2 ]         = 'H91'
```

9.17.2 Definition von Hilfsfunktionen

Aufgabe

Parametrierung der Hilfsfunktionsspezifischen Maschinendaten für eine Maschine mit folgender Konfiguration:

Spindeln

- Spindel 1: Masterspindel
- Spindel 2: Zweite Spindel

Getriebestufen

- Spindel 1: 5 Getriebestufen
- Spindel 2: keine Getriebestufen

Schaltfunktionen für Kühlwasser Ein/Aus

- Spindel 1
 - "Ein" = M50
 - "Aus" = M51
- Spindel 2
 - "Ein" = M52
 - "Aus" = M53

Anforderungen

Spindel 1 (Masterspindel)

Hinweis

Standardzuordnungen

- Die Hilfsfunktionen M3, M4, M5, M70 und M1=3, M1=4, M1=5, M1=70 der Spindel 1 (Masterspindel) sind standardmäßig der 2. Hilfsfunktionsgruppe zugeordnet.
 - Alle S- und S1-Werte der Spindel 1 (Masterspindel) sind standardmäßig der 3. Hilfsfunktionsgruppe zugeordnet.
-
- Nach Satzsuchlauf soll die zuletzt programmierte Getriebestufe ausgegeben werden. Die folgenden Hilfsfunktionen werden dazu der 9. Hilfsfunktionsgruppe zugeordnet:
 - M40, M41, M42, M43, M44, M45
 - M1=40, M1=41, M1=42, M1=43, M1=44, M1=45
 - Die Hilfsfunktionen M3, M4, M5, M70 und M1=3, M1=4, M1=5, M1=70 (2. Hilfsfunktionsgruppe) sowie S- und S1-Werte (3. Hilfsfunktionsgruppe) sollen folgendes Ausgabeverhalten haben:
 - Ausgabedauer einen OB40-Zyklus (Schnelle Quittierung)
 - Ausgabe vor der Bewegung
 - Die Hilfsfunktionen zum Getriebeumschalten M40 bis M45 und M1=40 bis M1=45 (9. Hilfsfunktionsgruppe) sollen folgendes Ausgabeverhalten haben:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung)
 - Ausgabe vor der Bewegung

Spindel 2

- In einem Satz darf nur eine M-Funktion zur Drehrichtungsumschaltung programmiert werden. Nach Satzsuchlauf soll die zuletzt programmierte Drehrichtung ausgegeben werden. Die folgenden Hilfsfunktionen werden dazu der 10. Hilfsfunktionsgruppe zugeordnet:
 - M2=3, M2=4, M2=5, M2=70
- Alle S2-Werte werden der 11. Hilfsfunktionsgruppe zugeordnet.
- Die Hilfsfunktionen M2=3, M2=4, M2=5, M2=70 (10. Hilfsfunktionsgruppe) sowie S2-Werte (11. Hilfsfunktionsgruppe) sollen folgendes Ausgabeverhalten haben:
 - Ausgabedauer einen OB40-Zyklus (Schnelle Quittierung)
 - Ausgabe vor der Bewegung

Kühlwasser

- Ein- und Ausschalten in einem Teileprogrammsatz ist nicht zulässig. Nach Satzsuchlauf soll das Kühlwasser ein- bzw. ausgeschaltet werden. Die folgenden Hilfsfunktionen werden dazu z. B. der 12. bzw. 13. Hilfsfunktionsgruppe zugeordnet:
 - 12. Hilfsfunktionsgruppe: M50, M51
 - 13. Hilfsfunktionsgruppe: M52, M53
- Die Hilfsfunktionen M50, M51 (12. Hilfsfunktionsgruppe) und M52, M53 (13. Hilfsfunktionsgruppe) sollen folgendes Ausgabeverhalten haben:
 - Ausgabedauer einen OB1-Zyklus (Normale Quittierung)
 - Ausgabe vor der Bewegung

Parametrierung der Maschinendaten

Die Parametrierung der Maschinendaten erfolgt über eine entsprechende Programmierung innerhalb eines Teileprogramms:

Programmcode	Kommentar
\$MN_AUXFU_MAXNUM_GROUP_ASSIGN=21	; Anzahl anwenderdefinierter Hilfsfunktionen pro Kanal
\$MN_AUXFU_GROUP_SPEC[1]='H22'	; Ausgabeverhalten der 2.Hilfsfunktionsgruppe
\$MN_AUXFU_GROUP_SPEC[2]='H22'	; Ausgabeverhalten der 3.Hilfsfunktionsgruppe
\$MN_AUXFU_GROUP_SPEC[8]='H21'	; Ausgabeverhalten der 9.Hilfsfunktionsgruppe
\$MC_AUXFU_ASSIGN_TYPE[0]="M"	; Beschreibung der 1.Hilfsfunktion: M40
\$MC_AUXFU_ASSIGN_EXTENSION[0]=0	
\$MC_AUXFU_ASSIGN_VALUE[0]=40	
\$MC_AUXFU_ASSIGN_GROUP[0]=9	
	; ... (entsprechend für 2.-5.Hilfsfunktion)
\$MC_AUXFU_ASSIGN_TYPE[5]="M"	; Beschreibung der 6.Hilfsfunktion: M45
\$MC_AUXFU_ASSIGN_EXTENSION[5]=0	
\$MC_AUXFU_ASSIGN_VALUE[5]=45	
\$MC_AUXFU_ASSIGN_GROUP[5]=9	
\$MC_AUXFU_ASSIGN_TYPE[6]="M"	; Beschreibung der 7.Hilfsfunktion: M1=40
\$MC_AUXFU_ASSIGN_EXTENSION[6]=1	
\$MC_AUXFU_ASSIGN_VALUE[6]=40	
\$MC_AUXFU_ASSIGN_GROUP[6]=9	
	; . . . (entsprechend für 8.-11.Hilfsfunktion)
\$MC_AUXFU_ASSIGN_TYPE[11]="M"	; Beschreibung der 12.Hilfsfunktion: M1=45
\$MC_AUXFU_ASSIGN_EXTENSION[11]=1	
\$MC_AUXFU_ASSIGN_VALUE[11]=45	
\$MC_AUXFU_ASSIGN_GROUP[11]=9	
\$MN_AUXFU_GROUP_SPEC[9] = 'H22'	; Ausgabeverhalten der 10.Hilfsfunktionsgruppe

Programmcode	Kommentar
\$MC_AUXFU_ASSIGN_TYPE[12]="M"	; Beschreibung der 13.Hilfsfunktion: M2=3
\$MC_AUXFU_ASSIGN_EXTENSION[12]=2	
\$MC_AUXFU_ASSIGN_VALUE[12]=3	
\$MC_AUXFU_ASSIGN_GROUP[12]=10	
\$MC_AUXFU_ASSIGN_TYPE[13]="M"	; Beschreibung der 14.Hilfsfunktion: M2=4
\$MC_AUXFU_ASSIGN_EXTENSION[13]=2	
\$MC_AUXFU_ASSIGN_VALUE[13]=4	
\$MC_AUXFU_ASSIGN_GROUP[13]=10	
\$MC_AUXFU_ASSIGN_TYPE[14]="M"	; Beschreibung der 15.Hilfsfunktion: M2=5
\$MC_AUXFU_ASSIGN_EXTENSION[14]=2	
\$MC_AUXFU_ASSIGN_VALUE[14]=5	
\$MC_AUXFU_ASSIGN_GROUP[14]=10	
\$MC_AUXFU_ASSIGN_TYPE[15]="M"	; Beschreibung der 16.Hilfsfunktion: M2=70
\$MC_AUXFU_ASSIGN_EXTENSION[15]=2	
\$MC_AUXFU_ASSIGN_VALUE[15]=70	
\$MC_AUXFU_ASSIGN_GROUP[15]=10	
\$MN_AUXFU_GROUP_SPEC[10]='H22'	; Spezifikation der 11.Hilfsfunktionsgruppe
\$MC_AUXFU_ASSIGN_TYPE[16]="S"	; Beschreibung der 17.Hilfsfunktion: S2=<alle Werte>
\$MC_AUXFU_ASSIGN_EXTENSION[16]=2	
\$MC_AUXFU_ASSIGN_VALUE[16]=-1	
\$MC_AUXFU_ASSIGN_GROUP[16]=11	
\$MN_AUXFU_GROUP_SPEC[11]='H21'	; Spezifikation der 12.Hilfsfunktionsgruppe
\$MC_AUXFU_ASSIGN_TYPE[17]="M"	; Beschreibung der 18.Hilfsfunktion: M50
\$MC_AUXFU_ASSIGN_EXTENSION[17]=0	
\$MC_AUXFU_ASSIGN_VALUE[17]=50	
\$MC_AUXFU_ASSIGN_GROUP[17]=12	
\$MC_AUXFU_ASSIGN_TYPE[18]="M"	; Beschreibung der 19.Hilfsfunktion: M51
\$MC_AUXFU_ASSIGN_EXTENSION[18]=0	
\$MC_AUXFU_ASSIGN_VALUE[18]=51	
\$MC_AUXFU_ASSIGN_GROUP[18]=12	
\$MN_AUXFU_GROUP_SPEC[12]='H21'	; Spezifikation der 13.Hilfsfunktionsgruppe
\$MC_AUXFU_ASSIGN_TYPE[19]="M"	; Beschreibung der 20.Hilfsfunktion: M52

9.18 Datenlisten

Programmcode	Kommentar
\$MC_AUXFU_ASSIGN_EXTENSION[19]=0	
\$MC_AUXFU_ASSIGN_VALUE[19]=52	
\$MC_AUXFU_ASSIGN_GROUP[19]=13	
\$MC_AUXFU_ASSIGN_TYPE[20]="M"	; Beschreibung der 21.Hilfsfunktion: M53
\$MC_AUXFU_ASSIGN_EXTENSION[20]=0	
\$MC_AUXFU_ASSIGN_VALUE[20]=53	
\$MC_AUXFU_ASSIGN_GROUP[20]=13	

9.18 Datenlisten

9.18.1 Maschinendaten

9.18.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10713	M_NO_FCT_STOPRE	M-Funktion mit Vorlaufstopp
10714	M_NO_FCT_EOP	M-Funktion für Spindel aktiv nach NC-RESET
10715	M_NO_FCT_CYCLE	Durch Unterprogramm zu ersetzende M-Funktion
11100	AUXFU_MAXNUM_GROUP_ASSIGN	Maximale Anzahl anwenderdefinierbarer Hilfsfunktionen pro Kanal
11110	AUXFU_GROUP_SPEC	Gruppenspezifisches Ausgabeverhalten
11450	SEARCH_RUN_MODE	Verhalten nach Satzsuchlauf

9.18.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20110	RESET_MODE_MASK	Festlegung der Steuerungsgrundstellung nach Teilprogramm-Start
20112	START_MODE_MASK	Festlegung der Steuerungs-Grundstellung nach Hochlauf und bei Reset bzw. Teilprogrammende
20270	CUTTING_EDGE_DEFAULT	Grundstellung der Werkzeugschneide ohne Programmierung
20800	SPF_END_TO_VDI	Unterprogrammende / Halt an PLC
22000	AUXFU_ASSIGN_GROUP	Gruppenzuordnung von anwenderdefinierten Hilfsfunktionen
22010	AUXFU_ASSIGN_TYPE	Typ von anwenderdefinierten Hilfsfunktionen

Nummer	Bezeichner: \$MC_	Beschreibung
22020	AUXFU_ASSIGN_EXTENSION	Adresserweiterung für anwenderdefinierte Hilfsfunktionen
22030	AUXFU_ASSIGN_VALUE	Wert von anwenderdefinierten Hilfsfunktionen
22035	AUXFU_ASSIGN_SPEC	Ausgabeverhalten von anwenderdefinierten Hilfsfunktionen
22040	AUXFU_PREDEF_GROUP	Gruppenzuordnung von vordefinierten Hilfsfunktionen
22050	AUXFU_PREDEF_TYPE	Typ von vordefinierten Hilfsfunktionen
22060	AUXFU_PREDEF_EXTENSION	Adresserweiterung für vordefinierte Hilfsfunktionen
22070	AUXFU_PREDEF_VALUE	Wert von vordefinierten Hilfsfunktionen
22080	AUXFU_PREDEF_SPEC	Ausgabeverhalten von vordefinierten Hilfsfunktionen
22100	AUXFU_QUICK_BLOCKCHANGE	Satzwechselverzögerung bei schnellen Hilfsfunktionen
22110	AUXFU_H_TYPE_INT	Typ von H-Hilfsfunktionen
22200	AUXFU_M_SYNC_TYPE	Ausgabezeitpunkt M-Funktionen
22210	AUXFU_S_SYNC_TYPE	Ausgabezeitpunkt S-Funktionen
22220	AUXFU_T_SYNC_TYPE	Ausgabezeitpunkt T-Funktionen
22230	AUXFU_H_SYNC_TYPE	Ausgabezeitpunkt H-Funktionen
22240	AUXFU_F_SYNC_TYPE	Ausgabezeitpunkt F-Funktionen
22250	AUXFU_D_SYNC_TYPE	Ausgabezeitpunkt D-Funktionen
22252	AUXFU_DL_SYNC_TYPE	Ausgabezeitpunkt DL-Funktionen
22254	AUXFU_ASSOC_M0_VALUE	Zusätzliche M-Funktion für Programm-Halt
22256	AUXFU_ASSOC_M1_VALUE	Zusätzliche M-Funktion für bedingten Halt
22530	TOCARR_CHANGE_M_CODE	M-Code bei Werkzeugträgerwechsel
22532	GEOAX_CHANGE_M_CODE	M-Code bei Umschaltung der Geometrieachsen
22534	TRAFO_CHANGE_M_CODE	M-Code bei Transformationswechsel
22560	TOOL_CHANGE_M_CODE	Hilfsfunktion für Werkzeugwechsel

9.18.2 Signale

9.18.2.1 Signale an Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Assoziiertes M01 aktivieren	DB21,DBX30.5	DB320x.DBX14.5

9.18.2.2 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
M-Funktion 1 - 5 Änderung	DB21,DBX58.0-4	DB250x.DBX4.0-4
M-Funktion 1 - 5 nicht decodiert	DB21,DBX59.0-4	-
S-Funktion 1 - 3 Änderung	DB21,DBX60.0-2	DB250x.DBX6.0

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
S-Funktion 1 - 3 Quick	DB21,DBX60.4-6	-
T-Funktion 1 - 3 Änderung	DB21,DBX61.0-2	-
T-Funktion 1 - 3 Quick	DB21,DBX61.4-6	-
D-Funktion 1 - 3 Änderung	DB21,DBX62.0-2	DB250x.DBX10.0
D-Funktion 1 - 3 Quick	DB21,DBX62.4-6	-
DL-Funktion Änderung	DB21,DBX63.0	-
DL-Funktion Quick	DB21,DBX63.4	-
H-Funktion 1 - 3 Änderung	DB21,DBX64.0-2	DB250x.DBX12.0-2
H-Funktion 1 - 3 Quick	DB21,DBX64.4-6	-
F-Funktion 1 - 6 Änderung	DB21,DBX65.0-5	-
M-Funktion 1 - 5 Quick	DB21,DBX66.0-4	-
F-Funktion 1 - 6 Quick	DB21,DBX67.0-5	-
erweiterte Adresse M-Funktion 1 (16 Bit-Int)	DB21,DBB68-69	DB250x.DBB3004
M-Funktion 1 (DInt)	DB21,DBB70-73	DB250x.DBD3000
erweiterte Adresse M-Funktion 2 (16 Bit-Int)	DB21,DBB74-75	DB250x.DBB3012
M-Funktion 2 (DInt)	DB21,DBB76-79	DB250x.DBD3008
erweiterte Adresse M-Funktion 3 (16 Bit-Int)	DB21,DBB80-81	DB250x.DBB3020
M-Funktion 3 (DInt)	DB21,DBB82-85	DB250x.DBD3016
erweiterte Adresse M-Funktion 4 (16 Bit-Int)	DB21,DBB86-87	DB250x.DBB3028
M-Funktion 4 (DInt)	DB21,DBB88-91	DB250x.DBD3024
erweiterte Adresse M-Funktion 5 (16 Bit-Int)	DB21,DBB92-93	DB250x.DBB3036
M-Funktion 5 (DInt)	DB21,DBB94-97	DB250x.DBD3032
erweiterte Adresse S-Funktion 1 (16 Bit-Int)	DB21,DBB98-99	DB250x.DBB4004
S-Funktion 1 (Real)	DB21,DBB100-103	DB250x.DBD4000
erweiterte Adresse S-Funktion 2 (16 Bit-Int)	DB21,DBB104-105	DB250x.DBB4012
S-Funktion 2 (Real)	DB21,DBB106-109	DB250x.DBD4008
erweiterte Adresse S-Funktion 3 (16 Bit-Int)	DB21,DBB110-111	DB250x.DBB4020
S-Funktion 3 (Real)	DB21,DBB112-115	DB250x.DBD4016
erweiterte Adresse T-Funktion 1 (16 Bit-Int)	DB21,DBB116-117	-
T-Funktion 1 (Int)	DB21,DBB118-119	DB250x.DBD2000
erweiterte Adresse T-Funktion 2 (16 Bit-Int)	DB21,DBB120-121	-
T-Funktion 2 (Int)	DB21,DBB122-123	-
erweiterte Adresse T-Funktion 3 (16 Bit-Int)	DB21,DBB124-125	-
T-Funktion 3 (Int)	DB21,DBB126-127	-
erweiterte Adresse D-Funktion 1 (8 Bit-Int)	DB21,DBB128	-
D-Funktion 1 (8 Bit-Int)	DB21,DBB129	DB250x.DBD5000
erweiterte Adresse D-Funktion 2 (8 Bit-Int)	DB21,DBB130	-
D-Funktion 2 (8 Bit-Int)	DB21,DBB131	-
erweiterte Adresse D-Funktion 3 (8 Bit-Int)	DB21,DBB132	-
D-Funktion 3 (8 Bit-Int)	DB21,DBB133	-
erweiterte Adresse DL-Funktion (8 Bit-Int)	DB21,DBB134	-

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
DL-Funktion (Real)	DB21,DBB136	-
erweiterte Adresse H-Funktion 1 (16 Bit-Int)	DB21,DBB140-141	DB250x.DBB6004
H-Funktion 1 (Real oder DInt)	DB21,DBB142-145	DB250x.DBD6000
erweiterte Adresse H-Funktion 2 (16 Bit-Int)	DB21,DBB146-147	DB250x.DBB6012
H-Funktion 2 (REAL oder DInt)	DB21,DBB148-151	DB250x.DBD6008
erweiterte Adresse H-Funktion 3 (16 Bit-Int)	DB21,DBB152-153	DB250x.DBB6020
H-Funktion 3 (Real oder DInt)	DB21,DBB154-157	DB250x.DBD6016
erweiterte Adresse F-Funktion 1 (16 Bit-Int)	DB21,DBB158-159	-
F-Funktion 1 (Real)	DB21,DBB160-163	-
erweiterte Adresse F-Funktion 2 (16 Bit-Int)	DB21,DBB164-165	-
F-Funktion 2 (Real)	DB21,DBB166-169	-
erweiterte Adresse F-Funktion 3 (16 Bit-Int)	DB21,DBB170-171	-
F-Funktion 3 (Real)	DB21,DBB172-175	-
erweiterte Adresse F-Funktion 4 (16 Bit-Int)	DB21,DBB176-177	-
F-Funktion 4 (Real)	DB21,DBB178-181	-
erweiterte Adresse F-Funktion 5 (16 Bit-Int)	DB21,DBB182-183	-
F-Funktion 5 (Real)	DB21,DBB184-187	-
erweiterte Adresse F-Funktion 6 (16 Bit-Int)	DB21,DBB188-189	-
F-Funktion 6 (Real)	DB21,DBB190-193	-
Dynamische M-Funktion: M00 - M07	DB21,DBB194	DB250x.DBB1000
Dynamische M-Funktion: M08 - M15	DB21,DBB195	DB250x.DBB1001
Dynamische M-Funktion: M16 - M23	DB21,DBB196	DB250x.DBB1002
Dynamische M-Funktion: M24 - M31	DB21,DBB197	DB250x.DBB1003
Dynamische M-Funktion: M32 - M39	DB21,DBB198	DB250x.DBB1004
Dynamische M-Funktion: M40 - M47	DB21,DBB199	DB250x.DBB1005
Dynamische M-Funktion: M48 - M55	DB21,DBB200	DB250x.DBB1006
Dynamische M-Funktion: M56 - M63	DB21,DBB201	DB250x.DBB1007
Dynamische M-Funktion: M64 - M71	DB21,DBB202	DB250x.DBB1008
Dynamische M-Funktion: M72 - M79	DB21,DBB203	DB250x.DBB1009
Dynamische M-Funktion: M80 - M87	DB21,DBB204	DB250x.DBB1010
Dynamische M-Funktion: M88 - M95	DB21,DBB205	DB250x.DBB1011
Dynamische M-Funktion: M96 - M99	DB21,DBX206.0-3	DB250x.DBB1012.0-3
Assoziiertes M00/M01 aktiv	DB21,DBX318.5	DB330x.DBX4002.5

9.18.2.3 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
F-Funktion für Positionierachse (REAL)	DB31,DBB78-81	-

9.18.2.4 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
M-Funktion für Spindel (Int)	DB21,DBB86-87	DB370x.DBD0000
S-Funktion für Spindel (Real)	DB21,DBB88-91	DB370x.DBD0004

10.1 Kurzbeschreibung

Kanal

Ein Kanal der NC stellt die kleinste Einheit für das manuelle Verfahren von Achsen und die automatische Abarbeitung von Teileprogrammen dar. Ein Kanal befindet sich zu einem Zeitpunkt immer in einer bestimmten Betriebsart, z. B. AUTOMATIK, MDA oder JOG. Ein Kanal kann als eigenständige NC betrachtet werden.

Betriebsartengruppe (BAG)

Ein Kanal gehört immer einer Betriebsartengruppe an. Eine Betriebsartengruppe kann auch aus mehreren Kanälen bestehen.

Das Kennzeichen einer Betriebsartengruppe ist, dass sich alle Kanäle der Betriebsartengruppe zu einem Zeitpunkt immer in der gleichen Betriebsart, z. B. AUTOMATIK, MDA oder JOG, befinden. Dies wird durch die NC-interne Betriebsartenlogik sichergestellt.

Eine Betriebsartengruppe kann als eigenständige mehrkanalige NC betrachtet werden.

Kanallücken

Bei der Konfiguration der Kanäle können Platzhalter-Kanäle vorgesehen werden, um die Konfiguration über Maschinen einer Serie möglichst einheitlich anzulegen und nur die tatsächlich benutzten Kanäle zu aktivieren.

Programmtest

Zum Testen oder Einfahren eines neuen Teileprogramms gibt es folgende Möglichkeiten:

- Programmbearbeitung ohne Sollwertausgaben
- Programmbearbeitung im Einzelsatzbetrieb
- Programmbearbeitung mit Probelaufvorschub
- Teileprogrammsätze ausblenden
- Satzsuchlauf mit oder ohne Berechnung

Satzsuchlauf

Über Satzsuchlauf gibt es folgende Programmsimulationen zum Suchen bestimmter Programmstellen:

- Typ 1 ohne Berechnung an Kontur
- Typ 2 mit Berechnung an Kontur

10.1 Kurzbeschreibung

- Typ 4 mit Berechnung an Satzende
- Typ 5 Selbsttätiger Start der angewählten Programmstelle mit Berechnung aller erforderlichen Daten aus der Vorgeschichte
- Automatischer Start eines ASUPs nach Satzsuchlauf
- Kaskadierter Satzsuchlauf
- Kanalübergreifender Satzsuchlauf im Modus "Programmtest"

Programmbetrieb

Programmbetrieb liegt dann vor, wenn in der Betriebsart AUTOMATIK oder MDA Teileprogramme bzw. Teileprogrammsätze abgearbeitet werden. Während der Abarbeitung kann dabei der Programmablauf durch PLC-Nahtstellensignale und Kommandos beeinflusst werden.

Für jeden Kanal können Grundstellungen über kanalspezifische Maschinendaten vorgegeben werden. Diese Grundstellungen wirken u. a. auf die G-Gruppen sowie auf die Hilfsfunktionsausgabe.

Die Anwahl eines Teileprogramms kann nur erfolgen, wenn sich der betreffende Kanal im Reset-Zustand befindet.

Außerdem werden alle weiteren Programmabläufe mit PLC-Nahtstellensignalen und den entsprechenden Kommandos behandelt.

- Starten des Teileprogramms bzw. des Teileprogrammsatzes
- Teileprogrammberechnung und Programmbeeinflussung
- RESET-Kommando, Programmzustand und Kanalzustand
- Reaktionen auf Bedienungs- und Programmaktionen
- Ereignisgesteuerte Programmaufrufe

Asynchrone Unterprogramme (ASUP), Interruptroutinen

Mit Hilfe von Interrupteingängen ist die NC in der Lage, die aktuelle Teileprogrammbearbeitung zu unterbrechen und auf hochpriorisierte Ereignisse in Interruptroutinen/ASUPs zu reagieren.

Einzelsatz

In der Funktion Einzelsatz kann der Anwender das Teileprogramm satzweise abarbeiten.

Es gibt 3 Einstellungsarten der Funktion Einzelsatz:

- SLB1: = IPO-Einzelsatz
- SLB2: = Decodiereinzelsatz
- SLB3: = Stopp im Zyklus

Basis-Satzanzeige

Zur bestehenden Satzanzeige können über eine zweite, der Basis-Satzanzeige, alle Sätze angezeigt werden, die eine Aktion an der Maschine bewirken werden.

Die tatsächlich angefahrenen Endpositionen werden als Absolutposition dargestellt. Die Positionswerte beziehen sich wahlweise auf das Werkstückkoordinatensystem (WKS) oder auf das Einstellbare Nullpunkt-System (ENS).

Programm von Extern abarbeiten

Bei der Bearbeitung komplexer Werkstücke kommt es vor, dass der Speicher der NC für die Programme nicht ausreicht. Mit der Funktion "Abarbeiten von Extern" oder "EES (Execution from External Storage)" können Teileprogramme von einem externen Speicher (z. B. von der Festplatte) aufgerufen und abgearbeitet werden.

Verhalten nach POWER ON, Reset, ...

Das Verhalten der Steuerung nach:

- Hochlauf (POWER ON)
- Reset/Teileprogramm-Ende
- Teileprogramm-Start

kann für Funktionen wie z. B. G-Befehle, Werkzeuglängenkorrektur, Transformation, Mitschleppverbände, Tangentiale Nachführung, Programmierbare Synchronspindel, für bestimmte Systemeinstellungen durch Maschinendaten verändert werden.

Unterprogrammaufruf mit M-, T- bzw. D-Funktionen

Für bestimmte Anwendungen kann es vorteilhaft sein, M- T- bzw. D-Funktionen sowie einige NC-Sprachbefehle (*SPOS*, *SPOSA*) durch einen Unterprogrammaufruf zu ersetzen. Dies kann z. B. für den Aufruf der Werkzeugwechselroutine genutzt werden.

Mit entsprechenden Maschinendaten können Unterprogramme mit M- T- bzw. D-Funktionen z. B. für einen Getriebestufenwechsel definiert und beeinflusst werden.

Programmlaufzeit / Werkstückzähler

Zur Unterstützung des Bearbeiters an der Werkzeugmaschine werden Informationen zur Programmlaufzeit und zur Werkstückzählung bereitgestellt.

Die dabei definierte Funktionalität ist **nicht identisch mit Funktionen der Werkzeugverwaltung** und besonders für NC-Systeme ohne Werkzeugverwaltung vorgesehen.

10.2 Betriebsartengruppe (BAG)

Betriebsartengruppe

In einer Betriebsartengruppe (BAG) werden mehrere Kanäle einer NC zu einer Bearbeitungseinheit zusammengefasst. Sie stellt somit im Prinzip eine eigenständige "NC" innerhalb einer NC dar.

Eine BAG ist im Wesentlichen dadurch gekennzeichnet, dass sich alle ihr zugeordneten Kanäle zu einem Zeitpunkt immer in der gleichen Betriebsart (AUTOMATIK, JOG, MDA) befinden.

Hinweis

Im weiteren Verlauf dieser Beschreibung wird von einer BAG und einem Kanal ausgegangen. Funktionen, die mehrere Kanäle benötigen (z. B. Funktion "Achstausch"), sind beschrieben in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; "K5: Kanalübergreifende Programmkoordinierung" und "K10: Kanalübergreifender Achstausch"

Parametrierung

Die Zuordnung eines Kanals zu einer BAG erfolgt über das Maschinendatum:

MD10010 \$MN_ASSIGN_CHAN_TO_MODE_GROUP[<Kanal-Index>] = BAG-Nummer

Alle Kanäle mit der gleichen BAG-Nummer bilden eine BAG.

Hinweis

Eine NC umfasst standardmäßig immer eine BAG mit einem Kanal. Die Parametrierung einer NC ohne Kanal ist nicht möglich.

BAG-spezifische NC/PLC-Nahtstellensignale

Die BAG-spezifischen NC/PLC-Nahtstellensignale liegen im Datenbaustein DB11.

Die BAG-spezifische NC/PLC-Nahtstelle umfasst im Wesentlichen folgende Nahtstellensignale:

- Anforderungssignale PLC → NC
 - BAG-Reset
 - BAG-Stop Achsen plus Spindel
 - BAG-Stop
 - Betriebsartenwechsel Sperre
 - Betriebsart: JOG, MDA, AUTOM.
 - Einzelsatz: Typ A, Typ B
 - Maschinenfunktion: REF, REPOS, TEACH IN, INC x
- Statussignale NC → PLC
 - Betriebsart-Strobe: JOG, MDA, AUTOMATIK
 - Maschinenfunktion-Strobe: REF, REPOS, TEACH IN
 - Alle Kanäle im Reset-Zusand
 - BAG reseted
 - BAG betriebsbereit
 - aktive Betriebsart: JOG, MDA, AUTOMATIK
 - aktive Maschinenfunktion: REF, REPOS, TEACH, INC x

Aktivierte und nicht aktivierte Kanäle, Kanallücken

Aktivierter Kanal

Ein Kanal mit einer BAG-Nummer $\neq 0$ ist ein aktiver Kanal.

Nicht aktivierter Kanal

Ein Kanal mit BAG-Nummer 0 ist ein nicht aktivierter Kanal. Er belegt Steuerungs-intern keinen Speicherplatz.

Kanallücken

Ein Kanal mit BAG-Nummer 0 ist nicht nur ein nicht aktivierter Kanal. Er stellt in der Abfolge der Kanäle auch eine sogenannte Kanallücke dar.

Der Vorteil von Kanallücken ist, dass damit innerhalb einer Serie von bauähnlichen Maschinen die Konfigurationsdaten weitgehend gleich gehalten werden können. Für die spezifische Inbetriebnahme werden dann nur die Kanäle aktiviert, die für die vorliegende Maschine

benötigt werden. Der nicht belegte Speicher steht dann als zusätzlicher Anwenderspeicher zur freien Verfügung.

Tabelle 10-1 Beispiel

Maschinendatum	Bedeutung
MD10010 \$MN_ASSIGN_CHAN_TO_MODE_GROUP[0] = 1	Kanal 1, BAG 1
MD10010 \$MN_ASSIGN_CHAN_TO_MODE_GROUP[1] = 2	Kanal 2, BAG 2
MD10010 \$MN_ASSIGN_CHAN_TO_MODE_GROUP[2] = 0	Kanal 3, nicht vorhanden
MD10010 \$MN_ASSIGN_CHAN_TO_MODE_GROUP[3] = 1	Kanal 4, BAG 1
MD10010 \$MN_ASSIGN_CHAN_TO_MODE_GROUP[4] = 2	Kanal 5, BAG 2

Siehe auch

K2: Achsen, Koordinatensysteme, Frames (Seite 713)

S1: Spindeln (Seite 1295)

10.2.1 BAG-Stop

Funktion

Über die folgenden NC/PLC-Nahtstellensignale werden in allen Kanälen der BAG die Verfahrbewegungen der Achsen bzw. Achsen und Spindeln angehalten, sowie die Teileprogrammabarbeitung unterbrochen:

DB11 DBX0.5 (BAG-Stop)

DB11 DBX0.6 (BAG-Stop Achsen plus Spindeln)

10.2.2 BAG-Reset

Funktion

Die Anforderung zum BAG-Reset erfolgt über das BAG-spezifische NC/PLC-Nahtstellensignal:

DB11 DBX0.7 = 1 (BAG-Reset)

Wirkung

Wirkung auf die Kanäle der BAG:

- Die Teileprogrammaufbereitung (Vorlauf) wird gestoppt.
- Alle Achsen und Spindeln werden entlang ihrer Beschleunigungskennlinien ohne Konturverletzung auf Stillstand abgebremst.
- Die noch nicht an die PLC ausgegebenen Hilfsfunktionen werden nicht mehr ausgegeben.
- Die Vorlaufzeiger werden auf die Unterbrechungsstelle und die Satzzeiger auf den Anfang der jeweiligen Teileprogramme gesetzt.
- Alle Grundstellungen (z. B. der G-Befehle) werden auf die parametrisierten Werte eingestellt.
- Alle Alarme mit Löschkriterium "Kanal-Reset" werden gelöscht.

Wenn alle Kanäle der BAG im Reset-Zustand sind, dann:

- werden alle Alarme mit Löschkriterium "BAG-Reset" gelöscht.
- werden der Abschluss des BAG-Resets und die Betriebsbereitschaft der BAG an der NC/PLC-Nahtstelle angezeigt:
DB11 DBX6.7 = 1 (alle Kanäle im Reset-Zustand)
DB11 DBX6.3 = 1 (BAG betriebsbereit)

10.3 Betriebsarten und Betriebsartenwechsel

Eindeutige Betriebsart

Alle Kanäle einer Betriebsartengruppe (BAG) befinden sich immer in derselben Betriebsart:

- AUTOMATIK
- JOG
- MDA

Sind einzelne Kanäle unterschiedlichen Betriebsartengruppen zugeordnet, so erfolgt über eine Kanalumschaltung auch eine Umschaltung auf die entsprechende BAG. Auf diese Weise ist über eine Kanalumschaltung ein Betriebsartenwechsel möglich.

Betriebsarten

Es stehen folgende Betriebsarten zur Verfügung:

- **AUTOMATIK**
Automatisches Abarbeiten von Teileprogrammen:
 - Teileprogrammtest
 - Alle Kanäle der BAG können gleichzeitig aktiv sein.
- **JOG in AUTOMATIK**
JOG in AUTOMATIK ist eine Erweiterung der Betriebsart AUTOMATIK mit dem Ziel der Bedienungsvereinfachung. JOG kann ausgeführt werden, ohne AUTOMATIK zu verlassen, wenn Randbedingungen dies zulassen.

- **JOG**
Manuelles Verfahren von Achsen über Verfahrtasten der Maschinensteuertafel oder über ein an der Maschinensteuertafel angeschlossenes Handrad:
 - Kanalspezifische Signale und Verriegelungen werden bei Bewegungen, die per ASUP oder über statische Synchronaktionen ausgeführt werden, beachtet.
 - Kopplungen werden berücksichtigt.
 - Jeder Kanal in der BAG kann aktiv sein.
- **MDA**
Manual Data Automatic (Eingabe der Sätze erfolgt über die Bedienoberfläche):
 - Eingeschränktes Abarbeiten von Teileprogrammen und Teileprogrammteilen.
 - Teileprogrammtest
 - Maximal 1 Kanal pro BAG kann aktiv sein (nur bei TEACH IN).
 - Achsen können nur in unterlagerten Maschinenfunktionen wie JOG, REPOS oder TEACHIN manuell verfahren werden.

Gültig für alle Betriebsarten

Betriebsarten-übergreifende Synchronaktionen

Übergreifend für alle Betriebsarten, können modale Synchronaktionen per IDS für folgende Funktionen parallel zum Kanal abgearbeitet werden:

- Kommandoachsfunktionen
- Spindelfunktionen
- Technologiezyklen

Anwahl

Über die Bedienoberfläche kann der Anwender die gewünschte Betriebsart mit Hilfe von Softkeys anwählen.

Diese Anwahl (AUTOMATIK, MDA oder JOG) wird an die NC/PLC-Nahtstelle zur PLC weitergeleitet, aber noch nicht aktiviert:

DB11 DBX4.0, 0.1, 0.2 (Betriebsart-Strobe)

Aktivierung und Prioritäten

Die Betriebsart der BAG wird aktiviert über die NC/PLC-Nahtstelle:

DB11 DBX0.0, 0.1, 0.2 (Betriebsart)

Werden mehrere Betriebsarten zur gleichen Zeit angewählt, besteht folgende Priorität:

Priorität	Betriebsart	BAG-Signal (NC → PLC)
1. Priorität, hoch	JOG	DB11 DBX0.2
2. Priorität, mittel	MDA	DB11 DBX0.1
3. Priorität, niedrig	AUTOMATIK	DB11 DBX0.0

Anzeige

Die aktuelle Betriebsart der BAG wird angezeigt über die NC/PLC-Nahtstelle:

DB11 DBX6.0, 0.1, 0.2 (aktive Betriebsart)

BAG-Signal (NC → PLC)	Aktive Betriebsart
DB11 DBX6.2	JOG
DB11 DBX6.1	MDA
DB11 DBX6.0	AUTOMATIK

Maschinenfunktionen

Innerhalb einer Betriebsart können Maschinenfunktionen angewählt werden, die ebenfalls innerhalb der BAG gelten:

- Maschinenfunktionen innerhalb der Betriebsart **JOG**
 - REF (Referenzpunktfahren)
 - REPOS (Repositionieren)
 - JOG-Retract (Rückzugsbewegung in Werkzeugrichtung)
- Maschinenfunktionen innerhalb der Betriebsart **MDA**
 - REF (Referenzpunktfahren)
 - REPOS (Repositionieren)
 - TEACHIN (Ein-Teachen von Achspositionen)

NC/PLC-Nahtstellensignale

- DB11 DBX5.0, 0.1, 0.2 (Maschinenfunktion-Strobe): Anforderung
- DB11 DBX1.0, 0.1, 0.2 (Maschinenfunktion): Aktivierung
- DB11 DBX7.0-2 (aktive Maschinenfunktion): Rückmeldung

Kanal-Zustände

- **Kanal-Reset**
Die Maschine befindet sich im Grundzustand. Dieser wird über das PLC-Programm vom Maschinenhersteller z. B. nach dem Einschalten oder nach Programmende definiert.
- **Kanal aktiv**
Ein Programm ist gestartet, die Programmabarbeitung oder Referenzpunktfahren läuft.
- **Kanal unterbrochen**
Das laufende Programm oder Referenzpunktfahren sind unterbrochen worden.

Funktionen innerhalb von Betriebsarten

Die Betriebsarten werden ergänzt durch anwenderspezifische Funktionen. Die Funktionen sind technologie- und maschinenunabhängig und können aus den drei Kanalzuständen "Kanal-Reset", "Kanal aktiv" oder "Kanal unterbrochen" gestartet und/oder bearbeitet werden.

Randbedingung für die Maschinenfunktion TEACH IN

TEACH IN ist nicht zulässig für Führungs- oder Leitachsen eines aktiven Achsverbundes, z. B. bei:

- Gantry-Achsverbund oder ein Gantry-Achsenpaar
- Mitschleppverbund von Leitachse mit Folgeachse

JOG in AUTOMATIK

JOG in Betriebsart AUTOMATIK ist zulässig, wenn die BAG im Zustand "RESET" und die Achse JOG-fähig ist. Zustand "RESET" für die BAG bedeutet:

- Alle Kanäle sind im Zustand "RESET"
- Alle Programme sind abgebrochen
- In keinem Kanal ist DRF aktiv

Eine Achse ist **JOG-fähig** wenn sie in keinem der folgenden Zustände ist:

- PLC-Achse als konkurrierende Positionierachse (Anforderung der Achse von der PLC)
- Kommandoachse (die Achse wurde von einer Synchronaktion programmiert und die Bewegung ist noch nicht zu Ende)
- Drehende Spindel (Spindel die über RESET hinaus dreht)
- Asynchrone Pendelachse

Hinweis: Die Eigenschaft "JOG-fähig" ist unabhängig von der Funktion "JOG in AUTOMATIK".

Aktivierung

Die Funktion "JOG in AUTOMATIK" ist aktivierbar mit dem Maschinendatum:

MD10735 \$MN_JOG_MODE_MASK

- Vor POWER ON muss folgendes Maschinendatum gesetzt sein:
MD10735 \$MN_JOG_MODE_MASK, Bit 0 = 1
- Der Anwender schaltet nach AUTO (PLC-Anwenderschnittstelle DB11 DBX0.0 = 0→1 Flanke). "JOG in AUTOMATIK" ist damit aktiv, wenn vorher der NC in allen Kanälen der BAG den Kanalzustand "RESET" und den Programmzustand "Abgebrochen" hatte. Zudem muss die dann betroffene Achse "JOG-fähig" sein. DRF muss abgeschaltet sein oder werden.
- In allen Kanälen der BAG ohne den Kanalzustand "RESET" und den Programmzustand "Abgebrochen" wird RESET ausgelöst oder das laufende Programm kommt mit M30/M2 zu Ende.
- Die betroffene Achse wird automatisch "JOG-fähig" gemacht (z. B. Achstauch: PLC → NC).

Hinweis: In den meisten Anwendungsfällen sind die zu verfahrenenden Achsen "JOG-fähig" und mit dem Umschalten nach AUTOMATIK ist damit auch "JOG in AUTOMATIK" aktiv.

Merkmale

- Die Tasten +/- leiten eine JOG-Bewegung ein, die BAG wird **intern** nach JOG geschaltet. (Kurz "Intern-JOG").
- Die Bewegungen der Handräder leiten eine JOG-Bewegung ein, die BAG wird **intern** nach JOG geschaltet, außer es ist DRF aktiv.
- Eine begonnene JOG-Bewegung ist erst zu Ende, wenn die Endposition des Inkrements (falls dies eingestellt worden war) erreicht wurde, oder die Bewegung mit "Restweglöschen" abgebrochen wurde.
Damit kann ein Inkrement mit Stopp angehalten und mit Start wieder bis zum Ende gefahren werden. Solange bleibt der NC im "Intern-JOG". Ein teilweise abgefahrenes Inkrement ist möglich, muss aber nicht mit Stopp unterbrochen sein. Es gibt einen Modus, bei dem eine Wegnahme der Fahrtaste zur Unterbrechung innerhalb eines Inkrements führt.
- Ohne eine JOG-Bewegung verhält sich "JOG in AUTOMATIK" wie "Automatik". Insbesondere startet die Start-Taste das angewählte Teileprogramm und der entsprechende Softkey des HMI löst einen Satzsuchlauf aus.
- Mit einer aktiven JOG-Bewegung ist der NC intern in JOG und eine Satzsuchlaufanforderung wird abgelehnt und der Start startet das Teileprogramm nicht. Der Start startet ggf. das verbliebene Inkrement oder er bleibt ohne Wirkung.
- Solange eine Achse der BAG in JOG verfahren wird, bleibt die BAG intern in JOG.
Bemerkung: Diese Phase kann mit der JOG-Bewegung einer Achse beginnen und mit dem Ende der JOG-Bewegung einer *anderen* Achse enden.
- Eine Achse mit aktiver JOG-Bewegung darf keinem Achstausch unterworfen werden. (Die Achse könnte die BAG wechseln). Der NC verriegelt diesen Achstausch-Versuch.
- Die PLC-Anwenderschnittstelle zeigt die Betriebsart "Automatik" an:
 - DB11 DBX6.0, 6.1, 6.2 = 1
 - DB11 DBX7.0, 7.1, 7.2 = 0
- Die NC/PLC-Nahtstelle zeigt an, wenn in "JOG in AUTOMATIK" sich die BAG in "BAG–RESET" befindet.
 - DB11 DBX6.4 (BAG reseted, BAG 1)
 - DB11 DBX26.4 (BAG reseted, BAG 2)
 - DB11 DBX46.4 (BAG reseted, BAG 3)
- Die NC/PLC-Nahtstelle zeigt an, wenn in "JOG in AUTOMATIK" die NC intern nach "Intern-JOG" geschaltet hat.
 - DB11 DBX6.5 (NC intern JOG aktiv, BAG 1)
 - DB11 DBX26.5 (NC intern JOG aktiv, BAG 2)
 - DB11 DBX46.5 (NC intern JOG aktiv, BAG 3)

Randbedingungen

"JOG in AUTOMATIK" kann intern nur dann nach JOG wechseln, wenn die BAG im Zustand "BAG–RESET" ist. D. h. mitten in einem gestoppten Programm kann nicht unmittelbar gejoggt werden. Der Benutzer kann joggen, wenn er in dieser Situation die JOG-Taste oder die Reset-Taste *in allen Kanälen* der BAG drückt.

Mit der Anwahl von Automatik werden die INC-Tasten abgeschaltet und der Anwender kann/muss die INC-Tasten erneut drücken, um ein von ihm gewünschtes Inkrement anzuwählen. Schaltet der NC nach "Intern-JOG", so bleibt das angewählte Inkrement erhalten.

Versucht der Anwender die Geo- oder Orientierungsachsen zu joggen, so wechselt der NC auch nach "Intern-JOG" und die Bewegung wird ausgeführt. Dabei können mehrere Achsen physisch bewegt werden, die alle "JOG-fähig" sein müssen.

Nach der JOG-Bewegung schaltet der NC "Intern-JOG" wieder ab und wählt damit AUTO wieder an. Der Interne Modewechsel wird verzögert nach dem Bewegungsende durchgeführt. Damit werden unnötig viele Schaltvorgänge vermieden, die z. B. beim Handrad auftreten könnten. Die PLC darf sich nur auf das PLC-Signal "Intern-JOG-Aktiv" verlassen.

Der NC schaltet auch dann nach "Intern-JOG", wenn die Achse keine Freigabe hat.

Siehe auch

R1: Referenzieren (Seite 1243)

10.3.1 Überwachungen und Verriegelungen der einzelnen Betriebsarten

Kanalzustand bestimmt Überwachungen

Überwachungen in den Betriebsarten

In den einzelnen Betriebsarten sind unterschiedliche Überwachungen aktiv. Diese Überwachungen sind Technologie- und Maschinen-unabhängig.

In jeder Betriebsart, in Abhängigkeit des Betriebszustandes, ist ein Teil der Überwachungen aktiv. Welche Überwachungen in welcher Betriebsart und in welchem Betriebszustand aktiv sind, wird durch den Kanalzustand bestimmt.

Verriegelungen in den Betriebsarten

In den einzelnen Betriebsarten können unterschiedliche Verriegelungen aktiv sein. Diese Verriegelungen sind Technologie- und Maschinen-unabhängig.

In jeder Betriebsart, in Abhängigkeit des Betriebszustandes, können fast alle Verriegelungen aktiviert werden.

10.3.2 Betriebsartenwechsel

Einführung

Ein Betriebsartenwechsel wird über die BAG-Nahtstelle (DB11) angefordert und aktiviert. Eine Betriebsartengruppe befindet sich entweder in der Betriebsart AUTOMATIK, JOG oder MDA, d. h. mehrere Kanäle einer Betriebsartengruppe können gleichzeitig keine unterschiedlichen Betriebsarten annehmen.

Ob die angeforderte Betriebsart erreichbar ist und wie diese durchgeführt wird, ist maschinenspezifisch über das PLC-Programm projektierbar.

Hinweis

Die Betriebsart wird erst dann steuerungsintern gewechselt, wenn der "Kanalzustand aktiv" nicht mehr vorliegt. Zum fehlerfreien Betriebsartenwechsel müssen aber alle Kanäle einen zulässigen Betriebszustand eingenommen haben.

Mögliche Betriebsartenwechsel

Die möglichen Betriebsartenwechsel können Sie der folgenden Tabelle für einen Kanal entnehmen:

	AUTOMATIK		JOG			MDA			
				AUTO	MDA		manuelles Verfahren		AUTO
	Reset	unterbr.	Reset	unterbr.	unterbr.	Reset	unterbr.	aktiv	unterbr.
AUTOMATIK			X	X		X			
JOG	X	X				X	X		X
MDA	X	X	X		X				

Die mit "X" gekennzeichneten Positionen sind mögliche Betriebsartenwechsel.

Sonderfälle

- **Fehler bei Betriebsartwechsel**
Wenn eine Betriebsartenwechsel-Anforderung vom System abgewiesen wurde, erfolgt die Fehlermeldung "BA-Wechsel erst nach NC-Stop möglich". Diese Fehlermeldung kann gelöscht werden, ohne den Kanalzustand zu ändern.
- **Betriebsartenwechsel-Sperre**
Mit Hilfe des Nahtstellensignals:
DB11 DBX0.4 (Betriebsartenwechsel-Sperre)
kann ein Wechseln der Betriebsart verhindert werden.
Es wird dabei schon die Betriebsartenwechsel-Anforderung unterdrückt.
Der Anwender muss eine Meldung über die aktive Sperre für den Bediener selber projektieren. Vom System wird dafür nichts abgesetzt.
- **Betriebsartenwechsel von MDA nach JOG**
Sind nach einem Betriebsartenwechsel von MDA nach JOG alle Kanäle der BAG im Reset-Zustand, so wechselt die NC von JOG nach AUTO. In diesem Zustand können die Teileprogrammbefehle `START` oder `INIT` durchgeführt werden.
Ist nach dem Betriebsartenwechsel ein Kanal der BAG nicht mehr im Reset-Zustand, so wird der Teileprogrammbefehl `START` in dieser Situation mit dem Alarm 16952 abgelehnt.

10.4 Kanal

In einem Kanal der NC werden die vom Anwender vorgegebenen Teileprogramme abgearbeitet.

Folgende Eigenschaften charakterisieren einen Kanal:

- Ein Kanal ist immer einer Betriebsartengruppe (Seite 480) (BAG) zugeordnet.
- Zu einem Zeitpunkt, kann ein Kanal immer nur genau ein Teileprogramm abarbeiten.
- Einem Kanal sind Maschinen-, Geometrie- und Zusatzachsen sowie Spindeln zugeordnet. Nur diese können über die im Kanal abgearbeiteten Teileprogramm verfahren werden.
- Ein Kanal besteht aus den internen Einheiten
 - Vorlauf (Programmdekodierung und Satzaufbereitung)
 - Hauptlauf (Bahn- und Achsinterpolation).
- Ein Kanal hat eine Schnittstelle zur PLC. Über diese NC/PLC-Nahtstelle können vom PLC-Anwenderprogramm aus diverse kanalspezifische Zustandsdaten gelesen und Anforderung an den Kanal geschrieben werden.
- In einem Kanal wirken kanalspezifische Werkzeugkorrektur (Seite 1475).
- In einem Kanal wirken kanalspezifische Koordinatensysteme (Seite 734).
- Jeder Kanal hat standartmäßig einen eindeutigen Kanalnamen. Der vorgegebene Kanalname kann über Maschinendaten verändert werden:
`MD20000 $MC_CHAN_NAME = "<Kanalname>"`

Mehrere Kanäle können zu einer Betriebsartengruppe (BAG) zusammengefasst werden. Die Kanäle einer BAG befinden sich immer in der gleichen Betriebsart (AUTOMATIK, JOG, MDA).

Kanalkonfiguration

Kanäle können über folgendes Maschinendatum mit einem eigenen Kanalnamen belegt werden:

MD20000 \$MC_CHAN_NAME (Kanalname)

Die verschiedenen Achsen werden per Maschinendatum den vorhandenen Kanälen zugeordnet. Für eine Achse/Spindel kann es zu einer Zeit immer nur einen sollwertgebenden Kanal geben. Der Istwert der Achse/Spindel kann von mehreren Kanälen gleichzeitig gelesen werden. Die Achse/Spindel muss dem jeweiligen Kanal bekannt sein.

Des Weiteren können noch folgende kanalspezifische Einstellungen über Maschinendaten definiert werden:

- Löschstellungen bzw. Programmiergrundstellungen von G-Gruppen über das Maschinendatum:
MD20150 \$MC_GCODE_RESET_VALUES (Löschstellung der G-Gruppen)
- Hilfsfunktionsgruppen bezüglich der Zusammenstellung und des Ausgabezeitpunkts
- Transformationsbedingungen zwischen den Maschinenachsen und den Geometrieachsen
- Weitere Einstellungen für die Abarbeitung eines Teileprogramms

Änderung der Kanalzuordnung

Eine Änderung der Kanalkonfiguration kann online nicht auf programmtechnischem Wege in einem Teileprogramm oder über das PLC-Anwenderprogramm erfolgen. Änderungen in der Konfiguration müssen über die Maschinendaten erfolgen. Die Änderungen sind erst nach einem erneuten POWER ON wirksam.

Containerachsen und Linkachsen

Ein Achscontainer fasst eine Gruppe von Achsen eines Containers zusammen. Diese Achsen werden als Containerachsen bezeichnet. Hierbei wird einer Kanalachse ein Zeiger auf einem Containerslot (Ringpuffer-Platz innerhalb des entsprechenden Containers) zugeordnet. In diesem Slot befindet sich zwischenzeitlich eine der in diesem Container zusammengefassten Achsen.

Maschinenachsen im Achscontainer müssen zu jedem Zeitpunkt jeweils genau **einer** Kanalachse zugewiesen sein.

Link-Achsen können fest einem Kanal oder dynamischen (Achscontainer-Switch) mehreren Kanälen der lokalen, oder der anderen NCU zugeordnet werden. Sie sind aus Sicht einer der Kanäle, an deren NCU die Achse nicht physikalisch angeschlossen ist, eine nicht lokale Achse.

Die herbeigeführte Zuordnung der Linkachsen zu einem Kanal erfolgt:

- Für feste Zuordnung über Maschinendatum:
Direktes logisches Maschinenachsbild auf Linkachsen zeigen lassen.
- Für dynamische Zuordnung:
Achscontainerslot - Maschinendatum auf Linkachsen zeigen lassen.

Weitere Informationen zu Link- und Containerachsen finden Sie in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Mehrere Bedientafeln an mehreren NCUs, Dezentrale Systeme (B3)

Nahtstellensignale

Die Signale des 1. Kanals liegen in der NC/PLC-Nahtstelle im DB21, die vom Kanal 2 im DB22. Der oder die Kanäle können von der PLC bzw. NC überwacht bzw. beeinflusst werden.

Kanalspezifische Technologie-Angabe

Für jeden Kanal kann die verwendete Technologie angegeben werden:

MD27800 \$MC_TECHNOLOGY_MODE

Im Auslieferungszustand sind standardmäßig Maschinendaten für die Technologie Fräsen aktiv.

Spindelfunktionen über PLC

Spindelfunktionen können neben dem Funktionsbaustein FC18 auch über axiale NC/PLC-Nahtstellensignale parallel zu laufenden Teileprogrammen gestartet und gestoppt werden.

Voraussetzungen:

- Kanalzustand: "Unterbrochen" oder "RESET"
- Programmzustand: "Unterbrochen" oder "Abgebrochen"

Folgende Funktionen können von der PLC über Nahtstellensignale gesteuert werden:

- Stopp (entspricht M5)
- Start mit Drehrichtung rechts (entspricht M3)
- Start mit Drehrichtung links (entspricht M4)
- Getriebestufe auswählen
- Positionieren (entspricht M19)

Bei mehreren Kanälen wird die von der PLC gestartete Spindel in dem Kanal aktiv, dem sie zum Startzeitpunkt zugeordnet ist.

Weitere Informationen zur speziellen Spindelschnittstelle siehe Kapitel "S1: Spindeln (Seite 1295)".

PLC-kontrollierte Einzelachsvorgänge

Statt von einem Kanal, kann eine Achse auch von der PLC kontrolliert werden. Dazu fordert die PLC über die NC/PLC-Nahtstelle die Achse von der NC an:

DB31, ... DBX28.7 = 1 (PLC kontrolliert Achse)

Folgende Funktionen können vom PLC aus kontrolliert und beeinflusst werden:

- Ablauf von Achse/Spindel abbrechen (entspricht Restweg löschen)
- Achse/Spindel stoppen bzw. unterbrechen

- Achse/Spindel weiterfahren (Bewegungsablauf fortsetzen)
- Achse/Spindel in den Grundzustand zurücksetzen

Weitere Informationen zum kanalspezifischen Signalaustausch (PLC → NC) siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)".

Die genaue Funktionalität von autarken Einzelachsvorgängen ist beschrieben in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2)

10.4.1 Startsperr, global und kanalspezifisch

Funktion

In ShopMill / ShopTurn kann der Start eines Programms standardmäßig nur im Bereich Maschine ausgeführt werden. Der Start in den anderen Bereichen, z.B. Werkzeuge, wird über eine Startsperr (PI-Dienst) verriegelt.

Die Startsperr automatisierte Abläufe kann über ein NC/PLC-Nahtstellensignal aufgehoben werden.

NC/PLC-Nahtstellensignale

Startsperr aufheben

Mit dem Nahtstellensignal wird eine anstehende Startsperr aufgehoben:

DB21, ... DBX7.5 (Startsperr aufheben)

Maschinendaten

Mit dem Maschinendatum wird eingestellt, dass ein Alarm angezeigt wird, falls bei gesetzter Startsperr ein Start angefordert wird:

MD11411 \$MN_ENABLE_ALARM_MASK, Bit 6

BTSS-Variablen

Variable	Beschreibung
startLockState	Status der globalen Startsperr
chanStartLockState	Status der kanalspezifischen Startsperr
startRejectCounter	Startzähler bei globaler Startsperr
startLockCounter	Startzähler bei kanalspezifischer Startsperr

PI-Dienste

Setzen der Startsperrung (_N_STRTLK)

Mit dem PI-Dienst _N_STRTLK kann die Startsperrung global bzw. kanalspezifisch gesetzt werden.

- Globale Startsperrung
Ist die globale Startsperrung gesetzt, wird der Start eines NC-Programms in allen Kanälen unterbunden. Ein bereits laufendes NC-Programm bleibt unbeeinflusst. Erst der nächste Start eines NC-Programms wird unterbunden.
- Kanalspezifische Startsperrung
Durch die kanalspezifische Startsperrung wird nur in der Betriebsart AUTOMATIK und im Kanalzustand "Reset" der Start eines NC-Programms unterbunden.

Rücksetzen der globalen bzw. kanalspezifischen Startsperrung (_N_STRTUL)

Mit dem PI-Dienst _N_STRTUL kann die Startsperrung globale bzw. kanalspezifische Startsperrung zurückgesetzt werden.

10.5 Programmtest

Zum Testen bzw. Einfahren eines neuen Teileprogramms gibt es mehrere Steuerungsfunktionen. Durch die Verwendung dieser Funktionen wird eine Gefährdung der Maschine während der Testphase bzw. der Zeitaufwand dafür stark verringert. Es ist möglich, mehrere Programmtestfunktionen gleichzeitig zu aktivieren, um ein besseres Ergebnis zu bekommen.

Testmöglichkeiten

Es werden folgende Testmöglichkeiten beschrieben:

- Programmbearbeitung ohne Sollwertausgaben
- Programmbearbeitung im Einzelsatzbetrieb
- Programmbearbeitung mit Probelaufvorschub
- Teileprogrammsätze ausblenden
- Satzsuchlauf mit oder ohne Berechnung

10.5.1 Programmbearbeitung ohne Sollwertausgaben

Funktion

Im Zustand "Programmtest" wird ein Teileprogramm abgearbeitet, ohne dass Achs- oder Spindel-Sollwerte ausgegeben werden.

Damit kann der Anwender die programmierten Achspositionen sowie die Hilfsfunktionsausgaben eines Teileprogramms kontrollieren. Außerdem kann diese Programmsimulation als erweiterter Syntax-Check verwendet werden.

Anwahl

Die Anwahl der Funktion erfolgt über die Bedienoberfläche im Menü "Programmbeeinflussungen".

Mit der Anwahl wird folgendes Nahtstellensignal gesetzt:

DB21, ... DBX25.7 (Programmtest angewählt)

Die Funktion wird damit noch nicht aktiviert.

Aktivierung

Die Aktivierung der Funktion erfolgt über das Nahtstellensignal:

DB21, ... DBX1.7 (Programmtest aktivieren)

Anzeige

Als Rückmeldung des aktiven Programmtests wird auf der Bedienoberfläche das entsprechende Feld invers geschaltet und in der PLC das folgende Nahtstellensignal gesetzt:

DB21, ... DBX33.7 (Programmtest aktiv)

Programmstart und Programmablauf

Das Teileprogramm kann bei aktiver Programmtest-Funktion gestartet und abgearbeitet (inkl. Hilfsfunktionsausgaben, Wartezeiten, G-Befehlsausgaben etc) werden über das Nahtstellensignal:

DB21, ... DBX7.1 (NC-Start)

Sicherheitsfunktionen wie Software-Endschalter, Arbeitsfeldbegrenzungen sind weiterhin gültig.

Der einzige Unterschied zum normalen Programmablauf besteht darin, dass für alle Achsen (einschließlich Spindeln) intern **Achsensperre** gegeben ist. Die Maschinenachsen bewegen sich also nicht, die Istwerte werden intern aus den nicht ausgegebenen Sollwerten generiert. Die programmierten Geschwindigkeiten bleiben unverändert. Das bedeutet, dass die Positions- und Geschwindigkeitsangaben auf der Bedienoberfläche genau denen einer normalen Teileprogrammbearbeitung entsprechen. Die Lageregelung wird dabei nicht

unterbrochen, so dass nach Abschalten der Funktion die Achsen nicht referenziert werden müssen.

Hinweis

Die Signale für Genauhalt DB31, ... DBX60.6/60.7 (Genauhalt grob/fein) spiegeln den tatsächlichen Zustand an der Maschine wieder.

Während des Programmtests würden sie nur weggenommen, wenn die Achse aus ihrer (während des Programmtests konstanten) Sollposition weggedrückt würde.

Mit dem Signal DB21, ... DBX33.7 (Programmtest aktiv) kann sowohl das PLC-Programm als auch das Teileprogramm mit der Variablen \$P_ISTEST entscheiden, wie während des Tests auf diese Signale zu reagieren oder zu verzweigen ist.

Hinweis

Probelaufvorschub

Die "Programmbearbeitung ohne Achsbewegungen" kann auch zusammen mit der Funktion "Probelaufvorschub" aktiviert werden. Dadurch werden Teileprogrammabschnitte mit einem kleinen programmierten Vorschubwert in einem kürzeren Zeitabschnitt abgearbeitet.

Hinweis

Werkzeugverwaltung

Aufgrund der Achsensperre wird die Belegung eines Werkzeug-Magazins beim Programmtest nicht verändert. Über eine PLC-Applikation muss sichergestellt werden, dass die Konsistenz zwischen den Daten der Werkzeugverwaltung und dem Magazin nicht verloren geht. Auf den Toolbox-Disketten finden Sie dazu beim PLC-Grundprogramm ein Beispiel.

10.5.2 Programmbearbeitung im Einzelsatzbetrieb

Funktion

Bei der "Programmbearbeitung im Einzelsatzbetrieb" stoppt die Teileprogrammbearbeitung nach jedem Programmsatz. Falls eine Fräser- oder Schneiden-Radiuskorrektur angewählt ist, so stoppt die Bearbeitung nach jedem von der Steuerung eingefügten Zwischensatz.

Der Programmzustand wechselt auf "Programmzustand angehalten".

Der Kanalzustand bleibt auf aktiv.

Mit NC-Start gelangt der nächste Teileprogrammsatz zur Abarbeitung.

Anwendung

Der Anwender kann damit ein Teileprogramm Satz für Satz abarbeiten und die einzelnen Bearbeitungsschritte kontrollieren. Wenn er den abgearbeiteten Teileprogrammsatz für korrekt befunden hat, kann er den nächsten Satz anfordern.

Einzelsatztypen

Es wird zwischen folgenden Einzelsatztypen unterschieden:

- **Dekodier-Einzelsatz**
Bei diesem Einzelsatztyp werden alle Sätze des Teileprogramms (auch die reinen Rechensätze ohne Verfahrbewegungen) nacheinander durch "NC-Start" abgearbeitet.
- **Aktions-Einzelsatz (Grundeinstellung)**
Bei diesem Einzelsatztyp werden nur die Sätze einzeln abgearbeitet, die Aktionen (Verfahrbewegungen, Hilfsfunktionsausgaben usw.) auslösen.
Auch die bei der Dekodierung zusätzlich generierten Sätze (z. B. bei Fräser-Radiuskorrektur an spitzen Winkeln) werden einzeln abgefahren.
Bei Rechensätzen wird dagegen die Abarbeitung nicht angehalten, da diese keine Aktionen auslösen.

Die Festlegung des Einzelsatztyps erfolgt über die Bedienoberfläche im Menü "Programmbeeinflussungen".

VORSICHT

Funktionsbesonderheit bei Einzelsatztypen-Serie

Bei einer Serie von G33/G34/G35-Sätzen ist Einzelsatz nur dann wirksam, wenn "Probelaufvorschub" angewählt ist.

Rechensätze werden nicht im Einzelschritt bearbeitet (nur beim Dekodier-Einzelsatz).

SBL2 ist auch bei G33/G34/G35 unwirksam.

Anwahl

Die Anwahl des Einzelsatzbetriebs ist möglich:

- Über die Maschinensteuertafel (Taste "Single Block")
- Über die Bedienoberfläche
Zur genauen Vorgehensweise siehe:
Literatur:
Bedienungshandbuch der installierten HMI-Applikation

Aktivierung

Die Aktivierung der Funktion erfolgt durch das PLC-Grundprogramm über das Nahtstellensignal:

DB21, ... DBX0.4 (Einzelsatz aktivieren)

Anzeige

Als Rückmeldung des aktiven Einzelsatzbetriebs wird auf der Bedienoberfläche das entsprechende Feld in der Statuszeile invers geschaltet.

Sobald die Teileprogrammbearbeitung wegen des Einzelsatzbetriebs einen Teileprogrammsatz abgearbeitet hat, wird das folgende Nahtstellensignal gesetzt:

DB21, ... DBX35.3 (Programmzustand unterbrochen)

Abarbeiten ohne Einzelsatz-Stopp

Für bestimmte Programmabläufe kann trotz angewählten Einzelsatzbetriebs ein Abarbeiten ohne Einzelsatz-Stopp eingestellt werden, z. B. für:

- Interne ASUPs
- Anwender-ASUPs
- Zwischensätze
- Satzsuchlauf sammelsätze (Aktionssätze)
- Init-Sätze
- Unterprogramme mit `DISPLOF`
- Nicht-reorganisierbare Sätze
- Nicht-repositionierbare Sätze
- Wiederanfahr Satz ohne Verfahrinformation
- Werkzeuganfahr Satz

Die Einstellung erfolgt über das Maschinendatum:

MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK (Einzelsatz-Stopp verhindern)

Literatur:

Listenhandbuch Ausführliche Beschreibung der Maschinendaten

10.5.3 Programmbearbeitung mit Probelaufvorschub

Funktion

Die Funktion "Programmbearbeitung mit Probelaufvorschub" dient zum Testen der Koordination des im Kanal angewählten NC-Programms mit z. B. dem PLC-Anwenderprogramm, externen Signalen und / oder anderen Kanälen der NC. Das NC-Programm wird dabei exakt, entsprechend den programmierten Befehlen, Funktionen und Verfahrgeschwindigkeiten abgearbeitet. Um während des Tests die Programmbearbeitungszeit zu verkürzen, können die Verfahrbewegungen durch Aktivierung des Probelaufvorschubs schneller ausgeführt werden. Statt des programmierten Vorschubs wirkt bei z. B. G01, G02, G03, G33, G34, G35, G95 dann der Probelaufvorschub.

Der Probelaufvorschub wird auf der Bedienoberfläche in den Betriebsarten AUTOMATIK und MDA angewählt und kann dann bei einer Automatikunterbrechung oder am Ende eines Satzes aktiviert werden. Die Aktivierung erfolgt vom PLC-Anwenderprogramm aus über die NC/PLC-Nahtstelle.

Weitere Informationen zur Vorschubbeeinflussung siehe Kapitel "V1: Vorschübe (Seite 1413)".

ACHTUNG

Zerstörung von Werkzeug oder Werkstück durch zu hohe Schnittgeschwindigkeiten

Wird bei aktiver Funktion "Programmbearbeitung mit Probelaufvorschub" eine Werkstückbearbeitung durchgeführt, können die sich dabei ergebenden Schnittgeschwindigkeiten außerhalb des zulässigen Bereichs liegen und zu Zerstörungen am Werkzeug und/oder Werkstück führen.

Parametrierung

Probelaufvorschub

Der im Kanal wirksame Probelaufvorschub wird eingestellt mit:

SD42100 \$SC_DRY_RUN_FEED = <Probelaufvorschub>

Hinweis

Umdrehungsvorschubs

Der Probelaufvorschubwert gilt auch anstelle des programmierten Umdrehungsvorschubs in Programmsätzen mit G95.

Probelaufeinstellungen

Das Auswahlkriterium für den bei "Programmbearbeitung mit Probelaufvorschub" im Kanal wirksamen Vorschub wird eingestellt mit:

SD42101 \$SC_DRY_RUN_FEED_MODE = <Wert>

<Wert>	Bedeutung
0	Als Probelaufvorschub wirkt das Maximum aus SD42100 und dem programmierten Vorschub.
1	Als Probelaufvorschub wirkt das Minimum aus SD42100 und dem programmierten Vorschub.
2	Als Probelaufvorschub wirkt SD42100.
10	Wie Wert 0, außer bei Gewindeschneiden (G33, G34, G35) und Gewindebohren (G331, G332, G63). Bei diesen Funktionen wirkt der programmierte Vorschub.
11	Wie Wert 1, außer bei Gewindeschneiden (G33, G34, G35) und Gewindebohren (G331, G332, G63). Bei diesen Funktionen wirkt der programmierte Vorschub.
12	Wie Wert 2, außer bei Gewindeschneiden (G33, G34, G35) und Gewindebohren (G331, G332, G63). Bei diesen Funktionen wirkt der programmierte Vorschub.

Aktivierung

Anwahl

Die Anwahl der Funktion erfolgt über die Bedienoberfläche über Bedienbereich "Maschine" > Betriebsart "AUTOMATIK" oder "MDA" > "Programmbeeinflussung" > "Probelaufvorschub".

Als Anforderung an das PLC-Anwenderprogramm die Funktion zu aktivieren wird folgendes NC/PLC-Nahtstellensignal gesetzt:

DB21, ... DBX24.6 = 1 (Probelaufvorschub angewählt)

Aktivierung

Zur Anforderung an die NC die Funktion zu aktivieren, muss das PLC-Anwenderprogramm folgendes NC/PLC-Nahtstellensignal setzen:

DB21, ... DBX0.6 = 1 (Probelaufvorschub aktivieren)

Rückmeldung

Als Rückmeldung für den Maschinenbediener, dass die Funktion in der NC aktiv ist, wird auf der Bedienoberfläche in der Statusanzeige des Bedienbereichs "Maschine" **DRY** angezeigt.

Literatur

Ausführliche Informationen zur Bedienung der Funktion "Programmbearbeitung mit Probelaufvorschub" finden sich in:

- Bedienhandbuch Drehen
- Bedienhandbuch Fräsen

Stichworte: Programmbeeinflussung, Probelaufvorschub, DRY

10.5.4 Teileprogrammsätze ausblenden

Funktion

Beim Testen bzw. Einfahren neuer Programme ist es hilfreich, wenn bestimmte Teileprogrammsätze für die Programmabarbeitung gesperrt bzw. ausgeblendet werden. Dazu müssen die betreffenden Sätze mit Schrägstrich gekennzeichnet werden.

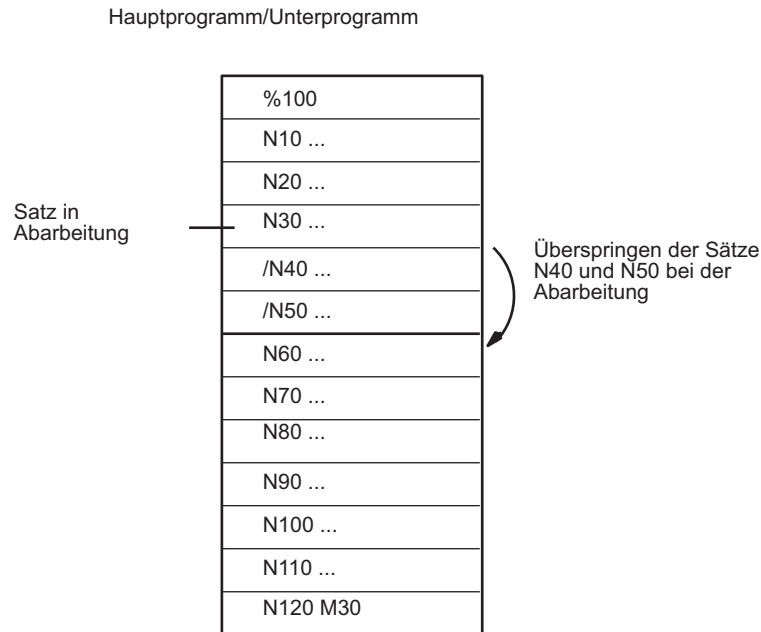


Bild 10-1 Ausblenden von Teileprogrammsätzen

Anwahl

Die Anwahl der Funktion erfolgt über die Bedienoberfläche im Menü "Programmbeeinflussungen".

Mit der Anwahl wird folgendes Nahtstellensignal gesetzt:

DB21, ... DBX26.0 (Satz ausblenden angewählt)

Die Funktion wird damit noch nicht aktiviert.

Aktivierung

Die Aktivierung dieser Funktion erfolgt über das Nahtstellensignal:

DB21, ... DBX2.0 (Satz ausblenden aktivieren)

Hinweis

Die Funktion "Teileprogrammsätze ausblenden" ist auch bei einem Satzsuchlauf aktiv.

Anzeige

Als Rückmeldung der aktivierten Funktion "Teileprogrammsätze ausblenden" wird auf der Bedienoberfläche das entsprechende Feld in der Statuszeile invers geschaltet.

10.6 Werkstücksimulation

Funktion

In der Werkstücksimulation wird das aktuelle Teileprogramm vollständig berechnet und das Ergebnis in der Bedienoberfläche grafisch dargestellt. Ohne die Maschinenachsen zu verfahren, wird so das Ergebnis der Programmierung kontrolliert. Falsch programmierte Bearbeitungsschritte werden frühzeitig erkannt und Fehlbearbeitungen am Werkstück verhindert.

Simulations-NC

Die Simulation nutzt eine eigene NC-Instanz (Simulations-NC). Deshalb muss vor einem Simulationsstart ein Abgleich des realen NC zum Simulations-NC erfolgen. Bei diesem Abgleich werden unter anderem sämtliche aktive Maschinendaten aus dem NC ausgelesen und in den Simulations-NC eingelesen. Unter den aktiven Maschinendaten befinden sich die NC- und Zyklenmaschinendaten.

Compile-Zyklen in Simulation (nur 840D sl)

Bis SW 4.4 werden keine, ab SW 4.4 nur ausgesuchte Compile-Zyklen (CC) bei der Werkstücksimulation unterstützt. Die Maschinendaten der unterstützten Compile-Zyklen werden einmalig nach Steuerungshochlauf abgeglichen. Ein Abgleich mit "Simulationsstart" findet nicht statt!

Hinweis

In Teileprogrammen können CC-spezifische Sprachbefehle und Maschinendaten von nicht unterstützten CCs **nicht** benutzt werden (siehe auch Absatz "CC-Befehle im Teileprogramm").

Sonderbewegungen von unterstützten CCs (OEM-Transformationen) werden unter Umständen falsch dargestellt.

CC-Befehle im Teileprogramm

Sprachbefehle im Teileprogramm von nicht unterstützten Compile-Zyklen (OMA1 ... OMA5, OEMIPO1/2, G810 ... G829, eigene Prozeduren und Funktionen) führen ohne individuelle Behandlung zur Alarmmeldung und zum Abbruch der Simulation.

Lösung: Die fehlenden CC-spezifischen Sprachelemente im Teileprogramm individuell behandeln (\$P_SIM Abfrage). Beispiel:

Programmcode	Kommentar
N1 G01 X200 F500	

Programmcode	Kommentar
IF (I==P_SIM)	
N5 X300	; Bei Simulation CC nicht aktiv.
ELSE	
N5 X300 OMA1=10	
ENDIF	

10.7 Satzsuchlauf Typ 1, 2 und 4

Funktion

Satzsuchlauf bietet die Möglichkeit, die Abarbeitung eines Teileprogramms von einem nahezu beliebigen Teileprogrammsatz aus zu beginnen.

Beim Satzsuchlauf erfolgt ein NC-interner Schnelldurchlauf ohne Verfahrbewegungen durch das Teileprogramm bis zum gewählten Zielsatz. Dabei wird versucht, möglichst exakt den Steuerungszustand zu erzielen, wie er sich am Zielsatz bei der normalen Teileprogrammabarbeitung (z. B. bezüglich Achspositionen, Spindeldrehzahlen, eingewechselte Werkzeuge, NC/PLC-Nahtstellensignale, Variablenwerte) ergibt, um mit möglichst wenigen manuellen Eingriffen ab dem Zielsatz mit der automatischen Teileprogrammabarbeitung fortfahren zu können.

Satzsuchlauftypen

- **Typ 1: Satzsuchlauf ohne Berechnung**
Satzsuchlauf ohne Berechnung dient zum schnellstmöglichen Auffinden eines Teileprogrammsatzes. Es wird dabei keinerlei Berechnung durchgeführt. Der Steuerungszustand am Zielsatz entspricht unverändert dem vor dem Start des Satzsuchlaufs.
- **Typ 2: Satzsuchlauf mit Berechnung an Kontur**
Satzsuchlauf mit Berechnung an Kontur dient dazu, um in beliebigen Situationen an die programmierte Kontur anfahren zu können. Mit NC-START wird die Anfangsposition des Zielsatzes bzw. die Endposition des Satzes vor dem Zielsatz angefahren. Dieser wird bis zur Endposition abgefahren. Die Abarbeitung erfolgt konturtreu.

- **Typ 4: Satzsuchlauf mit Berechnung an Satzendpunkt**
Satzsuchlauf mit Berechnung an Satzendpunkt dient dazu, um in beliebigen Situationen eine Zielposition (z. B. Werkzeugwechselposition) anfahren zu können. Angefahren wird der Endpunkt des Zielsatzes bzw. die nächste programmierte Position unter Verwendung der im Zielsatz gültigen Interpolationsart. Dies geschieht nicht konturtreu.
Es werden nur die im Zielsatz programmierten Achsen gefahren. Gegebenenfalls muss an der Maschine vor dem Start der weiteren automatischen Teileprogrammbearbeitung manuell über die Betriebsart "JOG-REPOS" eine kollisionsfreie Ausgangssituation geschaffen werden.
- **Typ 5: Satzsuchlauf mit Berechnung im Modus "Programmtest" (SERUPRO)**
SERUPRO (search run by programtest) ist ein Kanal-übergreifender Satzsuchlauf mit Berechnung. Die NC startet dabei das angewählte Teileprogramm im Modus "Programmtest". Mit Erreichen des Zielsatzes wird der Programmtest automatisch wieder abgewählt. Bei diesem Satzsuchlauftyp sind auch Interaktionen zwischen dem Kanal, in dem der Satzsuchlauf stattfindet, und Synchronaktionen und anderen Kanälen der NC möglich.
Siehe Kapitel "Satzsuchlauf Typ 5 (SERUPRO) (Seite 517)".

Hinweis

Weitere Erläuterungen zum Satzsuchlauf siehe Kapitel "Verhalten bei Satzsuchlauf (Seite 441)".

Folgeaktionen

Nach dem Abschluss eines Satzsuchlaufs können folgende Folgeaktionen erfolgen:

- Typ1 - Typ 5: Automatischer Start eines ASUPS
Mit dem Einwechseln des letzten Aktionssatzes kann ein Anwenderprogramm als ASUP gestartet werden.
- Typ1 - Typ 4: Kaskadierter Satzsuchlauf
Aus dem Zustand "Suchziel gefunden" kann ein weiterer Satzsuchlauf mit geänderter Suchzielangabe gestartet werden.

10.7.1 Funktionsbeschreibung

Prinzipieller Ablauf für Typ 2 oder Typ 4

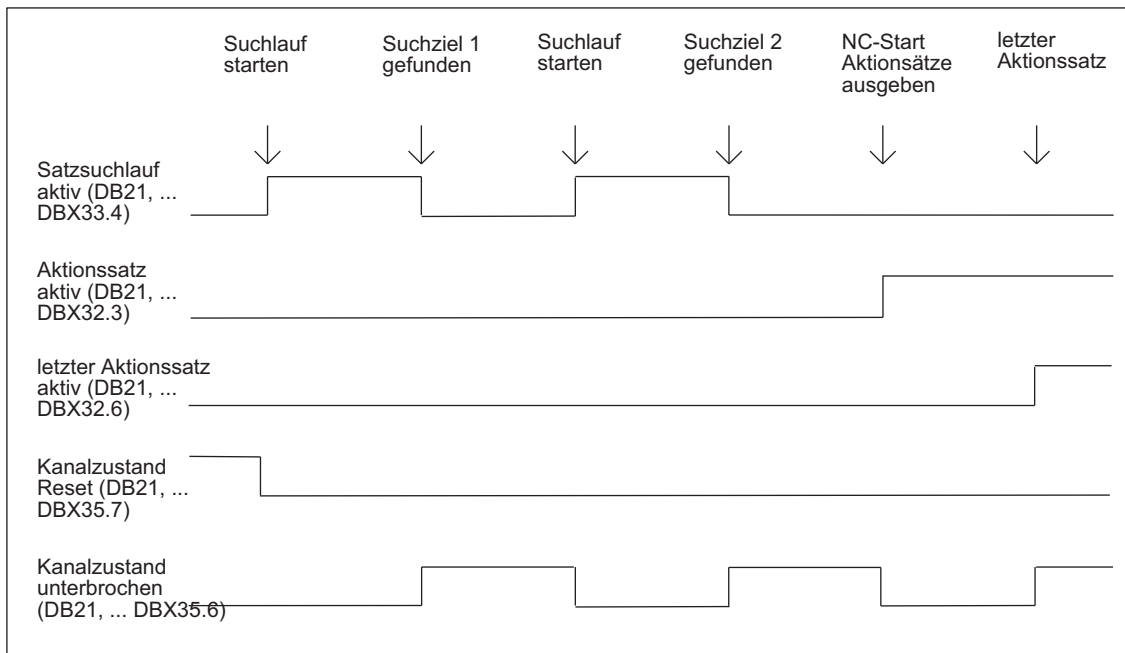
1. Anwender: Aktivierung des Satzsuchlaufs **Typ 2** oder **Typ 4** (Satzsuchlauf mit Berechnung an ...) über die Bedienoberfläche
2. Suchen des Zielsatzes mit Aufsammeln der Hilfsfunktionen
3. Stopp nach "Suchziel gefunden" → Anzeige des Alarms 10208 "Zur Programmfortsetzung NC-Start geben"
4. Anwender: NC-Start zum Abarbeiten der Aktionssätze → DB21, ... DBX7.1 = 1 (NC-Start)
5. Abarbeiten der Aktionssätze

6. Letzter Aktionssatz wird eingewechselt \Rightarrow Automatischer Start von /_N_CMA_DIR/_N_PROG_EVENT_SPF (Voreinstellung) als ASUP
7. Letzter ASUP-Satz (`REPOSA`) wird eingewechselt \Rightarrow DB21, ... DBX32.6 = 1 (letzter Aktionssatz aktiv)
8. **Optional:** Abarbeiten der anwenderspezifischen Anforderungen über PLC-Anwenderprogramm
9. Anzeige des Alarms 10208 "Zur Programmfortsetzung NC-Start geben"
10. Anwender: NC-Start zur Programmfortsetzung \Rightarrow DB21, ... DBX7.1 = 1 (NC-Start)

Suchziel nicht gefunden

Wird das Suchziel nicht gefunden, wird der Alarm 15370 "Suchziel bei Satzsuchlauf nicht gefunden" angezeigt und der Satzsuchlauf abgebrochen.

Zeitlicher Ablauf



Aktionssätze

Während eines Satzsuchlaufs **Typ 2** oder **Typ 4** (Satzsuchlauf mit **Berechnung** an ...) werden Aktionen wie z. B. Werkzeug- (T, D), Spindel- (S), Vorschub-Programmierung oder M-Funktionsausgaben aufgesammelt. Mit NC-Start zum Abarbeiten der Aktionssätze werden die aufgesammelten Aktionen an die PLC ausgegeben.

Hinweis

Mit NC-Start für die Aktionssätze, wird die während Satzsuchlauf **Typ 2** oder **Typ 4** (Satzsuchlauf mit **Berechnung** an ...) aufgesammelte Spindelprogrammierung (S-Wert, M3 / M4 / M5 / M19, SPOS) aktiv.

Der Anwender muss über das PLC-Anwenderprogramm sicherstellen, dass das Werkzeug betrieben werden kann oder die Spindelprogrammierung zurückgesetzt bzw. nicht ausgegeben wird:

DB31, ... DBX2.2 = 1 (Spindelreset) .

Randbedingungen

Aufsetzmodus nach Satzsuchlauf Typ 4

Erfolgt die erste Programmierung einer Achse nach einem Satzsuchlauf Typ 4 (Satzsuchlauf mit Berechnung an Satzendpunkt) inkrementell, kann der programmierte inkrementelle Wert auf den bis zum Suchziel aufgesammelten Positionswert oder auf den aktuellen Istwert der Achse addiert werden. Die Einstellung erfolgt über:

SD42444 \$SC_TARGET_BLOCK_INCR_PROG

Das Settingdatum wird mit NC-Start für die Ausgabe der Aktionssätze ausgewertet.

Einzelatz

Soll, nachdem das Suchziel bei Satzsuchlauf Typ 2 oder Typ 4 (Satzsuchlauf mit **Berechnung** an ...) gefunden wurde und aktiver Funktion "Einzelatz" (DB21, ... DBX0.4 == 1 (Einzelatz aktivieren)), nicht nach jedem Aktionssatz angehalten werden, kann dieses Verhalten über das Maschinendatum ausgeschaltet werden:

MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK, Bit 3 = 1 (Einzelatzunterdrückung bei Aktionssätzen)

Nahtstellensignal "Anfahratz aktiv"

Das Nahtstellensignal wird **nur** bei Satzsuchlauf **Typ 2** (Satzsuchlauf mit **Berechnung an Kontur**) gesetzt:

- DB21, ... DBX32.4 = 1 (Anfahratz aktiv)

Bei Satzsuchlauf **Typ 4** (Satzsuchlauf mit **Berechnung an Satzendpunkt**) wird das Nahtstellensignal nicht gesetzt, da hier kein Anfahratz erzeugt wird (Anfahratz gleich Zielsatz).

Interpolationsart des Zielsatzes

Bei Satzsuchlauf **Typ 4** (Satzsuchlauf mit **Berechnung an Satzendpunkt**) wird die Anfahrbewegung in der im Zielsatz gültigen Interpolationsart durchgeführt. Bei anderen

Interpolationsarten als Linearinterpolation (G0 oder G1), kann die Anfahrbewegung mit Alarm abgebrochen werden (z. B. Kreisendpunktfehler bei Kreisinterpolation G2 oder G3).

10.7.2 Satzsuchlauf im Zusammenhang mit weiteren NCK-Funktionen

10.7.2.1 ASUP nach und bei Satzsuchlauf

Satzsuchlauf Typ 2 und Typ 4: Synchronisation der Kanalachsen

Wenn nach einem Satzsuchlauf Typ 2 oder Typ 4 (Satzsuchlauf mit Berechnung an ...) ein ASUP gestartet wird, werden die Istpositionen aller Kanalachsen synchronisiert.

Auswirkung

Werden die nachfolgenden Systemvariablen im ASUP gelesen, enthalten sie folgende Werte:

- \$P_EP: Aktuelle Istposition der Kanalachse im WKS
- \$AC_RETPOINT: Aufgesammelte Satzsuchlaufposition der Kanalachse im WKS

Satzsuchlauf Typ 2: ASUP Abschluss

Bei Satzsuchlauf Typ 2 (Satzsuchlauf mit Berechnung an Kontur) muss zum Abschluss des ASUP der nachfolgende Befehl `REPOSA` (Wiederanfahren an die Kontur; linear; alle Kanalachsen) programmiert werden.

Auswirkung

- Alle Kanalachsen werden auf ihre im Satzsuchlauf aufgesammelte Suchlaufposition verfahren.
- \$P_EP == "aufgesammelte Satzsuchlaufposition der Kanalachse (WKS)"

Satzsuchlauf Typ 4: `REPOS`-Verhalten

Nach Satzsuchlauf Typ 4 (Satzsuchlauf mit Berechnung an Satzendpunkt) wird während des durch **Beginn** und **Ende** beschriebenen Zeitraums durch den Befehl `REPOS` kein automatisches Repositionieren ausgelöst:

- **Beginn**: NC/PLC-Nahtstellensignal DB21,... DBX32.6 == 1 (letzter Aktionssatz aktiv)
- **Ende**: Fortsetzen der Programmbearbeitung mit NC-Start.

Der Startpunkt der Anfahrbewegung sind die aktuellen Positionen der Kanalachsen zum Zeitpunkt zu dem NC-Start gegeben wird. Der Endpunkt ergibt sich durch die weiteren im Teileprogramm programmierten Verfahrbewegungen.

Bei Satzsuchlauf Typ 4 wird durch die NC keine Anfahrbewegung erzeugt.

Auswirkung:

- Die Systemvariable \$P_EP (programmierte Endposition) enthält nach Verlassen des ASUP die Istposition, auf die die Kanalachsen vom ASUP oder manuell (Betriebsart: JOG) positioniert wurden.
\$P_EP == "aktuelle Istposition der Kanalachse im WKS"

10.7.2.2 PLC-Aktionen nach Satzsuchlauf

Wenn alle Aktionssätze seitens der NC abgearbeitet sind und Aktionen seitens der PLC, z.B. Start eines ASUP um einen Werkzeugwechsel durchzuführen, oder des Bedieners, z.B. Überspeichern, möglich sind, wird folgendes kanalspezifisches NC/PLC-Nahtstellensignal gesetzt:

- DB21, ... DB32.6 = 1 (letzter Aktionssatz aktiv)

Zeitpunkt der Alarmausgabe parametrieren

Um den Bediener darauf hinzuweisen, dass zur Fortsetzung der Programmbearbeitung noch NC-Start im Kanal erforderlich ist, wird Alarm 10208 "Zur Programmfortsetzung NC-Start geben" angezeigt.

Wann der Alarm angezeigt wird, ist über das Maschinendatum einstellbar:

MD11450 \$MN_SEARCH_RUN_MODE, Bit 0 = <Wert>

<Wert>	Bedeutung
0	Mit dem Einwechseln des letzten Aktionssatzes nach Satzsuchlauf erfolgt: <ul style="list-style-type: none"> Die Bearbeitung des Teileprogramms wird gestoppt DB21, ... DB32.6 = 1 (letzter Aktionssatz aktiv) Alarm 10208 anzeigen
1	Mit dem Einwechseln des letzten Aktionssatzes nach Satzsuchlauf erfolgt: <ul style="list-style-type: none"> Die Bearbeitung des Teileprogramms wird gestoppt DB21, ... DB32.6 = 1 (letzter Aktionssatz aktiv) Alarm 10208 erst anzeigen, wenn DB21, ... DB32.6 == 1 (PLC-Aktion beendet)

Zusammen mit dem Alarm werden folgende Nahtstellensignale gesetzt:

- DB21, ... DB32.7 = 1 (NC-Alarm mit Bearbeitungsstillstand steht an)
- DB21, ... DB32.6 = 1 (NC-Alarm kanalspezifisch steht an)

10.7.2.3 Spindelfunktionen nach Satzsuchlauf**Steuerungsverhalten und Ausgabe**

Ob die beim Satzsuchlauf aufgesammelten Spindel-spezifischen Hilfsfunktionen automatisch im Aktionsatz oder anwenderspezifisch zu einem späteren Zeitpunkt an die PLC ausgegeben werden, ist einstellbar über

MD11450 \$MN_SEARCH_RUN_MODE, Bit 2 = <Wert>

<Wert>	Bedeutung
0	Die Ausgabe der bei Satzsuchlauf aufgesammelten Spindel-spezifischen Hilfsfunktionen (M3, M4, M5, M19, M70) erfolgt im Aktionssatz.
1	Die bei Satzsuchlauf aufgesammelten Spindel-spezifischen Hilfsfunktionen werden im Aktionssatz nicht ausgegeben. Die Ausgabe kann anwenderspezifisch zu einem späteren Zeitpunkt, z.B. in einem ASUP, vorgenommen werden. Die aufgesammelten Spindel-spezifischen Hilfsfunktionen werden dazu in Systemvariablen gespeichert:

Systemvariablen

Die beim Satzsuchlauf aufgesammelten Spindel-spezifischen Hilfsfunktionen werden in folgenden Systemvariablen gespeichert:

Systemvariable	Beschreibung
\$P_SEARCH_S [<n>]	Zuletzt programmierte Spindeldrehzahl bzw. Schnittgeschwindigkeit
\$P_SEARCH_SDIR [<n>]	Zuletzt programmierte Spindeldrehrichtung
\$P_SEARCH_SGEAR [<n>]	Zuletzt programmierte Getriebestufen M-Funktion
\$P_SEARCH_SMODE [<n>]	Zuletzt programmierte Spindelbetriebsart
\$P_SEARCH_SPOS [<n>]	Zuletzt mittels M19, SPOS bzw. SPOSA programmierte Spindelposition bzw. Verfahrenweg
\$P_SEARCH_SPOSMODE [<n>]	Zuletzt mittels M19, SPOS bzw. SPOSA programmierter Positionsanfahrmode
<n>: Spindelnummer	

Zur späteren Ausgabe der Spindel-spezifischen Hilfsfunktionen können die Systemvariablen z.B. in einem ASUP gelesen und nach Ausgabe der Aktionssätze ausgegeben werden:

DB21, ... DBX32.6 == 1 (Letzter Aktionssatz aktiv)

Hinweis

Die Inhalte der Systemvariablen \$P_S, \$P_DIR und \$P_SGEAR können nach Satzsuchlauf durch Synchronisationsvorgänge verloren gehen.

Für weiterführende Informationen zu ASUP, Satzsuchlauf und Aktionssätzen siehe Kapitel "Ausgabeunterdrückung von Spindel-spezifischen Hilfsfunktionen (Seite 446)" und "Programmtest (Seite 494)".

10.7.2.4 Lesen von Systemvariablen bei Satzsuchlauf

In Teileprogrammen können über Systemvariable Werte aus den Bereichen Vorlauf, Hauptlauf oder Servo/Antrieb gelesen werden:

\$P_...	Vorlauf-bezogene Systemvariablen, enthalten programmierte Werte
\$A_...	Hauptlauf-bezogene Systemvariable,n enthalten aktuelle Werte
\$V_...	Servo/Antrieb-bezogene Systemvariablen, enthalten aktuelle Werte

Da während eines Satzsuchlaufs vom Typ 2 und Typ 4 (Satzsuchlauf **mit Berechnung** an ...) keine Sätze in den Hauptlauf gelangen, werden Hauptlauf- und Servo/Antrieb-bezogene Systemvariablen während des Satzsuchlaufs nicht verändert. Gegebenenfalls muss für diese Systemvariablen durch Abfrage ob ein Satzsuchlauf aktiv ist \$P_SEARCH (Satzsuchlauf aktiv) im NC-Programm eine Sonderbehandlung für Satzsuchlauf durchgeführt werden.

Vorlauf-bezogene Systemvariable liefern in allen Suchlaufarten korrekte Werte.

10.7.3 Automatischer Start eines ASUP nach Satzsuchlauf

Parametrierung

Wirksamsetzen der Funktion

Der automatische ASUP-Start nach Satzsuchlauf wird durch folgende MD-Einstellung aktiviert:

MD11450 \$MN_SEARCH_RUN_MODE, Bit 1 = 1

Zu aktivierendes Programm

In der Grundstellung wird nach dem Satzsuchlauf mit dem Einwechseln des letzten Aktionssatzes das Programm **_N_PROG_EVENT_SPF** aus dem Verzeichnis **_N_CMA_DIR** als ASUP aktiviert. Soll ein anderes Programm aktiviert werden, muss im folgenden Maschinendatum der Name dieses Anwenderprogramms eingetragen sein:

MD11620 \$MN_PROG_EVENT_NAME

Verhalten bei gesetzter Einzelsatzbearbeitung

Über das kanalspezifische Maschinendatum wird eingestellt, ob das aktivierte ASUP trotz gesetzter Einzelsatzbearbeitung ohne Unterbrechung abgearbeitet wird, oder die Einzelsatzbearbeitung wirksam sein soll:

MD20106 \$MC_PROG_EVENT_IGN_SINGLEBLOCK, Bit 4 = <Wert>

<Wert>	Bedeutung
0	Einzelsatzbearbeitung ist wirksam.
1	Einzelsatzbearbeitung wird unterdrückt.

Verhalten bei gesetzter Einlesesperre

Über das kanalspezifische Maschinendatum wird eingestellt, ob das ASUP trotz gesetzter Einlesesperre (DB21, ... DBX6.1 == 1) komplett abgearbeitet wird, oder die Einlesesperre wirksam sein soll:

MD20107 \$MC_PROG_EVENT_IGN_INHIBIT, Bit 4 = <Wert>

<Wert>	Bedeutung
0	Einlesesperre ist wirksam.
1	Einlesesperre wird unterdrückt.

Hinweis

Weitere Informationen zur Parametrierung von MD11620, MD20108 und MD20107 siehe Kapitel "Parametrierung (Seite 577)".

Programmierung

Durch welches Ereignis das ASUP gestartet wurde, ist in der Systemvariablen \$P_PROG_EVENT hinterlegt. Bei automatischer Aktivierung nach Satzsuchlauf liefert \$P_PROG_EVENT den Wert "5".

Ablauf

Ablauf zum automatischen Start eines ASUP nach Satzsuchlauf:

1. Anwender: Aktivierung des Satzsuchlaufs **Typ 2** oder **Typ 4** (Satzsuchlauf **mit Berechnung** an ...) über die Bedienoberfläche
2. Suchen des Zielsatzes mit Aufsummieren der Hilfsfunktionen
3. Stopp nach "Suchziel gefunden ⇒ Anzeige des Alarms 10208 "Zur Programmfortsetzung NC-Start geben"
4. Anwender: NC-Start zum Abarbeiten der Aktionssätze ⇒ DB21, ... DBX7.1 = 1 (NC-Start)
5. Abarbeiten der Aktionssätze
6. Letzter Aktionssatz wird eingewechselt ⇒ Automatischer Start von /_N_CMA_DIR/_N_PROG_EVENT_SPF (Voreinstellung) als ASUP
7. Letzter ASUP-Satz (REPOSA) wird eingewechselt ⇒ DB21, ... DBX32.6 = 1 (letzter Aktionssatz aktiv)
8. **Optional:** Abarbeiten der anwenderspezifischen Anforderungen über PLC-Anwenderprogramm
9. Anzeige des Alarms 10208 "Zur Programmfortsetzung NC-Start geben"

Hinweis

Mit MD11450 \$MN_SEARCH_RUN_MODE, Bit 0 == 1, wird Alarm 10208 erst nach Freigabe durch das PLC-Anwenderprogramm (DB21, ... DBX1.6 = 1 (PLC-Aktion beendet)) ausgegeben.

10. Anwender: NC-Start zur Programmfortsetzung ⇒ DB21, ... DBX7.1 = 1 (NC-Start)

10.7.4 Kaskadierter Satzsuchlauf

Funktionalität

Mit der Funktion "Kaskadierter Satzsuchlauf" ist es möglich aus dem Zustand "Suchziel gefunden" einen weiteren Satzsuchlauf zu starten. Die Kaskadierung kann nach jedem gefundenen Suchziel beliebig oft fortgesetzt werden und ist für folgende Satzsuchlauffunktionen anwendbar:

- Satzsuchlauf Typ 1 ohne Berechnung
- Satzsuchlauf Typ 2 mit Berechnung an Kontur
- Satzsuchlauf Typ 3 mit Berechnung an Satzendpunkt

Hinweis

Nur wenn das Suchziel gefunden wurde, kann aus der gestoppten Programmbearbeitung ein weiterer "Kaskadierter Satzsuchlauf" gestartet werden.

Aktivierung

Der "Kaskadierte Satzsuchlauf" wird im bestehenden Maschinendatum projiziert:
MD11450 \$MN_SEARCH_RUN_MODE

- Mit Bit 3 = 0 (FALSE) wird der kaskadierte Satzsuchlauf frei geschaltet (d. h. mehrfache Suchzielvorgaben sind möglich).
- Aus Kompatibilitätsgründen kann der kaskadierte Satzsuchlauf mit Bit 3 = 1 (TRUE) gesperrt werden. Voreingestellt ist der kaskadierte Satzsuchlauf Bit 3 = 0.

Ablaufverhalten

Suchziel gefunden, Satzsuchlauf erneut starten

Mit dem Erreichen des Suchziels wird die Programmbearbeitung gestoppt und das Suchziel als aktueller Satz angezeigt. Nach jedem gefundenen Suchziel ist ein neuer Satzsuchlauf beliebig oft wiederholbar.

Suchzielvorgaben ändern

Vor jeden Satzsuchlaufstart können Suchzielangabe und Satzsuchlauffunktion geändert werden.

Beispiel: Ablauf einer Bearbeitungssequenz mit kaskadiertem Satzsuchlauf

- RESET
- Satzsuchlauf bis Suchziel 1
- Satzsuchlauf bis Suchziel 2 → "Kaskadierter Satzsuchlauf"

- NC-Start für Ausgabe der Aktionssätze → Alarm 10208
- NC-Start → Fortsetzen der Programmbearbeitung

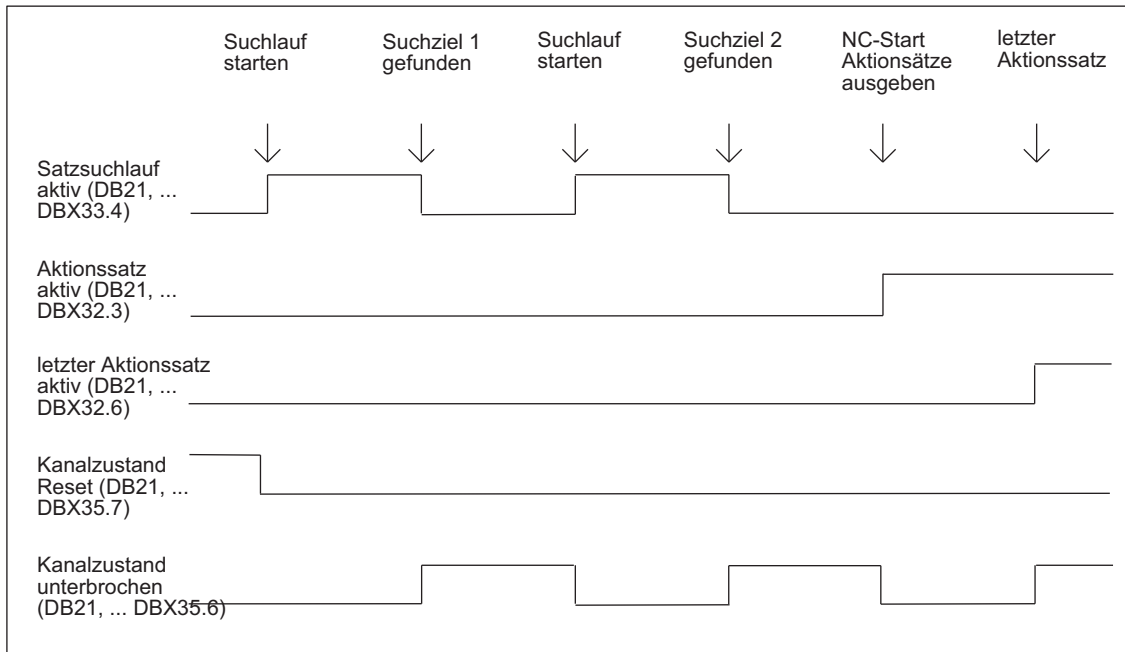


Bild 10-2 Zeitlicher Ablauf Nahtstellensignale

10.7.5 Beispiele zum Satzsuchlauf mit Berechnung

Auswahl

Wählen Sie aus den folgenden Beispielen den Satzsuchlauftyp aus, der Ihrer Problemstellung am Besten entspricht.

Satzsuchlauf Typ 4 mit Berechnung an Satzendpunkt

Beispiel mit automatischem Werkzeugwechsel nach Satzsuchlauf bei aktiver Werkzeugverwaltung:

1. Maschinendaten setzen:
MD11450 \$MN_SEARCH_RUN_MODE auf 1
MD11602 \$MN_ASUP_START_MASK Bit 0 = 1 (ASUP-Start aus gestopptem Zustand)
2. ASUP "SUCHLAUF_ENDE" von PLC über FB4 anwählen (siehe auch Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)").
3. Teileprogramm "WERKSTUECK_1" laden und anwählen.
4. Satzsuchlauf auf Satzendpunkt Satznummer N220.
5. HMI meldet "Suchziel gefunden".
6. NC-Start für die Ausgabe der Aktionssätze.

7. Mit dem PLC-Signal:
DB21... DB32.6 (letzter Aktionssatz aktiv)
startet die PLC über FC9 das ASUP "SUCHLAUF_ENDE" (siehe auch Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)").
8. Nach ASUP-Ende (auswertbar z. B. über die zu definierende M-Funktion M90, siehe Beispiel Satz N1110) setzt die PLC das Signal:
DB21, ... DBX1.6 (PLC-Aktion beendet).
Alternativ kann auch das NC/PLC-Nahtstellensignal:
DB21-DB30 DBB318 Bit 0 (ASUP ist angehalten)
abgefragt werden.
Dadurch wird der Alarm 10208 angezeigt, d. h. es können jetzt weitere Aktionen seitens des Bedieners erfolgen.
9. Manuelle Eingriffe durch Bediener (JOG, JOG-REPOS, Überspeichern).
10. Teileprogramm mit NC-Start fortsetzen.

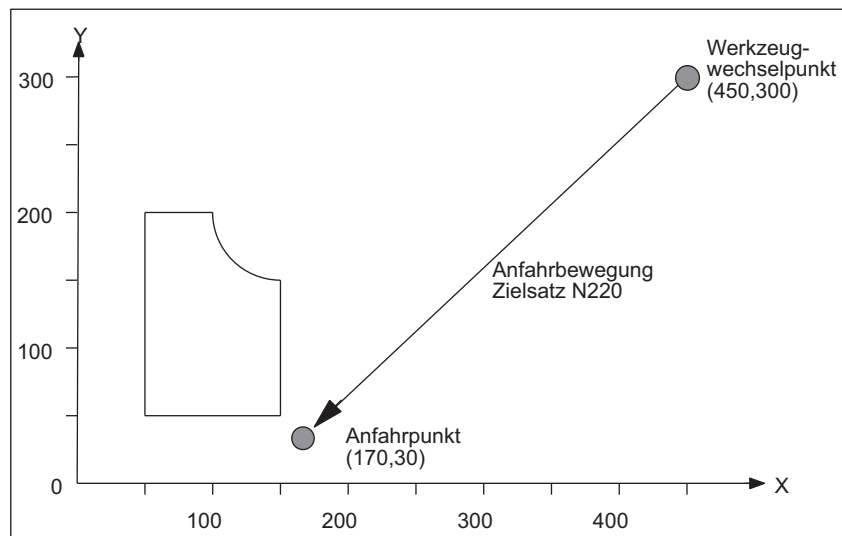


Bild 10-3 Anfahrbewegung bei Satzsuchlauf auf Satzendpunkt (Zielsatz N220)

Hinweis

"Satzsuchlauf an Kontur" mit Zielsatz N220 würde eine Anfahrbewegung zum Werkzeugwechsellpunkt (Startpunkt des Zielsatzes) erzeugen.

Satzsuchlauf Typ 2 mit Berechnung an Kontur

Beispiel mit automatischem Werkzeugwechsel nach Satzsuchlauf bei aktiver Werkzeugverwaltung:

1. bis 3. wie Beispiel zum Satzsuchlauf Typ 4
4. Satzsuchlauf an Kontur Satznummer N260
5. bis 10. wie Beispiel zum Satzsuchlauf Typ 4

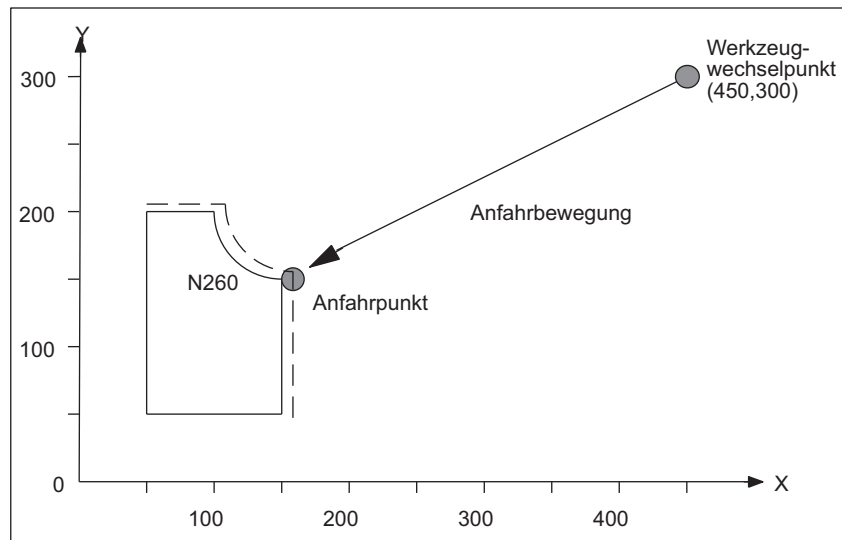


Bild 10-4 Anfahrbewegung bei Satzsuchlauf an Kontur (Zielsatz N260)

Hinweis

"Satzsuchlauf auf Satzendpunkt " mit Zielsatz N260 würde zu Alarm 14040 (Kreisendpunktfehler) führen.

Teilprogramme für Typ 4 und Typ 2**PROC WERKSTUECK_1**

Programmcode	Kommentar
;Hauptprogramm	
...	
;Bearbeitung Konturabschnitt 1 mit Werkzeug "FRAESER_1"	
...	
N100 G0 G40 X200 Y200	; Abwahl Radiuskorrektur
N110 Z100 D0	; Abwahl Längskorrektur
;Ende Konturabschnitt 1	
;	
;Bearbeitung Konturabschnitt 2 mit Werkzeug "FRAESER_2"	
N200 T="FRAESER_2"	; Werkzeug vorwählen
N210 WZW	; Werkzeugwechselroutine aufrufen
N220 G0 X170 Y30 Z10 S3000 M3 D1	; Anfahr Satz Konturabschnitt 2
N230 Z-5	; Zustellung
N240 G1 G64 G42 F500 X150 Y50	; Startpunkt der Kontur
N250 Y150	
N260 G2 J50 X100 Y200	
N270 G1 X50	

10.7 Satzsuchlauf Typ 1, 2 und 4

Programmcode	Kommentar
N280 Y50	
N290 X150	
N300 G0 G40 G60 X170 Y30	; Abwahl Radiuskorrektur
N310 Z100 D0	; Abwahl Längenkorrektur
Ende Konturabschnitt 2	
...	
M30	
PROC WZW	
;Werkzeugwechselroutine	
N500 DEF INT TNR_AKTIV	; Variable für aktive T-Nummer
N510 DEF INT TNR_VORWAHL	; Variable für vorgewählte T-Nummer
N520 TNR_AKTIV = \$TC_MPP6[9998,1]	; T-Nummer des aktiven Werkzeugs lesen
N530 GETSELT(TNR_VORWAHL)	; T-Nummer des vorgewählten Werkzeugs lesen
;	
;Werkzeugwechsel nur ausführen, wenn Werkzeug noch nicht aktiv ist	
N540 IF TNR_AKTIV == TNR_VORWAHL GOTOF ENDE	
N550 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	; Werkzeugwechselpunkt anfahren
N560 M6	; Werkzeugwechsel ausführen
;	
ENDE: M17	
PROC SUCHLAUF_ENDE SAVE	
;ASUP für den Aufruf der Werkzeugwechselroutine nach Satzsuchlauf	
N1000 DEF INT TNR_AKTIV	; Variable für aktive T-Nummer
N1010 DEF INT TNR_VORWAHL	; Variable für vorgewählte T-Nummer
N1020 DEF INT TNR_SUCHLAUF	; Variable für im Satzsuchlauf ermittelte
	telte
	; T-Nummer
N1030 TNR_AKTIV = \$TC_MPP6[9998,1]	; T-Nummer des aktiven Werkzeugs lesen
N1040 TNR_SUCHLAUF = \$P_TOOLNO	; durch Satzsuchlauf ermittelte T-Nummer lesen
N1050 GETSELT(TNR_VORWAHL)	; T-Nummer des vorgewählten Werkzeugs lesen
N1060 IF TNR_AKTIV ==TNR_SUCHLAUF GOTOF ASUP_ENDE	
N1070 T = \$TC_TP2[TNR_SUCHLAUF]	; T-Anwahl über Werkzeugnamen
N1080 WZW	; Werkzeugwechselroutine aufrufen
N1090 IF TNR_VORWAHL == TNR_SUCHLAUF GOTOF ASUP_ENDE	
N1100 T = \$TC_TP2[TNR_VORWAHL]	; T-Vorwahl restaurieren über Werkzeugnamen
ASUP_ENDE:	
N1110 M90	; Rückmeldung an PLC
N1120 REPOSA	; ASUP Ende

10.7.6 Randbedingungen

10.7.6.1 Kompressor-Funktionen (COMPON, COMPCURV, COMPCAD)

- Wenn bei Satzsuchlauf **Typ 2** oder **Typ 4** (Satzsuchlauf **mit Berechnung** an ...) der Zielsatz in einem Programmabschnitt liegt, in dem eine Kompressor-Funktion (COMPON, COMPCURV, COMPCAD) aktiv ist, werden beim Wiederaufahren an die Kontur Positionen auf der durch den **Kompressor** berechneten Bahn angefahren. Diese Positionen müssen nicht exakt mit den Positionen auf der im Teileprogramm programmierten Bahn übereinstimmen.
- Wenn bei der Komprimierung im Teileprogramm programmierte Sätze entfallen, werden diese Sätze im Satzsuchlauf nicht als Zielsatz gefunden ⇒ Alarm 15370 "Suchziel nicht

10.8 Satzsuchlauf Typ 5 (SERUPRO)

10.8.1 Funktionsbeschreibung

Der Suchlauftyp 5, Satzsuchlauf mit Berechnung im Modus "Programmtest" (SERUPRO, "Search-Run by Programtest") ermöglicht einen kanalübergreifenden Satzsuchlauf mit Berechnung zu einem wählbaren Unterbrechungspunkt. Dazu werden während SERUPRO, unter Beachtung vorhandener Programmkoordinierungsbefehle, alle zur Programmfortsetzung in den unterbrochenen Kanälen erforderlichen Zustandsdaten ermittelt und anschließend NC und PLC in einen Zustand versetzt, der die Programmfortsetzung erlaubt.

Vor dem Wiederaufahren an die Kontur mit anschließendem Fortsetzen der Programmbearbeitung, können über ein automatisch gestartetes anwenderspezifisches ASUP alle noch eventuell erforderlichen Ausgangszustände erzeugt werden.

Kanäle

In Verbindung mit HMI ist SERUPRO für folgende Kanäle vorgesehen:

- Nur für den aktuellen SERUPRO-Kanal (1)
- Für alle Kanäle mit gleichem Werkstücknamen wie der SERUPRO-Kanal (2)
- Für alle Kanäle mit gleicher BAG wie der SERUPRO-Kanal (3)
- Für alle Kanäle der NCU (4)

Die Auswahl des Kanalumfangs für SERUPRO erfolgt über die Projektierungsdatei **maschine.ini**, in dem Abschnitt [BlockSearch]:

Section [BlockSearch]	Satzsuchlauf-Funktion für HMI freigeben und Satzsuchlauf-Konfiguration auswählen
SeruproEnabled=1	;SERUPRO Softkey bei HMI verfügbar. Standardwert ist (1)
SeruproConfig=1	;Nummer (1) bis (4) der oben genannten Kanalgruppierung. Standardwert ist (1)

10.8 Satzsuchlauf Typ 5 (SERUPRO)

Alle anderen mit SERUPRO gestarteten Kanäle werden im Modus "Self-Acting Serupro" betrieben. Nur der Kanal, in dem auch ein Zielsatz ausgewählt wurde, kann mit Satzsuchlauf im Modus SERUPRO gestartet werden.

Aktivierung

Die Aktivierung von SERUPRO erfolgt über HMI. Mit den Softkey "Prog.Test Kontur" wird SERUPRO bedient.

SERUPRO verwendet REPOS, um den Zielsatz anzufahren.

Zeitlicher Ablauf von SERUPRO

1. Über HMI wird Softkey "Pog. Test Kontur" und das Suchziel bedient.
2. Die NC startet jetzt selbsttätig das angewählte Programm im Modus "Programmtest".
 - Achsen werden dabei nicht verfahren.
 - Hilfsfunktionen \$A_OUT und die direkte PLC-IO werden ausgegeben.
 - Die Hilfsfunktionen des Zielsatzes werden nicht ausgegeben.
3. Die NC stoppt am Beginn des Zielsatzes, wählt intern Programmtest ab, und zeigt die Stopp-Bedingung "Warten: Suchziel gefunden" an.
4. Liegt das anwenderspezifische ASUP "PROG_EVENT.SPF" vor, wird es automatisch gestartet.
5. Mit dem nächsten NC-Start erfolgt ein Wiederanfahren an die Kontur (REPOS). Der REPOS-Vorgang erfolgt über ein System-ASUP und kann über die Funktion "Editierbares ASUP" erweitert werden.

Randbedingungen für Satzsuchlauf SERUPRO

Die Funktion SERUPRO darf nur in der Betriebsart "Automatik" aktiviert und im Programmzustand (Kanalzustand RESET) abgebrochen werden.

Startet im Normalbetrieb **nur** der PLC gemeinsam mehrere Kanäle, so kann dies durch SERUPO in jedem Kanal simuliert werden.

Bei Maschinendatum-Einstellung:

MD10708 \$MN_SERUPRO_MASK, Bit 1 = 0

bricht der Alarm 16942 "Kanal %1 Start-Programmbefehl Aktion %2<ALNX> nicht möglich" die Simulation dann ab, wenn der Teileprogrammbefehl `START` verwendet wird.

Das Maschinendatum:

MD10707 \$MN_PROG_TEST_MASK

erlaubt das Abschalten im gestoppten Zustand und hat keinen Einfluss auf den SERUPRO-Vorgang. Die Standardvorbelegung erlaubt das Abschalten nur im RESET-Zustand.

Hinweis

Nach dem Abschalten von Programmtest beginnt ein REPOS-Vorgang, für den dieselben Einschränkungen wie beim SERUPRO-Anfahren gelten. Auftretende Beeinträchtigungen lassen sich mit einem ASUP verhindern.

SERUPRO-Verhalten beeinflussen

Für die nachfolgend beispielhaft aufgeführten Funktionen kann das Verhalten von SERUPRO NC-spezifisch vorgegeben werden:

- Programmierter Halt (M0)
- Programmkoordinierungsbefehl START
- Group-SERUPRO
- Kanalübergreifendes Beenden von SERUPRO
- Override

MD10708 \$MN_SERUPRO_MASK = <Verhalten bei SERUPRO>

Kanalspezifische Grundstellungen für SERUPRO

Mit dem nachfolgenden Maschinendatum werden im Normalfall die kanalspezifischen Grundstellungen nach Teileprogrammstart festgelegt:

MD20112 \$MC_START_MODE_MASK= <Grundstellungen>

Für SERUPRO können eigene Grundstellungen welche die Grundstellungen aus MD20112 ersetzen, vorgegeben werden:

MD22620 \$MC_START_MODE_MASK_PRT = <SERUPRO-Grundstellungen>

Die SERUPRO-Grundstellungen müssen explizit freigegeben werden über:

MD22621 \$MC_ENABLE_START_MODE_MASK_PRT = 1

NC/PLC-Nahtstellensignal "Satzsuchlauf via Programmtest ist aktiv"

Der Satzsuchlauf via Programmtest wird angezeigt über das NC/PLC-Nahtstellensignal: DB21, ... DBX318.1 == 1

Das Nahtstellensignal ist gesetzt vom Start des Satzsuchlaufs bis zum Einwechseln des Zielsatzes in den Hauptlauf.

Zum benutzerdefinierten ASUP nach dem SERUPRO-Vorgang

Hinweis

Entschließt sich der Maschinenhersteller nach dem SERUPRO-Vorgang ein ASUP gemäß Punkt 7. zu starten, so muss Folgendes beachtet werden:

Gestoppter Zustand nach Punkt 6.:

Die Maschinendaten:

MD11602 \$MN_ASUP_START_MASK

und

MD11604 \$MN_ASUP_START_PRIO_LEVEL

erlauben es dem NC, das ASUP aus dem gestoppten Zustand selbsttätig über FC9-Baustein zu starten.

Quittierung vom FC9 erst nach Beendigung des REPOS-Satz:

Das ASUP kann erst vom FC9-Baustein mit "Asup Done" als beendet gemeldet werden, wenn auch der REPOS-Satz beendet ist.

Abwahl des vorgesehenen REPOS-Vorgangs nach Punkt 8.:

Der Start des ASUPs wählt den vorgesehenen REPOS-Vorgang ab!

Daher sollte das ASUP mit REPOSA abgeschlossen werden, um den REPOS-Vorgang beizubehalten.

Löschen eines unerwünschten REPOS-Vorgangs:

Der unerwünschte REPOS-Vorgang wird gelöscht, indem das ASUP mit M17 oder RET beendet wird.

Besondere Behandlung von ASUP:

Grundsätzlich wird ein ASUP, das mit REPOS endet und aus einem gestoppten Zustand heraus gestartet wird, besonders behandelt.

Das ASUP stoppt selbsttätig vor dem REPOS-Satz und zeigt dies an über:

DB21, ... DBX318.0 (ASUP angehalten)

Automatischer ASUP-Start

Das unter dem Pfad:

/_N_CMA_DIR/_N_PROG_EVENT_SPF

liegende ASUP, wird automatisch mit dem Maschinendatum:

MD11450 \$MN_SEARCH_RUN_MODE, Bit1 = 1

im SERUPRO-Anfahren nach folgenden Ablauf gestartet:

1. Der SERUPRO-Vorgang ist komplett durchgeführt.
2. Der Anwender löst "NC-Start" aus.
3. Das ASUP wird gestartet.
4. Die NC stoppt vor dem Teileprogrammbefehl REPOS selbsttätig und die Meldung "Zur Programmfortsetzung NC-Start geben" erscheint.

5. Der Anwender drückt zum zweiten Mal "NC-Start".
6. Die NC führt die Wiederanfahrbewegung aus und setzt das Teileprogramm mit dem Zielsatz fort.

Hinweis

Der automatische ASUP-Start mit MD11450 benötigt **Starts**, um das Programm fortzusetzen.

Damit nähert sich das Ablaufverhalten an andere Satzsuchlauftypen an.

10.8.2 Wiederanfahren an die Kontur (REPOS)

Mit der Funktion "Wiederanfahren an die Kontur" (REPOS) kann eine unterbrochene Bearbeitung an der Unterbrechungsstelle fortgesetzt werden. Im Gegensatz zu REPOS ermöglicht SERUPRO das "Nachholen" oder "Wiederholen" eines Programmabschnittes. Dazu wird, nachdem SERUPRO den Zielsatz gefunden hat, die Kontur an der mit REPOS-Mode wählbaren Stelle angefahren und die Bearbeitung fortgesetzt.

SERUPRO: REPOS Verhalten einstellen

Das REPOS-Verhalten, d. h. das Verhalten im Wiederanfahrtsatz, wird über folgendes Maschinendatum eingestellt:

MD11470 \$MN_REPOS_MODE_MASK = <REPOS-Mode>

<REPOS-Mode>		
Bit	Wert	Bedeutung
0	0	Ein unterbrochene Verweilzeit wird wiederholt
	1	Ein unterbrochene Verweilzeit fortgesetzt
1	-	Reserviert
2	0	DB31, ... DBX10.0 (REPOS Verzögerung) wird im Wiederanfahrtsatz nicht beachtet.
	1	DB31, ... DBX10.0 (REPOS Verzögerung) wird im Wiederanfahrtsatz beachtet.
3	0	SERUPRO: Im Wiederanfahrtsatz nur Bahnachsen verfahren
	1	SERUPRO: Im Wiederanfahrtsatz Bahn- und Positionierachsen gleichzeitig verfahren
4	0	REPOS: Im Wiederanfahrtsatz nur Bahnachsen verfahren
	1	REPOS: Im Wiederanfahrtsatz Bahn- und Positionierachsen gleichzeitig verfahren
5	0	Während der Unterbrechung geänderte Vorschübe und Spindeldrehzahlen wirken erst ab dem ersten Teileprogrammsatz nach der Unterbrechungsstelle
	1	Während der Unterbrechung geänderte Vorschübe und Spindeldrehzahlen wirken ab der Unterbrechungsstelle, d. h. sie werden bereits im Restsatz gültig und werden damit vorgezogen. Dieses Verhalten bezieht sich auf jeden REPOS-Vorgang.

<REPOS-Mode>		
Bit	Wert	Bedeutung
6	0	SERUPRO: Im Wiederanfahrtsatz werden neutrale Achsen und positionierende Spindeln als Bahnachsen verfahren.
	1	SERUPRO: Im Wiederanfahrtsatz werden neutrale Achsen und positionierende Spindeln als Kommandoachsen verfahren. Neutrale Achsen und positionierende Spindeln werden nach SERUPRO repositioniert. Für Neutrale Achsen, die nicht repositioniert werden dürfen, muss die REPOS Verzögerung gesetzt werden: DB31, ... DBX10.0 = 1 (REPOS Verzögerung)
7	0	Die REPOS Verzögerung ist nicht freigegeben.
	1	Die REPOS Verzögerung ist freigegeben. Achsen mit aktiver REPOS Verzögerung (DB31, ... DBX10.0 == 1), die keine Geometrie- oder Orientierungsachsen sind, werden beim Wiederanfahren nicht verfahren.

**VORSICHT****Kollisionsgefahr**

MD11470 \$MN_REPOS_MODE_MASK, Bit 3 oder Bit 4 = 1

Es liegt in der alleinigen Verantwortung des Anwenders sicherzustellen, dass es beim gleichzeitigen Verfahren der Achsen im Wiederanfahrtsatz zu keiner Kollision an der Maschine kommt.

Wiederanfahren mit gesteuertem REPOS

Der REPOS-Mode kann für die Bahnachsen über die NC/PLC-Nahtstelle vorgegeben werden:

DB21, ... DBX31.0 - 2 (REPOS-Mode)

Der REPOS-Mode wird im NC-Programm programmiert und bestimmt das Anfahrverhalten (siehe Kapitel "Wiederanfahren an die Kontur mit gesteuertem REPOS (Seite 530)").

Das REPOS-Verhalten einzelner Achsen kann auch über NC/PLC-Nahtstellensignale gesteuert werden und muss über das Maschinendatum freigegeben werden:

MD11470 \$MN_REPOS_MODE_MASK.BIT 2 = 1

Bahnachsen können nicht einzeln beeinflusst werden. Bei allen weiteren Achsen, die keine Geometrieachsen sind, kann REPOS einzelner Achsen vorübergehend verhindert und auch verschoben werden. Über NC/PLC-Nahtstellensignal werden die einzelnen Kanalachsen, die

REPOS herausfahren möchte, zu einem späteren Zeitpunkt wieder freigegeben oder weiter gesperrt.

**GEFAHR****Kollisionsgefahr**

Durch das Signal DB31, ... DBX2.2 (Restweg löschen) ergibt sich folgendes gefährliches Verhalten, wenn die Funktion "Repositionieren einzelner Achsen verhindern" angewählt ist:

MD11470 \$MN_REPOS_MODE_MASK.Bit 2 == 1

Solange eine Achse nach der Unterbrechung inkrementell programmiert wird, fährt die NC andere Positionen an als ohne Unterbrechung.

Siehe nachfolgendes Beispiel: Achse wird inkrementell programmiert

Beispiel: Rundachse A wird inkrementell programmiert

Die Rundachse A ist die vierte Maschinenachse.

- Die Rundachse A steht vor dem REPOS-Vorgang auf 11°
Im Unterbrechungssatz, d. h. im Zielsatz von SERUPRO, soll die Rundachse A auf 27° verfahren.
Beliebig viele Sätze später wird die Rundachse A inkrementell um 5° verfahren:
N1010 POS[A]=IC(5) FA[A]=1000
Bei gesetztem Nahtstellensignal DB34 DBX10.0 = 1 (REPOS Verzögerung) verfährt die Rundachse A im REPOS-Vorgang nicht und wird mit N1010 nach 32° gefahren. Der Benutzer muss evtl. den Weg von 11° nach 27° bewusst quittieren.

**GEFAHR****Kollisionsgefahr**

Da die Achse nach der Unterbrechung inkrementell programmiert wird, verfährt sie nach 16° statt 32°.

- Achsen einzeln starten
Das REPOS-Verhalten für das SERUPRO-Anfahren mit mehreren Achsen wird angewählt mit:
MD11470 \$MN_REPOS_MODE_MASK.BIT 3 = 1
Die NC beginnt SERUPRO-Anfahren mit einem Satz, der **alle** Positionierachsen auf das programmierte Ende und die Bahnachse auf den Zielsatz fährt.
Der Benutzer startet die einzelnen Achsen, indem er die Vorschubfreigaben entsprechend anwählt. Anschließend wird der Zielsatz abgefahren.
- Positionierachsen im Wiederanfahrtsatz repositionieren
Positionierachsen werden nicht im Restsatz, sondern im Wiederanfahrtsatz repositioniert und betreffen nicht nur den Satzsuchlauf über Programmtest beim SERUPRO-Anfahren:
MD11470 \$MN_REPOS_MODE_MASK.Bit 3 = 1: für Satzsuchlauf über Programmtest (SERUPRO)
MD11470 \$MN_REPOS_MODE_MASK.Bit 4 = 1: für jedes REPOS

Hinweis

Ist weder Bit 3 noch Bit 4 gesetzt, werden in dieser Phase Achsen, die keine Bahnachsen sind im Restsatz repositioniert.

Achse mit REPOS-Verschiebung verzögert anfahren

Ist das achsspezifische Nahtstellensignal DB31, ... DBX10.0 (REPOS Verzögerung) gesetzt, wird mit der positiven Flanke von DB21, ... DBX31.4 (REPOS Mode Änderung) die REPOS-Verschiebung für diese Achse erst mit ihrer nächsten Programmierung herausgefahren.

Ob diese Achse gerade einer REPOS-Verschiebung unterliegt, kann über Synchronaktionen mit \$AA_REPOS_DELAY gelesen werden.



VORSICHT

Kollisionsgefahr

DB31, ... DBX10.0 (REPOS Verzögerung) hat auf Maschinenachsen, die eine Bahn bilden, keine Auswirkung.

Ob eine Achse eine Bahnachse ist, kann über DB31, ... DBX76.4 (Bahnachse) gelesen werden.

Übernahmezeitpunkt der REPOS-Signale

Mit der positiven Flanke von DB21, ... DBX31.4 (REPOS Mode Änderung) werden folgende REPOS-Signale in die NC übernommen:

- Kanalspezifisch: DB21, ... DBX31.0 - 2 (REPOS Mode)
- Achsspezifisch: DB31, ... DBX10.0 (REPOS Verzögerung)

Die Pegel der REPOS-Signale beziehen sich auf den aktuellen Hauptlaufsatz. Es werden folgende zwei Fälle unterschieden:

1. Im Hauptlauf befindet sich **ein** Wiederanfahrtsatz eines gerade aktiven REPOS-Vorgangs. Der laufende REPOS-Vorgang wird abgebrochen, neu aufgesetzt, und die REPOS-Verschiebungen werden durch die oben genannten REPOS-Signale beeinflusst.
2. Im Hauptlauf befindet sich **kein** Wiederanfahrtsatz eines gerade aktiven REPOS-Vorgangs. Jeder zukünftige REPOS-Vorgang, der an den aktuellen Hauptlaufsatz wieder anfahren möchte, wird durch die oben genannten REPOS-Signale beeinflusst.

Hinweis

In einem laufenden ASUP wirkt DB21, ... DBX31.4 (REPOS Mode Änderung) nicht auf das abschließende REPOS, außer das Signal wird zufällig zum Zeitpunkt gesetzt zu dem die REPOS-Sätze abgearbeitet werden.

Im 1. Fall ist das Signal nur im gestoppten Zustand erlaubt.

Verhalten bei RESET:

- Die NC hat das PLC-Signal bereits quittiert:
DB21, ... DBX31.4 == 1 (REPOS Mode Änderung) **UND**
DB21, ... DBX319.0 == 1 (REPOS Mode Änderung Quittung)
Tritt in dieser Situation ein Kanal-Reset auf, wird der aktive REPOS-Mode gelöscht:
DB21, ... DBX319.1 - 3 = 0 (Aktiver REPOS-Mode)
 - Die NC hat das PLC-Signal noch nicht quittiert:
DB21, ... DBX31.4 == 1 (REPOS Mode Änderung) **UND**
DB21, ... DBX319.0 == 0 (REPOS Mode Änderung Quittung)
Tritt in dieser Situation ein Kanal-Reset auf, wird die Quittierung der REPOS Mode Änderung und der aktive REPOS-Mode gelöscht:
DB21, ... DBX319.0 = 0 (REPOS Mode-Änderung Quittung)
DB21, ... DBX319.1 - 3 = 0 (Aktiver REPOS-Mode)
-

Mit NC/PLC-Nahtstellensignalen SERUPRO-Anfahren beeinflussen

Das SERUPRO-Anfahren kann mit DB21, ... DBX31.4 (REPOS Mode Änderung) und den dazugehörigen Signalen in den folgenden Phasen eingesetzt werden:

- Von "Suchziel gefunden" bis "Start des SERUPRO-ASUPs"
- Von "SERUPRO-ASUP stoppt selbsttätig vor dem REPOS" bis "Zielsatz wird abgearbeitet"

Während das SERUPRO-ASUP z. B. im Programmteil vor dem REPOS abgearbeitet wird, wirkt das Nahtstellensignal nicht auf SERUPRO-Anfahren.

REPOS Abläufe mit NC/PLC-Nahtstellensignalen

REPOS mit NC/PLC-Nahtstellensignalen steuern

REPOS-Verschiebungen lassen sich mit den folgenden NC/PLC-Nahtstellensignalen beeinflussen:

- DB21, ... DBX31.0 - 2 (REPOS Mode)
- DB21, ... DBX31.4 (REPOS Mode Änderung)
- DB31, ... DBX10.0 (REPOS Verzögerung)
- DB31, ... DBX72.0 (REPOS Verzögerung)

REPOS-Quittierungssignale

Mit folgenden NC/PLC-Nahtstellensignalen werden Funktionen, die das REPOS-Verhalten über PLC beeinflussen, **von der NC** quittiert:

- DB21, ... DBX319.0 (REPOS Mode-Änderung Quittung)
- DB21, ... DBX319.1 - 3 (Aktiver REPOS Mode)
- DB21, ... DBX319.5 (REPOS Verzögerung Quittung)
- DB31, ... DBX70.0 (REPOS Verschiebung)
- DB31, ... DBX70.1 (REPOS Verschiebung gültig)
- DB31, ... DBX70.2 (REPOS Verzögerung Quittung)
- DB31, ... DBX76.4 (Bahnachse)

Weitere Informationen hierzu finden Sie unter "REPOS-Verschiebung in der Nahtstelle".

REPOS-Quittierungsvorgänge

Wird von der NC eine "REPOS Mode Änderung" erkannt (DB21, ... DBX31.4 == 1), wird diese von der PLC mit DB21, ... DBX319.0 = 1 quittiert.

Hinweis

Hat die NC das Nahtstellensignal DB21, ... DBX31.4 (REPOS Mode Änderung) noch nicht mit dem Nahtstellensignal DB21, ... DBX319.0 (REPOS Mode-Änderung Quittung) quittiert, führte ein Kanal-Reset in dieser Situation zum Programmabbruch, und REPOS, mit dem der REPOS Mode beeinflusst werden soll, wird nicht durchgeführt.

Ein von PLC vorgegebener REPOS Mode wird von der NC mit den folgenden Nahtstellensignalen quittiert:

- DB21, ... DBX319.1 - 3 (Aktiver REPOS Mode)
- DB31, ... DBX10.0 (REPOS Verzögerung)
- DB31, ... DBX70.2 (REPOS Verzögerung Quittung)

Beispiel

- Zeitpunkt ②: Ein NC-Programm wird in Satz N20 mit NC-Stop angehalten. Alle Achsen werden über ihre parametrisierte Bremsrampe bis zum Stillstand abgebremst.
- Zeitpunkt ③: Nachdem vom PLC-Anwenderprogramm der "REPOS Mode" gesetzt wurde, übernimmt die NC den REPOS Mode mit der 0/1-Flanke von "REPOS Mode-Änderung".
- Zeitpunkt ④: "REPOS Mode Änderung Quittung" bleibt so lange gesetzt, bis das ASUP ausgelöst wird
- Zeitpunkt ⑤: Der REPOS-Vorgang wird im ASUP begonnen.
- Zeitpunkt ⑥: Der Restsatz des ASUPs wird wieder eingewechselt.

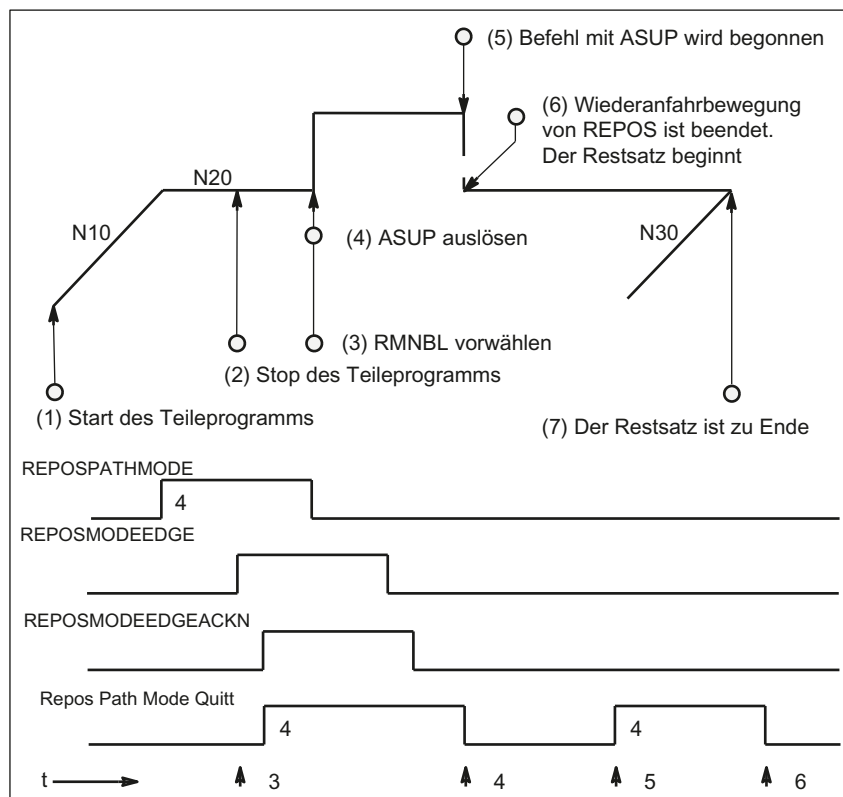


Bild 10-5 REPOS Ablauf im Teileprogramm mit zeitlichen Quittierungssignalen von NC

NC setzt Quittierung erneut

Phase, in der REPOSPATHMODE weiterhin wirkt (Restsatz des im → Zeitpunkt (2) gestoppten Programms ist noch nicht zu Ende ausgeführt).

Sobald die REPOS-Wiederanfahrbewegung des ASUP bearbeitet wird, setzt der NC den "Repos Path Mode Quitt" erneut (→ Zeitpunkt (5)). Sollte kein REPOSPATHMODE über NC/ PLC-Nahtstellensignal vorgewählt worden sein, wird der programmierte REPOS-Mode angezeigt.

"Repos Path Mode Quitt" wird mit dem Einwechseln des Restsatzes (→ Zeitpunkt (6)) zurückgenommen. Der dem im → Zeitpunkt (2) folgende Teileprogrammsatz N30 wird fortgesetzt.

Das Nahtstellensignal:

DB31, ... DBX70.2 (REPOS Verzögerung Quittung) ist analog definiert.

DB31, ... DBX70.1 (REPOS Verschiebung gültig) = 1, wenn:

DB21, ... DBX319.1-319.3 (Aktiver REPOS Mode) = 4 (RMNBL).

Gültige REPOS-Verschiebung

Mit dem Ende des SERUPRO-Vorgangs kann der Anwender die REPOS-Verschiebung über das NC/PLC-Nahtstellensignal Achse/Spindel (NC→PLC):
DB31, ... DBX70.0 (REPOS Verschiebung) auslesen.

Dieses Signal hat folgende Auswirkung für diese Achse:

Wert 0: Keine REPOS-Verschiebung wird herausgefahren werden.

Wert 1: Eine REPOS-Verschiebung wird herausgefahren werden.

Gültigkeitsbereich

Das Nahtstellensignal:

DB31, ... DBX70.0 (REPOS Verschiebung)

wird mit dem Ende des SERUPRO-Vorganges versorgt.

Mit dem Start eines SERUPRO-ASUP oder dem automatischen ASUP-Start wird die REPOS-Verschiebung ungültig.

REPOS-Verschiebung im Gültigkeitsbereich aktualisieren

Zwischen SERUPRO-Ende und -Start kann mit Moduswechsel die Achse in JOG verfahren werden.

Der Anwender fährt die REPOS-Verschiebung händisch mit JOG heraus, um das NST DB31, ... DBX70.0 (REPOS Verschiebung) auf den Wert 0 zu setzen.

Im Gültigkeitsbereich kann die Achse auch über FC18 verfahren werden, wobei das NST DB31, ... DBX70.0 (REPOS Verschiebung) ständig aktualisiert wird.

Gültigkeitsbereich anzeigen

Der Gültigkeitsbereich von der REPOS-Verschiebung wird angezeigt mit dem Nahtstellensignal:

DB31, ... DBX70.1 (REPOS Verschiebung gültig)

Es wird angegeben, ob eine gültige Berechnung vorliegt:

Wert 0: REPOS-Verschiebung dieser Achse ist korrekt berechnet.

Wert 1: REPOS-Verschiebung dieser Achse ist nicht berechenbar, das REPOS liegt in der Zukunft, z. B. am ASUP-Ende, oder kein REPOS aktiv.

REPOS-Verschiebung nach einem Achstausch

Mit dem Sammelsignal DB21, ... DBX319.5 (REPOS Verzögerung) kann festgestellt werden, ob eine gültige REPOS-Verschiebung stattgefunden hat:

Wert 0: Alle Achsen, die von diesem Kanal aktuell kontrolliert werden, haben entweder keine REPOS-Verschiebung oder ihre REPOS-Verschiebungen sind ungültig.

Wert 1: Sonstige.

REPOS-Verschiebung bei synchronisierter Synchronspindelkopplung

Beim Wiederanfahren mit SERUPRO wird an die Unterbrechungsstelle wieder vorgelaufen. War eine Synchronspindelkopplung bereits synchronisiert, dann existiert keine REPOS-Verschiebung der Folgespindel und es steht auch kein Synchronisationsweg an. Die Synchronisationssignale bleiben gesetzt.

Suchziel gefunden beim Satzwechsel

Das achsspezifische NC/PLC-Nahtstellensignal DB31, ... DBX76.4 (Bahnachse) ist 1, wenn die Achse Teil des Bahnverbundes ist.

Dieses Signal zeigt den Zustand des aktuell zu bearbeitenden Satzes beim Satzwechsel an. Spätere Zustandsänderungen werden nicht berücksichtigt.

Wenn der SERUPRO-Vorgang mit "Suchziel gefunden" beendet ist, bezieht sich DB31, ... DBX76.4 (Bahnachse) auf den Zielsatz.

10.8.2.1 Wiederanfahren an die Kontur mit gesteuertem REPOS

Nachdem mit SERUPRO der Zielsatz gefunden wurde, erfolgt vor dem Fortsetzen des unterbrochenen Programms ein REPOS-Vorgang zum Wiederanfahren der Kontur. Standardmäßig ist der REPOS-Mode "Wiederanfahren an den Satzanfangspunkt des Zielsatzes" (RMBBL) aktiv. Über die NC/PLC-Nahtstelle kann der REPOS-Mode anwenderspezifisch vorgegeben werden:

DB21, ... DBX31.0 - .2 (REPOS-Mode)

Literatur:

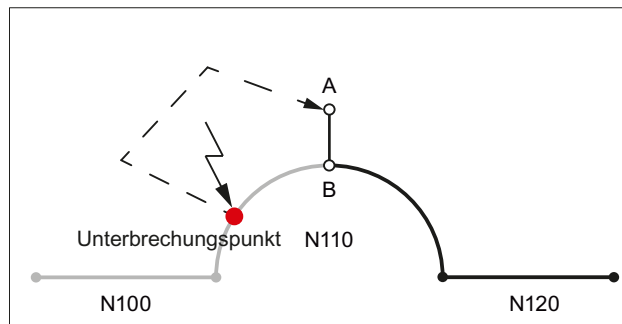
Eine ausführliche Beschreibung der Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

REPOS-Mode: Wiederanfahren an den nächstliegenden Bahnpunkt (RMNBL)

Im REPOS-Mode **RMNBL** wird von der REPOS-Startposition aus der nächstliegende Punkt der Kontur angefahren.

Beispiel

Die Programmunterbrechung erfolgte an einer beliebigen Stelle im Satz N110. Die Achsen wurden anschließend, z.B. manuell, auf Position (A) verfahren. Nachdem SERUPRO den Zielsatz N110 gefunden hat, erfolgt der REPOS-Vorgang mit REPOS-Mode **RMNBL**. Von der REPOS-Startposition (A) aus ist Punkt (B) der nächstliegende Punkt der Kontur. Mit Erreichen des Punktes (B) ist der REPOS-Vorgang abgeschlossen. Ab Punkt (B) wird wieder die programmierte Kontur des unterbrochenen Programms abgefahren.



REPOS-Mode über NC/PLC-Nahstelle vorgeben

Der REPOS-Mode kann über folgendes NC/PLC-Nahtstellensignal vorgegeben werden:
DB21, ... DBX31.0 - .2 (REPOS-Mode)

Hinweis

RMNBL ist eine allgemeine REPOS-Erweiterung und nicht nur auf SERUPRO beschränkt.

RMIBL und RMBBL verhalten sich bei SERUPRO identisch.

DB21, ... DBX31.0 - .2 (REPOS-Mode) beeinflusst nur die Verfahrbewegung der Bahnachsen.

Das Verhalten der anderen Achsen kann einzeln mit dem Nahtstellensignal DB31, ... DBX10.0 (REPOSDELAY) verändert werden. Die REPOS-Verschiebung wird nicht sofort, sondern erst mit ihrer nächsten Programmierung herausgefahren.

Weitere Informationen zur Programmierung des Wiederanfahrpunkts siehe unter:

Literatur:

Programmierhandbuch Arbeitsvorbereitung; Bahnverhalten, Kapitel: Wiederanfahren an Kontur

10.8.3 Suchlaufvorgang beschleunigen

Maschinendateneinstellungen

Die Abarbeitungsgeschwindigkeit des SERUPRO-Vorgangs kann über das nachfolgende Maschinendatum beeinflusst werden

MD22600 \$MC_SERUPRO_SPEED_MODE = <Wert>

<Wert>	Bedeutung
0	Programmtest mit Satzsuchlauf-/Probelaufvorschub-Geschwindigkeit: <ul style="list-style-type: none"> • Achsen: MD22601 \$MC_SERUPRO_SPEED_FACTOR * Probelaufvorschub • Spindeln: MD22601 \$MC_SERUPRO_SPEED_FACTOR * programmierte Drehzahl Dynamische Begrenzungen von Achsen / Spindeln werden nicht beachtet.
1	Programmtest mit programmierter Geschwindigkeit: <ul style="list-style-type: none"> • Achsen: Probelaufvorschub • Spindeln: programmierte Drehzahl Dynamische Begrenzungen von Achsen / Spindeln werden beachtet.
2	Programmtest mit Probelaufvorschub Unter Programmtest wird mit der programmierten Geschwindigkeit / Drehzahl gefahren. Dynamische Begrenzungen von Achsen / Spindeln werden beachtet.
3	Programmtest mit Satzsuchlauf-Geschwindigkeit Unter Programmtest wird mit folgender Geschwindigkeit gefahren: <ul style="list-style-type: none"> • Achsen: MD22601 \$MC_SERUPRO_SPEED_FACTOR * programmierter Vorschub • Spindeln: MD22601 \$MC_SERUPRO_SPEED_FACTOR * programmierte Drehzahl. Dynamische Begrenzungen von Achsen / Spindeln werden nicht beachtet. Hinweis Bei aktivem Umdrehungsvorschub (z.B. G95) wird der programmierte Vorschub nicht mit MD22601 \$MC_SERUPRO_SPEED_FACTOR multipliziert, sondern nur die programmierte Spindeldrehzahl. Dadurch ergibt sich auch hier eine Erhöhung der effektiven Bahngeschwindigkeit um MD22601 \$MC_SERUPRO_SPEED_FACTOR.

Randbedingungen

Hauptlaufachsen

MD22600 \$MC_SERUPRO_SPEED_MODE wirkt bei SERUPRO auf folgende Hauptlaufachsen:

- PLC-Achsen
- Kommando-Achsen
- Positionierachsen
- Pendelachsen

Synchronaktionen**ACHTUNG****Aktionen vom Synchronaktion werden bei SERUPRO u.U. nicht ausgeführt**

Da während SERUPRO intern andere Aktualwerte (z.B. Achspositionen) erzeugt werden als im normalen Programmablauf, kann es bei SERUPRO vorkommen, dass Bedingungen von Synchronaktionen die Aktualwerte (z.B. Achspositionen) abprüfen, nicht mehr als wahr (TRUE) erkannt werden und demzufolge der Aktionsteil der Synchronaktion nicht ausgeführt wird.

Umdrehungsvorschub

Auswirkungen von MD22601 \$MC_SERUPRO_SPEED_FACTOR bei DryRun:

- Umschalten von G95/G96/G961/G97/G971 auf G94
- Gewindebohren und Gewindeschneiden: normale DryRun-Geschwindigkeit.

Gewindebohren ohne Ausgleichsfutter

- Bei "Gewindebohren ohne Ausgleichsfutter" (G331/G332), wird die Spindel lagegeregelt im Bahnverbund interpoliert. Dabei ist die Bohrtiefe (Linearachse), die Gewindesteigung und Drehzahl (Spindel) vorgegeben.
Während DryRun ist die Geschwindigkeit von der Linearachse vorgegeben, die Drehzahl bleibt konstant und die Gewindesteigung wird angepasst.
Nach SERUPRO ergibt sich für die Spindel eine zum Normalbetrieb abweichende Position, weil die Spindel bei SERUPRO weniger oft gedreht hat.

10.8.4 SERUPRO-ASUP**SERUPRO-ASUP Besonderheiten**

Während SERUPRO-ASUP sind Besonderheiten zu beachten bei:

- Referenzpunktfahren: Referenzieren per Teileprogramm G74
- Werkzeugverwaltung: Werkzeugwechsel und Magazindaten
- Spindelhochlauf: Beim Start eines SERUPRO-ASUP

G74 Referenzpunkt fahren

Beindet sich zwischen dem Programmanfang und dem Suchziel die Anweisung G74 (Referenzpunktfahren), so wird dies von der NC ignoriert.

SERUPRO-Anfahren berücksichtigt diese Anweisung G74 nicht!

Werkzeugverwaltung

Bei aktiver Werkzeugverwaltung (WZV) wird folgende Einstellung empfohlen:

MD20310 \$MC_TOOL_MANAGEMENT_MASK BIT 20 = 0

Damit wird das während des SERUPRO-Vorgangs erzeugte WZV-Kommando **nicht** an die PLC ausgegeben!

Das WZV-Kommando wirkt sich wie folgt aus:

- Die NC quittiert die Kommandos selbst.
- Es werden keine Magazindaten verändert.
- Werkzeug-Daten werden nicht verändert.

Ausnahme:

Der Werkzeug-Zustand des im Testbetrieb aktivierten Werkzeuges kann den Zustand 'aktiv' annehmen. Damit kann nach dem SERUPRO-Vorgang das falsche Werkzeug auf der Spindel sein.

Abhilfe:

Der Anwender startet ein SERUPRO-ASUP, das real verfahren wird. Vor dem Start kann der Anwender ein ASUP starten, welches das korrekte Werkzeug einwechselt.

SERUPRO-Vorgang: Funktionalität: Im Ablauf vom Punkt 2. bis 6.

SERUPRO-ASUP: Funktionalität: Im Ablauf Punkt 7.

Außerdem muss im Maschinendatum MD20310 \$MC_TOOL_MANAGEMENT_MASK das Bit 11 = 1 gesetzt werden, da das ASUP ggf. eine T-Anwahl wiederholen muss.

Anlagen mit Werkzeugverwaltung und Nebenspindel unterstützen SERUPRO nicht!

Beispiel

Werkzeugwechselunterprogramm

Programmcode	Kommentar
PROC L6	; Werkzeugwechselroutine
N500 DEF INT TNR_AKTUELL	; Variable für aktive T-Nummer
N510 DEF INT TNR_VORWAHL	; Variable für vorgewählte T-Nummer
	; Aktuelles Werkzeug ermitteln
N520 STOPRE	; Im Programmtest-Betrieb wird
N530 IF \$P_ISTEST	; aus dem Programmkontext
N540 TNR_AKTUELL = \$P_TOOLNO	; das "aktuelle" Werkzeug gelesen.
N550 ELSE	; Andernfalls wird das Werkzeug der Spindel ausgelesen.
N560 TNR_AKTUELL = \$TC_MPP6[9998,1]	; T-Nummer des Werkzeugs auf der Spindel lesen
N570 ENDIF	
N580 GETSELT(TNR_VORWAHL)	; T-Nummer des vorgewählten Werkzeugs der Masterspindel lesen
	; Werkzeugwechsel nur ausführen, wenn Werkzeug noch nicht aktuell ist
N590 IF TNR_AKTUELL <> TNR_VORWAHL	; Werkzeugwechselpunkt anfahren
N600 G0 G40 G60 G90 SUPA X450 Y300	
Z300 D0	
N610 M206	; Werkzeugwechsel ausführen
N620 ENDIF	

Programmcode	Kommentar
N630 M17	

ASUP für den Aufruf der Werkzeugwechselroutine nach Satzsuchlauf-Typ 5

Programmcode	Kommentar
PROC ASUPWZV2	
N1000 DEF INT TNR_SPINDEL	; Variable für aktive T-Nummer
N1010 DEF INT TNR_VORWAHL	; Variable für vorgewählte T-Nummer
N1020 DEF INT TNR_SUCHLAUF	; Variable für im Satzsuchlauf ermittelte T-Nummer
N1030 TNR_SPINDEL = \$TC_MPP6[9998,1]	; T-Nummer des Werkzeugs auf der Spindel lesen
N1040 TNR_SUCHLAUF = \$P_TOOLNO	; Durch Satzsuchlauf ermittelte T-Nummer lesen, ; d. h. dieses Werkzeug bestimmt die ; momentane Werkzeugkorrektur.
N1050 GETSELT(TNR_VORWAHL)	; T-Nummer des vorgewählten Werkzeugs lesen
N1060 IF TNR_SPINDEL == TNR_SUCHLAUF GOTOF AS- UP_ENDE1	
N1070 T = \$TC_TP2[TNR_SUCHLAUF]	; T-Anwahl über Werkzeugnamen
N1080 L6	; Werkzeugwechselroutine aufrufen
N1085 ASUP_ENDE1:	
N1090 IF TNR_VORWAHL == TNR_SUCHLAUF GOTOF AS- UP_ENDE	
N1100 T = \$TC_TP2[TNR_VORWAHL]	; T-Vorwahl restaurieren über Werkzeugnamen
N1110 ASUP_ENDE:	
N1110 M90	; Rückmeldung an PLC
N1120 REPOSA	; ASUP Ende

In beiden Programmen PROC L6 und PROC ASUPWZV2 wird der Werkzeugwechsel mit M206 anstelle mit M6 programmiert.

Das ASUP-Programm "ASUPWZV2" benutzt verschiedene Systemvariablen, die einerseits den Programmfortschritt (\$P_TOOLNO) kennen und andererseits den aktuellen Zustand der Maschine (\$TC_MPP6[9998,1]) darstellen.

Spindelhochlauf

Beim Start des SERUPRO-ASUPs wird die Spindel nicht auf die im Programm vorgesehene Drehzahl hochgedreht, denn mit SERUPRO-ASUP soll nach dem Werkzeugwechsel, das neue Werkzeug auf die richtige Werkstückposition korrigiert werden.

Ein Spindelhochlauf wird mit SERUPRO-ASUP wie folgt durchgeführt:

- SERUPRO-Vorgang ist komplett beendet.
- Der Anwender startet über den FC-9 Baustein das SERUPRO-ASUP mit dem ggf. die Spindel hochgedreht wird.

- Der Start nach M0 im ASUP verändert den Spindelzustand nicht.
- SERUPRO-ASUP stoppt vor dem REPOS-Teileprogrammsatz selbsttätig.
- Der Anwender drückt START.
- Die Spindel dreht auf den Zielsatzzustand hoch, falls im ASUP die Spindel nicht anders programmiert worden war.

Hinweis

Anpassungen für REPOS der Spindeln:

Bei Anpassungen für SERUPRO-Anfahren und der Spindelfunktionalität sind Übergänge von Drehzahlsteuerbetrieb und Positionierbetrieb zu beachten.

Weitere Informationen zu den Betriebsartenwechsel von Spindeln siehe Kapitel "Betriebsarten (Seite 1296)".

10.8.5 Self-Acting SERUPRO

Self-Acting SERUPRO

Die kanalspezifische Funktion "Self-Acting SERUPRO" erlaubt einen SERUPRO-Ablauf **ohne** vorher ein Suchziel in einem Programm der abhängigen SERUPRO-Kanäle definiert zu haben.

Außerdem kann ein spezieller Kanal, der "seruproMasterChan", für **jedes** "Self-Acting SERUPRO" definiert werden. In diesem kann ein Suchziel definiert werden.

Die Funktion "Self-Acting SERUPRO" unterstützt den kanalübergreifenden Satzsuchlauf SERUPRO.

Funktion

Mit den Vorgang "Self-Acting SERUPRO" kann kein Suchziel gefunden werden. Wird das Suchziel nicht erreicht, so wird auch kein Kanal angehalten. In bestimmten Situationen wird aber trotzdem der Kanal vorübergehend angehalten. Dabei wird der Kanal in der Regel auf einen anderen Kanal warten. Beispiele hierfür sind: Wait-Marken, Kopplungen oder Achstausch.

Warte-Phase tritt auf:

In dieser Warte-Phase prüft die NC den Kanal "seruproMasterChan", ob dieser ein Suchziel erreicht hat. Die Warte-Phase wird bei Nichterreichen eines Suchziels wieder verlassen.

Wird das Suchziel erreicht, so wird der SERUPRO-Vorgang auch in diesem Kanal beendet. Der Kanal "seruproMasterChan" muss im normalen SERUPRO-Modus gestartet worden sein.

Keine Warte-Phase tritt auf:

"Self-Acting SERUPRO" wird durch das M30 des Teileprogramms beendet.

Der Kanalzustand befindet sich danach wieder im RESET-Zustand.

Ein SERUPRO-Anfahren findet nicht statt.

Start einer Gruppe von Kanälen

Wird eine Gruppe von Kanälen nur mit "Self-Acting SERUPRO" gestartet, so werden alle Kanäle mit "RESET" beendet.

Ausnahmen:

Ein Kanal wartet auf einen Partnerkanal, der überhaupt nicht gestartet wurde.

Ein kanalübergreifender Satzsuchlauf kann wie folgt durchgeführt werden:

- Der Anwender wählt über HMI die Kanäle aus, die zusammenarbeiten müssen (Kanalgruppe).
- Aus der Kanalgruppe wählt der Anwender einen besonders wichtigen Kanal aus, für den er ein Suchziel explizit anwählen möchte (Zielkanal).
- Der HMI startet dann SERUPRO auf dem Zielkanal und "Self-Acting SERUPRO" auf dem Rest der Kanalgruppe.

Der Vorgang ist beendet, wenn **jeder** betroffene Kanal "seruproActive" gelöscht hat.

"Self-Acting SERUPRO" akzeptiert keinen Masterkanal auf einer anderen NCU.

Aktivierung

Die Aktivierung von "Self-Acting SERUPRO" erfolgt über HMI als Satzsuchlauf Start für Satzsuchlauf-Typ 5 für den Zielkanal "seruproMasterChan".

Für die vom Zielkanal gestarteten abhängigen Kanäle wird kein Suchziel angegeben.

10.8.6 Programmabschnitt für Wiederaufsetzen sperren

Programmierter Unterbrechungszeiger

Ist aufgrund von fertigungs- und/oder prozesstechnischen Gründen abzusehen, dass bei einem Programmabbruch innerhalb eines bestimmten Programmabschnitts kein Wiederaufsetzen möglich ist, kann dieser Programmabschnitt für den Zielsatz eines eventuellen Satzsuchlaufs gesperrt werden.

Erfolgt nach einem Programmabbruch innerhalb des für das Wiederaufsetzen gesperrten Programmabschnitts ein Satzsuchlauf auf die Unterbrechungsstelle, wird von der Steuerung dazu der letzte ausführbare Satz (Hauptlaufsatz) vor dem Beginn des gesperrten Bereichs als Zielsatz (Haltesatz) verwendet.

Programmierung

Syntax

```
IPTRLOCK ( )
```

Funktionalität

Markiert den Anfang des Programmabschnitts, ab dem Wiederaufsetzen gesperrt ist. Als Zielsatz für einen Satzsuchlauf mit "Wiederaufsetzen an der Unterbrechungsstelle" wird ab jetzt, bis zur Freigabe mit `IPTRUNLOCK`, der nächste ausführbare Satz (Hauptlaufsatz) verwendet, in dem `IPTRLOCK` aktiv wird. Dieser Satz wird im weiteren Verlauf als **Haltesatz** bezeichnet.

Wirksamkeit: modal

Syntax

`IPTRUNLOCK()`

Markiert das Ende des für ein Wiederaufsetzen gesperrten Programmabschnitts. Als Zielsatz für einen Satzsuchlauf mit "Wiederaufsetzen an der Unterbrechungsstelle" wird ab dem nächsten ausführbaren Satz (Hauptlaufsatz) in dem `IPTRLOCK` aktiv wird, wieder der aktuelle Satz verwendet. Dieser Satz wird im weiteren Verlauf als **Freigabesatz** bezeichnet.

Wirksamkeit: modal

Beispiel

Programmcode	Kommentar
...	
N010 IPTRLOCK()	; Gesperrter Bereich: Anfang
N020 R1=R1+1	
N030 G4 F1	; Haltesatz
...	; Gesperrter Bereich
N200 IPTRUNLOCK()	; Gesperrter Bereich: Ende
N220 R1=R1+1	
N230 G4 F1	; Freigabesatz
...	

Randbedingungen

- `IPTRLOCK` wirkt innerhalb eines Programms (*.MPF, *.SPF) maximal bis zum Programmende (M30, M17, RET). Mit dem Programmende wird implizit `IPTRUNLOCK` aktiv.
- Eine mehrfache Programmierung von `IPTRLOCK` innerhalb eines Programms wirkt nicht kumulativ. Mit der ersten Programmierung von `IPTRUNLOCK` innerhalb des Programms oder mit Erreichen des Programmendes, werden alle vorausgegangenen `IPTRLOCK`-Aufrufe abgeschlossen.
- Erfolgt ein Unterprogrammaufruf innerhalb eines gesperrten Bereiches, ist auch für diese und alle eventuell folgenden Unterprogrammebenen das Wiederaufsetzen gesperrt. Die Sperre kann innerhalb des aufgerufenen Unterprogramms auch durch explizites Programmieren von `IPTRUNLOCK` nicht aufgehoben werden.

Beispiel: Verschachtelung gesperrter Programmabschnitte in zwei Programmebenen

Durch die Aktivierung der Wiederaufsetzsperre in PROG_1, ist Wiederaufsetzen auch für PROG_2 und alle eventuell folgenden Programmebenen gesperrt.

Programmcode	Kommentar
PROC PROG_1	; Programm 1
...	
N010 IPTRLOCK()	
N020 R1=R1+1	
N030 G4 F1	; Haltesatz
...	; Gesperrter Bereich: Anfang
N040 PROG_2	; Gesperrter Bereich
...	; Gesperrter Bereich: Ende
N050 IPTRUNLOCK()	
N060 R2=R2+2	
N070 G4 F1	; Freigabesatz
...	

Programmcode	Kommentar
PROC PROG_2	; Programm 2
N210 IPTRLOCK()	; Unwirksam aufgrund von Programm 1
...	
N250 IPTRUNLOCK()	; Unwirksam aufgrund von Programm 1
...	
N280 RET	; Unwirksam aufgrund von Programm 1

Beispiel 3: Mehrfache Programmierung von IPTRLOCK

Programmcode	Kommentar
PROC PROG_1	; Programm 1
...	
N010 IPTRLOCK()	
N020 R1=R1+1	
N030 G4 F1	; Haltesatz
...	; Gesperrter Bereich: Anfang
N150 IPTRLOCK()	; Gesperrter Bereich
...	; Gesperrter Bereich
N250 IPTRLOCK()	; Gesperrter Bereich
...	; Gesperrter Bereich: Ende
N360 IPTRUNLOCK()	
N370 R2=R2+2	
N380 G4 F1	; Freigabesatz
...	

Systemvariable

Über die Systemvariable \$P_IPTRLOCK kann der Status des aktuellen Satzes ermittelt werden:

\$P_IPTRLOCK	Bedeutung
FALSE	Der aktuelle Satz befindet sich nicht innerhalb eines für das Wiederaufsetzen gesperrten Programmabschnitts
TRUE	Der aktuelle Satz befindet sich innerhalb eines für das Wiederaufsetzen gesperrten Programmabschnitts

Automatische funktionsspezifische Wiederaufsetzsperre

Für verschiedene Kopplungen kann das Ein/Ausschalten der Wiederaufsetzsperre kanalspezifisch automatisch mit dem Ein/Ausschalten der Kopplung erfolgen:

MD22680 \$MC_AUTO_IPTR_LOCK, Bit x

Bit	Wert	Bedeutung
0	Elektronisches Getriebe (EGON / EGOF)	
	1	Automatischen Wiederaufsetzsperre ist aktiv
	0	Automatischen Wiederaufsetzsperre ist nicht aktiv
1	Axiale Leitwertkopplung (LEADON / LEADOF)	
	1	Automatischen Wiederaufsetzsperre ist aktiv
	0	Automatischen Wiederaufsetzsperre ist nicht aktiv

Dieser Programmbereich beginnt mit dem letzten ausführbaren Satz **vor** dem Einschalten und endet mit dem Ausschalten.

Der automatische Unterbrechungszeiger wird bei Kopplungen, die über Synchronaktionen ein- bzw. ausgeschaltet wurden, nicht aktiv.

Beispiel: Axiale Leitwertkopplung automatisch suchunfähig erklären:

Programmcode	Kommentar
N100 G0 X100	
N200 EGON(Y, "NOC", X, 1, 1)	; Suchunfähiger Programmabschnitt beginnt.
N300 LEADON(A, B, 1)	
...	
N400 EGOF(Y)	
...	
N500 LEADOF(A, B)	; Suchunfähiger Programmabschnitt endet.
N600 G0 X200	

Ein Programmabbruch im suchunfähigen Programmabschnitt (N200 - N500) versorgt den Unterbrechungszeiger stets mit N100.

ACHTUNG

Unerwünschter Zustand durch Funktionsüberschneidungen

Bei einer Überschneidung der Funktionen "Programmierbarer Unterbrechungszeiger" und "Automatischer Unterbrechungszeiger" über Maschinendatum wählt die NC den größtmöglichen suchunfähigen Bereich.

Ein Programm kann eine Kopplung fast über die gesamte Laufzeit benötigen. Der automatische Unterbrechungszeiger würde damit immer auf den Programmanfang weisen und die Funktion SERUPRO würde de facto nutzlos.

10.8.7 Verhalten bei Power On, Betriebsartenwechsel und RESET

SERUPRO ist bei Power On inaktiv. Während SERUPRO ist der Betriebsartenwechsel erlaubt. RESET bricht SERUPRO ab, der intern gewählte Programmtest ist wieder ausgewählt. SERUPRO ist nicht mit anderen Satzsuchlauftypen kombinierbar.

10.8.8 Randbedingungen

10.8.8.1 STOPRE im Zielsatz

Alle satzübergreifenden Einstellungen erhält der STOPRE-Satz aus dem vorangegangenen Satz und kann damit Bedingungen vor dem eigentlichen Satz für die folgenden Fälle berücksichtigen:

- Aktuell bearbeitete Programmzeile mit dem Hauptlauf synchronisieren.
- Satzübergreifende Einstellungen für SERUPRO ableiten, um z. B. beim Anfahren von SERUPRO diese REPOS-Bewegung zu beeinflussen.

Beispiel: Durch Sollwertvorgabe einer X-Achse eine Z-Achse positionieren.

Wenn der Satz "G1 F100 Z=\$AA_IM[X]" interpretiert wird, so sorgt der vorangestellte STOPRE-Satz für die Synchronisation mit dem Hauptlauf. Damit wird über \$AA_IM der korrekte Sollwert der X-Achse gelesen, um die Z-Achse auf die gleiche Position zu fahren.

Beispiel: Externe Nullpunktverschiebung lesen und richtig einrechnen.

```
N10 G1 X1000 F100
N20 G1 X1000 F500
N30 G1 X1000 F1000
N40 G1 X1000 F5000
```

N50 SUPA G1 F100 X200	; externe Nullpunktverschiebung nach 200 fahren
N60 G0 X1000	
N70 ...	

Durch ein implizites STOPRE vor N50 kann der NC die aktuelle Nullpunktverschiebung lesen und richtig einrechnen.

Bei einem SERUPRO-Vorgang auf das Suchziel N50 wird im SERUPRO-Anfahren auf den impliziten STOPRE repositioniert und die Geschwindigkeit ermittelt sich aus N40 mit F5000.

Impliziter Vorlaufstopp

Situationen, in denen ein impliziter Vorlaufstopp abgesetzt wird:

1. In allen Sätzen in denen einer der folgenden Variablenzugriffe vorkommt:
 - Programmierung einer Systemvariablen, die mit \$A... beginnt
 - Programmierung einer redefinierte Variable mit den Attribut SYN R / SYN RW
2. Bei folgenden Befehlen:
 - Teileprogrammbefehl MEACALC, MEASURE
 - Programmierung von SUPA (Frames und Online-Korrekturen unterdrücken)
 - Programmierung CTABDEF (Beginn der Kurventabellendefinition)
 - Teileprogrammbefehl WRITE/DELETE (Datei schreiben/löschen)
 - vor dem ersten WRITE/DELETE-Befehl einer Sequenz solcher Befehle
 - Teileprogrammbefehl EXTCALL
 - Teileprogrammbefehl GETSELT, GETEXET
 - bei Werkzeugwechsel und aktiver Werkzeugfeinkorrektur FTOCON
3. Bei folgender Kommando-Bearbeitung:
 - Endbearbeitung eines Satzsuchlaufs Typ 1 ("Satzsuchlauf ohne Berechnung")
 - Endbearbeitung eines Satzsuchlaufs Typ 2 mit Berechnung ("Satzsuchlauf an Konturendpunkt")

10.8.8.2 SPOS im Zielsatz

Ist eine Spindel mit M3 / M4 programmiert und wird im Zielsatz nach SPOS umgeschaltet, ist am Ende des SERUPRO-Vorganges (Status "Suchziel gefunden") die Spindel nach SPOS umgeschaltet.

DB31, ... DBX84.5 = TRUE (aktive Spindelbetriebsart: Positionierbetrieb)

10.8.8.3 Fahren auf Festanschlag (FXS)

Beim Wiederanfahren an die Kontur (REPOS) wird die Funktion "Fahren auf Festanschlag" (FXS) automatisch wiederholt. Dabei wird jede Achse berücksichtigt. Als Moment wird das zuletzt vor dem Suchziel programmierte Moment angewendet.

Systemvariable

Die Systemvariablen für "Fahren auf Festanschlag" haben bei SERUPRO folgende Bedeutung:

- \$AA_FXS: Fortschritt der Programmsimulation
- \$VA_FXS: Realer Maschinenzustand

Die beiden Systemvariablen haben **außerhalb** der Funktion SERUPRO immer dieselben Werte.

ASUP

Für SERUPRO kann ein anwenderspezifisches ASUP aktiviert werden.

Literatur

Ausführliche Informationen zum Satzsuchlauf SERUPRO finden sich in Kapitel "Ausführliche Beschreibung (Seite 316)".

10.8.8.4 Fahren mit begrenztem Moment/Kraft (FOC)

Beim Wiederauffahren an die Kontur (REPOS) wird die Funktion "Fahren mit begrenztem Moment/Kraft" (FOC) automatisch wiederholt. Dabei wird jede Achse berücksichtigt. Als Moment wird das zuletzt vor dem Suchziel programmierte Moment angewendet.

Systemvariable

Die Systemvariablen für "Fahren mit begrenztem Moment/Kraft" haben bei SERUPRO folgende Bedeutung:

- \$AA_FOC: Fortschritt der Programmsimulation
- \$VA_FOC: Realer Maschinenzustand

Randbedingung

Ein wechselnder Momentenverlauf kann beim Wiederauffahren an die Kontur **nicht** realisiert werden.

Beispiel

Ein Programm fährt eine Achse X von Position 0 nach 100 und schaltet alle 20 Inkremente für jeweils 10 Inkremente "Fahren mit begrenztem Moment/Kraft" (FOC) ein. Dieser Momentenverlauf wird in der Regel mit satzweisem FOC erzeugt und kann beim Wiederauffahren an die Kontur (REPOS) nicht nachvollzogen werden. Stattdessen wird gemäß der letzten Programmierung die Achse X von 0 nach 100 mit oder ohne begrenztem Moment/Kraft fahren.

Literatur

Ausführliche Informationen zum Satzsuchlauf SERUPRO finden sich in Kapitel "Ausführliche Beschreibung (Seite 316)".

10.8.8.5 Synchronspindel

Synchronspindel ist simulierbar.

Der Synchronspindelbetrieb mit einer Leitspindel und beliebig vielen Folgespindeln ist in allen vorhandenen Kanälen mit SERUPRO simulierbar.

Weitere Informationen zu Synchronspindeln entnehmen Sie bitte:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Synchronspindel (S3)

10.8.8.6 Kopplungen und Master-Slave

Soll- und Istwert-Kopplungen

Der SERUPRO-Vorgang ist eine Programmsimulation im Modus Programmtest mit dem Soll- und Istwertkopplungen simulierbar sind.

Festlegungen für Simulation von EG

Für die Simulation von EG werden damit folgende Festlegungen getroffen:

1. Es wird immer mit Sollwertkopplung simuliert.
2. Sind nur einige Leitachsen, d. h. nicht alle Leitachsen unter SERUPRO, wird die Simulation mit Alarm 16952 "ResetClear/NoStart" abgebrochen. Dies kann bei kanalübergreifenden Kopplungen auftreten.
3. Achsen, die aus NC-Sicht nur einen Geber haben und von Extern bewegt werden, können nicht korrekt simuliert werden. Insbesondere dürfen diese Achsen nicht in Kopplungen einfließen.



VORSICHT

Fehlerhafte Simulation

Um Kopplungen korrekt simulieren zu können, müssen die Kopplungen vorher abgeschaltet werden.

Dies kann mit Maschinendatum MD10708 \$MA_SERUPRO_MASK erfolgen.

Festlegungen für Achskopplungen

Der SERUPRO-Vorgang simuliert Achskopplungen stets unter der Annahme von Sollwertkopplung. Damit werden Endpunkte für **alle** Achsen berechnet, die als Zielpunkte für SERUPRO-Anfahren benutzt werden. Zugleich ist die Kopplung bereits mit "Suchziel gefunden" aktiv. Der Weg vom aktuellen Punkt bis zum Endpunkt wird bei SERUPRO-Anfahren unter aktiver Kopplung durchgeführt.

LEADON

Für die Simulation von Axialen Leitwertkopplungen ergeben sich damit folgende Festlegungen:

1. Es wird immer mit Sollwertkopplung simuliert.
2. SERUPRO-Anfahren erfolgt mit aktiver Kopplung und einer überlagerten Bewegung der Folgeachse, um den simulierten Zielpunkt zu erreichen.

Die allein durch die Kopplung bewegte Folgeachse kann den Zielpunkt nicht immer erreichen. Es wird im SERUPRO-Anfahren eine Überlagerte lineare Bewegung für die Folgeachse berechnet, um den simulierten Punkt anzufahren!

Simulierten Zielpunkt für LEAD mit JOG erreichen

Zum Zeitpunkt "Suchziel gefunden" ist die Kopplung insbesondere für die JOG-Bewegungen bereits aktiv. Bei nicht erreichten Zielpunkt kann beim SERUPRO-Anfahren die Folgeachse mit aktiver Kopplung und einer überlagerten Bewegung auf den Zielpunkt verfahren werden.

Hinweis

Weitere Informationen zum Wiederaufahren von Achskopplungen siehe Kapitel "Wiederaufahren an die Kontur (REPOS) (Seite 521)".

Master-Slave

Nach Abschluss des Satzsuchlaufs kann ein System ASUP automatisch gestartet werden. In diesem hat der Anwender die Möglichkeit den Kopplungszustand und die zugehörigen Achspositionen nachträglich zu beeinflussen. Die dazu erforderlichen Informationen werden über folgende Systemvariable bereitgestellt:

Systemvariable	Beschreibung
\$P_SEARCH_MASLD[<Slaveachse>]	Positionsversatz zwischen Slave- und Masterachse zum Schließzeitpunkt der Kopplung.
\$AA_MASL_STAT[<Slaveachse>]	Aktueller Zustand einer Master-Slave Kopplung
\$P_SEARCH_MASLC[<Slaveachse>]	Status: der Zustand der Kopplung wurde im Satzsuchlauf verändert
Die Systemvariablen werden beim Einschalten der Kopplung mit MASLON gelöscht.	

Hinweis

Die gekoppelten Achsen müssen sich zum Zeitpunkt des Satzsuchlaufs im gleichen Kanal befinden.

Literatur

Weitere Informationen zur Master-Slave-Kopplung finden sich in:

Funktionshandbuch Sonderfunktionen; Kapitel "TE3: Drehzahl-/Drehmomentkopplung, Master-Slave"

Beispiel

- **System-ASUP**

- Pfad und Name: /_N_CMA_DIR/PROGEVENT.SPF
- Masterachse: X
- Slaveachse: Y

Programmcode	
PROG PROGEVENT	
N10 IF(((\$S_SEARCH_MASLC[Y] <> 0) AND (\$AA_MASL_STAT[Y] <> 0))	
N20 MASLOF(Y)	
N30 SUPA Y = \$AA_IM[X] - \$P_SEARCH_MASLD[Y]	
N40 MASLON(Y)	
N50 ENDIF	
N60 REPOSA	
...	
RET	

- **Maschinendaten**

Damit das ASUP automatisch startet, müssen folgende Maschinendaten gesetzt werden:

- NC-spezifische Maschinendaten:
 - MD11604 \$MN_ASUP_START_PRIO_LEVEL = 100
 - MD11450 \$MN_SEARCH_RUN_MODE = 'H02'
- Kanal-spezifisch für den Kanal in dem das ASUP gestartet wird, oder allgemein für alle Kanäle:
 - MD20105 \$MC_PROG_EVENT_IGN_REFP_LOCK, Bit<n> = TRUE
n: Für alle benötigten ereignisgesteuerten Programmaufrufe (Prog-Events)
 - MD20115 \$MC_IGNORE_REFP_LOCK_ASUP, Bit<n> = TRUE
n: Für alle **benötigten Anwender**-Interrupts

ACHTUNG**System-Interrupts**

Mit MD20115 \$MC_IGNORE_REFP_LOCK_ASUP, Bit 8 bis 31 werden die System-Interrupts freigegeben.

Durch Bit 8 / Interrupt 9 wird ein ASUP gestartet, welches Verfabrbewegungen beinhaltet.

Achskopplungen

- Beschleunigung der Abarbeitungsgeschwindigkeit und Leitachse und Folgeachsen in unterschiedlichen Kanälen
Bei einer Leitachse, deren Folgeachsen einem anderen Kanal als dem Kanal der Leitachse zugeordnet sind, wirkt die Einstellung zur Beschleunigung der Abarbeitungsgeschwindigkeit (MD22601 (Seite 531)\$MC_SERUPRO_SPEED_FACTOR) nicht:
- Mitschleppen
Die Funktion Mitschleppen (TRAILON) wird von SERUPRO unterstützt.
Weitere Informationen zu Mitschleppen mit TRAILON, TRAILOF finden sich in:
Literatur:
 - Funktionshandbuch Sonderfunktionen; Achskopplungen (M3)
 - Programmierhandbuch Arbeitvorbereitung; Kapitel "Achskopplungen"
- Gantry-Achsen
Die Funktion Gantry-Achsen wird von SERUPRO unterstützt.
Weitere Informationen zur Funktionalität von Gantry-Achsen finden sich in:
Literatur:
Funktionshandbuch Sonderfunktionen; Kapitel "G1: Gantry-Achsen"
- Tangentialsteuerung
Die Funktion tangentiales Nachführung einzelner Achsen wird von SERUPRO unterstützt.
Weitere Informationen zur Tangentialsteuerung finden sich in:
Literatur:
Funktionshandbuch Sonderfunktionen; Kapitel "T3: Tangentialsteuerung"

10.8.8.7 Achsfunktionen

SERUPRO Bedingungen

Bei Achsfreigabe, Autarken Achsvorgängen und Achstausch müssen die besonderen Bedingungen für SERUPRO berücksichtigt werden.

Achsfreigabe

Das axiale NST DB31, ... DBX3.7 ("Programmtest Achs-/Spindel Freigabe") beeinflusst die Achsfreigaben, wenn an die Maschine keine Reglerfreigabe gegeben werden soll oder kann und wirkt nur während Programmtest oder SERUPRO aktiv ist.

Es besteht die Möglichkeit über das Nahtstellensignal PLC → NC DB31, ... DBX3.7 (Programmtest Achs-/Spindel Freigabe) diese Freigabe zu geben. Fehlt bei

Programmtest oder SERUPRO die reale Reglerfreigabe, so hat das folgenden Einfluss auf Achsen/Spindeln:

- Sobald der simulierte Programmablauf eine Achse/Spindel verfahren möchte, wird die Meldung "Warten auf Achsfreigabe" oder "Warten auf Spindelfreigabe" angezeigt und die Simulation stoppt.
- Wird während einer simulierten Verfahrbewegung das NC/PLC-Nahtstellensignal DB31, ... DBX3.7 (Programmtest Achs-/Spindel Freigabe) wieder weggenommen, wird der Alarm 21612: "Kanal %1 Achse %2 NC/PLC-Nahtstellensignal 'Reglerfreigabe' während der Bewegung zurückgesetzt" ausgelöst.

Autarke Achsvorgänge

Autarke Einzelachsvorgänge sind vom PLC kontrollierte Achsen, die bei SERUPRO auch simuliert werden. Damit wird während SERUPRO wie im normalen Ablauf, die PLC die Kontrolle einer Achse übernehmen oder abgeben. Diese Achse kann gegebenenfalls auch über FC18 verfahren werden. Die PLC übernimmt die Kontrolle der Achse **vor** dem Anfahrtsatz und ist für die Positionierung dieser Achse verantwortlich. Dies gilt für alle Satzsuchlauftypen.

Weitere Informationen zu autarken Einzelachsvorgängen entnehmen Sie bitte:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2)

Achstausch

Problem: Ein Programm verfährt eine Achse und gibt sie vor dem Zielsatz mit WAITP(X) ab. Damit unterliegt X nicht dem REPOS und die Achse wird beim SERUPRO-Anfahren nicht berücksichtigt.

Über das Maschinendatum MD11470 \$MN_REPOS_MODE_MASK kann für SERUPRO-REPOS folgendes Verhalten erzielt werden:

Die neutralen Achsen werden als "Kommando-Achsen" im SERUPRO-REPOS verfahren. Die Achse interpoliert ohne Bahnzusammenhang, auch wenn sie zuletzt als Bahnachse programmiert worden war. In diesem Fall ergibt sich die Geschwindigkeit aus MD32060 \$MA_POS_AX_VELO. Nach dem SERUPRO-Anfahren ist diese Achse wieder neutral.

Neutrale Achsen, die trotzdem nicht repositioniert werden dürfen, müssen mit dem axialen NC/PLC-Nahtstellensignal "REPOSDELAY" beaufschlagt werden. Damit wird ihre REPOS Bewegung gelöscht.

Beispiel:

Nach SERUPRO wird via Technologiezyklen in der Synchronaktion eine Achse bewusst bewegt. Die Kommando-Achsen werden immer im Anfahrtsatz und nie im Zielsatz bewegt. Der

Zielsatz kann erst eingewechselt werden, wenn alle Kommando-Achsen zu Ende gefahren sind.

**VORSICHT****Die PLC kontrollierte Achse wird nicht repositioniert**

Achsen, die mit RELEASE(X) vor dem Zielsatz freigegeben worden sind, werden nicht repositioniert.

10.8.8.8 Getriebestufenwechsel

Abläufe

Der Getriebestufenwechsel (GSW) erfordert von der NC physikalische Bewegungen, um einen neuen Gang einlegen zu können.

Im SERUPRO-Vorgang ist ein Getriebestufenwechsel nicht erforderlich und wird wie folgt durchgeführt:

Manche Getriebe können nur NC-geführt gewechselt werden, da entweder die Achse pendeln, oder vorher eine bestimmte Position angefahren werden muss.

Der Getriebestufenwechsel kann im MD35035 \$MA_SPIND_FUNCTION_MASK mit den Bits 0 bis 2 selektiv für DryRun, Programmtest und SERUPRO unterdrückt werden.

Der GSW muss dann im REPOS nachgeholt werden und funktioniert auch, wenn die zugehörige Achse am Zielsatz im "Drehzahlsteuerbetrieb" sein soll. Andernfalls wird der automatische GWS mit einem Alarm abgelehnt, wenn im Teileprogramm zwischen dem GWS und dem Zielsatz die Achse u. a. in eine Transformation oder Kopplung aufgenommen worden war.

Hinweis

Weitere Informationen zum Getriebestufenwechsel bei DryRun, Programmtest und SERUPRO siehe Kapitel "S1: Spindeln (Seite 1295)".

10.8.8.9 Überlagerte Bewegung

Nur SERUPRO

Werden "Überlagerte Bewegungen" verwendet, so kann nur der Satzsuchlauf über Programmtest (SERUPRO) verwendet werden, da dabei die überlagerten Bewegungen im Hauptlauf entsprechend interpoliert werden. Dies gilt insbesondere für \$AA_OFF.

Geschwindigkeitsprofil statt maximale Achsgeschwindigkeit

Im Programmtest muss ein Geschwindigkeitsprofil verwendet werden, welches es erlaubt "Überlagerte Bewegungen" im Hauptlauf zu interpolieren. Es kann somit nicht mit der maximalen Achsgeschwindigkeit interpoliert werden.

Die Achsgeschwindigkeit wird im Modus "Probelaufvorschub" über SD42100 \$SC_DRY_RUN_FEED eingestellt.

Die Geschwindigkeit des SERUPRO-Vorgangs wird über MD22600 \$MC_SERUPRO_SPEED_MODE ausgewählt.

10.8.8.10 NC/PLC-Nahtstellensignale

REPOS-Verschiebung vorhanden

Hat sich während SERUPRO für eine Achse eine REPOS-Verschiebung ergeben, wird dies mit Abschluss des SERUPRO-Vorganges über die axiale NC/PLC-Nahtstelle angezeigt:

DB31, ...DBX70.0 == 1 (REPOS-Verschiebung vorhanden)

Gültigkeitsbereich der REPOS-Verschiebung

Mit dem Start eines SERUPRO-ASUP oder NC-Start zum Fortsetzen der Bearbeitung, wird die REPOS-Verschiebung ungültig:

DB31, ... DBX70.1 == 1 (REPOS-Verschiebung ungültig)

Zwischen dem Abschluss des SERUPRO-Vorgangs und NC-Start zum Fortsetzen der Bearbeitung, kann die Achse in der Betriebsart JOG manuell oder über das PLC-Anwenderprogramm mittels FC 18 verfahren werden. Wird dabei die REPOS-Verschiebung vollständig herausgefahren, wird das Nahtstellensignal zurückgesetzt.

10.8.8.11 Flexibilisierung der Grundeinstellung

Grundeinstellung / SERUPRO-Grundeinstellung

Mit dem Maschinendatum MD20112 \$MC_START_MODE_MASK wird die Grundstellung der Steuerung bei Teileprogrammstart bezüglich der G-Codes (insbesondere die aktuelle Ebene und einstellbare Nullpunktverschiebung), Werkzeuglängenkorrektur, Transformation und Achskopplungen festgelegt. Speziell für den SERUPRO-Vorgang besteht die Möglichkeit mit MD22620 \$MC_ENABLE_START_MODE_MASK_PRT eine zum normalen Teileprogrammstart abweichende Grundeinstellung zu wählen. Die neue Einstellung muss dabei hinterlegt sein im Maschinendatum:

MD22620 \$MC_START_MODE_MASK_PRT

Die Bedeutung der jeweiligen Bits von MD22620 ist identisch mit denen von MD20112 \$MC_START_MODE_MASK.

Beispiel:

Beim Teileprogrammstart bleibt die Synchronspindelkopplung zu Beginn des SERUPRO Vorgangs erhalten.

```
$MC_START_MODE_MASK = 'H400'  
$MC_START_MODE_MASK_PRT = 'H00'
```

```
; nicht projektierte Syn-  
chronspindelkopplung  
; wird ausgeschaltet  
; bleibt aktiv
```

<code>\$MC_ENABLE_START_MODE_MASK_PRT = 'H01'</code>	<code>; \$MC_START_MODE_MASK_PRT wird bei SERUPRO ausgewertet statt \$MC_START_MODE_MASK</code>
--	---

10.8.8.12 Kompressor-Funktionen (COMPON, COMPCURV, COMPCAD)

- Wenn bei Satzsuchlauf **Typ 2** oder **Typ 4** (Satzsuchlauf mit **Berechnung** an ...) der Zielsatz in einem Programmabschnitt liegt, in dem eine Kompressor-Funktion (COMPON, COMPCURV, COMPCAD) aktiv ist, werden beim Wiederanfahren an die Kontur Positionen auf der durch den **Kompressor** berechneten Bahn angefahren. Diese Positionen müssen nicht exakt mit den Positionen auf der im Teileprogramm programmierten Bahn übereinstimmen.
- Wenn bei der Komprimierung im Teileprogramm programmierte Sätze entfallen, werden diese Sätze im Satzsuchlauf nicht als Zielsatz gefunden ⇒ Alarm 15370 "Suchziel nicht

10.8.9 Systemvariable

Übersicht der für SERUPRO relevanten Systemvariablen:

Systemvariable	Bedeutung
\$AC_ASUP, Bit 20	ASUP Aktivierungsgrund: \$AC_ASUP, Bit 20 == 1 ⇒ System-ASUP aktiv, Grund: SERUPRO-Suchziel erreicht
\$AC_SERUPRO	SERUPRO-Status: \$AC_SERUPRO == 1 ⇒ SERUPRO ist aktiv
\$P_ISTEST	Programmtest-Status: SERUPRO aktiv ⇒ \$P_ISTEST == 1
\$P_SEARCHL	Zuletzt aktiver Satzsuchlaufftyp: \$P_SEARCHL == 5 von SERUPRO Anfang bis Reset oder Programmende
\$AC_REPOS_PATH_MODE	REPOS-Mode zum Wiederanfahren der Kontur nach erfolgtem SERUPRO

10.9 Programmbetrieb

Definition

Programmbetrieb liegt vor, wenn in den Betriebsarten AUTOMATIK bzw. MDA NC-Programme bzw. NC-Programmsätze abgearbeitet werden.

NC/PLC-Nahtstellensignale

Der Programmbetrieb kann vom PLC-Anwenderprogramm aus über BAG- und Kanal-spezifische NC/PLC-Nahtstellensignale beeinflusst werden bzw. gibt entsprechende Rückmeldungen an das PLC-Anwenderprogramm zurück.

Eine funktionspezifische Übersicht der NC/PLC-Nahtstellensignale findet sich in:

Literatur

- Funktionshandbuch Grundfunktionen; Kapitel "Z1: NC/PLC-Nahtstellensignale
- Funktionshandbuch Erweiterungsfunktionen; Kapitel "Z2: NC/PLC-Nahtstellensignale
- Funktionshandbuch Sonderfunktionen; Kapitel "Z3: NC/PLC-Nahtstellensignale

Eine Übersicht aller NC/PLC-Nahtstellensignale findet sich in:

Literatur

Listenhandbuch NC-Variable und Nahtstellensignale; Kapitel "Nahtstellensignale - Übersicht"

10.9.1 Grundstellungen

Für jeden Kanal können Grundstellungen im über kanalspezifische Maschinendaten vorgegeben werden. Diese Grundstellungen wirken u. a. auf die G-Gruppen sowie auf die Hilfsfunktionsausgabe.

Hilfsfunktionsausgabe

Die zeitliche Ausgabe der Hilfsfunktionen kann über die Maschinendaten AUXFU_x_SYNC_TYPE (MD22200, 22210, 22220, 22230, 22240, 22250, 22260), (Ausgabezeitpunkt der M, S, T, H, F, D, E-Funktionen) vorbestimmt werden. Nähere Erläuterungen dazu siehe Kapitel "H2: Hilfsfunktionsausgaben an PLC (Seite 399)".

G-Gruppen

Für jede der vorhandenen G-Gruppen kann über MD20150 \$MC_GCODE_RESET_VALUES (Löschstellung der G-Gruppen) eine Programmiergrundstellung vorgegeben werden. Diese Programmiergrundstellung ist automatisch beim Programmstart bzw. im Reset-Zustand wirksam, bis sie durch einen G-Befehl derselben G-Gruppe abgewählt wird.

Über das MD22510 \$MC_GCODE_GROUPS_TO_PLC (G-Befehle, die bei Satzwechsel/RESET an Nahtstelle NC-PLC ausgegeben werden) kann die Ausgabe der G-Befehle an die PLC-Nahtstelle aktiviert werden.

Literatur

Eine Auflistung der G-Gruppen mit den zugehörigen G-Befehlen finden sich in:

Programmierhandbuch Grundlagen

Grundkonfigurationen des NC-Sprachumfangs bei SINUMERIK solution line

Für SINUMERIK 840D sl können bestimmte Grundkonfigurationen des NC-Sprachumfangs über Maschinendaten projektierbar generiert werden. Damit wird speziell für den Anwender unter Berücksichtigung seiner benötigten Optionen und Funktionen der NC-Sprachumfang einheitlich auf Ihn zugeschnitten konfiguriert.

10.9.1.1 Maschinendaten

NC-Sprachumfang

Wie mit Sprachbefehlen nicht aktiver Optionen und Funktionen verfahren werden soll, wird mit folgendem Maschinendatum eingestellt:

MD10711 \$MN_NC_LANGUAGE_CONFIGURATION = <Wert>

Wert	Bedeutung
0	Alle Sprachbefehle sind bekannt. Erst bei der Ausführung wird erkannt, ob die benötigte Funktion aktiviert ist.
1	Alle Sprachbefehle sind bekannt. Sprachbefehle von nicht freigegebenen Optionen werden bei der Programminterpretation erkannt ⇒ Alarm 12553 "Option/Funktion ist nicht aktiv"
2	Es sind nur die Sprachbefehle der freigegebenen Optionen bekannt. Sprachbefehle von nicht freigegebene Optionen werden bei der Programminterpretation erkannt ⇒ Alarm 12550 "Name nicht definiert oder Option/Funktion nicht vorhanden"
3	Alle Sprachbefehle sind bekannt. Sprachbefehle nicht aktivierter Funktionen werden bei der Programminterpretation erkannt ⇒ Alarm 12553 "Option/Funktion ist nicht aktiv" Beispiel 1. Option für die Zylindermanteltransformation gesetzt 2. Die Zylindermanteltransformation ist im Maschinendatum MD24100 \$MC_TRAOF_TYPE_1 nicht aktiviert 3. Bereits die Programmierung des Befehls <code>TRACYL</code> führt zum Alarm 12553.
4	Es sind nur die Sprachbefehle der aktiven Funktionen bekannt. Sprachbefehle nicht aktiver Funktionen werden nicht erkannt ⇒ Alarm 12550 "Name nicht definiert oder Option/Funktion nicht vorhanden" Hinweis Ob der betreffende Sprachbefehle generell in der Siemens Sprache oder nur auf der entsprechenden Anlage nicht vorhanden ist, kann in diesem Fall nicht unterschieden werden.

10.9.1.2 Programmierung

Mit der Funktion "STRINGIS(...)" wird geprüft, ob die angegebene Zeichenkette als Element der NC-Programmiersprache im aktuellen Sprachumfang zur Verfügung steht.

Folgende Elemente der NC-Programmiersprache können geprüft werden:

- G-Befehle aller existierenden G-Gruppen
- DIN- oder NC-Adressen
- Funktionen
- Prozeduren
- Schlüsselworte
- Systemdaten wie Maschinendaten \$M... , Settingdaten \$S... oder Optionsdaten \$O...
- Systemvariable \$A... , \$V... , \$P...
- Rechenparameter R...
- Zyklennamen von aktivierten Zyklen

- GUD- und LUD-Variablen
- Makro-Namen
- Label-Namen

Definition

```
INT STRINGIS (STRING <Name>)
```

Syntax

```
<Rückgabewert> = STRINGIS (<Name>)
```

Bedeutung

STRINGIS () :	Prüffunktion mit Rückgabewert
<Name>:	Zu prüfende Zeichenkette

<Rückgabewert >:	Der Rückgabewert ist in den ersten drei Dezimalstellen codiert yxx		
	000	Der angegebene String ist kein Element des aktuellen Sprachumfangs ¹⁾	
	100	Der angegebene String ist Element des aktuellen Sprachumfangs, aber aktuell nicht programmierbar (Option/Funktion ist inaktiv)	
	2xx	Der angegebene String ist ein programmierbares Element aktuellen Sprachumfangs (Option/Funktion ist aktiv). Detailinformationen sind in der 1. und 2. Dezimalstelle xx enthalten:	
		xx	Bedeutung
		01	DIN-Adresse oder NC-Adresse ²⁾
		02	G-Befehl (z.B. G04, INVCW)
		03	Funktion mit Rückgabewert
		04	Funktion ohne Rückgabewert
		05	Schlüsselwort (z.B. DEFINE)
		06	Maschinen- (\$M...), Setting- (\$S...) oder Optionsdatum (\$O...)
		07	Systemparameter, z.B. Systemvariable (\$...) oder Rechenparameter (R...)
		08	Zyklus (Der Zyklus muss im NC geladen und die Zyklenprogramme aktiv sein ³⁾)
		09	GUD-Variable (Die GUD-Variable muss der in GUD-Definitionsdateien definierten und die GUD-Variablen aktiviert sein)
		10	Makroname (Das Makro muss in der Makro-Definitionsdateien definierten und Makros aktivierten sein) ⁴⁾
	11	LUD-Variable des aktuellen Teileprogramms	
	12	ISO G-Befehl (ISO Sprachmodus muss aktiv ist)	
400	Der angegebene String ist eine NC-Adresse, die weder als DIN-Adresse bzw. NC-Adresse (xx==01) noch als Makroname (xx==10) erkannt wurde und die nicht G oder R ist ²⁾		
y00	Keine spezifische Zuordnung möglich		

1) Steuerungs-abhängig ist unter Umständen nur eine Untermenge der Siemens NC-Sprachbefehle bekannt, z.B. SINUMERIK 802D sl. Auf diesen Steuerungen wird für Strings, die prinzipiell Siemens NC-Sprachbefehle sind, der Wert 0 zurückgegeben. Dieses Verhalten kann über MD10711 \$MN_NC_LANGUAGE_CONFIGURATION verändert werden. Bei MD10711 = 1 wird dann für Siemens NC-Sprachbefehle immer der Wert 100 zurückgegeben.

2) NC-Adressen sind folgende Buchstaben: A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Diese NC-Adressen können auch mit einer Adresserweiterung programmiert werden. Die Adresserweiterung kann bei der Prüfung mit STRINGIS angegeben werden. Beispiel: 201 == STRINGIS("A1").
Die Buchstaben: D, F, H, L, M, N, O, P, S, T sind NC-Adressen oder Hilfsfunktionen die anwenderdefiniert verwendet werden. Für sie wird immer der Wert 400 zurückgegeben. Beispiel: 400 == STRINGIS("D"). Diese NC-Adressen können bei der Prüfung mit STRINGIS nicht mit Adresserweiterung angegeben werden.
Beispiel: 000 == STRINGIS("M02"), aber 400 == STRINGIS("M").

3) Namen von Zyklenparametern können mit STRINGIS **nicht** geprüft werden.

4) Als Makro definierte Adress z.B. G, H, M, L werden als Makro identifiziert

Beispiele

In den folgenden Beispielen wird angenommen, dass das angegebene NC-Sprachelement, sofern nicht besonders vermerkt, in der Steuerung prinzipiell programmierbar ist.

1. Der String "T" ist als Hilfsfunktion definiert:

```
400 == STRINGIS ("T")  
000 == STRINGIS ("T3")
```

2. Der String "X" ist als Achse definiert:

```
201 == STRINGIS ("X")  
201 == STRINGIS ("X1")
```

3. Der String "A2" ist als NC-Adresse mit Erweiterung definiert:

```
201 == STRINGIS ("A")  
201 == STRINGIS ("A2")
```

4. Der String "INVCW" ist als benannter G-Befehl definiert:

```
202 == STRINGIS ("INVCW")
```

5. Der String "\$MC_GCODE_RESET_VALUES" ist als Maschinendatum definiert:

```
206 == STRINGIS (" $MC_GCODE_RESET_VALUES")
```

6. Der String "GETMDACT" ist eine NC-Sprachfunktion:

```
203 == STRINGIS ("GETMDACT ")
```

7. Der String "DEFINE" ist ein Schlüsselwort:

```
205 == STRINGIS ("DEFINE")
```

8. Der String "\$TC_DP3" ist ein Systemparameter (Werkzeuglängenkomponente):

```
207 == STRINGIS (" $TC_DP3")
```

9. Der String "\$TC_TP4" ist ein Systemparameter (Werkzeuggröße):

```
207 == STRINGIS (" $TC_TP4")
```

10. Der String "\$TC_MPP4" ist ein Systemparameter (Magazinplattzustand):

- Die Werkzeugmagazin-Verwaltung ist aktiv: 207 == STRINGIS (" \$TC_MPP4") ;
- Die Werkzeugmagazin-Verwaltung ist nicht aktiv: 000 == STRINGIS (" \$TC_MPP4")

Siehe auch unten Absatz: Werkzeugmagazin-Verwaltung.

11. Der String "MACHINERY_NAME" ist als GUD-Variable definiert:

```
209 == STRINGIS ("MACHINERY_NAME")
```

12. Der String "LONGMACRO" ist als Makro definiert:

```
210 == STRINGIS ("LONGMACRO")
```

13. Der String "MYVAR" ist als LUD-Variable definiert:

```
211 == STRINGIS ("MYVAR")
```

14. Der String "XYZ" ist kein im NC bekannter Befehl, GUD-Variable, Makro- oder Zyklus-Name:

```
000 == STRINGIS ("XYZ")
```


Randbedingungen

Werkzeugmagazin-Verwaltung

Ist die Funktion Werkzeugmagazin-Verwaltung nicht aktiv, liefert `STRINGIS()` für die Systemparameter der Werkzeugmagazin-Verwaltung, unabhängig vom Maschinendatum

- MD10711 \$MN_NC_LANGUAGE_CONFIGURATION

immer den Wert 000.

ISO Mode

Die Funktion "ISO Mode" ist aktiv, wenn:

Ist die Funktion "ISO Mode" aktiv, überprüft `STRINGIS()` den angegebenen String zuerst als SINUMERIK G-Befehl:

- MD18800 \$MN_MM_EXTERN_LANGUAGE (Aktivierung externer NC-Sprachen)
- MD10880 \$MN_MM_EXTERN_CNC_SYSTEM (zu adaptierendes Steuerungssystems)

Bei aktivem "ISO Mode" überprüft `STRINGIS()` die angegebene Zeichenkette zuerst auf Zugehörigkeit zum SINUMERIK G-Befehl.

Ist der String kein SINUMERIK G-Befehl, wird er anschließend auf Zugehörigkeit zum ISO G-Befehl überprüft.

Programmierte Umschaltung des ISO Mode

Programmierte Umschaltungen durch die Befehle `G290` (SINUMERIK Mode) und `G291` (ISO Mode) haben auf `STRINGIS()` keine Auswirkung.

10.9.2 Anwahl und Start eines NC-Programms

NC/PLC-Nahtstellensignale

Anwahl

Die Anwahl eines NC-Programms kann nur erfolgen, wenn sich der betreffende Kanal im Reset-Zustand befindet.

- DB21, ... DBX35.7 == 1 (Reset)

Start

Ein NC-Programm wird durch zwei verschiedene Ereignisse gestartet:

1. DB21, ... DBX7.1 = 1 (NC-Start), das Signal wird üblicherweise durch Betätigung der Taste "NC-Start" der Maschinensteuertafel ausgelöst.
2. Befehl `START` in einem NC-Programm eines anderen aktiven Kanals. Der Kanal muss sich dazu in der Betriebsart AUTOMATIK oder MDA und im Zustand "Reset" oder "unterbrochen" befinden
 - DB21, ... DBX35.7 == 1 (Reset)
 - DB21, ... DBX35.6 == 1 (unterbrochen)

Ausgangsbedingungen

Ein NC-Programm kann nur gestartet werden, wenn folgende Ausgangsbedingungen erfüllt sind.

- DB11 DBX4.4 == 1 (BAG-betriebsbereit)
- DB11 DBX0.7 == 0 (BAG-Reset)
- DB21, ... DBX1.7 == 0 (Programmtest aktivieren)
- DB21, ... DBX7.0 == 0 (NC-Start-Sperre)
- DB21, ... DBX7.2 == 0 (NC-Stop an Satzgrenze)
- DB21, ... DBX7.3 == 0 (NC-Stop)
- DB21, ... DBX7.4 == 0 (NC-Stop Achsen plus Spindel)
- DB21, ...:DBX7. 7 == 0 (Reset)
- DB10 DBX56.1 == 0 (Not-Halt)
- Es dürfen **keine** Achs- oder NC-spezifischen Alarm anstehen

Ausführung des Kommandos

Das Teileprogramm bzw. der Teileprogrammsatz wird automatisch abgearbeitet und die folgenden Nahtstellensignale werden gesetzt:

- DB21, ... DBX35.5 (Kanalzustand aktiv)
- DB21, ... DBX35.0 (Programmzustand läuft)

Das Programm wird solange bearbeitet, bis das Programmende erreicht bzw. der Kanal durch ein STOP- oder RESET-Kommando unterbrochen bzw. abgebrochen wird.

Literatur

Eine ausführliche Beschreibung der Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

Alarmer

Das START-Kommando ist unter bestimmten Umständen nicht wirksam und es tritt dann einer der folgenden Alarmer auf:

- 10200 "Kein NC-Start bei aktivem Alarm erlaubt"
- 10202 "Kein NC-Start bei aktivem Kommando erlaubt"
- 10203 "Kein NC-Start bei nicht referenzierten Achsen erlaubt"

Literatur:

Diagnosehandbuch Alarmer

10.9.3 Programmunterbrechung

NC/PLC-Nahtstellensignale

Voraussetzungen

Eine Programmunterbrechung wird nur ausgeführt, wenn der Kanal und das NC-Programm aktiv sind:

- DB21, ... D35.5 == 1 (Kanal: aktiv)
- DB21, ... D35.0 == 1 (Programm: läuft)

Programmunterbrechung

Über folgende Ereignisse kann eine Programmbearbeitung unterbrochen werden:

- DB21, ... DBX7.2 == 1 (NC-Stop an Satzgrenze)
- DB21, ... DBX7.3 == 1 (NC-Stop)
- DB21, ... DBX7.4 == 1 (NC-Stop Achsen plus Spindel)
- DB21, ... DBX2.0 == 1 (Einzelsatz)
- Programmierter Befehl M00 oder M01 im abgearbeiteten NC-Programm

Anschließend befindet sich der Kanal und das NC-Programm im Zustand "unterbrochen":

- DB21, ... D35.6 == 1 (Kanal unterbrochen)
- DB21, ... D35.3 == 1 (Programm unterbrochen)

Literatur

Eine ausführliche Beschreibung der Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

Ablauf

Mit dem Erkennen einer Programmunterbrechung werden folgende Aktionen durchgeführt:

- Unterbrechung der Programmbearbeitung:
 - An der nächsten Satzgrenze bei folgenden Ereignissen: "NC-Stop an Satzgrenze", M00, M01 oder Einzelsatz
 - Sofort: Alle anderen Ereignisse
- Die verfahrenden Achsen des Kanals, werden über Bremsrampe stillgesetzt. Das Abbremsen der Achsen kann sich dabei über mehrere Sätze erstrecken.
- Der Satzzeiger zeigt den aktuellen Satz an der Unterbrechungsstelle an.
- Die bis zur Unterbrechungsstelle noch nicht ausgegebenen Hilfsfunktionen werden nicht mehr ausgegeben.

Möglichkeiten im unterbrochenen Zustand

Während einer Teileprogrammunterbrechung können diverse Funktionen im Kanal ausgeführt werden z.B.:

- **Überspeichern**
Literatur
Bedienhandbuch für SINUMERIK Operate, Kapitel "Werkstück bearbeiten" > "Überspeichern"
- **Satzsuchlauf**
Literatur
Funktionshandbuch Grundfunktionen, Kapitel "BAG, Kanal, Programmbetrieb, Reset-Verhalten (K1)" > "Satzsuchlauf" bzw. "Satzsuchlauf Typ 5 SERUPRO"
- **Wiederanfahren an die Kontur (REPOS)**
Literatur
Funktionshandbuch Grundfunktionen, Kapitel "BAG, Kanal, Programmbetrieb, Reset-Verhalten (K1)" > "Satzsuchlauf Typ 5 SERUPRO" > "REPOS" > "Wiederanfahren an Kontur mit gesteuertem REPOS"
- **Orientierter Werkzeugrückzug**
Literatur
 - Programmierhandbuch Arbeitsvorbereitung, Kapitel "Werkzeugkorrekturen"
 - Funktionsbeschreibung Grundfunktionen, Kapitel "Werkzeugkorrekturen (W1)" > "Orientierbare Werkzeugträger" > ""
- **ASUP** (siehe Kapitel "Asynchrone Unterprogramme (ASUPs) (Seite 591)")
- **DRF-Funktion, Verschieben des Werkstücknullpunktes**
Literatur
Funktionshandbuch Erweiterungsfunktionen; manuelles Verfahren und manuelles Verfahren per Handrad (H1)
- **Fortsetzen des unterbrochenen NC-Programms**
 - Befehl `START` aus einem anderen Kanal
Literatur
Programmieranleitung Arbeitsvorbereitung, Kapitel "Flexible NC-Programmierung" > "Programmkoordinierung (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)"
 - NC/PLC-Nahtstelle:
DB21, ... DBX7.1 = 1 (NC-Start)

10.9.4 Kanal-Reset

Funktion

Durch einen Kanal-Reset wird ein in der Betriebsart Automatik ein in Abarbeitung befindliches NC-Programm bzw. in der Betriebsart MDA ein in Abarbeitung befindlicher Programmsatz abgebrochen.

Das NC-Programm bzw. der Programmsatz kann an der Unterbrechungsstelle nicht mehr fortgesetzt werden. Nach Abschluss des Kanal-Resets befinden sich alle Achsen und Spindeln des Kanals in Zustand "Genauhalt" (Ausnahme: Nachführbetrieb).

Ablauf

Durch NC-Reset werden folgende Aktionen im Kanal durchgeführt:

- Die Programmaufbereitung wird sofort gestoppt.
- Achsen und Spindeln werden über ihre parametrisierten Bremsrampen bis zum Stillstand abgebremst.
- Die noch nicht ausgegebenen Hilfsfunktionen des aktuellen Satzes werden nicht mehr an die PLC ausgegeben.
- Der Satzzeiger wird auf den Anfang des angewählten NC-Programms zurückgesetzt.
- Alle Reset-Alarme des Kanals werden aus der Anzeige gelöscht.

Regeln

- Ein Kanal-Reset wird in jedem Kanalzustand ausgeführt.
- Ein Kanal-Reset wird von keinem anderen Kommando abgebrochen.

NC/PLC-Nahtstellensignale

Anforderung: Kanal-Reset

Ein Kanal-Reset wird über folgende NC/PLC-Nahtstellensignale angefordert:

- DB21, ... DBX7.7 = 1 (Reset)

Anforderung: BAG-Reset

Ein BAG-Reset löst Kanal-Reset in allen Kanälen einer BAG aus.

Ein BAG-Reset wird über folgende NC/PLC-Nahtstellensignale angefordert:

- DB11, ... DBX0.7 = 1 (BAG-Reset)

Rückmeldung: Kanal-Reset abgeschlossen

- DB21, ... DBX35.7 == 1 (Kanalzustand Reset)

Rückmeldung: BAG-Reset abgeschlossen

- DB11, ... DBX6.5 == 1 (BAG reseted)

Literatur

Eine ausführliche Beschreibung der Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

10.9.5 Programmzustand

Für jeden Kanal wird der Zustand des angewählten NC-Programms in der Nahtstelle angezeigt.

In den Betriebsarten AUTOMATIK und MDA können alle Programmzustände auftreten. In allen anderen Betriebsarten oder Maschinenfunktionen ist der Programmzustand abgebrochen oder unterbrochen.

NC/PLC-Nahtstellensignale

An der NC/PLC-Nahtstelle (DB21, ...) werden folgende Programmzustände angezeigt:

- DB21, ... DBX35.4 ("abgebrochen")
- DB21, ... DBX35.3 ("unterbrochen")
- DB21, ... DBX35.2 ("angehalten")
- DB21, ... DBX35.1 ("warten")
- DB21, ... DBX35.0 ("läuft")

Eine ausführliche Beschreibung der Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

Zustandsänderungen

Der Programmzustand wird durch Kommandos, NC/PLC-Nahtstellensignale und Alarme beeinflusst. Ausgehend vom Programmzustand "läuft" zeigt die Tabelle den Folgezustand aufgrund des jeweiligen Ereignisses.

Ausgangszustand des Programms: "läuft"					
Ereignis	Folgezustand des Programms				
	ab-gebr.	un-terbr.	angeh.	warten	läuft
DB21, ... DBX7.7 (Reset)	x				
DB21, ... DBX7.3 (NC-Stop)			x		
DB21, ... DBX7.2 (NC-Stop an Satzgrenze)			x		
DB21, ... DBX7.4 (NC-Stop Achsen u. Spindeln)			x		
DB21, ... DBX6.1 (Einlesesperre)					x
DB21, ... DBX6.0 (Vorschub Halt)					x
DB21, ... DBX12.3 / 16.3 / 20.3 (Vorschub Halt, GeoAchse 1 / 2 / 3)					x
DB21, ... DBB4; Vorschuboverride = 0%					x
DB31, ... DBX4.3 (Vorschub Halt/Spindel Halt)					x
DB21, ... DBX194.2 / DBX197.6 (M02 / M30 im Satz)	x				
DB21, ... DBX194.0 / DBX194.1 (M00 / M01 im Satz)			x		
DB21, ... DBX0.4 (Einzelsatz)			x		
DB21, ... DBX6.2 (Restweg löschen)					x
Hilfsfunktion an PLC ausgegeben, aber noch nicht quittiert			x		
WAIT-Befehl im Programm				x	
Alarm mit Systemreaktion "NOREADY"		x			

10.9.6 Kanalzustand

Für jeden Kanal wird in allen Betriebsarten der Kanalzustand an der NC/PLC-Nahtstelle (DB21, ...) angezeigt..

NC/PLC-Nahtstellensignale

An der NC/PLC-Nahtstelle werden die Kanalzustände durch folgende Signale angezeigt:

- DB21, ... DBX35.7 ("Reset")
- DB21, ... DBX35.6 ("unterbrochen")
- DB21, ... DBX35.5 ("aktiv")

Eine ausführliche Beschreibung der Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

Zustandsänderungen

Der Kanalzustand wird durch Kommandos und NC/PLC-Nahtstellensignale beeinflusst. Ausgehend vom Kanalzustand "aktiv" zeigt die Tabelle den Folgezustand aufgrund des jeweiligen Ereignisses.

Ausgangszustand des Kanals: "aktiv"			
Ereignis	Folgezustand des Kanals		
	"Reset"	"unterbrochen"	"aktiv"
DB21, ... DBX7.7 (Reset)	x		
DB21, ... DBX7.3 (NC-Stop)		x	
DB21, ... DBX7.2 (NC-Stop an Satzgrenze)		x	
DB21, ... DBX7.4 (NC-Stop Achsen u. Spindeln)		x	
DB21, ... DBX6.1 (Einlesesperre)			x
DB21, ... DBX6.0 (Vorschub Halt)			x
DB21, ... DBX12/16/20.3 (Vorschub Halt, GeoAchse 1/2/3)			x
DB21, ... DBB4; Vorschuboverride = 0%			x
DB31, ... DBX4.3 (Vorschub Halt/Spindel Halt)			x
DB21, ... DBX194.2/DBX197.6 (M02/M30 im Satz)	x		
DB21, ... DBX194.0/DBX194.1 (M00/M01 im Satz)		x	
DB21, ... DBX0.4 (Einzelsatz)		x	
DB21, ... DBX6.2 (Restweg löschen)			x
Hilfsfunktion an PLC ausgegeben, aber noch nicht quittiert			x
WAIT-Befehl im Programm			x

Der Kanalzustand "aktiv" wird erreicht, wenn ein NC-Programm oder NC-Programmsatz abgearbeitet wird oder wenn in der Betriebsart JOG Achsen verfahren werden.

10.9.7 Reaktionen auf Bedienungs- und Programmaktionen

Zustandsübergänge

Die folgende Tabelle zeigt die Kanal- und Programmzustände, die nach bestimmten Bedien- oder Programmaktionen auftreten.

Im linken Teil der Tabelle sind die verschiedenen Zustände des Kanals und des im Kanal angewählten Programms und die aktive Betriebsart aufgeführt.

Im rechten Teil der Tabelle sind die Bedien-/Programmaktionen und der Folge-Zustand aufgeführt.

Zu- stand	Kanalzustand			Programmzustand					Betriebsart			Bedien- oder Programmaktion => Folge-Zustand
	R	U	A	R	U	S	W	A	A	M	J	
1		x						x	x			RESET ⇒ 4
2		x						x		x		RESET ⇒ 5
3		x						x			x	RESET ⇒ 6
4	x			x					x			NC-Start ⇒ 13; BA-Wechsel ⇒ 5 bzw. 6
5	x			x						x		NC-Start ⇒ 14; BA-Wechsel ⇒ 4 bzw. 6
6	x			x							x	Richtungstaste ⇒ 15; BA-Wechsel ⇒ 4 bzw. 5
7		x		x						x		NC-Start ⇒ 14
8		x		x							x	NC-Start ⇒ 15
9		x			x				x			NC-Start ⇒ 13; BA-Wechsel ⇒ 10 bzw. 11
10		x			x					x		NC-Start ⇒ 16; BA-Wechsel ⇒ 9 bzw. 11
11		x			x						x	Richtungstaste ⇒ 17; BA-Wechsel ⇒ 9 bzw. 10
12		x				x			x			NC-Start ⇒ 13; BA-Wechsel ⇒ 10 bzw. 11
13			x					x	x			NC-Stop ⇒ 12
14			x	x						x		NC-Stop ⇒ 7; bei Satzende ⇒ 5
15			x	x							x	NC-Stop ⇒ 8; bei JOG-Ende ⇒ 6
16			x		x					x		NC-Stop ⇒ 10; bei Satzende ⇒ 10
17			x		x						x	NC-Stop ⇒ 11; bei JOG-Ende ⇒ 11
18			x				x		x			RESET ⇒ 4; warten auf anderen Kanal ⇒ 18

Kanalzustand

R → abgebrochen

U → unterbrochen

A → läuft

Programmzustand

R → abgebrochen

U → unterbrochen

S → angehalten

W → wartet

A → läuft

Betriebsarten

A → Automatik

M → MDA

J → JOG

10.9.8 Zeitdiagramm-Beispiel für einen Programmablauf

Programmcode

```
N10 G01 G90 X100 M3 S1000 F1000 M170
N20 M0
```

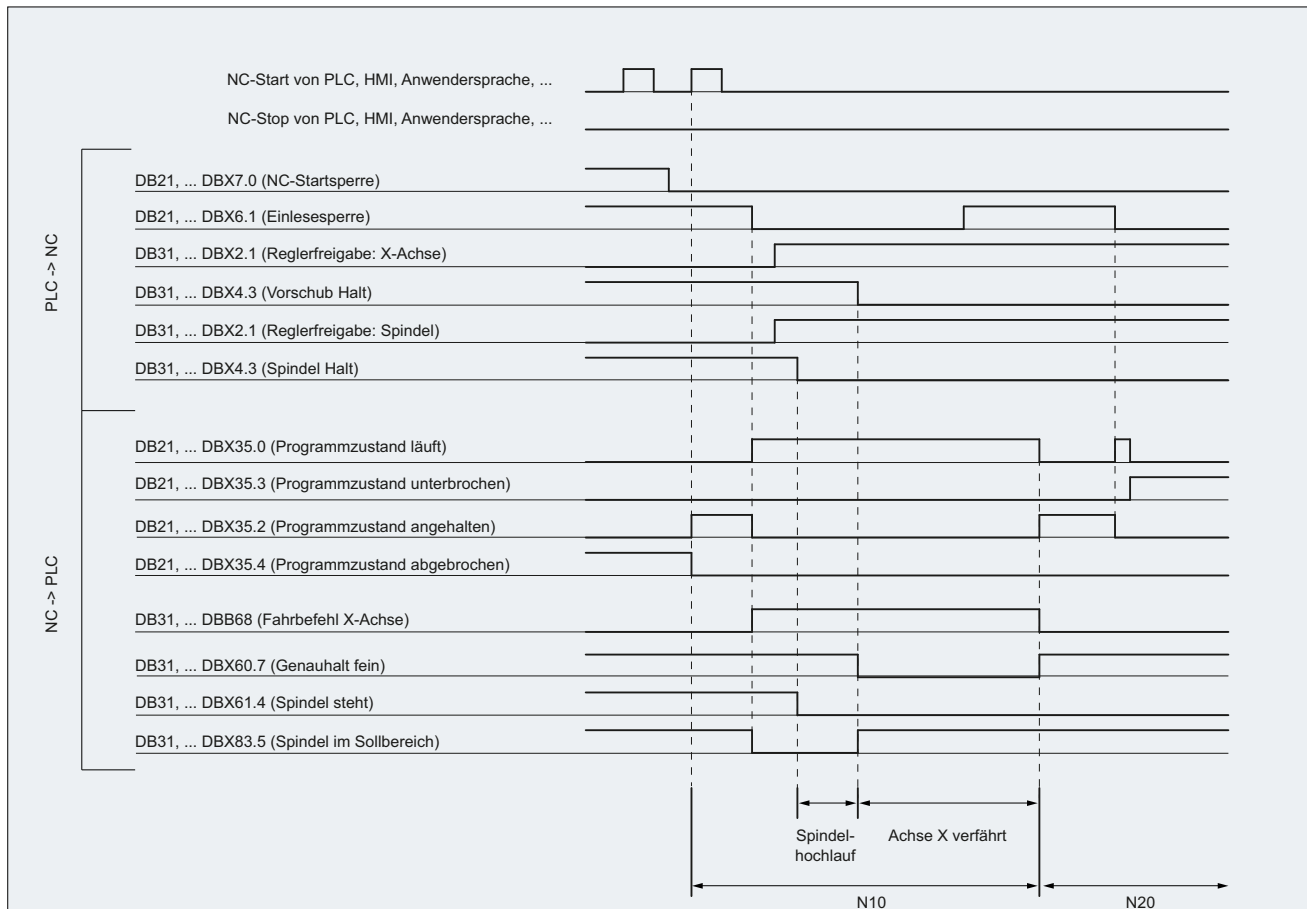


Bild 10-6 Signalverläufe während des Programms

10.9.9 Programmsprünge

10.9.9.1 Rücksprung auf Programmanfang (GOTOS)

Funktion

Mit der Funktion "Rücksprung auf den Programmanfang" kann aus einem Teileprogramm heraus an den Anfang des Programms zurückgesprungen werden. Das Programm wird daraufhin erneut abgearbeitet.

Im Vergleich zur Funktion "Programmsprünge auf Sprungmarken", mit der ebenfalls ein wiederholtes Abarbeiten eines Teileprogramms realisiert werden kann, bietet die Funktion "Rücksprung auf den Programmanfang" folgende Vorteile:

- Die Programmierung einer Sprungmarke am Programmanfang ist nicht notwendig.
- Der Programmneustart ist steuerbar durch das NC/PLC-Nahtstellensignal: DB21, ... DBX384.0 (Programmverzweigung steuern)
- Der Timer für die Programmlaufzeit kann beim Programmneustart auf "0" zurückgesetzt werden.
- Die Timer für die Werkstückzählung können beim Programmneustart um den Wert "1" erhöht werden.

Anwendungsbeispiel

Die Funktion findet Anwendung, wenn die Bearbeitung nachfolgender Werkstücke durch einen automatischen Programmneustart erfolgen soll, z. B. bei Drehmaschinen mit Stangenlader/-wechsler.

NC/PLC-Nahtstellensignale

Der Rücksprung erfolgt nur, wenn das folgende NC/PLC-Nahtstellensignal gesetzt ist:

DB21, ... DBX384.0 (Programmverzweigung steuern) = 1

Parametrierung

Programmlaufzeit

Die Laufzeit des angewählten NC-Programms wird in der Systemvariablen \$AC_CYCLE_TIME gespeichert. Beim Start eines neuen Programms wird die Systemvariable automatisch auf "0" zurückgesetzt (siehe Kapitel " Programmlaufzeit (Seite 691) ")

Über das folgende Maschinendatum kann eingestellt werden, dass die Systemvariable \$AC_CYCLE_TIME auch bei einem Programmneustart durch die Funktion "Rücksprung auf den Programmanfang" auf "0" zurückgesetzt wird:

MD27860 \$MC_PROCESSTIMER_MODE.Bit 8 = <Wert> (Aktivierung der Programmlaufzeit-Messung)

Bit	Wert	Bedeutung
8	0	\$AC_CYCLE_TIME wird durch die Funktion "Rücksprung auf den Programmanfang" nicht auf "0" zurückgesetzt.
	1	\$AC_CYCLE_TIME wird durch die Funktion "Rücksprung auf den Programmanfang" auf "0" zurückgesetzt.

Hinweis

Damit die Einstellung von Bit 8 wirksam werden kann, muss die Messung der aktuellen Programmlaufzeit aktiv sein (MD27860 Bit 1 = 1).

Werkstückzählung

Nachdem das Teileprogrammende (M02 / M30) erreicht ist, werden die aktivierten Werkstückzähler (\$AC_TOTAL_PARTS / \$AC_ACTUAL_PARTS / \$AC_SPECIAL_PARTS) um den Wert "1" erhöht (siehe Kapitel "Werkstückzähler (Seite 702)").

Über das folgende Maschinendatum kann eingestellt werden, dass die aktivierten Werkstückzähler auch bei einem Programmneustart durch die Funktion "Rücksprung auf den Programmanfang" hochgezählt werden:

MD27880 \$MC_PART_COUNTER.Bit <n> = <Wert> (Aktivierung der Werkstück-Zähler)

Bit	Wert	Bedeutung: Bei einem Programmneustart durch die Funktion "Rücksprung auf den Programmanfang" wird der Werkstückzähler:
7	0	\$AC_TOTAL_PARTS nicht hochgezählt.
	1	\$AC_TOTAL_PARTS hochgezählt.
11	0	\$AC_ACTUAL_PARTS nicht hochgezählt.
	1	\$AC_ACTUAL_PARTS hochgezählt.
15	0	\$AC_SPECIAL_PARTS nicht hochgezählt.
	1	\$AC_SPECIAL_PARTS hochgezählt.

Programmierung

Syntax

GOTOS

Bedeutung

GOTOS:	Rücksprung auf den Anfang des aktuellen Programms	
	Vorlaufstopp:	ja
	Wirksamkeit:	Satzweise

Beispiel

Programmierung	Kommentar
N10 ...	; Programmanfang
...	
IF ...	
N100 GOTOS	Rücksprung auf den Programmanfang (N10)
ENDIF	
...	
RET	

10.9.10 Programmteilmiederholungen

10.9.10.1 Programmierung

Die Programmteilmiederholung ermöglicht die Wiederholung von Programmteilen innerhalb eines NC-Programms.

Die zu wiederholenden Programmzeilen bzw. Programmbereiche werden durch Sprungmarken (Labels) gekennzeichnet.

Hinweis

Sprungmarken (Labels)

Sprungmarken stehen immer am Anfang eines Satzes. Wenn eine Programmnummer vorhanden ist, steht die Sprungmarke unmittelbar nach der Satznummer.

Für die Benennung von Sprungmarken gelten folgende Regeln:

- Anzahl an Zeichen:
 - mindestens 2
 - höchstens 32
 - Erlaubte Zeichen sind:
 - Buchstaben
 - Ziffern
 - Unterstriche
 - Die ersten beiden Zeichen müssen Buchstaben oder Unterstriche sein.
 - Nach dem Namen der Sprungmarke folgt ein Doppelpunkt (":").
-

Syntax

1. REPEATB: Einzelne Programmzeile wiederholen

```
<Sprungmarke>: ...  
...  
REPEATB <Sprungmarke> P=<n>
```

2. REPEAT + Sprungmarke: Programmbereich zwischen Sprungmarke und REPEAT-Anweisung wiederholen

```
<Sprungmarke>: ...  
...  
REPEAT <Sprungmarke> P=<n>
```

3. REPEAT + Sprungmarke_1 + Sprungmarke_2: Bereich zwischen zwei Sprungmarken wiederholen

```
<Start-Sprungmarke>: ...  
...  
<End-Sprungmarke>: ...
```

```
...
REPEAT <Start-Sprungmarke> <End-Sprungmarke> P=<n>
```

Hinweis

Die REPEAT-Anweisung mit den beiden Sprungmarken zu klammern, ist nicht möglich. Wird die <Start-Sprungmarke> vor der REPEAT-Anweisung gefunden und wird die <End-Sprungmarke> nicht vor der REPEAT-Anweisung erreicht, dann wird die Wiederholung zwischen <Start-Sprungmarke> und REPEAT-Anweisung durchgeführt.

4. REPEAT + Sprungmarke + ENDLABEL: Bereich zwischen Sprungmarke und ENDLABEL wiederholen:

```
<Sprungmarke>: ...
...
ENDLABEL: ...
...
REPEAT <Sprungmarke> P=<n>
```

Hinweis

Die REPEAT-Anweisung mit der <Sprungmarke> und dem ENDLABEL zu klammern, ist nicht möglich. Wird die <Sprungmarke> vor der REPEAT-Anweisung gefunden und wird ENDLABEL nicht vor der REPEAT-Anweisung erreicht, dann wird die Wiederholung zwischen <Sprungmarke> und REPEAT-Anweisung durchgeführt.

Bedeutung

REPEATB:	Befehl zum Wiederholen einer Programmzeile
REPEAT:	Befehl zum Wiederholen eines Programmbereichs
<Sprungmarke>:	<p>Die <Sprungmarke> kennzeichnet:</p> <ul style="list-style-type: none"> • REPEATB: Zu wiederholende Programmzeile • REPEAT: Beginn des zu wiederholenden Programmbereichs <p>Die mit der <Sprungmarke> gekennzeichnete Programmzeile kann vor oder nach der REPEAT-/REPEATB-Anweisung stehen. Gesucht wird zunächst in Richtung Programmanfang. Wird die Sprungmarke in dieser Richtung nicht gefunden, wird in Richtung Programmende gesucht.</p> <p>Ausnahme: Wenn der Programmbereich zwischen Sprungmarke und REPEAT-Anweisung wiederholt werden soll (siehe 2. unter Syntax), muss die mit der <Sprungmarke> gekennzeichnete Programmzeile vor der REPEAT-Anweisung stehen, da in diesem Fall nur in Richtung Programmanfang gesucht wird. Enthält die Zeile mit der <Sprungmarke> weitere Anweisungen, werden diese bei jeder Wiederholung erneut ausgeführt.</p>

ENDLABEL:	Schlüsselwort, welches das Ende eines zu wiederholenden Programmbe- reichs markiert Enthält die Zeile mit dem ENDLABEL weitere Anweisungen, werden diese bei jeder Wiederholung erneut ausgeführt. ENDLABEL kann mehrfach im Programm verwendet werden.	
P:	Adresse zur Angabe der Wiederholungsanzahl Hinweis: Ist kein P=<n> angegeben, wird der zu wiederholende Programmteil genau einmal wiederholt.	
<n>:	Wiederholungsanzahl	
	Typ:	INT
	Der zu wiederholende Programmteil wird <n> mal wiederholt. Nach der letz- ten Wiederholung wird das Programm mit der auf den REPEAT-/REPEATB- Befehl folgenden Programmzeile fortgesetzt.	

Beispiele

Beispiel 1: Einzelne Programmzeile wiederholen

Programmcode	Kommentar
N10 POSITION1: X10 Y20	
N20 POSITION2: CYCLE(0,,9,8)	; Positionszyklus
N30 ...	
N40 REPEATB POSITION1 P=5	; Führe Satz N10 fünfmal aus.
N50 REPEATB POSITION2	; Führe Satz N20 einmal aus.
N60 ...	
N70 M30	

Beispiel 2: Programmbereich zwischen Sprungmarke und REPEAT-Anweisung wiederholen

Programmcode	Kommentar
N5 R10=15	
N10 Begin: R10=R10+1	; Anfang des Programmabschnitts1
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	
N70 Z=10+R10	
N80 REPEAT BEGIN P=4	; Führe Bereich N10 bis N70 viermal aus.
N90 Z10	
N100 M30	

Beispiel 3: Bereich zwischen zwei Sprungmarken wiederholen

Programmcode	Kommentar
N5 R10=15	

Programmcode	Kommentar
N10 Begin: R10=R10+1	; Anfang des Programmabschnitts
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	
N70 END: Z=10	; Ende des Programmabschnitts
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	; Führe Bereich N10 bis N70 dreimal aus.
N110 Z10	
N120 M30	

Beispiel 4: Bereich zwischen Sprungmarke und ENDLABEL wiederholen

Programmcode	Kommentar
N10 G1 F300 Z-10	
N20 BEGIN1:	; Anfang Programmabschnitt 1
N30 X10	
N40 Y10	
N50 BEGIN2:	; Anfang Programmabschnitt 2
N60 X20	
N70 Y30	
N80 ENDLABEL: Z10	
N90 X0 Y0 Z0	
N100 Z-10	
N110 BEGIN3: X20	; Anfang Programmabschnitt 3
N120 Y30	
N130 REPEAT BEGIN3 P=3	; Führe Bereich N110 bis N120 dreimal aus.
N140 REPEAT BEGIN2 P=2	; Führe Bereich N50 bis N80 zweimal aus.
N150 M100	
N160 REPEAT BEGIN1 P=2	; Führe Bereich N20 bis N80 zweimal aus.
N170 Z10	
N180 X0 Y0	
N190 M30	

Beispiel 5: Fräsbearbeitung, Bohrposition mit verschiedenen Technologien bearbeiten

Programmcode	Kommentar
N10 ZENTRIERBOHRER()	; Zentrierbohrer einwechseln.
N20 POS_1:	; Anfang Programmabschnitt 1; Bohrpositionen 1
N30 X1 Y1	
N40 X2	
N50 Y2	
N60 X3 Y3	

Programmcode	Kommentar
N70 ENDLABEL:	
N80 POS_2:	; Anfang Programmabschnitt 2; Bohrpositionen 2
N90 X10 Y5	
N100 X9 Y-5	
N110 X3 Y3	
N120 ENDLABEL:	; Ende der Programmabschnitte 1 und 2
N130 BOHRER()	; Bohrzyklus
N140 GEWINDE(6)	; Gewindezyklus
N150 REPEAT POS_1	; Wiederhole Programmabschnitt ab POS_1 einmal bis ENDLABEL.
N160 BOHRER()	; Bohrzyklus.
N170 GEWINDE(8)	; Gewindezyklus.
N180 REPEAT POS_2	; Wiederhole Programmabschnitt ab POS_2 einmal bis ENDLABEL.
N190 M30	

Weitere Informationen

- Programmteilwiederholung kann geschachtelt aufgerufen werden. Jeder Aufruf belegt eine Unterprogrammebene.
- Ist während der Bearbeitung einer Programmteilwiederholung M17 oder RET programmiert, wird die Programmteilwiederholung abgebrochen. Das Programm wird mit dem auf die REPEAT-Zeile folgenden Satz fortgesetzt.
- In der aktuellen Programm-Anzeige wird die Programmteilwiederholung als eigene Unterprogrammebene angezeigt.
- Wird während der Programmteil-Bearbeitung Ebenenabbruch ausgelöst, wird das Programm nach dem Aufruf der Programmteilmbearbeitung fortgesetzt.
Beispiel:

Programmcode	Kommentar
N5 R10=15	
N10 BEGIN: R10=R10+1	; Breite
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; Ebenenabbruch
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Programmbearbeitung fortsetzen.
N130 M30	

- Kontrollstrukturen und Programmteilwiederholung können kombiniert genutzt werden. Es sollte jedoch keine Überschneidungen geben. Eine Programmteilwiederholung sollte innerhalb eines Kontrollstruktur-Zweigs liegen bzw. eine Kontrollstruktur innerhalb einer Programmteilwiederholung.
- Bei der Mischung von Sprüngen und Programmteilwiederholung werden die Sätze rein sequentiell abgearbeitet. Erfolgt z. B. ein Sprung aus einer Programmteilwiederholung, wird solange bearbeitet, bis das programmierte Programmteilende gefunden wird.
Beispiel:

Programmcode

```

N10 G1 F300 Z-10
N20 BEGIN1:
N30 X=10
N40 Y=10
N50 GOTOF BEGIN2
N60 ENDLABEL:
N70 BEGIN2:
N80 X20
N90 Y30
N100 ENDLABEL: Z10
N110 X0 Y0 Z0
N120 Z-10
N130 REPEAT BEGIN1 P=2
N140 Z10
N150 X0 Y0
N160 M30

```

Hinweis

Die REPEAT-Anweisung sollte hinter den Verfahrssätzen stehen.

10.9.11 Ereignisgesteuerter Programmaufruf (PROG_EVENT)

10.9.11.1 Funktion

Funktion

Die Funktion "Ereignisgesteuerte Programmaufrufe" (PROG_EVENT) startet beim Auftreten eines ausgewählten Ereignisses in der NC ein anwenderspezifisches NC-Programm (PROG_EVENT-Programm).

Anwendungsbeispiele

Grundeinstellung von Funktionen oder Initialisierungen von System- oder Anwendervariablen.

Ereignisse

Die Auswahl der auslösenden Ereignisse erfolgt über das Maschinendatum MD20108 \$MC_PROG_EVENT_MASK (siehe Kapitel "Parametrierung (Seite 577)").

PROG_EVENT-Programm

Über das Maschinendatum MD11620 \$MN_PROG_EVENT_NAME wird der Name des PROG_EVENT-Programms eingestellt (siehe Kapitel "Parametrierung (Seite 577)").

Das PROG_EVENT-Programm wird in dem Kanal ausgeführt, in dem das Ereignis aufgetreten ist.

Das PROG_EVENT-Programm wird mit niedrigster Priorität ausgeführt und kann daher von einem Anwender-ASUP unterbrochen werden.

Bearbeitungsabläufe

Bearbeitungsablauf bei Aktivierung durch Ereignis: Programm-Start

Ausgangszustand:

Kanal: Reset-Zustand
Betriebsart: AUTOMATIK (optional: Überspeichern aktiv)
 MDA
 TEACH IN

1. Im Kanal erfolgt ein Programm-Start
2. Initialisierungssequenz mit Auswertung von:
 - MD20112 \$MC_START_MODE_MASK
3. Impliziter Aufruf des PROG_EVENT-Programms als Unterprogramm
4. Bearbeitung des Datenteils des Hauptprogramms
5. Bearbeitung des Programmteils des Hauptprogramms

Bearbeitungsablauf bei Aktivierung durch Ereignis: Programm-Ende

Ausgangszustand

Kanal: Aktiv
Betriebsart: AUTOMATIK (optional: Überspeichern aktiv)
 MDA
 TEACH IN

1. Im Kanal wird im ausgeführten Programm der Programmende-Satz eingewechselt.
2. Programm-Ende wird ausgeführt, Auswertung folgender Maschinendaten:
 - MD20110 \$MC_RESET_MODE_MASK
 - MD20150 \$MC_GCODE_RESET_VALUES
 - MD20152 \$MC_GCODE_RESET_MODE

3. Impliziter Aufruf des PROG_EVENT-Programms als ASUP
4. Reset wird im Kanal ausgeführt, Auswertung folgender Maschinendaten:
 - MD20110 \$MC_RESET_MODE_MASK
 - MD20150 \$MC_GCODE_RESET_VALUES
 - MD20152 \$MC_GCODE_RESET_MODE

Ablauf bei Aktivierung durch Ereignis: Kanal-Reset

Ausgangszustand:

Kanal: beliebig

Betriebsart: beliebig

1. Steuerung aktiviert Reset-Sequenz mit Auswertung der Maschinendaten:
 - MD20110 \$MC_RESET_MODE_MASK
 - MD20150 \$MC_GCODE_RESET_VALUES
 - MD20152 \$MC_GCODE_RESET_MODE
2. Impliziter Aufruf des PROG_EVENT-Programms als ASUP
3. Steuerung aktiviert Reset-Sequenz mit Auswertung der Maschinendaten:
 - MD20110 \$MC_RESET_MODE_MASK
 - MD20150 \$MC_GCODE_RESET_VALUES
 - MD20152 \$MC_GCODE_RESET_MODE

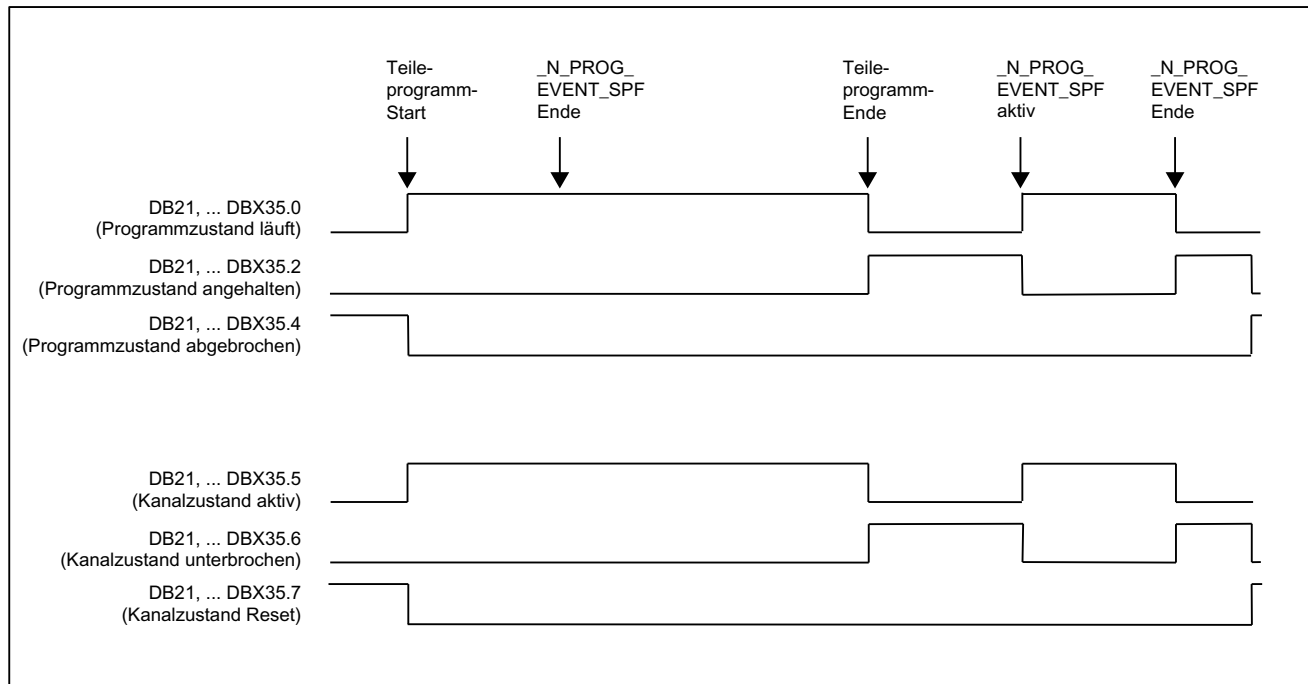
Ablauf bei Aktivierung durch Ereignis: Hochlauf der NC

1. Steuerung aktiviert nach Hochlauf Reset-Sequenz mit Auswertung der Maschinendaten:
 - MD20110 \$MC_RESET_MODE_MASK
 - MD20150 \$MC_GCODE_RESET_VALUES
 - MD20152 \$MC_GCODE_RESET_MODE
2. Impliziter Aufruf des PROG_EVENT-Programms als ASUP
3. Steuerung aktiviert Reset-Sequenz mit Auswertung der Maschinendaten:
 - MD20110 \$MC_RESET_MODE_MASK
 - MD20150 \$MC_GCODE_RESET_VALUES
 - MD20152 \$MC_GCODE_RESET_MODE

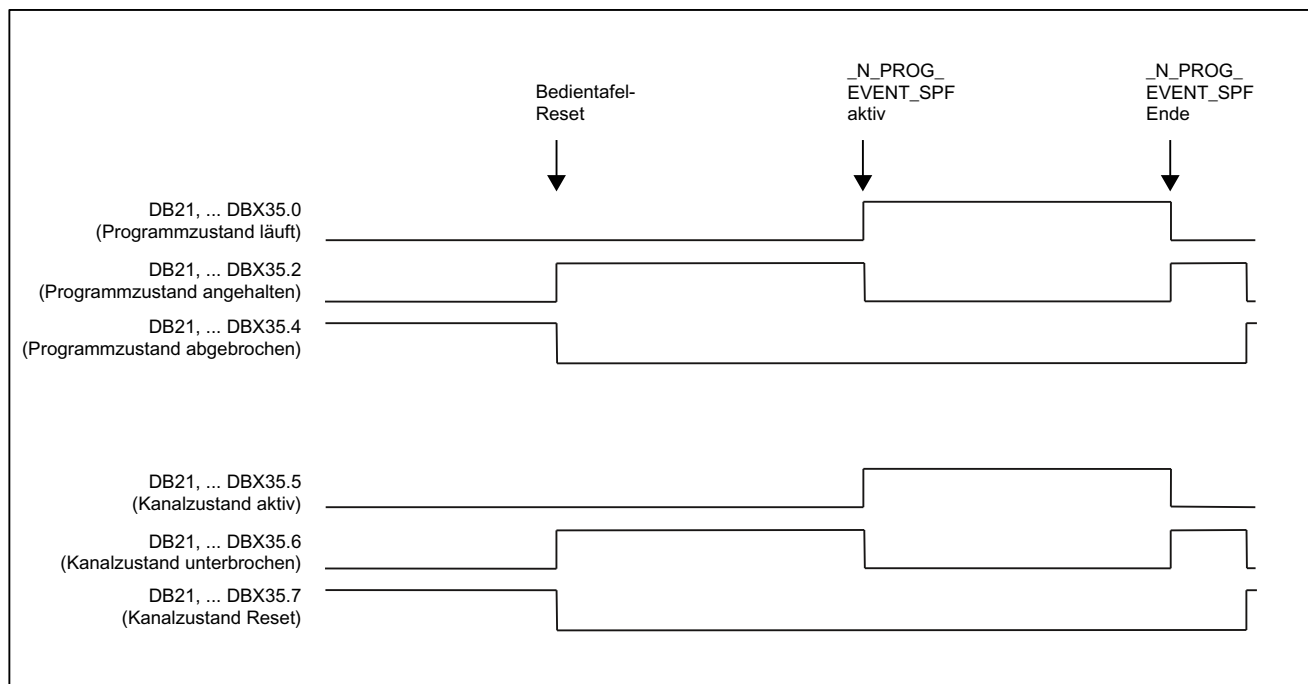
NC/PLC-Nahtstellensignale: Verlauf von "Programmzustand" und "Kanalzustand"

Die nachfolgenden Diagramme zeigen den Verlauf der verschiedenen NC/PLC-Nahtstellensignale für "Programmzustand" und "Kanalzustand" während des Ablaufs des ereignisgesteuerten Programmaufrufs.

Signalverlauf bei Aktivierung durch Programm-Start und Programm-Ende



Signalverlauf bei Aktivierung durch Kanal-Reset



NC/PLC-Nahtstellensignale: DB21, ... DBX35.4 (Programmzustand abgebrochen) und DB21, ... DBX35.7 (Kanalzustand Reset)

- Die Nahtstellensignale werden erst gesetzt, wenn das PROG_EVENT-Programm wieder beendet ist.
- Die Nahtstellensignale werden nicht gesetzt zwischen:
 - Programm-Ende und dem Start des PROG_EVENT-Programms
 - Kanal-Reset und dem Start des PROG_EVENT-Programms

NC/PLC-Nahtstellensignale: DB21, ... DBX376.0 ... 4 (Auslösende Ereignisse)

Das auslösende Ereignis wird über die Nahtstellensignale DB21, ... DBX376.0 ... 4 angezeigt:

Bit	Wert	Ereignis
0	1	Programm-Start aus Kanalzustand "Reset"
1	1	Programm-Ende
2	1	Kanal-Reset
3	1	Hochlauf der NC
4	1	1. Programm-Start nach Satzsuchlauf (siehe "Automatischer Start eines ASUP nach Satzsuchlauf (Seite 510)")

Wird das PROG_EVENT-Programm beendet oder mit Kanal-Reset abgebrochen, werden die Nahtstellensignale wieder gelöscht.

Die Nahtstellensignale stehen mindestens für die Dauer eines PLC-Zyklus an.

10.9.11.2 Parametrierung**Ereignissauswahl**

Durch welche Ereignisse das PROG_EVENT-Programm gestartet wird, wird kanalspezifisch eingestellt über das Maschinendatum:

MD20108 \$MC_PROG_EVENT_MASK.Bit <n> = 1

Bit	<Wert>	Bedeutung: Ereignis
0	1	Teileprogramm-Start
1	1	Teileprogramm-Ende
2	1	Kanal-Reset
3	1	Hochlauf
5	1	Safety-Programm im Hochlauf

Hinweis

MD20108 \$MC_PROG_EVENT_MASK wird in der Simulation nicht ausgewertet.

PROG_EVENT-Programm

Das PROG_EVENT-Programm (Voreinstellung: _N_PROG_EVENT_SPF) muss geladen und freigegeben sein.

Defaulteinstellung

Defaultmäßig wird bei einem Ereignis das Anwenderprogramm _N_CMA_DIR/_N_PROG_EVENT_SPF ausgeführt.

Das PROG_EVENT-Programm muss geladen und freigegeben sein.

Anwenderspezifische Einstellung

Soll bei einem Ereignis ein anderes PROG_EVENT-Programm als das aus der Defaulteinstellung ausgeführt werden, muss der Name die NC-Programms in folgendes Maschinendatum eingetragen werden:

MD11620 \$MN_PROG_EVENT_NAME = <Programmname>

Suchpfad

Das PROG_EVENT-Programm muss sich in einem der folgenden Zyklen-Verzeichnis befinden. Beim Eintreten des parametrisierten Ereignisses wird dann folgender Suchpfad durchlaufen:

1. /_N_CUS_DIR/ (Anwender-Zyklen)
2. /_N_CMA_DIR/ (Hersteller-Zyklen)
3. /_N_CST_DIR/ (Standard-Zyklen)

Das erste gefundene Programm mit dem in MD11620 \$MN_PROG_EVENT_NAME angegebenen Namen wird ausgeführt.

Hinweis

- Der angegebene Programmname wird syntaktisch wie bei einem Unterprogrammname geprüft, d. h. die ersten beiden Zeichen müssen Buchstaben oder Unterstriche sein (keine Ziffern). Präfix (_N_) und Suffix (_SPF) des Programmnamens werden, wenn nicht angegeben, automatisch ergänzt.
- Es greifen die Schutzmechanismen, wie sie allgemein für Zyklen aktivierbar sind (Schutzstufen für Schreiben, Lesen usw.).

Verhalten beim Starten eines Anwender-ASUP

Das Verhalten der Funktion "Ereignisgesteuerter Programmaufruf" beim Starten eines Anwender-ASUP aus dem Kanalzustand Reset kann kanalspezifisch eingestellt werden mit dem Maschinendatum:

MD20109 \$MC_PROG_EVENT_MASK_PROPERTIES.Bit 0 = <Wert>

Bit	Wert	Bedeutung
0	0	Bei Eintreten eines mit MD20108 parametrisierten Ereignisses wird das PROG_EVENT-Programms gestartet.
	1	Bei Eintreten eines mit MD20108 parametrisierten Ereignisses wird das PROG_EVENT-Programms nicht gestartet.

Verhalten bei gesetzter Einzelsatzbearbeitung

Das Verhalten der Funktion "Ereignisgesteuerter Programmaufruf" bei gesetzter Einzelsatzbearbeitung kann für jedes auslösende Ereignis kanalspezifisch mit folgendem Maschinendatum eingestellt werden:

MD20106 \$MC_PROG_EVENT_IGN_SINGLEBLOCK.Bit <n> = <Wert>

Bit	Wert	Bedeutung: Im PROG_EVENT-Programm wird die Einzelsatzbearbeitung bei Ereignis:
0	0	Teileprogramm-Start: wirksam
	1	Teileprogramm-Start: unterdrückt
1	0	Teileprogramm-Ende: wirksam
	1	Teileprogramm-Ende: unterdrückt
2	0	Kanal-Reset: wirksam
	1	Kanal-Reset: unterdrückt
3	0	Hochlauf: wirksam
	1	Hochlauf: unterdrückt

Ist die Einzelsatzbearbeitung unterdrückt, wird das PROG_EVENT-Programm ohne Unterbrechung abgearbeitet.

Hinweis

- MD20106 \$MC_PROG_EVENT_IGN_SINGLEBLOCK wirkt für alle Einzelsatzbearbeitungstypen.
- Die Einzelsatzbearbeitung im PROG_EVENT-Programm kann durch folgende Einstellung grundsätzlich ausgeschaltet werden:
MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK, Bit 0 = 1 (Einzelsatz-Stopp verhindern)
Die Einstellungen im MD20106 \$MC_PROG_EVENT_IGN_SINGLEBLOCK sind dann unwirksam.

Verhalten bei aktiver Einlesesperre

Das Verhalten der Funktion "Ereignisgesteuerter Programmaufruf" bei gesetzter Einlesesperre (DB21, ... DBX6.1 = 1) kann für jedes auslösende Ereignis kanalspezifisch mit folgendem Maschinendatum eingestellt werden:

MD20107 \$MC_PROG_EVENT_IGN_INHIBIT.Bit <n> = <Wert>

Bit	Wert	Bedeutung: Im PROG_EVENT-Programm wird die Einlesesperre bei Ereignis:
0	0	Teileprogramm-Start: wirksam Wird das PROG_EVENT-Programm mit Befehl RET beendet, wird dafür kein ausführbaren Satz erzeugt
	1	Teileprogramm-Start: unterdrückt Wird das PROG_EVENT-Programm mit Befehl RET beendet, wird dafür ein ausführbaren Satz, analog zu M17, erzeugt.
1	0	Teileprogramm-Ende: wirksam
	1	Teileprogramm-Ende: unterdrückt

Bit	Wert	Bedeutung: Im PROG_EVENT-Programm wird die Einlesesperre bei Ereignis:
2	0	Kanal-Reset: wirksam
	1	Kanal-Reset: unterdrückt
3	0	Hochlauf: wirksam
	1	Hochlauf: unterdrückt

Aktualisierung der Anzeige des Programm- und Kanalzustandes unterdrücken

Um ein Flackern der Anzeige des Programm- und Kanalzustandes an der Bedienoberfläche des HMI zu vermeiden, kann für die Ausführung des im Allgemeinen kurzen PROG_EVENT-Programms die Aktualisierung der Anzeige unterdrückt werden. In der Anzeige bleibt dann der Programm- und Kanalzustand vor Aktivierung des PROG_EVENT-Programms anstehen.

MD20192 \$MC_PROG_EVENT_IGN_PROG_STATE.Bit <n> = <Wert>

Bit	Wert	Bedeutung: Während der Abarbeitung des PROG_EVENT-Programms wird die Aktualisierung der Anzeige des Programm- und Kanalzustands bei Ereignis:
1	0	Teileprogramm-Ende: Nicht unterdrückt
	1	Teileprogramm-Ende: Unterdrückt
2	0	Kanal-Reset: Nicht unterdrückt
	1	Kanal-Reset: Unterdrückt
3	0	Hochlauf: Nicht unterdrückt
	1	Hochlauf: Unterdrückt

Hinweis

Die Systemvariablen \$AC_STAT und \$AC_PROG bleiben von dieser Funktion unbeeinflusst, d. h. im laufenden ereignisgesteuerten Anwenderprogramm werden \$AC_STAT auf "aktiv" und \$AC_PROG auf "läuft" gesetzt sein.

Die NC/PLC-Nahtstellensignale DB21, ... DBX35.0-7 ("Programmzustand ..." und "Kanalzustand ...") bleiben ebenfalls unbeeinflusst.

Verhalten bei DB21, ... DBX7.2 / 3 / 4 (NC-Stop ...)

Das Verhalten der Funktion "Ereignisgesteuerter Programmaufruf" bei "NC-Stop", "NC-Stop an Satzgrenze" und "NC-Stop Achsen plus Spindel" kann für die auslösenden Ereignisse Teileprogramm-Ende, Kanal-Reset und Hochlauf kanalspezifisch mit folgendem Maschinendatum eingestellt werden ::

MD20193 \$MC_PROG_EVENT_IGN_STOP.Bit <n> = <Wert>

Bit	Wert	Bedeutung: Das PROG_EVENT-Programm wird bei NC-Stop und Ereignis
1	0	Teileprogramm-Ende: angehalten/verhindert
	1	Teileprogramm-Ende: vollständig abgearbeitet
2	0	Kanal-Reset: angehalten/verhindert
	1	Kanal-Reset: vollständig abgearbeitet

Bit	Wert	Bedeutung: Das PROG_EVENT-Programm wird bei NC-Stop und Ereignis
3	0	Hochlauf: angehalten/verhindert
	1	Hochlauf: vollständig abgearbeitet

Anwendungsbeispiel

Ein vom Bediener durch Betätigen der Taste NC-Stop während Kanal-Reset oder Hochlauf verursachter Flankenwechsel des Nahtstellensignals DB21, ... DBX7.3 (NC-Stop) wird bei der Abarbeitung des PROG_EVENT-Programms ignoriert und so ein unerwünschtes Anhalten an der Maschine verhindert werden.

Hinweis

Eine Programmierung von `DELAYFSTON` / `DELAYFSTOF` im PROG_EVENT-Programm kann das mit MD20193 einstellbare Verhalten nicht ersetzen. NC-Stop kann vor Ausführung des Befehls `DELAYFSTON` noch eine Unterbrechung bewirken.

10.9.11.3 Programmierung

PROG_EVENT-Programm

Programmende

Soll das Anwenderprogramm durch Teileprogramm-Start aktiviert werden, muss folgendes beachtet werden:

- Das Anwenderprogramm muss mit `M17` bzw. `RET` beendet werden.
- Ein Rücksprung mittels `REPOS`-Befehl ist nicht zulässig.

Satzanzeige

Die Anzeige in der aktuellen Satzanzeige kann durch das `DISPLOF`-Attribut in der `PROC`-Anweisung unterdrückt werden.

Kommunikation zum PLC-Anwenderprogramm

Über Anwender-M-Funktionen, die im PROG_EVENT-Programm geschrieben werden, kann das PLC-Anwenderprogramm, z.B. über den Bearbeitungsstatus des PROG_EVENT-Programms, informiert werden.

Systemvariable

Abfrage des auslösenden Ereignisses

Das auslösenden Ereignis kann im PROG_EVENT-Programm mit folgender Systemvariablen abgefragt werden:

<Wert> = \$P_PROG_EVENT (Ereignisgesteuerter Programmaufruf aktiv)

Wert	Bedeutung: Aktivierung durch
1	Teileprogramm-Start
2	Teileprogramm-Ende
3	Kanal-Reset
4	Hochlauf
5	Nach Ausgabe des letzten Aktionssatzes nach Satzsuchlauf (siehe "Automatischer Start eines ASUP nach Satzsuchlauf (Seite 510)")

Abfrage des aktuellen Kanals

Der Kanal in dem das PROG_EVENT-Programm abgearbeitet wird, kann mit folgender Systemvariablen ermittelt werden:

<Wert> = \$P_CHANNO (Abfrage der aktuellen Kanalnummer)

Hinweis

Das PROG_EVENT-Programm wird in dem Kanal abgearbeitet, in dem das auslösende Ereignis aufgetreten ist.

Hochlauf ist ein Ereignis, das in allen Kanälen gleichzeitig auftritt.

10.9.11.4 Randbedingungen

Not-Halt / Alarm

Steht bei Kanal-Reset oder nach Hochlauf ein Not-Halt oder ein BAG / NC-spezifischer Alarm an, wird das PROG_EVENT-Programm erst nach der Not-Halt- bzw. Fehler-Quittierung in allen betroffenen Kanälen bearbeitet.

Hinweis

Das Ereignis "Hochlauf" tritt in allen Kanälen gleichzeitig auf.

10.9.11.5 Beispiele

Beispiel 1: Aufruf des PROG_EVENT-Programms bei allen Ereignissen

Parametrierung

MD20108 \$MC_PROG_EVENT_MASK = 'H0F'

Aufruf von _N_PROG_EVENT_SPF
bei:

- Teileprogramm-Start
- Teileprogramm-Ende
- Kanal-Reset
- Hochlauf

Programmierung

Programmcode	Kommentar
PROC PROG_EVENT DISPLOF	
; Bearbeitung für Teileprogramm-Start	
IF (\$P_PROG_EVENT==1)	
MY_GUD_VAR=0	; GUD-Variable initialisieren.
RET	
ENDIF	
; Bearbeitung für Teileprogramm-Ende und Kanal-Reset	
IF (\$P_PROG_EVENT==2) OR	
(\$P_PROG_EVENT==3)	
DRFOF	; DRF-Verschiebungen ausschalten.
IF \$MC_CHAN_NAME=="CHAN1"	
CANCEL(2)	; Modale Synchronaktion 2 löschen.
ENDIF	
RET	
ENDIF	
; Bearbeitung für Hochlauf	
IF (\$P_PROG_EVENT==4)	
IF \$MC_CHAN_NAME=="CHAN1"	
IDS=1 EVERY \$A_INA[1]>5.0 DO	
\$A_OUT[1]=1	
ENDIF	
RET	
ENDIF	
RET	

Beispiel 2: Aufruf des PROG_EVENT-Programms bei Kanal-Reset**Parametrierung**

MD20108 \$MC_PROG_EVENT_MASK = 'H04'

Aufruf von _N_PROG_EVENT_SPF
bei:

- Bedientafel-Reset

Programmierung

Programmcode	Kommentar
PROC PROG_EVENT DISPLOF	
N10 DRFOF	; DRF-Verschiebungen ausschalten
N20 M17	

Beispiel 3: Initialisierung der Funktion**Parametrierung**

Maschinendatenbelegung, Ausschnitt aus der Inbetriebnahmedatei (_N_INITIAL_INI)

Programmcode	Kommentar
...	
CHANDATA(3)	; Initialisierung für Kanal 3
\$MC_PROG_EVENT_IGN_INHIBIT='H01F'	
\$MC_PROG_EVENT_MASK='H04'	; Ereignis: Kanal-Reset
...	

Bedeutung

Das PROG_EVENT-Programm _N_CMA_DIR / _N_PROG_EVENT_SPF wird bei Kanal-Reset im 3. Kanal gestartet und bis zum Programmende abgearbeitet, unabhängig ob die Einlesesperre ein- oder ausgeschaltet ist.

10.9.12 Beeinflussung von Stopp-Ereignissen durch Stop-Delay-Bereiche**10.9.12.1 Funktion****Stop-Delay-Bereich**

Die Reaktion auf ein Stopp-Ereignis kann durch einen bedingt unterbrechbaren Bereich im NC-Programm beeinflusst werden. Ein solcher Programmbereich wird als Stop-Delay-Bereich bezeichnet.

Innerhalb von Stop-Delay-Bereichen soll nicht angehalten und auch der Vorschub nicht verändert werden. Ein etwaiger Stopp soll erst wirken, nachdem der Programmabschnitt zu Ende bearbeitet worden ist.

Dadurch ergeben sich folgende Vorteile:

- Der Programmabschnitt wird ohne Geschwindigkeitseinbruch bearbeitet.
- Bricht der Anwender, nachdem gestoppt ist, das Programm mit Reset ab, so ist der abgebrochene Programmsatz nach dem geschützten Bereich. Dieser Programmsatz eignet sich dann als Suchziel für einen nachfolgenden Satzsuchlauf.
- Solange ein Stop-Delay-Bereich bearbeitet wird, werden folgende Hauptlaufachsen nicht gestoppt:
 - Kommandoachsen
 - Positionierachsen, die mit POSA verfahren

Anwendung

Beispiel: Bearbeitung eines Gewindes.

Definition

Die Definition eines Stop-Delay-Bereichs erfolgt im Teileprogramm mittels der vordefinierten Prozeduren DELAYFSTON und DELAYFSTOF (siehe "Programmierung (Seite 588)").

Stopp-Ereignisse

Übersicht der Ereignisse, die einen Stopp zur Folge haben:

Ereignis	Reaktion
DB21, ... DBX7.7 (Reset) oder DB11 DBX20.7 (BAG-Reset)	Immediate
NC-Programm: M30	Alarm 16954
Anwender-Interrupt zum Start eines ASUP: FC 9 oder DB10 DBB1 (Setzen der digitalen NC-Eingänge von PLC)	Delayed
DB21, ... DBX6.2 / DB31, ... DBX2.2 (Restweg löschen) kanal- / achsspezifisch	Immediate
DB21, ... DBX6.3 (Unterprogramm-Durchlaufzahl löschen)	Delayed
DB21, ... DBX6.4 (Programmebenen Abbruch)	Delayed
Stopp wegen Einzelsatz <ul style="list-style-type: none"> • Im Stop-Delay-Bereich: NC stoppt am Ende des 1. Satzes außerhalb des Stop-Delay-Bereichs. • Einzelsatz bereits vor Stop-Delay-Bereich aktiv: NC hält an jeder Satzgrenze, d. h. auch im Stop-Delay-Bereich: DB21, ... DBX7.2 (NC-Stop an der Satzgrenze) Der Stop-Delay-Bereich ist damit abgewählt! 	Delayed
DB11 DBX21.7 (Einzelsatz Typ A einschalten)	Delayed
DB11 DBX21.6 (Einzelsatz Typ B einschalten)	Delayed
DB21, ... DBX7.4 (NC-Stop) und DB11 DBX0.6 (BAG-Stop Achsen plus Spindel)	Immediate
DB21, ... DBX7.3 (NC-Stop an Satzgrenze) und DB11 DBX0.5 (BAG-Stop)	Delayed
NC-Programm: Stopp wegen leerem Überspeicher-Puffer	Alarm 16954
NC-Programm: programmierter oder impliziter STOPRE	Alarm 16954
NC-Programm: M0 oder M1	Alarm 16954

Ereignis	Reaktion
DB21, ... DBX7.2 (NC-Stop an Satzgrenze)	Delayed
Unterprogramm-Ende sollte den Stop-Delay-Bereich immer abwählen.	Systemfehler
NC-Programm: WAITM	Alarm 16954
NC-Programm: WAITE	Alarm 16954
NC-Programm: INIT mit Parameter "S"	Alarm 16954
NC-Programm: MMC (STRING, CHAR)	Alarm 16954
DB21, ... DBB26 (Satz ausblenden ein/umschalten)	Delayed
DB21, ... DBB26 (Satz ausblenden ausschalten)	Delayed
DB21, ... DBX0.6 (DryRun einschalten)	Delayed
DB21, ... DBX0.6 (DryRun ausschalten)	Delayed
DB21, ... DBX6.1 (Einlesesperre)	Delayed
Alarm: Alarmprojektierung STOPATENDBYALARM	Immediate
Alarm: Alarmprojektierung STOPBYALARM	Immediate
Intern: Stopp nach Alarm bei leerem IPO-Puffer	Immediate
Intern: Stopp nach Alarm bei leerem IPO-Puffer	Immediate
NC-Programm: Alarm 16954 bei LFON	Alarm 16954
NC-Programm: WAITMC	Alarm 16954
NC-Programm: NEWCONF	Alarm 16954
NC-Programm: NEWCONF	Alarm 16954
BTSS: PI "_N_SETUDT"	Delayed
Erweitertes Stillsetzen und Rückziehen	Delayed
DB31, ... DBX3.0 (Externe Nullpunktverschiebung übernehmen)	Delayed
BTSS: PI "_N_FINDST" STOPRUN	Alarm 16955
Alarmer mit Reaktion NOREADY	Immediate
DB21, ... DBX7.2 == 1 (NC-Stop an Satzgrenze)	Delayed
DB21, ... DBX7.3 == 1 (NC-Stop)	
DB21, ... DBX7.4 == 1 (NC-Stop Achsen plus Spindel)	
DB21, ... DBX6.1 == 1 (Einlesesperre)	Delayed

Ereignis	Reaktion
DB21, ... DBX2.0 == 1 (Einzelsatz)	Delayed
Reaktion <ul style="list-style-type: none"> • Immediate Stoppt sofort, auch im Stop-Delay-Bereich. Wird als "hartes Stopp-Ereignis" bezeichnet. • Delayed Stopp (auch ein kurzfristiger) erfolgt erst nach dem Stop-Delay-Bereich. Wird als "sanftes Stopp-Ereignis" bezeichnet. • Alarm 16954 Programm wird abgebrochen, da im Stop-Delay-Bereich unerlaubte Programmbefehle verwendet worden sind. • Alarm 16955 Programm wird fortgesetzt, im Stop-Delay-Bereich hat eine unerlaubte Aktion stattgefunden. • Alarm 16957 Der Programmbereich (Stop-Delay-Bereich), der durch DELAYFSTON und DELAYFSTOF geklammert ist, konnte nicht aktiviert werden. Damit wirkt jeder Stopp sofort und wird nicht verzögert! Dies geschieht immer dann, wenn man in einen Stop-Delay-Bereich hineinbremsst, d. h. vor dem Stop-Delay-Bereich beginnt ein Bremsvorgang, der erst im Stop-Delay-Bereich endet. Betritt man den Stop-Delay-Bereich mit Override 0, so kann der Stop-Delay-Bereich ebenso nicht aktiviert werden (Beispiel: ein G4 vor dem Stop-Delay-Bereich erlaubt es dem Anwender, den Override auf 0 abzusenken. Der nächste Satz im Stop-Delay-Bereich beginnt dann mit Override 0 und die beschriebene Alarmsituation tritt auf.) Hinweis MD11411 \$MN_ENABLE_ALARM_MASK (Aktivierung von Warnungen) Bit 7 schaltet diesen Alarm erst ein.	

Hinweis

Es gibt NC-Ereignisse, die nur kurzfristig stoppen, um einen Umschaltvorgang durchzuführen und anschließend sofort wieder zu starten. Dazu zählt z. B. ein Interrupt, durch welchen das NC-Programm kurzfristig gestoppt wird, um das zugehörige ASUP zu starten. Auch diese Ereignisse sind im Stop-Delay-Bereich zulässig, werden aber auf dessen Ende verschoben und gelten demnach als "sanftes" Stopp-Ereignis.

Hinweis

Trifft ein "hartes" Stopp-Ereignis auf den "Stop-Delay-Bereich", so wird der "Stop-Delay-Bereich" komplett abgewählt! Das heißt, tritt in diesem Programmabschnitt ein weiterer beliebiger Stopp auf, so wird sofort angehalten. Erst eine Neuprogrammierung (erneutes DELAYFSTON) lässt einen neuen Stop-Delay-Bereich beginnen.

10.9.12.2 Parametrierung

Maschinendaten

Stopp-Verhalten bei G331/G332

Für das Gewindebohren ohne Ausgleichsfutter (G331, G332) ist das Stopp-Verhalten wie folgt einstellbar:

MD11550 \$MN_STOP_MODE_MASK

Bit	Wert	Bedeutung
0	0 (Default)	Bewirkt einen impliziten Stop-Delay-Bereich, wenn G331/G332 aktiv ist und zusätzlich eine Bahnbewegung oder eine Verweilzeit (G4) programmiert wurde.
	1	Ein Stopp während aktivem G331/G332 ist in folgenden Situationen möglich: <ul style="list-style-type: none"> • Bei Unterbrechung des Bahnsteuerbetriebs (G64). • Bei Programmierung einer Verweilzeit (G4) zwischen den G33x-Sätzen. Zur Definition eines Stop-Delay-Bereichs müssen die NC-Befehle DELAYFSTON und DELAYFSTOF benutzt werden.

10.9.12.3 Programmierung

Stop-Delay-Bereiche definieren (DELAYFSTON, DELAYFSTOF)

Zur Definition eines bedingt unterbrechbaren Bereichs im Teileprogramm (Stop-Delay-Bereich) dienen die vordefinierten Prozeduren DELAYFSTON und DELAYFSTOF.

Hinweis

In Synchronaktionen sind DELAYFSTON und DELAYFSTOF **nicht** zulässig!

Syntax

```
DELAYFSTON
...
DELAYFSTOF
```

Bedeutung

DELAYFSTON:	Beginn eines Stop-Delay-Bereichs definieren	
	Alleine im Satz:	ja
DELAYFSTOF:	Ende eines Stop-Delay-Bereichs definieren	
	Alleine im Satz:	ja

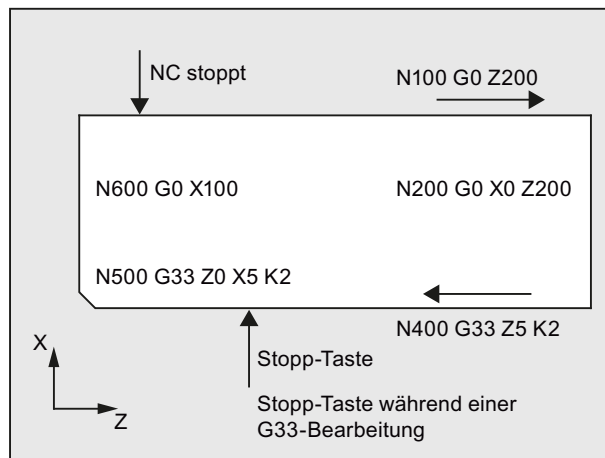
Programmierbeispiel

In einer Schleife wird folgender Programmblock wiederholt:

Programmcode

```
...
N99 MY_LOOP:
N100 G0 Z200
N200 G0 X0 Z200
N300 DELAYFSTON
N400 G33 Z5 K2 M3 S1000
N500 G33 Z0 X5 K3
N600 G0 X100
N700 DELAYFSTOF
N800 GOTOB MY_LOOP
...
```

Im folgenden Bild ist erkennbar, dass der Anwender im Stop-Delay-Bereich "Stopp" drückt, und die NC beginnt den Bremsvorgang außerhalb des Stop-Delay-Bereichs, d. h. im Satz N100. Damit kommt die NC im vorderen Bereich von N100 zum Halten.



Weitere Informationen

Unterprogrammende

Mit dem Ende des Unterprogramms, in dem DELAYFSTON gerufen wurde, wird implizit DELAYFSTOF aktiviert.

Schachtelung

Ruft Unterprogramm 1 in einem Stop-Delay-Bereich Unterprogramm 2, so ist Unterprogramm 2 komplett ein Stop-Delay-Bereich. Insbesondere ist DELAYFSTOF in Unterprogramm 2 wirkungslos.

Beispiel:

Programmcode	Kommentar
N10010 DELAYFSTON	; Sätze mit N10xxx Programmebene 1.
N10020 R1 = R1 + 1	
N10030 G4 F1	; Stop-Delay-Bereich beginnt.
...	
N10040 Unterprogramm2	
...	
...	; Interpretation des Unterprogramms 2.
N20010 DELAYFSTON	; Unwirksam, wiederholter Beginn, 2. Ebene.
...	
N20020 DELAYFSTOF	; Unwirksam, Ende in anderer Ebene.
N20030 RET	
N10050 DELAYFSTOF	; Stop-Delay-Bereichsende in gleicher Ebene.
...	
N10060 R2 = R2 + 2	
N10070 G4 F1	; Stop-Delay-Bereich endet. Stopps wirken ab jetzt unmittelbar.

Systemvariablen

Ob sich die Teileprogrammbearbeitung zum aktuellen Zeitpunkt in einem Stop-Delay-Bereich befindet, kann mittels folgender Systemvariablen abgefragt werden:

- im Teileprogramm mit \$P_DELAYFST
- in Synchronaktionen mit \$AC_DELAYFST

Wert	Bedeutung
0	Stop-Delay-Bereich nicht aktiv
1	Stop-Delay-Bereich aktiv

10.9.12.4 Randbedingungen

Überlappung

Überschneiden sich zwei Stop-Delay-Bereiche, einer aus DELAYFSTON/DELAYFSTOF und der andere aus MD11550 \$MN_STOP_MODE_MASK, so wird der größtmögliche Stop-Delay-Bereich gebildet.

Verweilzeit (G4)

G4 ist im Stop-Delay-Bereich zulässig. Andere Teileprogrammbefehle, die zu einem zum vorübergehenden Stopp führen (z. B. WAITM), sind nicht zulässig und lösen den Alarm 16954 aus.

Override

Wird der Override **vor** dem Stop-Delay-Bereich verändert, wird der Override **im** Stop-Delay-Bereich wirksam.

Wird der Override **im** Stop-Delay-Bereich verändert, wird die Änderung erst **nach** dem Stop-Delay-Bereich wirksam.

Hinweis

Override = 0

Wird der Override **vor** dem Stop-Delay-Bereich auf 0 gesetzt, kann der Stop-Delay-Bereich nicht aktiviert werden!

Vorschubsperr

DB21, ... DBX6.0 (Vorschubsperr) wirkt im Stop-Delay-Bereich nicht, erst nach dem Verlassen des Stop-Delay-Bereichs wird angehalten.

Einzelsatz

Schaltet man den Einzelsatz im Stop-Delay-Bereich ein, so stoppt der NC am Ende des ersten Satzes außerhalb des Stop-Delay-Bereichs.

Ist der Einzelsatz bereits vor dem Stop-Delay-Bereich angewählt, hält der NC an jeder Satzgrenze, d. h. auch im Stop-Delay-Bereich! **Der Stop-Delay-Bereich ist damit abgewählt.**

10.10 Asynchrone Unterprogramme (ASUPs)

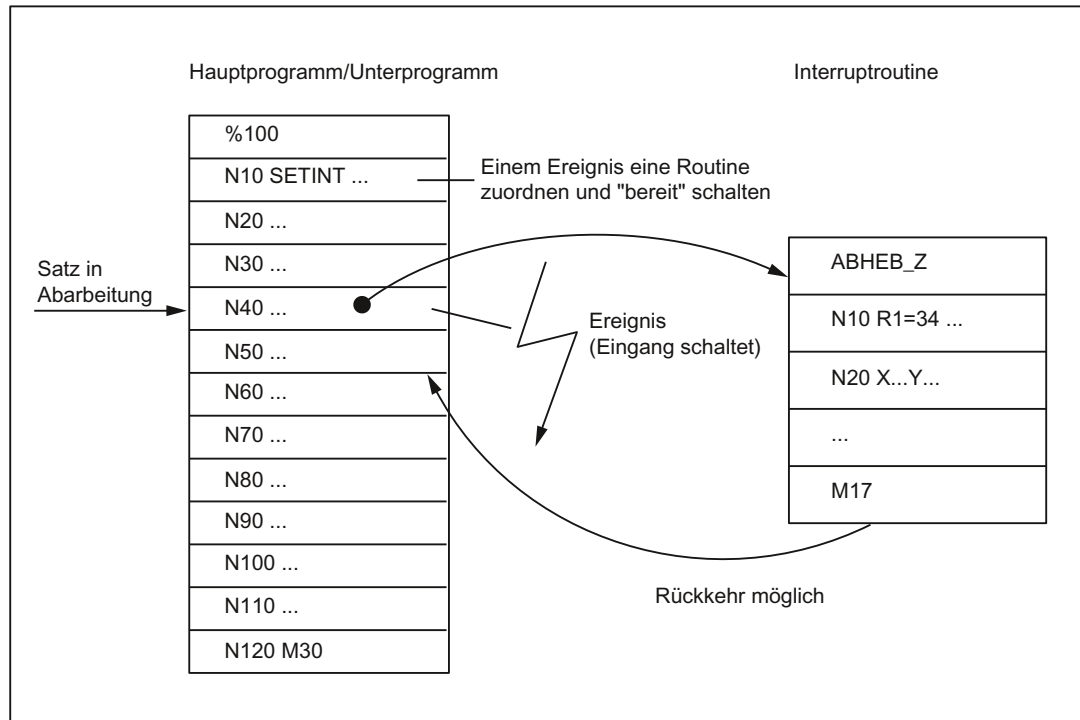
10.10.1 Funktion

Hinweis

Die im Folgenden verwendeten Begriffe "Asynchrones Unterprogramm", "ASUP" und "Interruptroutine" bezeichnen die gleiche Funktionalität.

Allgemein

Asynchrone Unterprogramme (ASUP) sind NC-Programme, die in einem NC-Kanal als Reaktion auf asynchrone Ereignisse (Interrupt-Eingangssignale, Prozess- bzw. Maschinenzustände) gestartet werden. Durch die Aktivierung eines ASUP wird ein in Abarbeitung befindliches NC-Programm unterbrochen. Mit dem ASUP-Ende kann das NC-Programm an der Unterbrechungsposition fortgesetzt werden.



Das im Kanal in Abarbeitung befindliche NC-Programm kann im Ganzen oder abschnittsweise vor einer Unterbrechung durch ein ASUP geschützt werden. Siehe Kapitel "Programmierung (SETINT, PRIO) (Seite 602)", Absatz "Flexible Programmierung".

Definition

Damit aus einem NC-Programm ein ASUP (Interruptroutine) wird, muss dem NC-Programm über den Befehl `SETINT` (siehe Kapitel "Programmierung (SETINT, PRIO) (Seite 602)") oder über den PI-Dienst "ASUP" (siehe Kapitel "PI-Dienst: ASUP (Seite 1010)") ein Interruptsignal zugeordnet werden.

Interruptsignale

- Als Interruptsignale stehen insgesamt 8 Peripherie-Eingänge zur Verfügung.
- Die Peripherie-Eingangssignale können über das PLC-Anwenderprogramm beeinflusst werden.
- Die ersten vier Peripherie-Eingänge sind die 4 schnellen Eingänge der NCU-Baugruppe. Die Signalzustände können über die NC/PLC-Nahtstelle im Datenbaustein DB10 gelesen werden. Über die NC/PLC-Nahtstelle im Datenbaustein DB10 können die Eingangssignale auch gesperrt werden.

Weitere Informationen zur PLC-Beeinflussung der schnellen NC-Eingänge (Interruptsignale) siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)".

Literatur:

Funktionshandbuch Erweiterungsfunktion; A4: Digitale und analoge NC-Peripherie

Aufruf

Im Programmbetrieb

Im Programmbetrieb, d. h. in der Betriebsart AUTOMATIK oder MDA, ist der Aufruf eines ASUP grundsätzlich immer möglich.

Außerhalb vom Programmbetrieb

Außerhalb des Programmbetriebs ist der Aufruf eines ASUP in folgenden Betriebsarten, Maschinenfunktionen und Zuständen möglich:

- JOG, JOG-REF
- MDA-Teach In, MDA-Teach In-REF, MDA-Teach In-JOG, MDA-REF, MDA-JOG
- AUTOMATIK, Programmzustand "gestoppt", "ready"
- Achszustand "Nicht referenziert"

Wird ein ASUP während JOG oder JOG-REF gestartet, wird die aktuelle Verfahrbewegung abgebrochen.

Aktivierung

Die Aktivierung eines ASUP erfolgt über:

- 0/1-Flanke des Interruptsignals, hervorgerufen durch eine 0/1-Flanke am zugehörigen schnellen NC-Eingang
- Aufruf des "Function Call ASUP" (siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)")
- Setzen eines Ausgangs über Synchronaktion, der über Kurzschluss auf einen Interrupt-Eingang parametrisiert ist (siehe "Beispiele (Seite 605)")

Literatur:

Funktionshandbuch Synchronaktionen

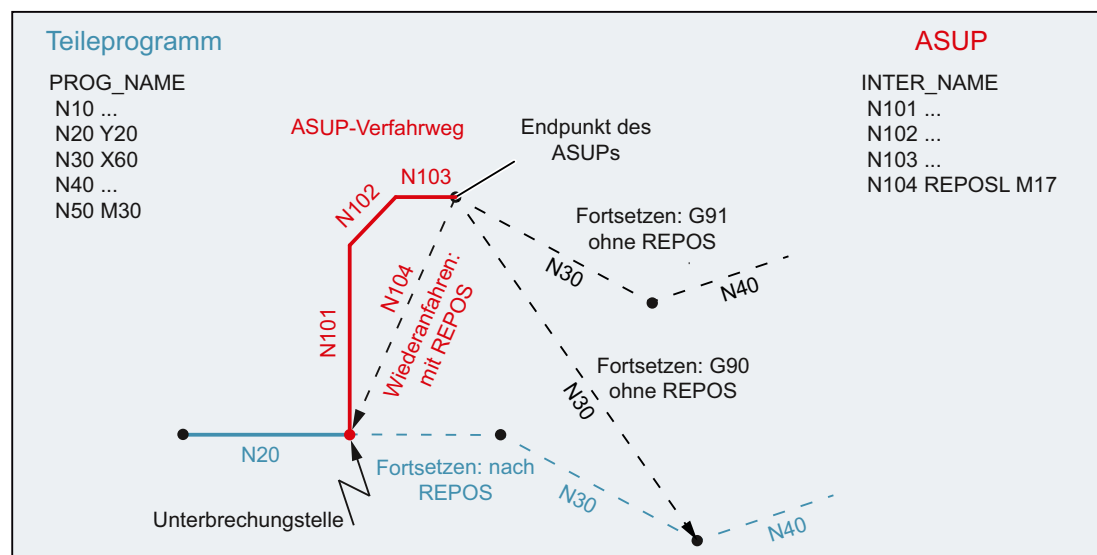
Anzeige

Die Aktivierung eines ASUP wird kanalspezifisch mit dem folgenden NC/PLC-Nahtstellensignal angezeigt:

DB21, ... DBX378.0 == 1 (ASUP aktiv)

10.10.1.1 Ablauf eines ASUPs im Programmbetrieb

1. Abbremsen der Achsen
Nach Aktivierung des ASUPs werden alle Maschinenachsen anhand der Bremsrampe (MD32300 \$MA_MAX_AX_ACCEL) bis zum Stillstand abgebremst und die Achspositionen abgespeichert.
2. Reorganisation
Zusätzlich zum Abbremsen der Achsen werden die vordekodierten Rechensätze bis zum Unterbrechungssatz zurückgerechnet, d. h. alle Variablen, Frames und G-Befehle bekommen den Wert zugeordnet, den sie ohne Vorausdekodierung des Teileprogramms an der Unterbrechungsstelle besitzen würden. Diese Werte werden auch wieder zwischengespeichert, um nach dem Ende des ASUPs wieder darauf zugreifen zu können. Ausnahmen, bei denen keine Reorganisation möglich ist:
 - Innerhalb von Gewindeschneidsätzen
 - Bei komplexen Geometrien (z. B. Spline oder Radiuskorrektur)
3. Abarbeiten des ASUPs
Nach Beendigung der Reorganisation wird automatisch das ASUP gestartet. Das ASUP wird wie ein normales Unterprogramm abgearbeitet (Schachtelungstiefe etc.).
4. Ende des ASUPs
Nachdem die Endkennung (M02, M30, M17) des ASUPs bearbeitet wurde, wird standardmäßig auf die Endposition des auf den Unterbrechungssatz folgenden Teileprogrammsatzes gefahren.
Falls ein Rückpositionieren auf den Unterbrechungspunkt erfolgen soll, muss am Ende des ASUPs eine REPOS-Anweisung programmiert werden, z. B.: N104 REPOS M17



10.10.1.2 ASUP mit REPOSA

Wird ein NC-Programm durch NC-Stop oder Alarm angehalten und anschließend vom PLC-Anwenderprogramm über FC9 ein ASUP mit REPOSA ausgelöst, ergibt sich typischerweise folgender Ablauf:

- Das ASUP bzw. die darin programmierten Verfahrbewegungen werden abgearbeitet:
 - Programmzustand: "Angehalten"
 - DB21, ... DBX318.0 (ASUP ist angehalten) = 1
- Vor dem Wiederauffahren an die Kontur (REPOS) wird erneut angehalten.
- Zum Wiederauffahren an die Kontur (REPOS) löst der Bediener NC-Start aus:
 - DB21, ... DBX318.0 (ASUP ist angehalten) = 0
 - Die Wiederauffahrbewegung wird ausgeführt.
- Mit dem Ende der Wiederauffahrbewegung wird das FC9-Quittungssignal "ASUP Done" gesetzt und das unterbrochene NC-Programm fortgesetzt.

Hinweis

Das NC/PLC-Nahtstellensignal DB21, ... DBX318.0 (ASUP ist angehalten) wird nur im folgenden Fall gesetzt:

Interrupt im Programmbetrieb und Kanalzustand "Unterbrochen".

Hinweis

Bei ASUPs ohne REPOS fallen das FC9-Quittungssignal "ASUP Done" und das Rücksetzen des NC/PLC-Nahtstellensignals DB21, ... DBX318.0 = 0 (ASUP ist angehalten) zeitlich zusammen.

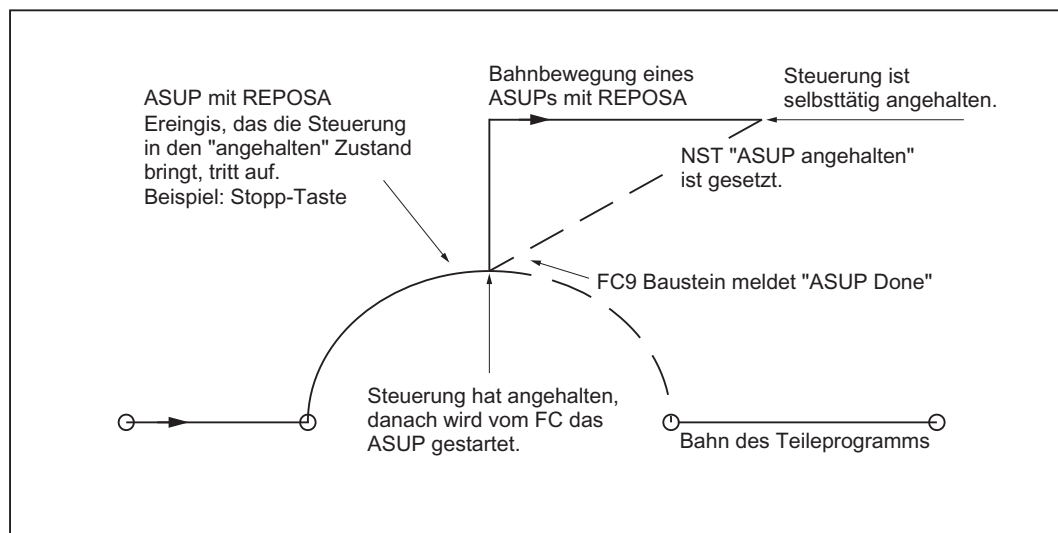


Bild 10-7 Schematischer Ablauf: ASUP mit REPOSA

10.10.1.3 NC-Verhalten

Die unterschiedlichen Reaktionen in den verschiedenen Zuständen von Kanal, BAG oder NC auf ein aktiviertes ASUP sind in der folgenden Tabelle aufgeführt:

Zustand	ASUP-Start	Reaktion der Steuerung
Programm ist aktiv	Interrupt, (PLC)	<ol style="list-style-type: none"> 1. Schnellabheben oder Achsen stoppen 2. Unterbrechung des Programms für die Dauer des ASUP 3. Anfahren der Unterbrechungsstelle, wenn REPOS im ASUP 4. Fortsetzung des Teileprogramms
Reset	Interrupt, (PLC)	<p>Das ASUP läuft ab wie ein Hauptprogramm. Am ASUP-Ende wird wieder Reset (ohne M30) ausgeführt. Der nächste Zustand der Steuerung hängt von folgenden Maschinendaten ab:</p> <p>MD20110 \$MC_RESET_MODE_MASK MD20112 \$MC_START_MODE_MASK</p> <p>Literatur: Funktionshandbuch Grundfunktionen; Achsen, Koordinatensysteme, Frames (K2), Kapitel: "Werkstücknahes Istwertsystem"</p>
Programmbetrieb (AUTOMATIK oder MDA) + Kanal gestoppt	Interrupt, (PLC)	<p>ASUP läuft ab. An seinem Ende wird wieder der Stopp-Zustand eingenommen.</p> <p>Wenn REPOS im ASUP:</p> <ul style="list-style-type: none"> • Die ASUP-Abarbeitung wird vor dem Anfahrssatz gestoppt. • Das Anfahren kann durch Start-Taste ausgelöst werden.
	Start-Taste	Nach ASUP-Ablauf läuft auch das gestoppte Programm weiter ab.
Handbetrieb + Kanal gestoppt	Interrupt, (PLC)	<p>Steuerung nimmt intern für den angesprochenen Kanal den Zustand "interne Programmbearbeitungsbetriebsart" ein (nach außen nicht merkbar) und aktiviert dann das ASUP. Die angewählte Betriebsart bleibt erhalten. Nach ASUP-Ende (M17) wird der ursprüngliche Zustand wieder eingenommen.</p>
JOG AUTO-Teach-In AUTO-Teach-Referenzp.	Interrupt, (PLC)	<p>Stopp der Bearbeitung, Auswerten von:</p> <p>MD11602 \$MN_ASUP_START_MASK MD11604 \$MN_ASUP_START_PRIO_LEVEL</p>
MDA-JOG, MDA-Teach-In, MDA-Teach-Referenzp.	Interrupt, (PLC)	<p>Ggf. internes Umschalten in "interne Programmbearbeitungsbetriebsart", Aktivierung des ASUP, Zustand vor ASUP-Start wieder herstellen.</p> <p>Ein zusammen mit SETINT definiertes LIFTFAST wird bei JOG nicht aktiviert.</p>
Handbetrieb + Kanal läuft	Interrupt, (PLC)	<p>Die gerade aktive Bewegung wird gestoppt. Der Restweg wird gelöscht. Der weitere Ablauf entspricht "Handbetrieb, Kanal gestoppt".</p>
Anwenderalarm 65500 - 65999, Kanal in Reset	Interrupt, (PLC)	<p>Wenn folgendes Maschinendatum eingestellt ist, wird das Anwender-ASUP trotz der Alarmreaktion "Verriegelung NC-Start" abgearbeitet:</p> <p>MD20194 \$MC_IGNORE_NONCSTART_ASUP</p>

Zustand	ASUP-Start	Reaktion der Steuerung
Abarbeitung von INITIAL.INI	nicht möglich	Es wird das Signal "Interruptanforderung nicht möglich" erzeugt.
Satzsuchlauf		
Alarm, der nicht durch NC-Start behoben werden kann		
Digitalisieren eingeschaltet		
Kanal im Fehlerzustand		
Anderer Anwenderalarm als 65500 - 65999, Kanal in Reset, MD20194 nicht aktiv	nicht möglich	Es wird das Signal "Die Anforderung wurde wegen eines Alarms abgebrochen" erzeugt.

10.10.2 Inbetriebnahme: Maschinendaten

10.10.2.1 NC-spez.: BAG-spezifische NC/PLC-Nahtstellensignale und Betriebsartenumschaltung

Mit dem Maschinendatum wird die Wirksamkeit der BAG-spezifischen NC/PLC-Nahtstellensignale des DB11 und die Kanäle in denen eine Umschaltung der Betriebsart ausgeführt wird, festgelegt:

MD11600 \$MN_BAG_MASK = <Wert>

Wert	Bedeutung
0	Die BAG-spezifischen NC/PLC-Nahtstellensignale des DB11 wirken . Eine interne Umschaltung der Betriebsart erfolgt in allen Kanälen der BAG.
1	Die BAG-spezifischen NC/PLC-Nahtstellensignale des DB11 wirken nicht . Eine interne Umschaltung der Betriebsart erfolgt nur in dem Kanal, in dem ein ASUP aktiv ist.
2	Die BAG-spezifischen NC/PLC-Nahtstellensignale des DB11 wirken . Eine interne Umschaltung der Betriebsart erfolgt nur in dem Kanal, in dem ein ASUP aktiv ist.
3	Die BAG-spezifischen NC/PLC-Nahtstellensignale des DB11 wirken nicht . Eine interne Umschaltung der Betriebsart erfolgt nur in dem Kanal, in dem ein ASUP aktiv ist.

Hinweis

Mehrkanalige Systeme

Soll bei mehrkanaligen Systemen die Funktion "Manuelles Verfahren während Unterbrechung eines ASUP in der Betriebsart JOG" (siehe unten) möglich sein, muss MD11600 \$MN_BAG_MASK auf den Wert "2" oder "3" gesetzt werden.

Siehe auch

Programmierung (SETINT, PRIO) (Seite 602)

10.10.2.2 NC-spez.: ASUP Startfreigabe

Mit dem Maschinendatum wird festgelegt, welche Stoppgründe bei einem ASUP-Start ignoriert werden:

MD11602 \$MN_ASUP_START_MASK, <Bit> = <Wert>

Bit	Wert	Bedeutung
0	0	Bei anstehendem Stopp wegen Stopp-Taste, M0 oder M01 wird das ASUP nicht gestartet.
	1	Trotz anstehendem Stopp wegen Stopp-Taste, M0 oder M01 wird das ASUP gestartet .
2	0	Beim Start eines ASUPs wirken bei anstehender Einlesesperre im Kanal die Einstellungen in den kanalspezifischen Maschinendaten: <ul style="list-style-type: none"> • MD20107 \$MC_PROG_EVENT_IGN_INHIBIT • MD20116 \$MC_IGNORE_INHIBIT_ASUP
	1	Der Start eines Anwender- oder System-ASUPs ist in allen Kanälen der NC auch bei anstehender Einlesesperre im jeweiligen Kanal freigegeben. Die Einstellungen in den kanalspezifischen Maschinendaten wirken nicht : <ul style="list-style-type: none"> • MD20107 \$MC_PROG_EVENT_IGN_INHIBIT • MD20116 \$MC_IGNORE_INHIBIT_ASUP Achtung Die Startfreigabe trotz anstehender Einlesesperre wirkt in allen Kanälen der NC. Sowohl für Anwender- als auch für System-ASUPs. Es wird daher dringend empfohlen statt der NC-spezifischen Freigabe die kanalspezifischen Freigaben zu verwenden.
3	0	Wurde in der Betriebsart JOG ein ASUP gestartet und durch NC-Stop unterbrochen, können in diesem Zustand Achsen nicht manuell verfahren werden.
	1	Wurde in der Betriebsart JOG ein ASUP gestartet und durch NC-Stop unterbrochen, können in diesem Zustand Achsen manuell verfahren werden. Durch NC-Start kann das ASUP fortgesetzt werden. Dabei erfolgt ein automatischer REPOS-Vorgang. Hinweis Ist in der Steuerung mehr als ein Kanal parametrierbar, muss zusätzlich folgendes Maschinendatum gesetzt werden: MD11600 \$MN_BAG_MASK, Bit 1 = 1

Manuelle Startfreigabe

Wird aufgrund der parametrierten Startfreigaben ein ASUP nicht **automatisch** gestartet, kann das ASUP dennoch über NC/PLC-Nahtstellensignal (DB21, ... DBX7.1) vom PLC-Anwenderprogramm oder manueller Betätigung von NC-Start gestartet werden.

Hinweis

Das ASUP für "Schnellabheben von der Kontur" (LIFTFAST) wird in jedem Fall gestartet.

10.10.2.3 NC-spez.: Wirksamkeit der parametrisierten Startfreigaben

Mit dem Maschinendatum wird eingestellt bis zu welcher ASUP-Priorität (Seite 602), ausgehend von der höchsten Priorität, die Einstellungen in MD11602 \$MN_ASUP_START_MASK wirksam sind:

MD11604 \$MN_ASUP_START_PRIO_LEVEL = <ASUP-Priorität>

Beispiel

MD11604 \$MN_ASUP_START_PRIO_LEVEL = 5

Die Einstellungen in MD11602 \$MN_ASUP_START_MASK sind wirksam für ASUPs der Prioritäten 1 → 5.

10.10.2.4 Kanalspez.: Startfreigabe trotz nicht referenzierter Achsen

Mit dem Maschinendatum wird eingestellt, bei welchen Interrupts das zugehörige ASUP trotz parametrierter Funktion "NC-Startsperre ohne Referenzpunkt" (MD20700 \$MC_REFP_NC_START_LOCK) gestartet wird:

MD20115 \$MC_IGNORE_REFP_LOCK_ASUP, Bit (1 - <Interrupt>) = TRUE

ACHTUNG**System-Interrupts**

Mit MD20115 \$MC_IGNORE_REFP_LOCK_ASUP, Bit 8 - 31 werden die den System-Interrupts zugeordneten ASUPs freigegeben.

Durch Bit 8 / Interrupt 9 wird ein ASUP gestartet, welches Verfahrbewegungen beinhaltet.

NC-spezifische ASUP-Startfreigabe

Ist MD11602 \$MN_ASUP_START_MASK, Bit 2 == TRUE, ist für alle Kanäle der NC die ASUP-Startfreigabe trotz parametrierter kanalspezifischer Funktion "NC-Startsperre ohne Referenzpunkt" (MD20700 \$MC_REFP_NC_START_LOCK) gegeben. Es wird daher **dringend empfohlen** statt der NC-spezifischen Freigabe die **kanalspezifischen** Freigabe zu verwenden.

10.10.2.5 Kanalspez.: Startfreigabe trotz Einlesesperre

Mit dem Maschinendatum wird eingestellt, bei welchen Interrupts das zugehörige ASUP trotz anstehender Einlesesperre im Kanal (DB21, ... DBX6.1) gestartet wird:

MD20116 \$MC_IGNORE_INHIBIT_ASUP, Bit (1 - <Interrupt>) = TRUE

ACHTUNG

System-Interrupts

Mit MD20116 \$MC_IGNORE_INHIBIT_ASUP, Bit 8 - 31 werden die den System-Interrupts zugeordneten ASUPs freigegeben.

Durch Bit 8 / Interrupt 9 wird ein ASUP gestartet, welches Verfahrbewegungen beinhaltet.

NC-spezifische ASUP-Startfreigabe

Ist MD11602 \$MN_ASUP_START_MASK, Bit 2 == TRUE, werden in allen Kanäle der NC die kanalspezifischen Einstellungen in MD20116 \$MC_IGNORE_INHIBIT_ASUP ignoriert. Es wird daher **dringend empfohlen** statt der NC-spezifischen Freigabe die **kanalspezifischen** Freigabe zu verwenden.

10.10.2.6 Kanalspez.: Kontinuierliche Abarbeitung trotz Einzelsatz

Mit dem Maschinendatum wird eingestellt, bei welchen Interrupts das zugehörige ASUP trotz aktiver Einzelsatzbearbeitung im Kanal (DB21, ... DBX0.4) **kontinuierlich**, d.h. ohne satzweise Unterbrechung, abgearbeitet wird:

MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP, Bit (1 - <Interrupt>) = TRUE

Randbedingungen

Die Einstellungen in MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP wirken nur bei Einzelsatz SBL1 (Hauptlauf-Einzelsatz).

ACHTUNG

System-Interrupts

Mit MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP, Bit 8 - 31 werden die den System-Interrupts zugeordneten ASUPs freigegeben.

Durch Bit 8 / Interrupt 9 wird ein ASUP gestartet, welches Verfahrbewegungen beinhaltet.

NC-spezifische ASUP-Startfreigabe

Ist MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK, Bit 1 == TRUE, werden in allen Kanäle der NC die kanalspezifischen Einstellungen in MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP ignoriert. Es wird daher **dringend empfohlen** statt der NC-spezifischen Freigabe die **kanalspezifischen** Freigabe zu verwenden.

10.10.2.7 Kanalspez.: Aktualisierung der Anzeige

Mit dem Maschinendatum wird eingestellt, dass keine Aktualisierung der Anzeige während der Ausführung von ASUPs durchgeführt wird, damit kein Flackern der Anzeige des Programm- und Kanalzustandes an der Bedienoberfläche bei der Ausführung von sehr kurzen ASUPs auftritt:

MD20191 \$MC_IGN_PROG_STATE_ASUP, Bit (1 - <Interrupt>) = TRUE

Hinweis

NC/PLC-Nahtstellensignal

Bei Ausführung eines ASUPs mit unterdrückter Anzeige wird folgendes NC/PLC-Nahtstellensignal gesetzt:

DB21, ... DBX378.1 = 1 ("Stilles" ASUP aktiv)

Systemvariable und NC/PLC-Nahtstellensignale

Die Systemvariablen und NC/PLC-Nahtstellensignale für Programm- und Kanalzustand werden durch die Unterdrückung der Anzeige während der Abarbeitung eines ASUP nicht beeinflusst:

- \$AC_STAT (Kanalzustand)
- \$AC_PROG (Programmzustand)
- DB21, ... DBX35.5 - 7 (Kanalzustand)
- DB21, ... DBX35.0 - 4 (Programmzustand)

10.10.3 Programmierung: Systemvariablen

10.10.3.1 REPOS-Möglichkeit (\$P_REPINF)

Im Zusammenhang mit ASUPs können Programmabläufe entstehen, für die es keine eindeutige Rückkehr zu einem Wiederanfahrpunkt an die Kontur (REPOS) gibt.

Über die Systemvariable kann im ASUP gelesen werden, ob REPOS möglich ist:

<Wert> = \$P_REPINF

Wert	Bedeutung
0	Repositionieren mit REPOS nicht möglich, weil: <ul style="list-style-type: none"> • Nicht im ASUP aufgerufen • ASUP aus Reset-Zustand ablief • ASUP aus JOG ablief
1	Repositionieren mit REPOS im ASUP möglich

10.10.3.2 Aktivierungsereignis (\$AC_ASUP)

Über die Systemvariablen \$AC_ASUP können bezüglich des Ereignisses, das zur Aktivierung des ASUPs geführt hat, folgende Informationen gelesen werden:

- Weswegen wurde ASUP aktiviert, z. B. Bit 0: Anwender-Interrupt "ASUP mit BIsync"
- Wodurch wurde das ASUP aktiviert, z. B. Bit 0: NC/PLC-Nahtstellensignal, Digitale-Analoge Schnittstelle
- Welche Art von Fortsetzung ist möglich, z.B. Bit 0: freiwählbar REORG oder RET

10.10.4 Programmierung (SETINT, PRIO)**Zuordnung: Interruptsignal zu NC-Programm**

Die Zuordnung eines NC-Programms zu einem Interruptsignal erfolgt mit dem Befehl SETINT. Das NC-Programm wird dadurch zu einem ASUP.

Syntax

```
SETINT(<n>) <NC-Programm>
```

Bedeutung

SETINT:	Zuordnung NC-Programm zu einem Interruptsignal	
<n>:	Nummer des Interruptsignals	
	Wertebereich:	0, 1, 2, ... 8
<NC-Programm>:	Programmname	

Beispiel

Programmcode	Kommentar
N20 SETINT(3) ABHEBEN_Z	; IF Eingang 3 == 1 ; THEN ASUP "ABHEBEN_Z" starten

Zusammen mit `SETINT` können zusätzlich folgende Anweisungen programmiert werden:

- **LIFTFAST**
Beim Eintreffen des Interruptsignals wird vor dem Start des ASUPs ein "Schnellabheben des Werkzeugs von der Kontur" durchgeführt. Die Bewegungsrichtung für das Schnellabheben wird durch die Programmanweisung `ALF` festgelegt.
- **BLSYNC**
Beim Eintreffen des Interruptsignals wird der laufende Programmsatz noch abgearbeitet und erst danach das ASUP gestartet.

Hinweis

Die Zuordnung Interruptsignal ↔ Teileprogramm wird unter folgenden Bedingungen gelöscht:

- Reset-Zustand des Kanals
 - `CLRINT`-Anweisung im Teileprogramm
-

Prioritäten

Falls in einem NC-Programm mehrere Interrupts durch `SETINT` aktiviert werden, müssen den zugeordneten NC-Programmen bzw. ASUPs unterschiedliche Prioritäten zugewiesen werden.

Syntax

`PRIO=<Wert>`

Bedeutung

<code>PRIO:</code>	Schlüsselwort zur Festlegung der Priorität des Interrupts
<code><Wert>:</code>	Priorität: 1, 2, 3 ... 128. 1 entspricht der höchsten Priorität.

Beispiel

Programmcode	Kommentar
N20 <code>SETINT(3) PRIO=2 ABHEBEN_Z</code>	; IF Eingang 3 == 1
	; THEN ASUP "ABHEBEN_Z" starten
N30 <code>SETINT(2) PRIO=3 ABHEBEN_X</code>	; IF Eingang 2 == 1
	; THEN ASUP "ABHEBEN_X" starten

Die ASUPs werden in der Reihenfolge der Prioritätswerte nacheinander abgearbeitet, wenn die Eingänge 2 und 3 gleichzeitig geschaltet haben:

1. "ABHEBEN_Z"
2. "ABHEBEN_X".

Weitere Interrupt-spezifische Befehle

Befehl	Bedeutung
SAVE	Wenn in einem ASUP der Befehl <code>SAVE</code> verwendet wird, dann werden mit Ende des ASUP die vor der Unterbrechung im unterbrochenen NC-Programm aktiven G-Befehle, Frames und Transformationen wieder wirksam.
DISABLE ENABLE	Durch Verwendung des Befehlpaars <code>DISABLE</code> <code>ENABLE</code> können Programabschnitte vor dem Unterbrechen durch ASUPs geschützt werden. Die über <code>SETINT</code> getroffene Zuordnung von Interruptsignal zu NC-Programm bleibt dabei erhalten. Das ASUP wird dann mit der nächsten 0/1-Flanke des Interruptsignals nach <code>ENABLE</code> gestartet.
CLRINT (<n>)	Zuordnung von Interruptsignals n zum mit <code>SETINT</code> zugeordneten NC-Programm löschen.

Literatur

Programmierhandbuch Arbeitsvorbereitung; Kapitel: "Flexible NC-Programmierung" > "Interruptroutine (ASUP)"

Siehe auch

Programmierung (Seite 607)

10.10.5 Randbedingungen

Betriebsartenübergreifender ASUP-Start

Zu prüfende Einstellungen

- MD11600 \$MN_BAG_MASK
- MD11604 \$MN_ASUP_START_PRIO_LEVEL
- Die Priorität der Interruptzuordnung

Empfohlene Einstellungen

NC-spezifische Maschinendaten:

- MD11600 \$MN_BAG_MASK = 'H3'

Hinweis

Bei dieser Einstellung ist zu beachten, dass die BAG-spezifischen NC/PLC-Nahtstellensignale des DB11 nicht mehr auf den Kanal wirken, in dem das ASUP abgearbeitet wird. Sollte dieses Verhalten nicht erwünscht sein, kann alternativ die Einstellung MD11600 \$MN_BAG_MASK = 'H2' verwendet werden (siehe "Inbetriebnahme: Maschinendaten (Seite 597)").

MD11602 \$MN_ASUP_START_MASK = 'H5'

- MD11604 \$MN_ASUP_START_PRIO_LEVEL = 7

Kanalspezifische Maschinendaten für den Kanal, in dem das ASUP gestartet wird, oder allgemein für alle Kanäle:

- MD20105 \$MC_PROG_EVENT_IGN_REFP_LOCK, Bit <n> = TRUE
<n>: Für alle benötigten ereignisgesteuerten Programmaufrufe (Prog-Events)
- MD20115 \$MC_IGNORE_REFP_LOCK_ASUP, Bit <n> = TRUE
<n>: Für alle **benötigten Anwender**-Interrupts

ACHTUNG

System-Interrupts

Mit MD20115 \$MC_IGNORE_REFP_LOCK_ASUP, Bit 8 ... 31 werden die System-Interrupts freigegeben.

Durch Bit 8 / Interrupt 9 wird ein ASUP gestartet, welches Verfahrbewegungen beinhaltet.

10.10.6 Beispiele

Aktivierung eines ASUP durch einen Interrupt aus einer Synchronaktion heraus

1. Zwei aktive digitale Ein-/Ausgangs-Bytes parametrieren:
 - MD10350 \$MN_FASTIO_DIG_NUM_INPUTS = 2
 - MD10360 \$MN_FASTIO_DIG_NUM_OUTPUTS = 2
2. Einen Kurzschluss mit ODER-Verknüpfung von Ausgang 9 auf Eingang 9 parametrieren:
 - Eingang 1, Eingangsbyte 2 = (Ausgang 1, Ausgangsbyte 2) **ODER** (HW-Eingangs-Signale 1, Eingangsbyte 2):
MD10361 \$MN_FASTIO_DIG_SHORT_CIRCUIT[0] = 'H0102B102'
3. Zuordnen des HW-Eingangsbytes zur Interrupt-Programmierung SETINT:
 - Eingangs-Byte 2:
MD21210 \$MC_SETINT_ASSIGN_FASTIN = 2
4. Eingang als ASUP-Auslöser definieren:
 - Eingang 1 im zweiten Eingangs-Bytes, d.h. absolut Eingang 9, startet Programm SYNCASUP
SETINT(1) PRIO=1 SYNCASUP
5. Synchronaktion zum Setzen des Ausgangs definieren:
 - Synchronaktion mit ID 1 setzt den Ausgang 9 immer dann auf 1, wenn der Wert des normierte Bahnparameters $\geq 0,5$ wird:
IDS=1 EVERY \$\$AC_PATHN ≥ 0.5 DO \$A_OUT[9]=1
Durch den Kurzschluss von Ausgang 9 auf Eingang 9 wird Interrupt 1 ausgelöst und das NC-Programm "SYNCASUP" als ASUP gestartet.

10.11 Anwenderspezifisches ASUP für RET und REPOS

10.11.1 Funktion

Funktion

Die Steuerungs-Software enthält ein Siemens-spezifisches ASUP zur Realisierung der Funktionen NC-Programmende (RET) und Wiederanfahren an die Kontur (REPOS). Das System-ASUPs kann vom Maschinenhersteller durch ein anwenderspezifisches ASUP ersetzt werden.



GEFAHR

Programmierfehler

Das Sicherstellen des fehlerfreien Inhalts des anwenderspezifischen ASUPs, welches das Siemens-spezifische ASUP ("ASUP.SYF") ersetzt, liegt in der alleinigen Verantwortung des Maschinenherstellers.

10.11.2 Parametrierung

Installation

ASUP-Name

Dem anwenderspezifischen ASUP muss folgender Namen gegeben werden:

- **_N_ASUP_SPF**

ASUP-Verzeichnisse

Das anwenderspezifischen ASUP "_N_ASUP_SPF" muss in einem der beiden Verzeichnisse abgelegt werden:

- **_N_CMA_DIR** (Hersteller-Verzeichnis)
- **_N_CUS_DIR** (Anwender-Verzeichnis)

Aktivierung und Suchpfad

Bei welchem Ereignis das anwenderspezifischen ASUP "_N_ASUP_SPF" aktiviert wird, wird in folgendem Maschinendatum über Bit 0 und 1 eingestellt.

Wo bei Aktivierung des anwenderspezifischen ASUP "_N_ASUP_SPF" mit der Suche begonnen wird, wird über Bit 2 eingestellt.

MD11610 \$MN_ASUP_EDITABLE, Bit 0, 1, 2 = <Wert>

Bit	Wert	Bedeutung
0 und 1	0	Weder bei Programmende (RET) noch bei Wiederanfahren an die Kontur (REPOS) wird das anwenderspezifische ASUP aktiviert.
	1	Bei RET wird das anwenderspezifische ASUP aktiviert. Bei REPOS wird das systemspezifische ASUP aktiviert.
	2	Bei RET wird das systemspezifische ASUP aktiviert. Bei REPOS wird das anwenderspezifische ASUP aktiviert.
	3	Bei RET und bei REPOS wird das anwenderspezifische ASUP aktiviert.
2	0	Das anwenderspezifische ASUP wird zuerst im Anwender-Verzeichnis _N_CUS_DIR gesucht.
	1	Das anwenderspezifische ASUP wird zuerst im Hersteller-Verzeichnis _N_CMA_DIR gesucht.

Festlegung einer Schutzstufe

Wenn ein anwenderspezifisches ASUP für RET und/oder REPOS benutzt werden soll (MD11610 \$MN_ASUP_EDITABLE ≠ 0), kann für die anwenderspezifische Routine "_N_ASUP_SPF" eine Schutzstufe festgelegt werden. Die Schutzstufe kann Werte im Bereich 0 - 7 annehmen. Die Einstellung erfolgt über das Maschinendatum:

MD11612 \$MN_ASUP_EDIT_PROTECTION_LEVEL = <Schutzstufe des anwenderspezifischen ASUP>

Details zu den Schutzstufen finden sich in:

Literatur:

Inbetriebnahmehandbuch; Schutzstufenkonzept

Verhalten bei gesetzter Einzelsatzbearbeitung

Über das folgende Maschinendatum wird eingestellt, ob trotz aktiver Einzelsatzbearbeitung die systemspezifischen ASUP und das anwenderspezifischen ASUP "_N_ASUP_SPF" ohne Unterbrechung abgearbeitet werden:

MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK, Bit 0 = <Wert>

Bit	Wert	Bedeutung
0	0	Bei aktivem Einzelsatzbetrieb wird in jedem ASUP-Satz angehalten.
	1	Bei aktivem Einzelsatzbetrieb wird das ASUP wird ohne Unterbrechung abgearbeitet .

10.11.3 Programmierung

Ermitteln der Ursache der ASUP-Aktivierung

Die Ursache, die zur Aktivierung des ASUP geführt hat, kann bitcodiert über die Systemvariable \$AC_ASUP gelesen werden.

Fortsetzung

Bei Verwendung der System-ASUP ist das Verhalten für die Fortsetzung nach Abarbeitung der Aktionen innerhalb des ASUP fest vorgegeben:

- System-ASUP1 → Fortsetzung mit `RET` (Unterprogrammrücksprung)
- System-ASUP2 → Fortsetzung mit `REPOS` (Repositionieren)

In der Beschreibung der Systemvariablen ist unter "Fortsetzung durch" für jede Ursache das Verhalten bezüglich der System-ASUP angegeben.

Hinweis

Fortsetzung bei anwenderspezifischem ASUP

Es wird empfohlen bei anwenderspezifischen ASUP die entsprechende Fortsetzung der System-ASUP beizubehalten.

Ursache: Betriebsartenwechsel (\$AC_ASUP, Bit 9 == 1)

Bei Betriebsartenwechsel ist die Fortsetzung abhängig vom Maschinendatum:

MD20114 \$MC_MODESWITCH_MASK (Unterbrechung MDA durch Betriebsartwechsel)

- Bit 0 == 0: System-ASUP1 → Fortsetzung mit `RET`
 - Bit 0 == 1: System-ASUP2 → Fortsetzung mit `REPOS`
-

Literatur

Eine ausführliche Beschreibung der Systemvariable findet sich in:

Listenhandbuch Systemvariable

10.12 ASUP-Start bei Anwenderalarmen durchführen

10.12.1 Funktion

Beschreibung

Ein ASUP kann in verschiedenen Situationen ausgelöst werden, entweder vom Anwender, vom System oder ereignisgesteuert. Ein Anwenderalarm mit der Alarmreaktion "Verriegelung NC-Start" verhindert in manchen Situationen den ASUP-Start. Bei anstehender Alarmreaktion wird ein Anwender-ASUP aus Reset mit dem Alarm 16906 quittiert.

Durch entsprechendes Setzen eines kanalspezifischen Maschinendatums ist es möglich ein ASUP zu starten und auszuführen, obwohl ein Anwenderalarm mit der Alarmreaktion

"Verriegelung NC-Start" aktiv ist. Die Alarmreaktion wird für den ASUP-Start überbrückt und lässt die Ausführung zu.

Hinweis

NC-Alarme mit der Alarmreaktion "Verriegelung NC-Start" werden durch die Überbrückung nicht beeinflusst. Ein Anwender-ASUP aus Reset ist nach wie vor nicht möglich und wird mit dem Alarm 16906 abgelehnt.

Überbrückbare Anwenderalarme

Das Nummerband für die überbrückbaren Anwenderalarme ist dafür wie folgt eingeteilt:

Nummernbereich	Auswirkung	Löschen
65000 - 65499	Anzeige, Verriegelung NC-Start	Reset
65500 - 65999	Anzeige, Verriegelung NC-Start (nicht für ASUPs bei eingestelltem MD20194)	Reset
66000 - 66999	Anzeige, Verriegelung NC-Start, Bewegungsstillstand nach Abarbeitung der vordecodierten Sätze	Reset
67000 - 67999	Anzeige	Cancel
68000 - 68999	Anzeige, Verriegelung NC-Start, sofortiger Interpolator-Stopp	Reset
69000 - 69999	Anzeige, Verriegelung NC-Start, Stopp am nächsten Satzende	Reset

Hinweis

Bei folgenden Alarmnummernbändern ist es zwar möglich die Alarmreaktion "Verriegelung NC-Start" zu überbrücken, die anderen Reaktionen der jeweiligen Alarme sorgen aber für Stopp-Bedingungen. Die Stopp-Bedingungen können mit der in diesem Kapitel beschriebenen Funktion nicht überbrückt werden:

- 66000 – 66999
- 68000 – 68999
- 69000 – 69999

10.12.2 Aktivierung

Einstellung

Über folgendes kanalspezifische Maschinendatum kann jeder ASUP-Kanal separat eingestellt werden:

MD20194 \$MC_IGNORE_NONCSTART_ASUP (ASUP-Start trotz anstehender Alarmreaktion "Verriegelung NC-Start" bei bestimmten Anwenderalarmen zulässig)

Eine Änderung der MD-Einstellung wird erst durch den Teileprogramm-Befehl `NEWCONF` oder über die Bedienoberfläche per Softkey wirksam.

Ablauf

Der normale Ablauf für den ASUP-Start sieht wie folgt aus:

- Maschinendatum entsprechend MD20194 \$MC_IGNORE_NONCSTART_ASUP setzen und per `NEWCONF` aktivieren.
- Teileprogramm starten.
Es erscheint ein Anwenderalarm aus den überbrückbaren Nummernbändern, z. B. Alarm 65500. Das kann aufgrund einer Synchronaktion oder durch einen Teileprogrammbefehl geschehen.
- Das Teileprogramm wird trotz Alarm bis zum Programmende `M02`, `M30` oder `M17` abgearbeitet.
Der Kanalzustand Reset ist aktiv.
- Ein nun gestartetes Anwender-ASUP aus Reset wird ausgeführt.

Hinweis

Ein ASUP-Start aus einem laufenden oder angehaltenen Teileprogramm ist auch bei anstehender Alarmreaktion "Verriegelung NC-Start" erlaubt. Es ist irrelevant, ob es sich dabei um einen Anwenderalarm oder einen NC-Alarm mit der Alarmreaktion "Verriegelung NC-Start" handelt.

Hinweis

Wenn während der Abarbeitung des Anwender-ASUPs das Teileprogramm angehalten wird, kann das ASUP nicht mehr durch die NC-Start-Taste fortgeführt werden, z. B. bei `M0` im ASUP oder bei Anwender-Stopp. Die Alarmreaktion "Verriegelung NC-Start" wird negativ mit dem Alarm 16906 abgelehnt. Der zuvor erzeugte Anwenderalarm kann nur noch mit Reset quittiert werden.

10.12.3 Beispiele

10.12.3.1 Anwender-ASUP aus Reset - Beispiel 1

In der Anwendung ist MD20194 nicht gesetzt.

Hauptprogramm

```
Programmcode
N10 G90 G0 Z10
N20 SETAL(65500)
N30 X100
N40 Z0
N50 M30
```

Anwender-ASUP

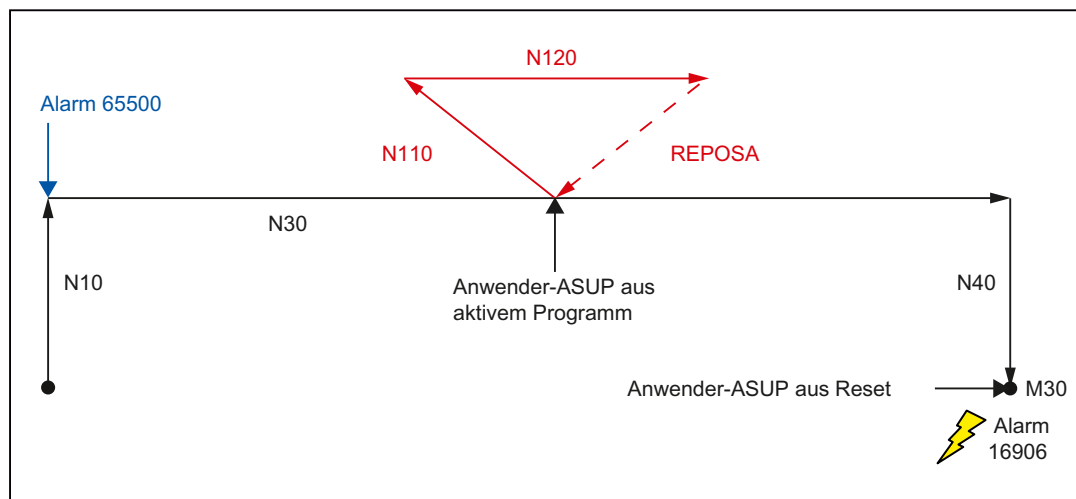
Programmcode

```

N110 G91 G0 X-10 Z5
N120 X20
N130 REPOSA

```

Ablauf



Der Satz **N10** wird abgearbeitet. Es erscheint der Alarm 65500, der die Alarmreaktionen "Anzeige" und "Verriegelung NC-Start" beinhaltet. Das Teileprogramm hält daraufhin nicht an. Der Satz **N30** wird eingewechselt und abgefahren. Wenn das Anwender-ASUP in der Satzmitte zum Einsatz kommt, wird es trotz anstehender Alarmreaktion "Verriegelung NC-Start" ausgeführt. Das **REPOSA** setzt wieder an der Programmunterbrechung an und fährt das Teileprogramm bis zum Programmende **M30** ab. Wenn das Anwender-ASUP jetzt zum Einsatz kommt, wird es mit dem Alarm 16906 abgelehnt. Der NC-Zustand befindet sich im Reset.

10.12.3.2 Anwender-ASUP aus Reset - Beispiel 2

Die Anwendung ist der Normalfall, MD20194 ist gesetzt.

Hauptprogramm

Programmcode

```

N4 $MC_IGNORE_NONCSTART_ASUP=1
N6 NEWCONF
N10 G90 G0 Z10
N20 SETAL(65500)
N30 X100
N40 Z0
N50 M30

```

Anwender-ASUP

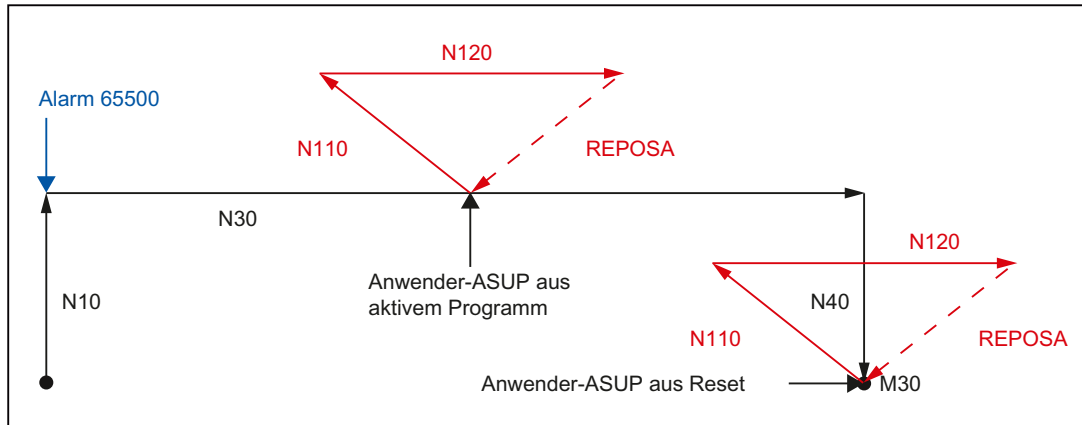
Programmcode

```

N110 G91 G0 X-10 Z5
N120 X20
N130 REPOSA

```

Ablauf



Das Maschinendatum MD20194 \$MC_IGNORE_NONCSTART_ASUP wird für ASUP-Kanal 1 gesetzt und per `NEWCONF` aktiviert. Der Satz `N10` wird abgearbeitet. Es erscheint der Alarm 65500, der die Alarmreaktionen "Anzeige" und "Verriegelung NC-Start" beinhaltet. Das Teileprogramm hält daraufhin nicht an. Der Satz `N30` wird eingewechselt und abgefahren. Wenn das Anwender-ASUP in der Satzmitte zum Einsatz kommt, wird es trotz anstehender Alarmreaktion "Verriegelung NC-Start" ausgeführt. Das `REPOSA` setzt wieder an der Programmunterbrechung an und fährt das Teileprogramm bis zum Programmende `M30` ab. Wenn das Anwender-ASUP jetzt zum Einsatz kommt, wird es aufgrund des gesetzten Maschinendatums MD20194 wiederholt ausgeführt.

10.12.3.3 Anwender-ASUP mit M0

In der Anwendung ist MD20194 gesetzt.

Hauptprogramm

Programmcode

```

N4 $MC_IGNORE_NONCSTART_ASUP=1
N6 NEWCONF
N10 G90 G0 Z10
N20 SETAL(65500)
N30 X100
N40 Z0
N50 M30

```


Anwender-ASUP

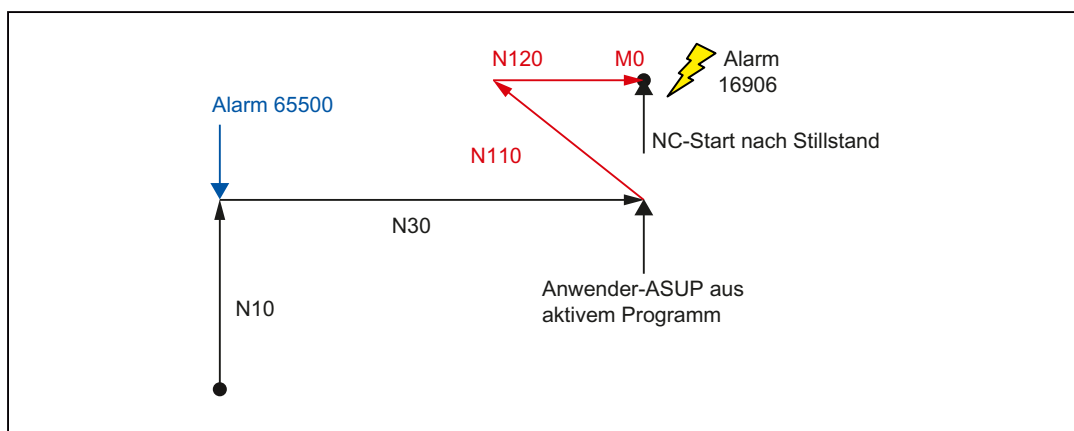
Programmcode

```

N110 G91 G0 X-10 Z5
N120 X10
N122 M0
N124 X10
N130 REPOSA

```

Ablauf



Das Maschinendatum MD20194 \$MC_IGNORE_NONCSTART_ASUP wird für ASUP-Kanal 1 gesetzt und per `NEWCONF` aktiviert. Der Satz `N10` wird abgearbeitet. Es erscheint der Alarm 65500, der die Alarmreaktionen "Anzeige" und "Verriegelung NC-Start" beinhaltet. Das Teileprogramm hält daraufhin nicht an. Der Satz `N30` wird eingewechselt und abgefahren. Wenn das Anwender-ASUP in der Satzmitte zum Einsatz kommt, wird es trotz anstehender Alarmreaktion "Verriegelung NC-Start" ausgeführt. An Satz `N122` hält das Teileprogramm wegen `M0` an. Das ASUP kann nicht mehr durch drücken der NC-Start-Taste fortgeführt werden. Die Alarmreaktion "Verriegelung NC-Start" wird mit dem Alarm 16906 abgelehnt. Der zuvor erzeugte Alarm 65500 kann nur noch mit Reset quittiert werden.

10.12.3.4 Anwender-ASUP mit Stopp

In der Anwendung ist MD20194 gesetzt.

Hauptprogramm

Programmcode

```

N4 $MC_IGNORE_NONCSTART_ASUP=1
N6 NEWCONF
N10 G90 G0 Z10
N20 SETAL(65500)
N30 X100
N40 Z0

```

Programmcode

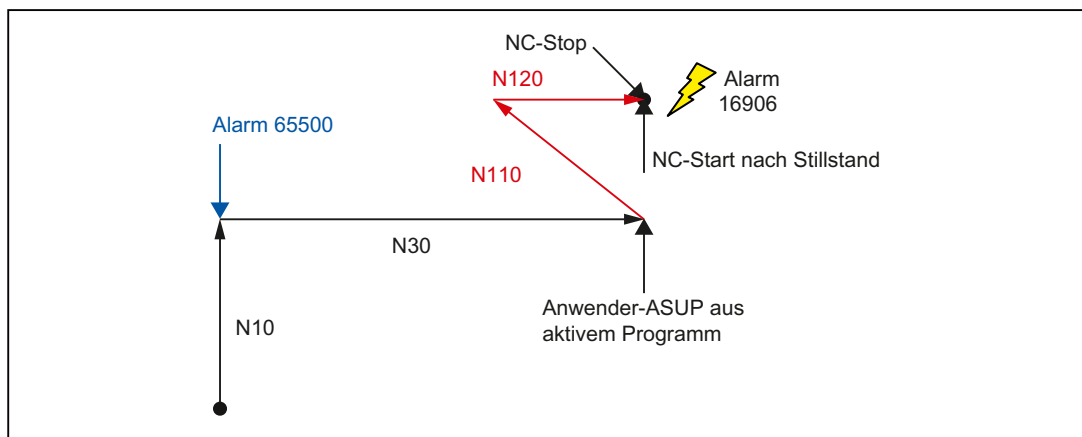
N50 M30

Anwender-ASUP**Programmcode**

N110 G91 G0 X-10 Z5

N120 X20

N130 REPOSA

Ablauf

Das Maschinendatum MD20194 \$MC_IGNORE_NONCSTART_ASUP wird für ASUP-Kanal 1 gesetzt und per `NEWCONF` aktiviert. Der Satz N10 wird abgearbeitet. Es erscheint der Alarm 65500, der die Alarmreaktionen "Anzeige" und "Verriegelung NC-Start" beinhaltet. Das Teileprogramm hält daraufhin nicht an. Der Satz N30 wird eingewechselt und abgefahren. Wenn das Anwender-ASUP in der Satzmitte zum Einsatz kommt, wird es trotz anstehender Alarmreaktion "Verriegelung NC-Start" ausgeführt. Wenn bei N120 in der Satzmitte die NC-Stop-Taste gedrückt wird, hält das ASUP an. Das ASUP kann nicht mehr durch drücken der NC-Start-Taste fortgeführt werden. Die Alarmreaktion "Verriegelung NC-Start" wird mit dem Alarm 16906 abgelehnt. Der zuvor erzeugte Alarm 65500 kann nur noch mit Reset quittiert werden.

10.12.3.5 Anwender-ASUP aus gestoppt

In der Anwendung ist MD20194 gesetzt oder nicht gesetzt.

Hauptprogramm**Programmcode**

N10 G90 G0 Z10

N20 SETAL(65500)

N30 X50

Programmcode

```

N35 M0
N38 X100
N40 Z0
N50 M30

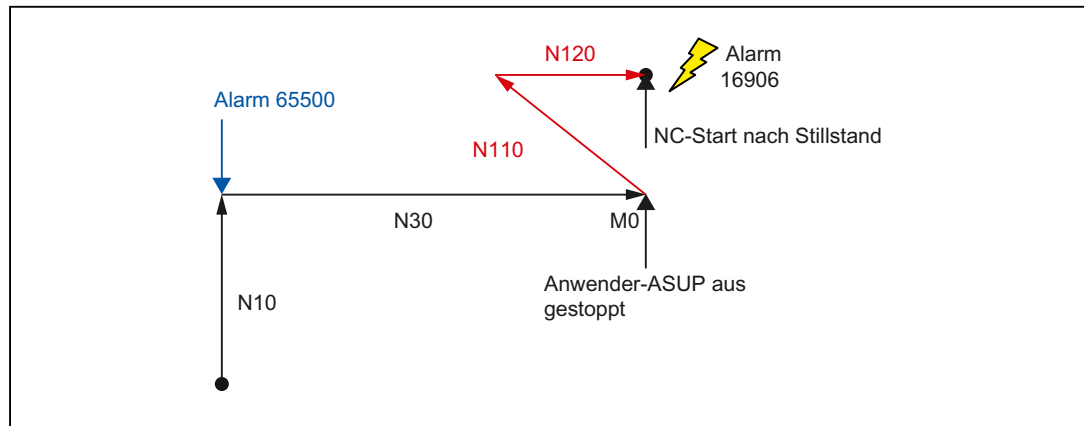
```

Anwender-ASUP**Programmcode**

```

N110 G91 G0 X-10 Z5
N120 X20
N130 REPOSA

```

Ablauf

Der Satz **N10** wird abgearbeitet. Es erscheint der Alarm 65500, der die Alarmreaktionen "Anzeige" und "Verriegelung NC-Start" beinhaltet. Das Teileprogramm hält daraufhin nicht an. Der Satz **N30** wird eingewechselt und abgefahren. An Satz **N35** hält das Teileprogramm wegen **M0** an. Der NC-Zustand befindet sich im Status gestoppt. Wenn das Anwender-ASUP jetzt zum Einsatz kommt, wird es trotz gestopptem NC-Zustand und anstehender Alarmreaktion "Verriegelung NC-Start" ausgeführt. Es ist dabei egal, ob das Maschinendatum MD20194 `$MC_IGNORE_NONCSTART_ASUP` gesetzt ist oder nicht. Nach Satz **N120** bleibt das ASUP vor `REPOSA` stehen. Die Wiederanfahrbewegung ist erst wieder mit dem nächsten NC-Start möglich. Der NC-Start wird aber aufgrund der Alarmreaktion "Verriegelung NC-Start" mit Alarm 16906 abgelehnt. Der zuvor erzeugte Alarm 65500 kann nur noch mit Reset quittiert werden.

10.13 Einzelsatz

Die Funktion "Einzelsatz" ermöglicht auf drei verschiedene Arten das satzweise Abarbeiten eines NC-Programms.

10.13 Einzelsatz

Folgende Einzelsatztypen stehen zur Verfügung:

- **"SBL1: Einzelsatz grob"** (Stopp nach jedem Hauptlauf-Satz)
Das NC-Programm bzw. die Bearbeitung wird nach jedem vollständig abgearbeiteten Hauptlaufsatz angehalten bzw. gestoppt.
- **"SBL2: Rechensatz"** (Stopp nach jedem decodierten Satz)
Das NC-Programm bzw. die Bearbeitung wird nach jedem decodierten Satz angehalten bzw. gestoppt.
Werden aus einem decodierten Satz mehrere Hauptlaufsätze erzeugt, wird nach jedem Hauptlaufsatz angehalten bzw. gestoppt.

Hinweis

Gewindeschneidsätze

Bei Gewindeschneidsätzen wird das NC-Programm nicht angehalten bzw. die Bearbeitung nicht gestoppt.

- **"SBL3: Einzelsatz fein"** (Stopp nach jedem decodierten Satz auch in einem Zyklus)
Wie SBL2, jedoch wird hier auch innerhalb eines Zyklus nach jedem Satz das NC-Programm angehalten bzw. die Bearbeitung gestoppt..

Bedienoberfläche: Anwahl des Einzelsatztyps

Die Anwahl des Einzelsatztyps SBL1, SBL2 oder SBL3 erfolgt über die HMI-Bedienoberfläche:

Bedienbereich "Maschine" > "Prog.beeinfl." > Menü: "Programmbeeinflussung"

10.13.1 Parametrierung

Nahtstellensignale

Aktivierung

Die Funktion "Einzelsatz" wird kanalspezifisch über das NC/PLC-Nahtstellensignal aktiviert:

DB21, ... DBX0.4 (Einzelsatz aktivieren)

Maschinendaten

Einzelsatzbearbeitung ausschalten (MD10702, MD20106, MD20117)

- Mit dem Maschinendatum kann für bestimmte Bearbeitungssituationen und Programmtypen eingestellt werden, dass trotz aktiver Einzelsatzfunktion nicht angehalten wird:

MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK

Hinweis

- Durch Programmierung (Seite 618) von `SBLON/SBLOF` innerhalb eines ASUPs oder Unterprogramms kann die Einzelsatzbearbeitung explizit ein/ausgeschaltet werden.
 - Bei Einzelsatztyp "SBL2: Rechensatz" wirkt das Maschinendatum nur bei System-ASUPs, Anwender-ASUPs und Unterprogrammen mit dem Attribut `DISPLOF`.
-

- Das Verhalten von ereignisgesteuerten Programmaufrufe (Prog-Events) bezüglich Einzelsatz wird eingestellt über:
MD20106 \$MC_PROG_EVENT_IGN_SINGLEBLOCK
- Das Verhalten von Interruptprogrammen (ASUP) bezüglich Einzelsatz wird eingestellt über:
MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP
Wird während des Einzelsatzes ein ASUP aktiviert, wird das ASUP vollständig abgearbeitet. Der Einzelsatz wirkt erst wieder nach dem ASUP bzw. im ersten Hauptlaufsatz, in dem keine Einzelsatz-Unterdrückung aktiviert ist. Ist beim Übergang von ASUP zu NC-Programm die Bahngeschwindigkeit zu groß, als dass noch im Folgesatz bis zum Stillstand abgebremst werden könnte, z. B. bei aktivem Bahnsteuerbetrieb G64, erfolgt der Bremsvorgang eventuell auch über mehrere noch folgende Sätze.

Hinweis

Durch Programmierung (Seite 618) von `SBLON` innerhalb eines ASUPs kann in diesem Fall die Einzelsatzbearbeitung **nicht** wieder eingeschaltet werden.

Settingdaten

Debugmode für Einzelsatz "SBL2: Rechensatz" (SD42200)

Bedingt durch die vorausseilende Dekodierung der Sätze, kann auf der Bedienoberfläche der Bezug zwischen der hauptlaufbezogenen aktuellen Satzanzeige und den angezeigten Variablenwerten verloren gehen. Es werden dann u. U. nicht plausible Variablenwerte angezeigt.

Mit dem folgenden kanalspezifischen Settingdatum kann eingestellt werden, dass bei aktivem Einzelsatz "SBL2: Rechensatz" bei jedem Satz ein Vorlaufstopp ausgeführt wird. Dadurch wird die vorausseilende Bearbeitung der Teileprogrammsätze unterdrückt und der Bezug zwischen aktueller Satzanzeige und Anzeige der Variablenwerte bleibt erhalten.

SD42200 \$SC_SINGLEBLOCK2_STOPRE (Debugmode für SBL2 aktivieren)

Hinweis

Konturabweichung

Beim Abarbeiten von Verfahrssätzen mit Einzelsatztyp "**SBL2: Rechensatz**" im **Debugmode** kann es zu Konturabweichungen kommen.

10.13.2 Programmierung

10.13.2.1 Einzelsatzbearbeitung aus/einschalten (SBLOF, SBLON)

Einzelsatzbearbeitung für das gesamte NC-Programm ausschalten

Ist das Ausschalten der Einzelsatzbearbeitung (SBLOF) in der ersten Zeile (PROC) eines **Hauptprogramms** programmiert, gilt dies bis zum Ende oder Abbruch des NC-Programms. Das NC-Programm wird dann bei Einzelsatz ohne anzuhalten abgearbeitet.

Ist das Ausschalten der Einzelsatzbearbeitung (SBLOF) in der ersten Zeile (PROC) eines **Unterprogramms** programmiert, gilt dies bis zum Ende oder Abbruch des NC-Programms. Mit dem programmierten Rücksprung-Befehl wird entschieden, ob am Ende des Unterprogramms angehalten wird oder nicht:

- Rücksprung mit M17: Stopp am Ende des Unterprogramms
- Rücksprung mit RET: **Kein** Stopp am Ende des Unterprogramms

Einzelsatzbearbeitung innerhalb des NC-Programms ausschalten

Ist das Ausschalten der Einzelsatzbearbeitung (SBLOF) in einem Satz innerhalb eines NC-Programms programmiert, wird die Einzelsatzbearbeitung ab diesem Satz bis zum nächsten programmierten Einschalten der Einzelsatzbearbeitung (SBLON) oder dem Ende der aktiven Unterprogrammebene ausgeschaltet.

Syntax

SBLOF
SBLON

Bedeutung

SBLOF:	Vordefinierte Prozedur zum Ausschalten der Einzelsatzbearbeitung	
	Alleine im Satz:	ja, im PROC-Satz möglich
	Wirksamkeit:	Modal
SBLON:	Vordefinierte Prozedur zum Einschalten der Einzelsatzbearbeitung	
	Alleine im Satz:	ja
	Wirksamkeit:	Modal

10.13.2.2 Randbedingungen

Einzelsatzunterdrückung und Satzanzeige

Die aktuelle Satzanzeige kann in Unterprogrammen mit `DISPLOF` unterdrückt werden. Wird `DISPLOF` zusammen mit `SBLOF` programmiert, wird bei Einzelsatz-Stopps innerhalb des Unterprogramms der Aufruf des Unterprogramms angezeigt.

Besonderheiten verschiedener Einzelsatztypen

- "SBL2: Rechensatz" UND MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK, Bit 12 == 1:
Im `SBLON`-Satz wird nicht angehalten.
- "SBL3: Einzelsatz fein": Der Befehl `SBLOF` wird unterdrückt

Einzelsatzunterdrückung für asynchrone Unterprogramme (ASUP)

Damit ein ASUP auch bei aktiver Einzelsatzbearbeitung ohne anzuhalten abgearbeitet wird, muss in der ersten Programmzeile des ASUPs `PROC` zusammen mit `SBLOF` programmiert werden.

Beispiel

Programmcode	Kommentar
N10 PROC ASUP1 SBLOF DISPLOF	
N20 IF \$AC_ASUP=='H200'	
N30 RET	; Kein REPOS bei BA-Wechsel.
N40 ELSE	
N50 REPOSA	; REPOS in allen übrigen Fällen.
N60 ENDIF	

10.13.2.3 Beispiele

Beispiel 1: Einzelsatzunterdrückung innerhalb eines Programms

Programmcode	Kommentar
N10 G1 X100 F1000	
N20 SBLOF	; Einzelsatz ausschalten
N30 Y20	
N40 M100	
N50 R10=90	
N60 SBLON	; Einzelsatz einschalten
N70 M110	
N80 ...	

Der Bereich zwischen N20 und N60 wird bei aktivem Einzelsatzbetrieb wie ein Satz behandelt.

Beispiel 2: Unterprogramm ohne anzuhalten**Hauptprogramm**

Programmcode	
...	
N100	G1 X10 G90 F200
N120	X-4 Y6
N130	CYCLE1
N140	G1 X0
N150	M30

Unterprogramm

Programmcode	Kommentar
N100 PROC CYCLE1 DISPLOF SBLOF	; Einzelsatz unterdrücken
N110 R10=3*SIN(R20)+5	
N120 IF (R11 <= 0)	
N130 SETAL(61000)	
N140 ENDIF	
N150 G1 G91 Z=R10 F=R11	
N160 M17	

Der Zyklus wird auch bei aktiver Einzelsatzbearbeitung vollständig abgearbeitet.

Beispiel 3: ASUP mit Einzelsatzunterdrückung und nicht sichtbar

Die Bearbeitung eines vom PLC-Anwenderprogramm gestarteten ASUPs soll auch bei aktiver Einzelsatzbearbeitung nicht unterbrochen werden. Zusätzlich soll das ASUP auch nicht sichtbar sein.

Programmcode	
N100 PROC NV SBLOF DISPLOF	; Einzelsatz- und Anzeigeunterdrückung
N110 CASE \$P_UIFRNUM OF	
0 GOTOF _G500	
1 GOTOF _G54	
2 GOTOF _G55	
3 GOTOF _G56	
4 GOTOF _G57	
DEFAULT GOTOF END	
N120 _G54: G54 D=\$P_TOOL T=\$P_TOOLNO	
N130 RET	
N140 _G54: G55 D=\$P_TOOL T=\$P_TOOLNO	
N150 RET	
N160 _G56: G56 D=\$P_TOOL T=\$P_TOOLNO	
N170 RET	
N180 _G57: G57 D=\$P_TOOL T=\$P_TOOLNO	
N190 RET	

Programmcode
N200 END: D=\$P_TOOL T=\$P_TOOLNO
N210 RET

Beispiel 4: Gezieltes Anhalten im Unterprogramm

Annahmen:

- Einzelsatzbearbeitung ist aktiv.
- MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK, Bit12 = 1

Hauptprogramm

Programmcode	Kommentar
N10 G0 X0	; Einzelsatz-Stopp
N20 X10	; Einzelsatz-Stopp
N30 CYCLE	; Vom Zyklus generierter Verfahrssatz.
N50 G90 X20	; Einzelsatz-Stopp
M30	

Unterprogramm

Programmcode	Kommentar
PROC CYCLE SBLOF	; Einzelsatzunterdrückung
N100 R0 = 1	
N110 SBLON	; Kein Einzelsatz-Stopp wegen MD10702, Bit12 = 1
N120 X1	; Einzelsatz-Stopp
N140 SBLOF	
N150 R0 = 2	
RET	

Beispiel 5: Einzelsatzunterdrückung bei Programmschachtelung

Annahme: Einzelsatzbearbeitung ist aktiv.

Programmcode	Kommentar
N10 X0 F1000	; Einzelsatz-Stopp
N20 UP1(0)	
PROC UP1(INT _NR) SBLOF	; Einzelsatz für UP1 ausschalten
N100 X10	
N110 UP2(0)	
PROC UP2(INT _NR)	
N200 X20	
N210 SBLON	; Einzelsatz einschalten
N220 X22	; Einzelsatz-Stopp
N230 UP3(0)	
PROC UP3(INT _NR)	

10.13 Einzelsatz

Programmcode	Kommentar
N300 SBLOF	; Einzelsatz ausschalten
N305 X30	
N310 SBLOF	; Einzelsatz einschalten
N320 X32	; Einzelsatz-Stopp
N330 SBLOF	; Einzelsatz ausschalten
N340 X34	
N350 M17	; Einzelsatz-Stopp (M17)
N240 X24	; Einzelsatz-Stopp (N210)
N250 M17	; Einzelsatz-Stopp (M17)
N120 X12	
N130 M17	; Einzelsatz-Stopp (M17)
N30 X0	; Einzelsatz-Stopp
N40 M30	; Einzelsatz-Stopp

10.13.3 BAG-spezifischer Einzelsatz Typ A / B

Beim BAG-spezifischen Einzelsatz wird in einem Kanal (Steuerkanal) das NC-Programm per Einzelsatz satzweise abgearbeitet. Im Steuerkanal muss Einzelsatz per NC/PLC-Nahtstellensignal (DB21 ... DBX0.4) aktiviert sein.

In den übrigen Kanälen der BAG (Abhängigen Kanälen) wird das jeweilige NC-Programm satzweise entsprechend dem BAG-spezifisch per NC/PLC-Nahtstellensignal gewähltem Einzelsatz Typ A oder B (DB11 DBX1.6 / 7) abgearbeitet. In den abhängigen Kanälen darf das NC/PLC-Nahtstellensignal (DB21 ... DBX0.4) nicht gesetzt sein.

Einzelsatz Typ A / B

- Einzelsatz Typ A: Stoppt der Steuerkanal, stoppen sofort auch die abhängigen Kanäle, vergleichbar mit NC-Stop.
- Einzelsatz Typ B: Stoppt der Steuerkanal, stoppen die abhängigen Kanäle am jeweiligen Satzende, vergleichbar mit NC-Stop an der Satzgrenze.

Nahtstellensignale**Steuerkanal**

- DB21 ... DBX0.4 (Einzelsatz aktivieren)

Alle Kanäle der BAG

- DB21 ... DBX7.1 (NC-Start)

BAG

- DB11 DBX1.6 (Einzelsatz Typ B)
- DB11 DBX1.7 (Einzelsatz Typ A)

Schematischer Ablauf für Einzelsatz Typ A

Voraussetzung: Alle Kanäle der BAG sind im Zustand "Reset" oder "Unterbrochen".

1. PLC-Anwenderprogramm: Einzelsatz im Steuerkanal anwählen, DB21 ... DBX0.4 = 1
2. PLC-Anwenderprogramm: Einzelsatz Typ A für die BAG anwählen, DB11 DBX1.7 = 1
3. PLC-Anwenderprogramm: Alle Kanäle der BAG starten, DB21 ... DBX0.4 = 1
4. Der Steuerkanal stoppt am Satzende.
5. Alle abhängigen Kanäle erhalten ein internes Signal zum sofortigen Stoppen der Bearbeitung.
6. Alle Kanäle der BAG sind im Zustand "Unterbrochen", wenn alle abhängigen Kanäle das jeweilige Ende der Bremsphase erreicht haben.

Schematischer Ablauf für Einzelsatz Typ B

Voraussetzung: Alle Kanäle der BAG sind im Zustand "Reset" oder "Unterbrochen".

1. PLC-Anwenderprogramm: Einzelsatz im Steuerkanal anwählen, DB21 ... DBX0.4 = 1
2. PLC-Anwenderprogramm: Einzelsatz Typ B für die BAG anwählen, DB11 DBX1.6 = 1
3. PLC-Anwenderprogramm: Alle Kanäle der BAG starten, DB21 ... DBX0.4 = 1
4. Der Steuerkanal stoppt am Satzende.
5. Alle abhängigen Kanäle erhalten ein internes Signal zum Stoppen der Bearbeitung am Satzende.
6. Alle Kanäle der BAG sind Zustand "Unterbrochen", wenn alle abhängigen Kanäle ihr jeweiliges Satzende erreicht haben.

10.13.4 Randbedingungen

10.13.4.1 Einzelsatztyp SBL2 und satzbezogene Synchronaktionen

Bei Einzelsatztyp **"SBL2: Rechensatz"** wird bei einer satzbezogenen Synchronaktion der nächste Stopp erst nach dem nächsten Hauptlaufsatz ausgeführt. Bei den zwischen der Synchronaktion und dem nächsten Hauptlaufsatz liegenden Vorlaufsätzen wird nicht angehalten.

10.13.4.2 Programmierter Halt (M0), Einzelsatz und Einzelsatztypumschaltung

Ausgangssituation: In einem Kanal ist ein NC-Programm durch ein darin programmiertes **M0** angehalten und im Kanal ist **Einzelsatz aktiv** (DB21, ... DBX0.4 == 1)

Wird in dieser Situation der **Einzelsatztyp** an der Bedienoberfläche **mehrfach** zwischen SBL1 oder SBL3 und SBL2 umgeschaltet, wird der Alarm 16922 "Maximale Schachteltiefe überschritten" angezeigt.

10.14 Programmbeeinflussung

Möglichkeiten

1. Funktionenwahl über Bedienoberfläche oder über PLC
2. Aktivierung von Ausblendeebenen
3. Größenanpassung des Interpolationspuffers
4. Darstellungsweise der Programmanzeige über eine zusätzliche Basis-Satzanzeige
5. Abarbeiten von extern (Puffergröße und -anzahl)
6. Abarbeiten von externen Unterprogrammen

10.14.1 Funktionenwahl über Bedienoberfläche oder PLC-Anwenderprogramm

Ablauf

Anwahl

Unter Bedienbereich "Automatik" > horizontaler Softkey "**Programmbeeinflussung**" können die Funktionen von der Bedienoberfläche aus angewählt werden. Durch HMI wird daraufhin ein Anforderungssignal in der HMI/PLC-Nahstelle gesetzt.

Aktivierung

Durch das PLC-Anwenderprogramm muss die Anforderung von HMI durch Setzen des zugehörigen Anforderungssignals an die NC weitergeleitet werden.

Anwenderspezifisch können die Funktionen auch unabhängig von der Bedienoberfläche direkt durch das PLC-Anwenderprogramm durch Setzen des zugehörigen Anforderungssignals an die NC aktiviert werden.

Rückmeldung

Die Aktivierung einer Funktion wird von der NC an die PLC zurückgemeldet (siehe Tabelle).

NC/PLC-Nahstellensignale

Tabelle 10-2 Programmbeeinflussung: Nahstellen-Signale

Funktion	Anwahlsignal	Aktivierungssignal	Rückmeldesignal
SKP "Ausblendsatz 0 bis 7"	DB21, ... DBX26.0 - 7	DB21, ... DBX2.0 - 7	---
SKP "Ausblendsatz 8 bis 9"	DB21, ... DBX27.0 - 1	DB21, ... DBX31.6 - 7	---
DRY "Probelauf Vorschub"	DB21, ... DBX24.6	DB21, ... DBX0.6	DB21, ... DBX318.6
RG0 "Korrektur Eilgang"	DB21, ... DBX25.3	DB21, ... DBX6.6	---
"Einzelsatz"	Vorwahl von SBL1, SBL2 oder SBL3 über Programmbeeinflussungsanzeige von HMI		

Funktion	Anwahlsignal	Aktivierungssignal	Rückmeldesignal
SBL1 "Hauptlauf-Einzelsatz"	Bedientafel HMI	DB21, ... DBX0.4	---
SBL2 "Dekodier-Einzelsatz"			
SBL3 "Im Zyklus"			
M01 "programmierter Halt"	DB21, ... DBX24.5	DB21, ... DBX0.5	DB21, ... DBX32.5
Assoziiertes M01	DB21, ... DBX24.4	DB21, ... DBX30.5	DB21, ... DBX318.5
DRF "Handrad-Verschiebung"	DB21, ... DBX24.3	DB21, ... DBX0.3	DB21, ... DBX33.3
PRT "keine Achsbewegungen, Programmtest"	DB21, ... DBX25.7	DB21, ... DBX1.7	DB21, ... DBX33.7

Literatur

- Listenhandbuch NC-Variablen und Nahtstellensignale
- Bedienungshandbuch HMI Advanced "Bedienbereich Maschine"

10.14.2 Aktivierung von Ausblendebeinen

Funktion

Sätze, die nicht bei jedem Programmlauf ausgeführt werden sollen, können ausgeblendet werden. Die Sätze, die ausgeblendet werden sollen, werden mit dem Zeichen "/" (Schrägstrich) vor der Satznummer gekennzeichnet.

Im Teileprogramm werden die Ausblendebeinen mit "/0" bis "/9" angegeben.

Pro Teileprogrammsatz kann nur eine Ausblendeebene angegeben werden.

Parametrierung

Die Anzahl der Ausblendebeinen wird festgelegt mit dem Maschinendatum:

MD51029 \$MM_MAX_SKP_LEVEL (Max. Anzahl der Ausblendebeinen im NC-Programm)

Programmierung

Sätze, die nicht bei jedem Programmlauf ausgeführt werden sollen (z. B. Programm-Einfahrsätze), können nach folgendem Schema ausgeblendet werden:

Programmcode	Kommentar
/N005	; Satz ausgeblendet, (DB21,... DBX2.0) 1.Ausblendeebene
/0 N005	; Satz ausgeblendet, (DB21,... DBX2.0) 1.Ausblendeebene
/1 N010	; Satz ausgeblendet, (DB21,... DBX2.1) 2. Ausblendeebene
/2 N020	; Satz ausgeblendet, (DB21,... DBX2.2) 3.Ausblendeebene
/3 N030	; Satz ausgeblendet, (DB21,... DBX2.3) 4.Ausblendeebene
/4 N040	; Satz ausgeblendet, (DB21,... DBX2.4) 5.Ausblendeebene
/5 N050	; Satz ausgeblendet, (DB21,... DBX2.5) 6.Ausblendeebene

Programmcode	Kommentar
/6 N060	; Satz ausgeblendet, (DB21,... DBX2.6), 7.Ausblendeebene
/7 N070	; Satz ausgeblendet, (DB21,... DBX2.7) 8.Ausblendeebene
/8 N080	; Satz ausgeblendet, (DB21,... DBX31.6) 9.Ausblendeebene
/9 N090	; Satz ausgeblendet, (DB21,... DBX31.7) 10.Ausblendeebene

Aktivierung

Die 10 Ausblendeebenen "/0" bis "/9" werden vom PLC durch Setzen der Nahtstellensignale PLC → NC aktiviert.

Die Aktivierung der Funktion von HMI über das Menü "Programmbeeinflussung" im Bedienbereich "Maschine" erfolgt:

- Für die Ausblendeebenen "/0" bis "/7":
über die Schnittstelle HMI → PLC DB21, ... DBB26 (Satzausblenden ausgewählt).
- Für die Ausblendeebenen "/8" bis "/9":
über die Schnittstelle HMI → PLC DB21, ... DBX27.0 bis DBX27.1.

Literatur:

Bedienhandbuch

Hinweis

Änderungen der auszublendenen Ebenen sind nur im STOP/RESET-Zustand der Steuerung möglich.

10.14.3 Größenanpassung des Interpolationspuffers

MD28060

Der kanalspezifische Interpolator arbeitet beim Teileprogrammablauf vorbereitete Sätze aus dem Interpolationspuffer ab. Die maximale Anzahl Sätze, die zu einem Zeitpunkt im Interpolationspuffer Platz finden sollen, wird durch das speicherkonfigurierende MD28060 \$MM_IPO_BUFFER_SIZE (Anzahl der NC-Sätze im IPO-Puffer(DRAM)) festgelegt. Für einige Anwendungen kann es sinnvoll sein, diesen Puffer nicht voll auszunutzen, um den "Abstand" zwischen Vorbereitung und Interpolation klein zu halten.

SD42990

Mit dem Settingdatum SD42990 \$SC_MAX_BLOCKS_IN_IPOBUFFER (Max. Anzahl Sätze im IPO-Puffer) kann dynamisch die Zahl der Sätze im Interpolationspuffer auf einen **kleineren** Wert als in MD28060 \$MC_MM_IPO_BUFFER_SIZE (Anzahl der NC-Sätze im IPO-Puffer (DRAM)) begrenzt werden, minimal auf 2 Sätze.

Werte des Settingdatums SD42990 \$SC_MAX_BLOCKS_IN_IPOBUFFER:

Wert	Wirkung
< 0	Keine Begrenzung des Interpolationspuffers aktiv. Der max. mögliche IPO-Puffer entsprechend MD 28060: MM_IPO_BUFFER_SIZE wird aktiviert.
oder 1	Der minimal zulässige IPO-Puffer mit 2 Sätzen wird aktiv.
< MM_IPO_BUFFER_SIZE	Der IPO-Puffer wird maximal mit der angegebenen Anzahl Sätze aktiviert.
>= MM_IPO_BUFFER_SIZE	Der IPO-Puffer wird mit der in MD 28060: MM_IPO_BUFFER_SIZE angegebenen Anzahl Sätze aktiviert.

Hinweis

Wenn das SD42990 \$SC_MAX_BLOCKS_IN_IPOBUFFER im Teileprogramm gesetzt wird, wird die Begrenzung des Interpolationspuffers sofort wirksam, wenn der Satz mit dem SD vom Interpreter in der Vorbereitung abgearbeitet wird.

D.h. gegebenenfalls wird die Begrenzung des IPO-Puffers einige Sätze vor der beabsichtigten Begrenzung wirksam (siehe auch das MD28070 \$MC_MM_NUM_BLOCKS_IN_PREP).

Will man das vermeiden, und die Begrenzung des IPO-Puffers soll satzsynchron wirksam werden, muss man vor dem Setzen des SD im Teileprogramm ein STOPRE (Vorlaufstopp) programmieren.

Gültigkeit

Das SD42990 \$SC_MAX_BLOCK_IN_IPOBUFFER hat globale, kanalspezifische Gültigkeit und kann auch in einem Teileprogramm verändert werden. Dieser veränderte Wert wird bei Programmende beibehalten. Soll dieses Settingdatum bei definierten Ereignissen wieder zurückgesetzt werden, muss dafür ein so genanntes ereignisgesteuertes Programm eingerichtet werden. Man könnte z. B. dieses Settingdatum bei RESET immer auf einen vordefinierten Wert setzen.

Anwendung

Die IPO-Pufferbegrenzung kann eingesetzt werden, wenn die Anzahl Sätze zwischen Satzaufbereitung und Interpolation möglichst klein sein soll, z. B. wenn im Teileprogramm Istpositionen gelesen und weiterverarbeitet werden sollen.

Beispiel

```

N10 ...
N20 ...
.....
N100 $SC_MAX_BLOCKS_IN_IPOBUFFER = 5           ; Begrenzung des IPO-Buffers auf 5
                                                NC-Sätze
N110 ...
N120 ...
.....

```

```
N200 $SC_MAX_BLOCKS_IN_IPOBUFFER = -1          ; Aufhebung der Begrenzung des IPO-  
                                                Puffers  
N210 ...  
.....
```

10.14.4 Darstellungsweise der Programmanzeige über eine zusätzliche Basis-Satzanzeige

Basis-Satzanzeige (nur bei ShopMill/ShopTurn)

Zur bestehenden Satzanzeige können über eine zweite, der so genannten Basis-Satzanzeige, alle Sätze angezeigt werden, die eine **Aktion an der Maschine** bewirken werden.

Vorausschauende Basis-Satzanzeige

Die tatsächlich angefahrenen Endpositionen werden als Absolutposition dargestellt. Die Positionswerte beziehen sich wahlweise auf das Werkstückkoordinatensystem (WKS) oder auf das Einstellbare Nullpunkt-System (ENS).

Die Anzahl der vorausschauend im Anzeigebuffer abgelegten Anzeigesätze hängt von der Anzahl der im jeweiligen Bearbeitungszustand aufbereiteten Sätze im Vorlauf des NCs ab. Wird ein Vorlaufstopp bearbeitet, so geht die Anzahl der Anzeigesätze gegen Null und steigt nach der Quittierung des Vorlaufstopps wieder an. Bei REORG-Ereignissen (z. B. Betriebsartenwechsel, ASUP-Start) werden die vorausschauend angelegten Anzeigesätze gelöscht und danach wieder aufbereitet.

Aufbereitete Werte

In der Basis-Satzanzeige aufbereitete Werte stimmen überein mit:

- Angewählten Werkzeugen
- Vorschub und der Spindeldrehzahl
- Tatsächlich angefahrenen Positionswerten

Ausnahmen:

Bei aktiver Werkzeugradiuskorrektur können Abweichungen auftreten.

Bei Modulo-Achsen wird in der Basis-Satzanzeige der programmierte Wert angezeigt, der auch außerhalb des Modulo-Bereichs liegen kann.

Hinweis

Grundsätzlich werden die Positionen im WKS oder im ENS dargestellt.

Die Funktion der Basis-Satzanzeige kann mit dem Settingdatum

SD42750 \$SC_ABSBLOCK_ENABLE (Basissatzanzeige freigeben) ein- oder ausgeschaltet werden.

10.14.5 Basis-Satzanzeige bei ShopMill/ShopTurn

Basis-Satzanzeige konfigurieren

Die Basis-Satzanzeige ist über folgende Maschinendaten konfigurierbar:

NC Maschinendaten für Basis-Satzanzeige	Bedeutung:
MD28400 \$MC_MM_ABSBLOCK	Basis-Satzanzeige aktivieren
MD28402 \$MC_MM_ABSBLOCK_BUFFER_CONF[2]	Größe des Anzeigebuffers
Anzeige Maschinendaten	einzustellende Positionswerte:
MD9004 \$MM_DISPLAY_RESOLUTION	für metrische Maßangabe
MD9011 \$MM_DISPLAY_RESOLUTION_INCH	für Inch Maßangabe
MD9010 \$MM_SPIND_DISPLAY_RESOLUTION	für Anzeigefeinheit Spindeln einzustellendes Koordinatensystem
MD9424 \$MM_MA_COORDINATE_SYSTEM	für Istwertanzeige WKS oder ENS

Diese Anzeigemaschinendaten werden in die NC-Maschinendaten MD17200 \$MN_GMMC_INFO_UNIT[0] (Globale HMI-Info) bis MD17200 \$MN_GMMC_INFO_UNIT[3] kopiert. Damit kann vom NC aus auf diese Anzeigemaschinendaten zugegriffen werden.

Aktivierung

Die Basis-Satzanzeige wird aktiviert durch das MD28400 \$MC_MM_ABSBLOCK (Satzanzeige mit Absolutwerten aktivieren) mittels Power On. Wird das MD28400 \$MC_MM_ABSBLOCK mit 1 beschrieben, so wird im Hochlauf ein kanalspezifischer Anzeigepuffer (FIFO) eingerichtet.

Größe des Anzeigebuffer (FIFO) = (MD28060 \$MC_MM_IPO_BUFFER_SIZE (Anzahl der NC-Sätze im IPO-Puffer) + MD28070 \$MC_MM_NUM_BLOCKS_IN_PREP (Anzahl der Sätze für die Satzaufbereitung)) multipliziert mit 128 Byte. Bei Standard-Maschinendateneinstellung entspricht dies einer Größe von 6 KByte.

Größe des Anzeigebuffers optimieren:

Der Speicherbedarf kann durch Angabe eines Wertes zwischen 128 und 512 optimiert werden. Die im Anzeigebuffer aufbereiteten Anzeigesätze werden über einen konfigurierbaren Upload-Puffer zum HMI transportiert.

Die maximale Größe des Upload-Puffers ergibt sich durch Multiplikation von (MD28402 \$MC_MM_ABSBLOCK_BUFFER_CONF[0] + MD28402 \$MC_MM_ABSBLOCK_BUFFER_CONF[1] + 1) mit der durch MD28400 \$MC_MM_ABSBLOCK projizierten Satzlänge.

Konfiguriert werden hier die Anzahl der Sätze **vor** dem aktuellen Satz im MD28402 \$MC_MM_ABSBLOCK_BUFFER_CONF[0] und **nach** dem aktuellen Satz im MD28402 \$MC_MM_ABSBLOCK_BUFFER_CONF[1].

Randbedingungen

Bei Überschreiten der in MD28400 \$MC_MM_ABSBLOCK projizierten Länge eines Anzeigesatzes wird dieser Anzeigesatz entsprechend abgeschnitten. Um dies darzustellen wird am Satzende der String "..." angefügt.

Für vorübersetzte Zyklen

(MD10700 \$MN_PREPROCESSING_LEVEL > 1 (Programmvorverarbeitungsstufe)) enthält der Anzeigesatz **nur** Achspositionen.

Weitere Randbedingungen für die Basis-Satzanzeige:

- Modale Synchronaktionssätze mit Absolutwerten werden nicht berücksichtigt.
- Die Basis-Satzanzeige ist während Satzsuchlauf mit oder ohne Berechnung deaktiviert.
- Polarkoordinatenprogrammierung wird nicht kartesisch dargestellt.

Radius / Durchmesser-Werte

Auf der Basis-Satzanzeige und der Positionsanzeige dargestellte Durchmesserwerte können zur internen Berechnung als Radius benötigt werden. Es bestehen folgende Möglichkeiten diese Werte für Maßangaben in Radius/Durchmesser gemäß G-Gruppe 29 zu beeinflussen:

- G-Befehl DIAMCYCOF (Erweiterung kanalspezifische Durchmesserprogrammierung)
Dieser G-Befehl schaltet die kanalspezifische Durchmesserprogrammierung während der Zyklusbearbeitung aus. Im Zyklus können somit Berechnungen immer im Radius erfolgen. Die Positionsanzeige und die Basis-Satzanzeige erfolgt weiterhin entsprechend dem Zustand der Durchmesserprogrammierung vor DIAMCYCOF.
In der Basis-Satzanzeige bleibt der zuletzt angezeigte Wert erhalten.
- G-Befehl DIACYCOFA[AX] (achsspezifische Durchmesserprogrammierung)
Dieser G-Befehl schaltet die achsspezifische Durchmesserprogrammierung während der Zyklusbearbeitung aus. Im Zyklus können somit Berechnungen immer im Radius erfolgen. In der Positionsanzeige und in der Basis-Satzanzeige erfolgt weiterhin entsprechend dem Zustand bezüglich Durchmesserprogrammierung vor DIACYCOFA[AX].
In der Basis-Satzanzeige bleibt der zuletzt angezeigte Wert erhalten.
- MD27100 \$MC_ABSBLOCK_FUNCTION_MASK (Satzanzeige mit Absolutwerten parametrieren)

Bit0 = 1 Sollwerte der Planachse werden in der Basis-Satzanzeige grundsätzlich als Durchmesserwerte angezeigt.

Verhalten bei aktivem Kompressor

Bei aktivem Kompressor mit G-Gruppe 30 ungleich COMPOF werden zwei Anzeigesätze generiert. Der

- Erste enthält den G-Befehl des aktiven Kompressors.
- Zweite enthält den String "..." als Zeichen dafür, dass Anzeigesätze fehlen.

Beispiel:

```
G0 X10 Y10 Z10      ; Satz der noch für die Basis-Satzanzeige aufbereitet wird
COMPCAD             ; Kompressor für optimierte Oberflächengüte (CAD-Prog.) ein
...                  ; String als Zeichen dafür, dass Anzeigesätze fehlen
```

Um Engpässe der NC-Leistung zu vermeiden, wird die Basis-Satzanzeige automatisch abgeschaltet. Als Zeichen dafür, dass Anzeigesätze fehlen, wird ein Anzeigesatz mit dem String "... " generiert.

Im Einzelsatz werden immer alle Anzeigesätze erzeugt.

10.14.6 Aufbau für einen DIN-Satz

Aufbau des Anzeigesatzes für einen DIN-Satz

Prinzipieller Aufbau des Anzeigesatzes für einen DIN-Satz

- Satznummer/Label
- G-Befehl der ersten G-Gruppe
(nur bei Änderung gegenüber dem letzten Maschinenfunktionssatz).
- Achspositionen
(Reihenfolge entsprechend MD20070 \$MC_AXCONF_MACHAX_USED
(Maschinenachsnummer gültig im Kanal)).
- Weitere modale G-Befehle
(nur bei Änderungen gegenüber den letzten Maschinenfunktionssatz).
- Weitere Adressen wie programmiert.

Der Anzeigesatz für die Basis-Satzanzeige wird direkt aus dem programmierten Teileprogrammsätzen nach folgenden Regeln abgeleitet:

- Makros werden expandiert.
- Ausblendkennungen und Kommentare entfallen.
- Satznummer und Labels werden aus dem Originalsatz übernommen, entfallen jedoch wenn DISPLOF aktiv ist.
- Die Anzahl der Nachkommastellen wird durch die Anzeigemaschinendaten MD9004, MD9010 und MD9011 über HMI festgelegt.

HMI-Anzeigemaschinendatum	Zugriff im NC Maschinendatum
MD9004 \$MM_DISPLAY_RESOLUTION	MD17200 \$MN_GMMC_INFO_NO_UNIT[0]
MD9011 \$MM_DISPLAY_RESOLUTION_INCH	MD17200 \$MN_GMMC_INFO_NO_UNIT[1]
MD9010 \$MM_SPIND_DISPLAY_RESOLUTION	MD17200 \$MN_GMMC_INFO_NO_UNIT[2]
MD9424 \$MM_MA_COORDINATE_SYSTEM	MD17200 \$MN_GMMC_INFO_NO_UNIT[3]

- Programmierte Achs-Positionen werden als absolute Positionen in dem durch das MD9424 \$MM_MA_COORDINATE_SYSTEM (Koordinatensystem für Istwertanzeige) vorgegebenen Koordinatensystem (WKS / ENS) dargestellt.

Hinweis

Bei Modulo-Achsen entfällt die Modulo-Korrektur. Dadurch sind Positionen außerhalb des Modulo-Bereichs möglich und es ergibt sich zwangsläufig eine Differenz zur Positionsanzeige in der grundsätzlich modulo-gewandelt wird.

Beispiele

Gegenüberstellungen von Anzeigesatz (Originalsatz) zur Basis-Satzanzeige:

- **Programmierte Positionen** werden absolut dargestellt.
Die Adressen AP/RP werden mit ihren programmierten Werten dargestellt.

Originalsatz:	Anzeigesatz:
N10 G90 X10.123	N10 X10.123
N20 G91 X1	N20 X11.123

- **Zuweisungen von Adressen** (nicht DIN-Adressen) werden in der Form <adresse> = <konstante> dargestellt.

Originalsatz:	Anzeigesatz:
N110 R1 = -67.5 R2 = 7.5	
N130 Z = R1 RND = R2	N130 Z-67.5 RND = 7.5

- **Indizes von Adressen** (Adresserweiterungen) werden als Konstanten <adresse> [<konstante>] = <konstante> dargestellt.

Originalsatz:	Anzeigesatz:
N220 DEF AXIS AXIS_VAR = X	
N240 FA[AXIS_VAR] = R2	N240 FA[X] = 1000

- **DIN-Adressen ohne Adresserweiterung** werden in der Form <din_adresse> <konstante> dargestellt.

Originalsatz:	Anzeigesatz:
N410 DEF REAL FEED = 1.5	
N420 F = FEED	N420 F1.5

Für **H-Funktionen** gilt: Unabhängig von der Ausgabeart zur PLC (MD22110 \$MC_AUXFU_H_TYPE_INT (Typ von H-Hilfsfunktionen ist integer)) wird der jeweils programmierte Wert angezeigt.

- Für die **Werkzeug-Anwahl über T-Befehl** wird eine Anzeigeinformation der Form T<wert> bzw. T=<string> generiert. Wurde eine Adresserweiterung programmiert, so wird diese auch aufgelöst. Wurden mehrere Spindeln projektiert oder ist die Funktion "Werkzeugwechsel über Werkzeughalter" (MD20124 \$MC_TOOL_MANAGEMENT_TOOLHOLDER (Werkzeughalternummer)) aktiv, so wird die T-Nummer grundsätzlich mit Adresserweiterung ausgegeben. Wurde keine Adresserweiterung programmiert, wird die Nummer der Masterspindel bzw. des Mastertoolholders eingesetzt (T<spindel_nummer/tool_holder> =).
- Für die **Spindelprogrammierung** über S, M3, M4, M5, M19, M40 - M45 und M70 (bzw. MD 20094 \$MC_SPIND_RIGID_TAPPING_M_NR (M-Funktion für das Umschalten in den gesteuerten Achsbetrieben)) gilt bezüglich Adresserweiterung folgende Regelung: Wurde eine Adresserweiterung programmiert, so wird diese auch aufgelöst. Wurden mehrere Spindeln projektiert, so wird grundsätzlich die Adresserweiterung mit ausgegeben. Wurde keine Adresserweiterung programmiert, wird die Nummer der Masterspindel eingesetzt (S<spindel_nummer>=).
- **Indirekte G-Befehlprogrammierung** der Form G[<gruppe>] = <ausdruck> wird durch den entsprechenden G-Befehl ersetzt.

Originalsatz:	Anzeigesatz:
N510 R1=2	
N520 G[8]= R1	N520 G54

- **Modale G-Befehle**, die keinen ausführbaren Satz erzeugen werden aufgesammelt und mit dem Anzeigesatz des nächsten ausführbaren Satz angezeigt, sofern dies von der Syntax her erlaubt ist (DIN-Satz). Ist dies nicht der Fall (z. B. vordefinierter Unterprogrammaufruf TRANSMIT) so wird ein eigener Anzeigesatz mit den geänderten G-Befehlen vorangestellt.

Originalsatz:	Anzeigesatz:
N610 G64	G64
N620 TRANSMIT	N620 TRANSMIT

- Für **Teileprogrammzeilen** in denen die Adressen **F** und **FA** vorkommen, wird immer ein Anzeigesatz generiert (auch bei MD22240 \$MC_AUXFU_F_SYNC_TYPE = 3 (Ausgabezeitpunkt der F-Funktionen)).

Originalsatz:	Anzeigesatz:
N630 F1000	N630 F1000
N640 X100	N640 X100

- Die für die **Satzanzeige generierten Anzeigesätze** werden **direkt** von den programmierten Teileprogrammsätzen abgeleitet. Werden durch die Konturaufbereitung Zwischensätze erzeugt (z.B. Werkzeugradiuskorrektur G41/G42, Radius/Fase RNDM, RND, CHF, CHR), so erhalten diese Sätze die Anzeigeinformation des der Bewegung zugrunde liegenden Teileprogrammsatzes.

Originalsatz:	Anzeigesatz:
N710 Y157.5 G42	N710 Y157.5 G42
N720 Z-67.5 RND=7.5	N720 Z-67.5 RND=7.5

- Beim **EXECTAB-Befehl** (Abarbeiten einer Tabelle von Konturelementen) wird im Anzeigesatz der durch EXECTAB generierte Satz angezeigt.

Originalsatz:	Anzeigesatz:
N810 EXECTAB (KTAB[5])	N810 G01 X46.147 Z-25.38

- Beim **EXECSTRING-Befehl** wird im Anzeigesatz der durch EXECSTRING generierte Satz angezeigt.

Originalsatz:
N910 DEF STRING[40] PROGSTRING = "N905 M3 S1000 G94 Z100 F1000 G55"
N920 EXECSTRING (PROGSTRING)

Originalsatz:
N905 Z100 G55 G94 M3 S1000 F1000

10.14.7 Abarbeiten von Extern

Funktion

Mit der Funktion "Abarbeiten von Extern" können Programme, die aufgrund ihrer Größe nicht mehr im NC-Speicher ablegbar sind, von einem externen Programmspeicher abgearbeitet werden.

Hinweis

Geschützte Zyklen (_CPF Dateien) können **nicht** von einem externen Programmspeicher abgearbeitet werden.

Externe Programmspeicher

Externe Programmspeicher können sich auf folgenden Datenträgern befinden:

- Lokales Laufwerk
- Netzlaufwerk
- USB-Laufwerk

Hinweis

Als Schnittstelle zum Abarbeiten eines auf einem USB-Laufwerk befindlichen externen Programms dürfen nur die USB-Schnittstellen an der Bedientafelfront bzw. TCU verwendet werden.

ACHTUNG

Werkzeug-/Werkstückbeschädigung durch USB-FlashDrive

Es wird empfohlen zum Abarbeiten eines externen Unterprogramms kein USB-FlashDrive zu verwenden. Ein Kommunikationsabbruch zum USB-FlashDrive während der Abarbeitung des Teileprogramms durch Kontaktschwierigkeiten, Herausfallen, Abbrechen durch Anstoßen oder versehentliches Abziehen führt zum sofortigen Stop der Bearbeitung. Werkzeug und/oder Werkstück könnten dabei beschädigt werden.

Anwendungen

- **Direktes Abarbeiten von externen Programmen**
Prinzipiell können alle Programme angewählt und abgearbeitet werden, die über die Verzeichnis-Struktur der Bedienoberfläche im HMI-Modus "Abarbeiten von Extern" erreichbar sind.
- **Abarbeiten von externen Unterprogrammen aus dem Teileprogramm**
Aufgerufen wird das "externe" Unterprogramm durch den Teileprogrammbefehl `EXTCALL` mit Angabe eines Aufrufpfades (optional) und des Unterprogrammnamens (→ siehe "Abarbeiten von externen Unterprogrammen (EXTCALL) (Seite 636)").

Parametrierung

Für die Abarbeitung eines Programms im Modus "Abarbeiten von Extern" muss im dynamischen NC-Speicher ein Nachladespeicher (FIFO-Puffer) reserviert werden.

Größe des FIFO-Puffers

Die Größe des FIFO-Puffers wird eingestellt mit dem Maschinendatum:

MD18360 \$MN_MM_EXT_PROG_BUFFER_SIZE (FIFO-Puffer-Größe für Abarbeiten von Extern)

Hinweis

Programme mit Sprunganweisungen

Bei externen Programmen, die Spunganweisungen enthalten (GOTO, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDIF etc.), müssen die Sprungziele innerhalb des Nachladespeichers liegen.

Hinweis

ShopMill-/ShopTurn-Programme

ShopMill- und ShopTurn-Programme müssen wegen der am Dateiende angefügten Konturbeschreibungen vollständig im Nachladespeicher abgelegt sein.

Anzahl der FIFO-Puffer

Für alle Programme, die gleichzeitig im Modus "Abarbeiten von Extern" abgearbeitet werden, muss jeweils ein FIFO-Puffer zur Verfügung gestellt werden.

Die Anzahl der FIFO-Puffer wird eingestellt mit dem Maschinendatum:

MD18362 \$MN_MM_EXT_PROG_NUM (Anzahl der gleichzeitig von Extern abarbeitbaren Programmebenen)

Verhalten bei Reset, Power-On

Durch Reset und Power-On werden externe Programmaufrufe abgebrochen und die jeweiligen FIFO-Puffer gelöscht.

Ein von einem externen Programmspeicher angewähltes Hauptprogramm wird nach einem Power-On automatisch wieder angewählt, wenn der entsprechende Programmspeicher weiterhin zur Verfügung steht und die Bearbeitung von EXTCALL-Aufrufen im MD9106 \$MM_SERVER_EXTCALL_PROGRAMS aktiv geschaltet ist.

10.14.8 Abarbeiten von externen Unterprogrammen (EXTCALL)

Funktion

Bei der Bearbeitung komplexer Werkstücke können sich für die einzelnen Bearbeitungsschritte Programmsequenzen ergeben, die wegen ihres Speicherbedarfs nicht mehr direkt im NC-Speicher ablegbar sind.

In solchen Fällen hat der Anwender die Möglichkeit, mit der Teileprogrammanweisung EXTCALL die Programmsequenzen als Unterprogramme von einem externen Programmspeicher im Modus "Abarbeiten von Extern" auszuführen.

Voraussetzungen

Für das Abarbeiten externer Unterprogramme gelten folgende Voraussetzungen:

- Die Unterprogramme müssen über die Verzeichnis-Struktur der Bedienoberfläche erreichbar sein.
- Für jedes Unterprogramm muss im dynamischen NC-Speicher ein Nachladespeicher (FIFO-Puffer) reserviert sein.

Hinweis

Unterprogramme mit Sprunganweisungen

Bei externen Unterprogrammen, die Spunganweisungen enthalten (GOTO, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDIF etc.), müssen die Sprungziele innerhalb des Nachladespeichers liegen.

Die Größe des Nachladespeichers wird eingestellt über:

MD18360 MM_EXT_PROG_BUFFER_SIZE

ShopMill-/ShopTurn-Programme

ShopMill- und ShopTurn-Programme müssen wegen der am Dateiende angefügten Konturbeschreibungen vollständig im Nachladespeicher abgelegt sein.

Parametrierung

Der Pfad zum externen Unterprogrammverzeichnis kann voreingestellt werden mit dem Settingdatum:

SD42700 \$SC_EXT_PROG_PATH (Programmpfad für externen Unterprogrammaufruf EXTCALL)

Zusammen mit dem bei der Programmierung angegebenen Unterprogrammpfad bzw. -name ergibt sich daraus der Gesamtpfad des aufzurufenden Programms.

Programmierung

Der Aufruf eines externen Unterprogramms erfolgt über den Teileprogrammbefehl EXTCALL.

Syntax: `EXTCALL ("<Pfad/><Programmname>")`

Parameter:

<code><Pfad/></code> :	Absolute oder relative Pfadangabe (optional)
Typ:	STRING

<Programmname>: Der Programmname wird ohne Präfix "_N_" angegeben.
 Die Dateierweiterung ("MPF", "SPF") kann mit dem Zeichen "_" oder "." am Programmnamen angefügt werden (**optional**).
 Typ: STRING

Hinweis**Pfadangabe: Kurzbezeichnungen**

Bei der Pfadangabe können folgende Kurzbezeichnungen verwendet werden:

- **LOCAL_DRIVE**: für lokales Laufwerk
- **CF_CARD**: für CompactFlash-Card
- **USB**: für USB Front-Anschluss

CF_CARD: und **LOCAL_DRIVE**: sind alternativ verwendbar.

EXTCALL-Aufruf mit absoluter Pfadangabe

Wenn das Unterprogramm unter dem angegebenen Pfad existiert, dann wird es nach dem **EXTCALL**-Aufruf ausgeführt. Wenn es nicht existiert, dann wird die Programmausführung abgebrochen.

EXTCALL-Aufruf mit relativer Pfadangabe / ohne Pfadangabe

Bei einem **EXTCALL**-Aufruf mit relativer Pfadangabe bzw. ohne Pfadangabe werden die vorhandenen Programmspeicher nach folgendem Muster durchsucht:

- Wenn in SD42700 \$SC_EXT_PROG_PATH eine Pfadangabe voreingestellt ist, dann wird zuerst ausgehend von diesem Pfad nach der Angabe im **EXTCALL**-Aufruf (Programmname ggf. mit relativer Pfadangabe) gesucht. Der absolute Pfad ergibt sich dann durch Zeichenverkettung aus:
 - der in SD42700 voreingestellten Pfadangabe
 - dem Zeichen "/" als Trennzeichen
 - dem bei **EXTCALL** angegebenen Unterprogrammpfad bzw. -namen
- Wurde das aufgerufene Unterprogramm unter dem voreingestellten Pfad nicht gefunden, werden als nächstes die Verzeichnisse des Anwenderspeichers nach der Angabe im **EXTCALL**-Aufruf durchsucht.
- Die Suche endet, wenn das Unterprogramm erstmalig gefunden wurde. Sollte die Suche keinen Treffer ergeben, kommt es zum Programmabbruch.

Beispiel**Abarbeiten von lokalem Laufwerk**

Hauptprogramm:

Programmcode
N010 PROC MAIN
N020 ...
N030 EXTCALL ("SCHRUPPEN")

Programmcode

N040 ...
N050 M30

Externes Unterprogramm:

Programmcode

N010 PROC SCHRUPPEN
N020 G1 F1000
N030 X= ... Y= ... Z= ...
N040 ...
...
...
N999999 M17

Das Hauptprogramm "MAIN.MPF" befindet sich im NC-Speicher und ist zur Abarbeitung angewählt.

Das nachzuladende Unterprogramm "SCHRUPPEN.SPF" bzw. "SCHRUPPEN.MPF" befindet sich auf dem lokalen Laufwerk in dem Verzeichnis "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD".

Der Pfad zu dem Unterprogramm ist im SD42700 voreingestellt:

SD42700 \$SC_EXT_PROG_PATH = "LOCAL_DRIVE:WKS.DIR/WST1.WPD"

Hinweis

Ohne Pfadangabe im SD42700 müsste die EXTCALL-Anweisung für dieses Beispiel wie folgt programmiert werden:

```
EXTCALL ("LOCAL_DRIVE:WKS.DIR/WST1.WPD/SCHRUPPEN")
```

10.15 Abarbeiten vom externen Speicher (EES) (Option)

10.15.1 Funktion

Hinweis

Für die Nutzung der Funktion ist die lizenzpflichtige Option "CNC Anwenderspeicher erweitert" oder "Abarbeiten vom externen Speicher (EES)" erforderlich!

Funktion

Mit der Funktion EES (Execution from External Storage) hat der Anwender die Möglichkeit, Programme durch die NC **direkt von einem externen Speicher** abarbeiten zu lassen.

Folgende Laufwerke können als externer Speicher zur Verfügung stehen:

Laufwerk	Symbolischer Name ¹⁾	SINUMERIK 840D sl	SINUMERIK 828D
NC Extend (ehem. "Lokales Laufwerk")	CF_CARD LOCAL_DRIVE SYS_DRIVE	CF-Card der NCU oder Lokale Festplatte einer PCU	System-CF-Card der PPU
USER CF	CF_CARD LOCAL_DRIVE	-	User-CF-Card der PPU
Netzlaufwerk			
Statisch geführtes USB-Laufwerk			

- ¹⁾ Bei SINUMERIK 840D sl sind dem Laufwerk NC Extend die symbolischen Namen LOCAL_DRIVE, CF_CARD und SYS_DRIVE fest zugeordnet (⇒ NC Extend kann über LOCAL_DRIVE, CF_CARD und SYS_DRIVE adressiert werden).
Bei SINUMERIK 828D ist die Zuordnung der symbolischen Namen LOCAL_DRIVE, CF_CARD und SYS_DRIVE zu NC Extend projektierbar. So kann die System-CF-Card der PPU auch über die symbolischen Namen LOCAL_DRIVE und CF_CARD adressiert werden, falls z. B. kein USER CF-Laufwerk vorhanden ist.

ACHTUNG

Werkzeug-/Werkstückbeschädigung durch USB-FlashDrive

Ein USB-FlashDrive kann zum Abarbeiten eines externen Programms **nicht** empfohlen werden. Ein Kommunikationsabbruch zum USB-FlashDrive während der Abarbeitung des Programms durch Kontaktschwierigkeiten, Herausfallen, Abbrechen durch Anstoßen oder versehentliches Abziehen führt zum unkontrollierten Stop der Bearbeitung. Werkzeug und/oder Werkstück könnten dabei beschädigt werden.

Voraussetzungen

Für die Nutzung von EES gelten folgende Voraussetzungen:

- Die lizenzpflichtige Option "CNC Anwenderspeicher erweitert" oder "Abarbeiten vom externen Speicher (EES)" muss gesetzt sein.
- Die Laufwerke, die an der Steuerung als externe Speicher zum Einsatz kommen sollen, müssen als logische Laufwerke projiziert sein (siehe "Inbetriebnahme (Seite 642)").

Arbeitsmodus

Abhängig von der vorhandenen Option und der Laufwerksprojektierung sind verschiedene EES-Arbeitsmodi möglich. Der aktive Arbeitsmodus einer Steuerung wird über das Maschinendatum MD18045 \$MN_EES_MODE_INFO angezeigt:

MD18045	Arbeitsmodus	Option	Externe Speicher	
			SINUMERIK 840D sl	SINUMERIK 828D
= 0	EES nicht aktiv	-	-	
= 1	Lokales EES aktiv	6FC5800-0AP77-0YB0 CNC Anwenderspeicher erweitert	Die Nutzung von EES auf einer NCU ist auf den erweiterten Anwenderspeicher (100 MB) der CF-Card begrenzt. Bei Einsatz von EES auf einer PCU kann der gesamte freie Speicher von NC Extend genutzt werden.	Die Nutzung von EES ist auf den erweiterten Anwenderspeicher (100 MB) der System-CF-Card begrenzt.
			EES über Netzwerk oder USB ist nicht möglich.	
		6FC5800-0AP12-0YB0 zusätzlicher HMI-Anwenderspeicher auf CF-Card der NCU	Bei Einsatz von EES auf einer NCU kann durch die Zusatzoption der lokale Anwenderspeicher auf bis zu 6 GB erweitert werden (abhängig von MD9111). Bei Einsatz von EES auf einer PCU wird die Zusatzoption nicht benötigt.	-
= 2	Globales EES aktiv	6FC5800-0AP75-0YB0 Abarbeiten vom externen Speicher (EES)	EES ist für alle verfügbaren externen Speicher nutzbar.	EES ist für alle verfügbaren externen Speicher nutzbar. Hinweis: EES über Netzwerk benötigt zusätzlich die Freigabe der HMI-Funktion "Network drive management" (MD19730 Bit 2 = 1).
		6FC5800-0AP12-0YB0 zusätzlicher HMI-Anwenderspeicher auf CF-Card der NCU	Bei Einsatz von EES auf einer NCU kann durch die Zusatzoption der lokale Anwenderspeicher auf bis zu 6 GB erweitert werden (abhängig von MD9111). Bei Einsatz von EES auf einer PCU wird die Zusatzoption nicht benötigt.	-

MD18045	Arbeitsmodus	Option	Externe Speicher	
			SINUMERIK 840D sl	SINUMERIK 828D
= 5	Lokales EES aktiv + Globaler Teileprogramm- speicher	Wie MD18045 = 1, nur dass zusätzlich auf dem erweiterten Anwenderspeicher ein globaler Teileprogramm- speicher (Seite 644) eingerichtet ist.		
= 6	Globales EES aktiv + Globaler Teileprogramm- speicher	Wie MD18045 = 2, nur dass zusätzlich auf einem externen Speicher ein globaler Teile- programm- speicher (Seite 644) eingerichtet ist.		

Eigenschaften

Durch die Funktion EES können die Funktionen "Abarbeiten von Extern (Seite 634)" und "Abarbeiten von externen Unterprogrammen (EXTCALL) (Seite 636)" ersetzt werden.

Die Funktion EES bietet folgende Vorteile:

- Systemweit einheitliches Programmhandling
- Keine Einschränkungen hinsichtlich der verwendbaren Befehle
Die Einschränkungen bei "Abarbeiten von Extern" und "Abarbeiten von externen Unterprogrammen (EXTCALL)", z. B. keine Rückwärtssprünge, Begrenzung der Sprungweite von Sprungbefehlen durch die Größe des Nachladespeichers, sind bei EES aufgehoben.
- Programme können zwischen den verschiedenen Programmspeichern (NC, GDIR, externes Laufwerk) wesentlich einfacher verlagert werden.
- Praktisch keine Einschränkung bezüglich Teileprogrammgröße und Programmanzahl (diese sind nur durch die Kapazität der externen Datenablage limitiert)
- Einheitliche Syntax für den Unterprogramm-Aufruf, unabhängig vom Ablageort des Unterprogramms (kein EXTCALL-Aufruf erforderlich)
- Netzlaufwerke können gemeinsam von mehreren Stationen (PCU/NCU) verwendet werden. Voraussetzung dafür ist eine einheitliche Laufwerksprojektierung für diese Stationen. Daraus ergibt sich eine für alle Stationen einheitliche Sicht auf die Programme.
- Durch die einheitliche Sicht aller Stationen auf die externen Programmspeicher werden Änderungen an den dort abgelegten Programmen konsistent für alle Stationen wirksam.

10.15.2 Inbetriebnahme

10.15.2.1 Projektierung der Laufwerke

Vor Verwendung der Funktion EES müssen die an der Steuerung zum Einsatz kommenden Laufwerke vereinbart werden.

Literatur:

- Inbetriebnahmehandbuch Basesoftware und Bediensoftware
- Bedienhandbuch Universal/Drehen/Fräsen/Schleifen

Nach dem Aktivieren der neuen Laufwerksprojektierung können die Programme frei auf die verfügbaren Laufwerke verteilt werden.

Hinweis

In der neu erstellten Laufwerksprojektierung sind eventuell nicht mehr alle vorherigen Laufwerke verfügbar. Der Zugriff auf darauf befindliche Programme ist dann nicht mehr möglich.

Abhilfe: Programme dieser Ablagen vorher auf ein weiterhin zugreifbares Laufwerk kopieren.

Hinweis

Da geschützte Zyklen (_CPF Dateien) systembedingt nur aus dem NC-Teileprogrammspeicher abgearbeitet werden, können sie **nicht** zur Abarbeitung auf einem externen Programmspeicher abgelegt werden.

Der bisherige NC-Teileprogrammspeicher mit den Verzeichnissen MPF.DIR, SPF.DIR und WKS.DIR ist für den EES-Betrieb nicht zwingend erforderlich. Ein System kann auch ohne die Verwendung des NC-Teileprogrammspeichers projektiert werden.

ACHTUNG
Ausführung nicht sichtbarer Programme
Selbst wenn der NC-Teileprogrammspeicher aus der Laufwerksprojektierung entfernt wurde, ist er intern im System immer noch vorhanden. Insbesondere bedeutet dies, dass bei der Programmausführung gegebenenfalls dort noch existierende Programme aus dem SPF.DIR unbeabsichtigt zur Ausführung gelangen können.
Abhilfe: Wird das System bei der Laufwerksprojektierung ohne NC-Teileprogrammspeicher projektiert, sollten alle dort eventuell noch abgelegten Programme manuell gelöscht werden.

Soll der NC-Teileprogrammspeicher weiterhin genutzt werden, sollte er nicht komplett aus dem System entfernt, sondern bei Bedarf nur mit einer entsprechenden Schutzstufe versehen werden.

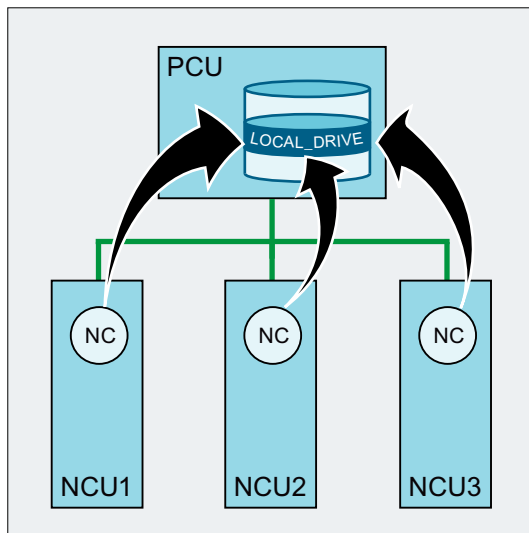
Laufwerke können von mehreren Stationen (PCU/NCU) gemeinsam genutzt werden. Mit einer einheitlichen Laufwerksprojektierung für diese Stationen erreicht man damit eine stationsunabhängige, einheitliche Sicht auf die Programme.

Hinweis

Die CF-Card einer NCU/PPU kann **nicht** von mehreren Stationen genutzt werden.

Beispiel:

Mehrere NCUs nutzen gemeinsam einen Programmspeicher auf der lokalen Festplatte der PCU (LOCAL_DRIVE).



Hinweis

Werden im EES-Betrieb an verschiedenen Stationen externe Programmspeicher gemeinsam verwendet, sind folgende Regeln zu beachten:

- Ein Programm kann **nicht** gleichzeitig von mehreren Stationen editiert werden.
- Programme, die sich in Abarbeitung befinden, können **nicht** mehr verändert werden.

Nur SINUMERIK 840D sl

Bei einem Betrieb mit externem HMI muss die Laufwerksprojektierung auf dem externen HMI erfolgen! Die Laufwerksprojektierung (logdrive.ini) muss aus dem externen HMI in die entsprechende NCU geladen werden. Für die Übertragung steht ein Softkey im Dialog der Laufwerksprojektierung zur Verfügung.

Bei Anlagen, an denen mehrere NCs zusammenarbeiten, muss die Laufwerksprojektierung für alle NCs gleich sein. Dies erreicht man durch Verteilen der Datei logdrive.ini auf alle in der Datei mmc.ini aufgeführten NCUs. Die dort vorhandenen Projektierungen werden dadurch überschrieben.

10.15.2.2 Globaler Teileprogrammspeicher (GDIR)

Bei der Vereinbarung der Laufwerke kann eines der Laufwerke als globaler Teileprogrammspeicher (GDIR) ausgezeichnet werden.

Literatur:

Bedienhandbuch; Kapitel: "Programme verwalten" > "Laufwerke einrichten"

Vom System werden automatisch die Verzeichnisse MPF.DIR, SPF.DIR und WKS.DIR auf dem Laufwerk, welches das GDIR beherbergt, angelegt. Diese drei Verzeichnisse bilden das GDIR.

Das GDIR spielt ausschließlich für die Funktion EES eine Rolle. Je nach Laufwerksprojektierung ersetzt oder erweitert das GDIR den NC-Teileprogrammspeicher. Das Einrichten eines GDIR ist für den EES-Betrieb allerdings nicht zwingend erforderlich.

Die Verzeichnisse und Dateien des GDIR können im Teileprogramm auf dieselbe Art wie im passiven Filesystem adressiert werden. Damit ist eine kompatible Verlagerung eines NC-Programms mit Pfadangaben aus dem passiven Filesystem ins GDIR möglich.

Das GDIR erweitert den Suchpfad für Unterprogramme, die ohne absolute Pfadangabe aufgerufen werden.

GDIR ersetzt den NC-Teileprogrammspeicher

Wenn der NC-Teileprogrammspeicher in den Verzeichnissen MPF.DIR, SPF.DIR und WKS.DIR komplett leer ist, ersetzt das GDIR den NC-Teileprogrammspeicher. Der bisherige NC-Suchpfad wird über das GDIR 1:1 abgebildet.

Anwahl des Hauptprogramms auf einer externen Ablage

Suchreihenfolge für die Unterprogramme:

1. Aktuelles Verzeichnis auf externer Ablage
2. SPF.DIR im GDIR-Speicher
3. Das mit CALLPATH referenzierte Laufwerk
4. Zyklen

GDIR erweitert den NC-Teileprogrammspeicher

Ist der NC-Teileprogrammspeicher in den Verzeichnissen MPF.DIR, SPF.DIR und WKS.DIR nicht leer, so ist die Suchreihenfolge für die Unterprogramme abhängig vom Ablageort des Hauptprogramms (aktives Verzeichnis).

Anwahl des Hauptprogramms im NC-Teileprogrammspeicher (MPF.DIR oder xxx.WPD in WKS.DIR)

Suchreihenfolge für die Unterprogramme:

1. Aktuelles Verzeichnis im NC-Teileprogrammspeicher
2. SPF.DIR im NC-Teileprogrammspeicher
3. Das mit CALLPATH referenzierte Laufwerk
4. Zyklen

Anwahl des Hauptprogramms auf für EES freigegebener externer Ablage

Suchreihenfolge für die Unterprogramme:

1. Aktuelles Verzeichnis auf externer Ablage
2. SPF.DIR im NC-Teileprogrammspeicher
3. SPF.DIR in GDIR
4. Das mit CALLPATH referenzierte Laufwerk
5. Zyklen

Hinweis

Zur Festlegung der Suchreihenfolge siehe auch MD11625
\$MN_FILE_ONLY_WITH_EXTENSION und MD11626 \$MN_CYCLES_ONLY_IN_CYCDIR!

Hinweis

Mit der CALLPATH-Anweisung kann auch auf ein externes Laufwerk verwiesen werden.

10.15.2.3 Einstellungen für Dateihandling im Teileprogramm bei EES

Anlagenweit eindeutige Programmnamen

Werden im EES-Betrieb an verschiedenen Stationen externe Programmspeicher gemeinsam verwendet, kann es bei parallel auf verschiedenen Stationen gleichzeitig durchgeführten Dateioperationen (WRITE, DELETE, ...) zu Zugriffskonflikten kommen. Zur Vermeidung derartiger Zugriffskonflikte wird empfohlen, auf jeder Station einen anlagenweit eindeutigen Namensraum für Dateinamen einzurichten.

Anlagenweit eindeutiger Namensraum

Ein anlagenweit eindeutiger Namensraum für Dateinamen wird z. B. erreicht durch Verknüpfung der Dateinamen mit dem in Maschinendaten parametrierbaren EES-spezifischen Namen der NC und der Kanalnummer des Kanals, in dem das Programm abgearbeitet wird. Die nachfolgende Programmierung erzeugt z. B. bei Abarbeitung des Programms den anlagenweit eindeutigen Dateinamen (MYFILE_NC1_1.SPF) durch Anhängen des EES-spezifischen Namens der NC (NC1) und der Kanalnummer (Kanal 1) an den Programmnamen.

```
$MN_EES_NC_NAME="NC1"
```

```
N10 DEF STRING[31] FILENAME
```

```
N20 FILENAME="MYFILE_" << $MN_EES_NC_NAME << "_" << $P_CHANNO <<  
".SPF"
```

Parametrierung

Der EES-spezifische Name der NC wird eingestellt im NC-spezifischen Maschinendatum:

```
MD10125 $MN_EES_NC_NAME = <NC-Name>
```

Hinweis

Anlagenweit eindeutiger Name der NC

Zur Vermeidung von Zugriffskonflikten muss der EES-spezifische Name der NC anlagenweit eindeutig sein. Die Verantwortung dafür liegt ausschließlich beim Anwender/Maschinenhersteller.

Beim Programmaufruf nur nach Dateien mit Dateikennung suchen

Um im EES-Betrieb die Programmsuche bei Unterprogrammaufrufen zu beschleunigen, wird empfohlen, die Suche auf Dateien **mit** Dateikennung (z. B. SPF, MPF etc.) zu beschränken:

MD11625 \$MN_FILE_ONLY_WITH_EXTENSION = 1

Hinweis

MD11625 hat keine Auswirkungen auf die Programmsuche beim Abarbeiten von externen Unterprogrammen mit EXTCALL.

Literatur:

Beschreibung des Suchpfades für den Unterprogrammaufruf siehe Programmierhandbuch Arbeitsvorbereitung.

Programme mit Interface nur in den Zyklenverzeichnissen suchen

Um im EES-Betrieb die Programmsuche bei Unterprogrammaufrufen zu beschleunigen, wird empfohlen, die Suche nach Unterprogrammen, die eine Interface-Beschreibung (mittels PROC-Anweisung) angelegt haben und deren Interface-Beschreibung aus einem der Zyklenverzeichnisse (CUS, CMA, CST) erzeugt wurde, auf die Zyklenverzeichnisse zu beschränken:

MD11626 \$MN_CYCLES_ONLY_IN_CYCDIR = 1

Hinweis

Auf Unterprogramme, deren Interface durch eine EXTERN-Erklärung angelegt wurde, hat MD11626 keine Auswirkung. Sie werden in allen Programmverzeichnissen gesucht.

ACHTUNG
Kein Sucherfolg bei Zyklen außerhalb der Zyklenverzeichnisse
Zyklen im aktuellen Verzeichnis und globalen Unterprogrammverzeichnis werden mit der Einstellung MD11626 = 1 nicht mehr gefunden!
Abhilfe: Zyklen immer in den Zyklenverzeichnissen ablegen.

10.15.2.4 Speicherkonfiguration

Endanwender-Programmspeicher im passiven Filesystem verkleinern

Bei aktivem EES kann gegebenenfalls der im passiven Filesystem für den Endanwender vorgesehene Programmspeicher verkleinert werden:

MD18352 \$MN_MM_U_FILE_MEM_SIZE (Endanwenderspeicher für Teileprogramme/Zyklen/Dateien)

Der frei werdende Speicher ist dann z. B. für Werkzeugdaten oder Herstellerzyklen (MD18353 \$MN_MM_M_FILE_MEM_SIZE) nutzbar.

Literatur:

Ausführliche Informationen zur Speicherkonfiguration siehe Funktionshandbuch Erweiterungsfunktionen

Nachladespeicher freigeben

Die Funktion EES kann die Funktionen "Abarbeiten von Extern" und "Abarbeiten von externen Unterprogrammen (EXTCALL)" ersetzen.

Um Unterprogramme aus Teileprogrammen mit EXTCALL-Aufrufen statt mit der Funktion "Abarbeiten von externen Unterprogrammen (EXTCALL)" mit EES abarbeiten zu lassen, müssen die EXTCALL-Aufrufe in CALL-Aufrufe umgewandelt und ggf. die Pfadangaben angepasst werden.

Nach vollständiger Umstellung können die für das "Abarbeiten von Extern" und "Abarbeiten von externen Unterprogrammen (EXTCALL)" benötigten Nachladespeicher (FIFO-Puffer) freigegeben werden:

MD18362 \$MN_MM_EXT_PROG_NUM (Anzahl der gleichzeitig von Extern abarbeitbaren Programmebenen) = 0

10.15.3 Randbedingungen

Teach In

Im EES-Betrieb ist die Anwendung der Funktion "Teach In" in der Betriebsart AUTOMATIK **nicht** möglich.

10.16 Process DataShare - Ausgabe auf ein externes Gerät/Datei

10.16.1 Funktion

Mit der Funktion "Process DataShare" ist es möglich, Daten aus einem Teileprogramm heraus auf ein externes Gerät / eine externe Datei zu schreiben, z. B. zum Zweck der Protokollierung von Produktionsdaten oder zur Steuerung von Zusatzaggregaten an einer Steuerung.

Verfügbarkeit

Die Funktion ist verfügbar:

- nur im realen NC (**nicht** in den Simulations-Softwares SNC und VNC).
- nur in Teileprogrammen (**nicht** in Synchronaktionen).
- parallel in allen Bearbeitungskanälen des NC für alle verfügbaren (projektierten) Ausgabegeräte.

Externe Geräte/Dateien

Externe Geräte/Dateien können sein:

- Dateien auf der lokalen CompactFlash Card
Mit lokaler CompactFlash Card ist der Speicher gemeint, auf den mit dem symbolischen Bezeichner LOCAL_DRIVE aus HMI heraus verwiesen wird. Bei SINUMERIK 840D sl ist dies das lokale Laufwerk, bei SINUMERIK 828D die Anwender CompactFlash Card.

Hinweis

Für die Ausgabe auf das Gerät LOCAL_DRIVE ist bei SINUMERIK 840D sl die Option "Zusätzlich xxx MB HMI-Anwenderspeicher auf CF-Card der NCU" erforderlich. Für SINUMERIK 828D muss eine Anwender CompactFlash Card vorhanden sein, eine Option ist hier nicht erforderlich.

- Dateien auf einem Netzlaufwerk
- V.24-Schnittstelle

Hinweis

Für die Ausgabe auf die V.24-Schnittstelle ist bei SINUMERIK 840D sl das NCU-Optionsmodul RS232-Schnittstelle erforderlich. Bei SINUMERIK 828D erfolgt die Ausgabe auf die integrierte V.24-Schnittstelle (Voraussetzung: MD51233 \$MNS_ENABLE_GSM_MODEM = 0).

Maximale Anzahl geöffneter externer Geräte

In einem Teileprogramm/Kanal kann auch mehr als ein externes Gerät/Datei belegt werden. Über alle NC-Kanäle hinweg können gleichzeitig maximal 10 Ausgabegeräte geöffnet sein. Zusätzlich gibt es noch zwei reservierte Einträge für Siemens-Zyklen.

Zu den Ausgabegeräten können gleichzeitig maximal 5 Aufträge aktiv sein.

Benutzungsmodus

Pro Ausgabegerät ist beim Öffnen des Geräts vorgebbbar, ob das Gerät exklusiv nur von einem Kanal aus genutzt werden darf oder ob es geteilt unter den Kanälen, die darauf ausgeben wollen, verwendet werden kann ("Shared"-Modus).

Verhalten bei Teileprogramm-Ende / Kanal-Reset

Mit Teileprogramm-Ende und Kanal-Reset werden alle im Kanal geöffneten externen Geräte/Dateien geschlossen.

Nutzung der Funktion zur Datenübertragung an die Steuerung

ACHTUNG

Datensicherheit

Wird die Funktion Process DataShare verwendet, um von einem externen Gerät über die Ethernet-Schnittstelle X130 Daten an die Steuerung zu schicken, besteht die Möglichkeit, dass die Daten auf der Steuerung durch Dritte verfälscht werden und nicht mehr konsistent sind. Es ist darauf zu achten, dass bei Verwendung der Funktion das Netzwerk gegen den Zugriff durch Dritte gesichert wurde.

10.16.2 Inbetriebnahme

Die Projektierung der zu verwendenden externen Geräte erfolgt in der Datei /oem/sinumerik/nck/extdev.ini bzw. /user/sinumerik/nck/extdev.ini. Sind beide Dateien vorhanden, haben die Einträge im User-Bereich Vorrang. Die Datei kann im Bedienbereich INBETRIEBNAHME unter SYSTEMDATEN/CF-Karte gepflegt werden.

Hinweis

Für die Verwendung von LOCAL_DRIVE und CYC_DRIVE ist keine Projektierung in der Datei extdev.ini erforderlich. Die beiden Geräte sind immer verfügbar, sobald die entsprechende Option gesetzt bzw. die Anwender CompactFlash Card vorhanden ist.

Im Abschnitt [ExternalDevices] der Datei extdev.ini werden die zu verwendenden externen Geräte definiert/aufgezählt. Als Geräte können ein serielles Device (/dev/v24) und bis zu neun Dateien oder Verzeichnisse (/dev/ext/1...9) angegeben werden. Die Notation für die Angabe ist Linux-Schreibweise. Zeilen, die mit ";" beginnen, sind Kommentar und werden überlesen.

Mit Ausnahme von /dev/v24 können die Geräte als Verzeichnispfad – abgeschlossen mit angehängtem "/" – oder als Dateipfad – also mit angehängtem voll qualifizierten Pfad, endend mit einem Dateinamen (ohne endendes "/") – vereinbart werden. Zu einem Gerät mit Verzeichnispfad muss bei Verwendung im Teileprogramm ein Dateiname (Pfad) mit angegeben werden.

Außer für /dev/v24 erfolgt die Definition eines Geräts durch die drei mit Komma getrennten Angaben für "Server", "Pfad" und dem optionalen "Schreibmodus".

Zu Dateien bzw. Verzeichnissen (wirkt dann auf alle Dateien in dem Verzeichnis) kann angegeben werden, ob die Datei nach dem Öffnen überschrieben werden soll ("O" = Overwrite) oder ob die Ausgaben an die Datei angehängt werden sollen ("A" = Append). Standardwert ist "A". Eine nichtexistierende Datei/Verzeichnis wird beim Öffnen neu angelegt.

Für das Gerät V.24-Schnittstelle werden nur die Einstellungen für Baudrate, Daten-Bits, Stopp-Bits, Parität, Protokoll und ggf. Ende in dieser Reihenfolge angegeben.

Für die Dateien, die auf LOCAL_DRIVE erzeugt/abgelegt werden, wird über das Datum LOCAL_DRIVE_MAX_FILESIZE eine maximale Dateigröße in Bytes – gültig einheitlich für alle Dateien – eingestellt. Die Dateigröße wird bei Ausführung eines EXTOPEN-Befehls im Append-

Modus überprüft. Optional kann mit dem Datum LOCAL_DRIVE_FILE_MODE der Schreibmodus ("O" = Overwrite, "A" = Append) festgelegt werden. Standardwert ist "A".

Hinweis

Eine Kopiervorlage für die Projektierungsdatei extdev.ini steht im Verzeichnis /siemens/sinumerik/nck zur Verfügung.

Hinweis

Änderungen an der Datei extdev.ini werden erst nach Neustart/Hochlauf des NC wirksam.

Hinweis

USB-Geräte

Für SINUMERIK 828D kann als Ziel eines der Geräte auch "usb" (ohne Partitionsangabe!) für ein frontseitiges USB-Gerät definiert werden. Das Gerät an USB kann aus dem Teileprogramm heraus nur mittelbar über einen symbolischen Gerätebezeichner "/dev/ext/x" angesprochen werden.

Bei SINUMERIK 840D sl können als USB-Geräte nur statisch verbundene USB-Schnittstellen einer TCU projektiert werden. Die Projektierung erfolgt über die Art SERVER:/PATH als Spezifikation für "Server" im obigen Sinne, wobei SERVER der TCU-Name ist, und /PATH die USB-Schnittstelle bezeichnet. Die jeweiligen USB-Schnittstellen einer TCU werden mit "dev0-0", "dev0-1", "dev1-0" angesprochen. Die Pfadangabe beginnt immer mit "/Partition", wobei die Partition durch ihre zweistellige Partitionsnummer oder ihren Partitionsnamen angegeben sein kann und ggf. mit einem Dateipfad bis zum gewünschten Ziel verlängert wird, also z. B.:

```
/dev/ext/8 = "TCU4:/dev0-0, /01/, A"
```

```
/dev/ext/8 = "TCU4:/dev0-0, /01/mydir.dir/"
```

```
/dev/ext/8 = "TCU4:/dev0-0, /myfirstpartition/Mydir.dir/myfile.txt, O"
```

Beispiel

```
[ExternalDevices]
```

```
; Kommentarzeile
```

```
; example for V24
```

```
; /dev/v24 = "9600, 8, 1, none, rts [, etx]"
```

```
; examples for network drives
```

```
; /dev/ext/1 = "[/USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE/, /, A"
```

```
; /dev/ext/2 = "[/USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE, /myfile.txt, O"
```

```
; /dev/ext/3 = "[/USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE, /mydir/, A"
```

```
; /dev/ext/4 = "SERVER:/dev0-0, /01/, A"
```

```
; ...
```

```
; SINUMERIK 828 only (USB)
```

```
; /dev/ext/9 = "usb, / [ , O]"
```

```
; default: Partitionsnummer = 1
```

```
; SIEMENS only
```

```
; /dev/cyc/1= "[/USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE, /mydir, A"
```

```
; /dev/cyc/2= "[/USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE/mydir, /, A"
```

```
LOCAL_DRIVE_MAX_FILESIZE = 50000
```

```
LOCAL_DRIVE_FILE_MODE = "O"
```

Wirksamkeit des EXTOPEN-Parameters <WriteMode>

Durch Angabe des Schreibmodus sowohl bei der Projektierung in der Datei extdev.ini als auch beim EXTOPEN-Aufruf kann es zu Rechtekonflikten kommen, die beim EXTOPEN-Aufruf ggf. mit Fehler quittiert werden:

Wert aus extdev.ini	Wert des EXTOPEN-Parameters		
	"OVR"	"APP"	-
"O"	O	Fehler	O
"A"	Fehler	A	A
-	O	A	A
Erläuterung: O: Modus "Überschreiben" ist wirksam. A: Modus "Anhängen" ist wirksam. Fehler: EXTOPEN-Aufruf wird mit Fehler quittiert.			

LOCAL_DRIVE: Dateiattribute

Die mit EXTOPEN auf LOCAL_DRIVE angelegten Dateien erhalten folgende Dateiattribute:

- Eigentümer: "user" Schreib-/Leserecht gesetzt
- Gruppe: "operator" Schreib-/Leserecht gesetzt

10.16.3 Programmierung

Das Schreiben von Daten aus einem Teileprogramm heraus auf ein externes Gerät/Datei erfolgt in drei Schritten:

1. Externes Gerät/Datei öffnen
Mit dem Befehl EXTOPEN wird das externe Gerät/Datei für den Kanal zum Schreiben geöffnet.
2. Daten schreiben
Das Ausgabedatum kann mit den Stringfunktionen der NC-Sprache, z. B. SPRINT, aufbereitet werden. Das Schreiben selbst erfolgt über den WRITE-Befehl.
3. Externes Gerät/Datei schließen
Mit dem Befehl EXTCLOSE oder durch Erreichen des Programmendes (M30) sowie bei Kanal-Reset wird das im Kanal belegte externe Gerät/Datei wieder freigegeben.

Syntax

```
DEF INT <Result>
DEF STRING[<n>] <Output>
...
EXTOPEN(<Result>,<ExtDev>,<SyncMode>,<AccessMode>,<WriteMode>)
...
<Output>="Ausgabe Daten"
WRITE(<Result>,<ExtDev>,<Output>)
...
EXTCLOSE(<Result>,<ExtDev>)
```

Bedeutung

EXTOPEN:	Vordefinierte Prozedur zum Öffnen eines externen Geräts/Datei	
<Result>:	Parameter 1: Ergebnisvariable	
	Anhand des Ergebnisvariablenwerts kann im Programm das Gelingen der Operation ausgewertet und entsprechend fortgefahren werden.	
	Typ:	INT
	Werte:	0 kein Fehler
		1 externes Gerät kann nicht geöffnet werden
		2 externes Gerät ist nicht projiziert
		3 externes Gerät mit ungültigem Pfad projiziert
		4 keine Zugriffsrechte für externes Gerät
		5 Benutzungsmodus: externes Gerät bereits "exklusiv" belegt
		6 Benutzungsmodus: externes Gerät bereits "shared" belegt
		7 Dateilänge größer als LOCAL_DRIVE_MAX_FILESIZE
		8 maximale Anzahl externer Geräte überschritten
		9 Option für LOCAL_DRIVE nicht gesetzt
		11 V.24-Schnittstelle bereits durch Easy-Message-Funktion belegt (nur 828D)
		12 Schreibmodus: Angabe widersprüchlich zu extdev.ini
		16 ungültiger externer Pfad programmiert
		22 externes Gerät nicht gemountet

<ExtDev>:	Parameter 2: Symbolischer Bezeichner für das zu öffnende externe Gerät/Datei	
	Typ:	STRING
	Der symbolische Bezeichner besteht aus:	
	1. dem logischen Gerätenamen	
	2. ggf. gefolgt von einem Dateipfad (angehängt mit "/").	
	Folgende logische Gerätenamen sind definiert:	
	"LOCAL_DRIVE":	Lokale CompactFlash Card (vordefiniert)
	"CYC_DRIVE":	reservierte Laufwerksangabe zur Verwendung in SIE-MENS-Zyklen (vordefiniert)
	"/dev/ext/1", ... "/dev/ext/9":	Verfügbare Netzlaufwerke Hinweis: Projektierung in der Datei extdev.ini erforderlich!
	"/dev/cyc/1", "/dev/cyc/2":	reservierte Laufwerksangaben zur Verwendung in SIE-MENS-Zyklen Hinweis: Projektierung in der Datei extdev.ini erforderlich!
"/dev/v24":	V.24-Schnittstelle Hinweis: Projektierung in der Datei extdev.ini erforderlich!	
Dateipfad:		
<ul style="list-style-type: none">• Zu "LOCAL_DRIVE" und "CYC_DRIVE" muss ein Dateipfad angegeben werden, z. B.: "LOCAL_DRIVE/my_dir/my_file.txt"• Die logischen Gerätenamen "/dev/ext/1...9" und "/dev/cyc/1...2" können per Projektierung:<ul style="list-style-type: none">– schon auf eine Datei verweisen, dann darf nur der logische Gerätenamen angegeben werden, z. B.: "/dev/ext/4"– oder auf ein Verzeichnis, dann muss ein Dateipfad angegeben werden, z. B.: "/dev/ext/5/my_dir/my_file.txt"• Zu "/dev/v24" darf kein Dateipfad angehängt sein.		
Hinweis: Für die logischen Gerätenamen "/dev/ext/1...9", "/dev/v24" und "/dev/cyc/1...2" wird Groß-/Kleinschreibung ignoriert, bei der Pfadangabe zu einer Datei ist Groß-/Kleinschreibung signifikant. Für "LOCAL_DRIVE" und "CYC_DRIVE" ist nur Großschreibung zulässig.		

<SyncMode>:	Parameter 3: Bearbeitungsmodus für die WRITE-Befehle zu diesem Gerät/Datei	
	Typ:	STRING
	Werte:	"SYN": Synchrones Schreiben Die Programmausführung wird angehalten, bis der Schreibvorgang abgeschlossen ist. Die erfolgreiche Beendigung des synchronen Schreibens kann durch Auswerten der Fehlervariablen des WRITE-Befehls überprüft werden.
		"ASYN": Asynchrones Schreiben Die Programmausführung wird durch den WRITE-Befehl nicht unterbrochen. Hinweis: Die Ergebnisvariable des WRITE-Befehls ist in diesem Modus nicht aussagekräftig und hat immer den Wert 0 (kein Fehler). Es gibt in diesem Modus keine Sicherheit, dass der WRITE-Befehl erfolgreich war.
<AccessMode>:	Parameter 4: Benutzungsmodus für dieses Gerät/Datei	
	Typ:	STRING
	Werte:	"SHARED": Gerät/Datei wird im "Shared"-Modus angefordert. Andere Kanäle können das Gerät mitverwenden, d. h. ebenfalls in diesem Modus öffnen.
		"EXCL": Gerät/Datei wird in dem Kanal exklusiv verwendet, kein anderer Kanal kann das Gerät mitverwenden.
<WriteMode>:	Parameter 5: Schreibmodus für die WRITE-Befehle zu diesem Gerät/Datei (optional)	
	Typ:	STRING
	Werte:	"APP": Anhängen Die Datei bleibt in ihrem Inhalt erhalten, Schreibaufrufe fügen an das Ende an.
		"OVR": Überschreiben Der Inhalt der Datei wird gelöscht und durch nachfolgende Schreibaufrufe neu erstellt.
	Hinweis: Mit diesem Parameter kann der in der Datei extdev.ini projektierte Schreibmodus nicht überschrieben werden. Im Konfliktfall wird der EXTOPEN-Aufruf mit Fehler quittiert.	

WRITE:	Vordefinierte Prozedur zum Schreiben der Ausgabedaten
--------	---

EXTCLOSE:	Vordefinierte Prozedur zum Schließen eines geöffneten externen Geräts/Datei		
<Result>:	Parameter 1: Ergebnisvariable		
	Typ:	INT	
	Werte:	0	kein Fehler
		16	ungültiger externer Pfad programmiert
		21	Fehler beim Schließen des externen Geräts
<ExtDev>:	Parameter 2: Symbolischer Bezeichner für das zu schließende externe Gerät/DateiBeschreibung siehe unter EXTOPEN! Hinweis: Der Bezeichner muss identisch zu dem im EXTOPEN-Aufruf angegebenen Bezeichner sein!		

Beispiel

Programmcode	
N10	DEF INT RESULT
N20	DEF BOOL EXTDEVICE
N30	DEF STRING[80] OUTPUT
N40	DEF INT PHASE
N50	EXTOPEN(RESULT,"LOCAL_DRIVE/my_file.txt","SYN","SHARED")
N60	IF RESULT > 0
N70	MSG("Fehler bei EXTOPEN:" << RESULT)
N80	ELSE
N90	EXTDEVICE=TRUE
N100	ENDIF
...	
N200	PHASE=4
N210	IF EXTDEVICE
N220	OUTPUT=SPRINT("Ende Phase: %D",PHASE)
N230	WRITE(RESULT,"LOCAL_DRIVE/my_file.txt",OUTPUT)
N240	ENDIF
...	

10.16.4 Randbedingungen

Einfluss auf Bahnsteuerbetrieb

Die Befehle EXTOPEN, WRITE und EXTCLOSE lösen jeweils Vorlaufstopp aus und unterbrechen damit den Bahnsteuerbetrieb.

Verhalten bei Satzsuchlauf

Während "Satzsuchlauf mit Berechnung" erfolgt mit WRITE keine Ausgabe. Es werden jedoch die EXTOPEN- und EXTCLOSE-Befehle aufgesammelt und -- nachdem das Suchziel erreicht wurde -- mit NC-Start wirksam gesetzt. Nachfolgende WRITE-Befehle finden damit dieselbe Umgebung wie bei der normalen Programmbearbeitung vor.

Bei Satzsuchlauf mit Berechnung im Modus "Programmtest" (SERUPRO) werden EXTOPEN, WRITE und EXTCLOSE wie bei der normalen Programmbearbeitung ausgeführt.

10.17 Systemeinstellungen für Hochlauf, RESET/Teileprogrammende und Teileprogramm-Start

Konzept

Das Verhalten der Steuerung kann über Maschinendaten für folgende Ereignissen eingestellt werden:

- Hochlauf (Power-On)
- Reset / Teileprogrammende
- Teileprogrammstart

Verhalten der Steuerung nach:	Einstellbar mit:
Hochlauf (Power-On) ^{*)}	MD20110 \$MC_RESET_MODE_MASK MD20144 \$MC_TRAFO_MODE_MASK MD20150 \$MC_GCODE_RESET_VALUES
RESET/Teileprogrammende	MD20110 \$MC_RESET_MODE_MASK MD20150 \$MC_GCODE_RESET_VALUES MD20152 \$MC_GCODE_RESET_MODE
Teileprogrammstart	MD20112 \$MC_START_MODE_MASK MD20110 \$MC_RESET_MODE_MASK
^{*)} siehe auch POWER ON (Seite 840)	

Systemeinstellungen nach Hochlauf

MD20110 \$MC_RESET_MODE_MASK, Bit 0 = 0 oder 1

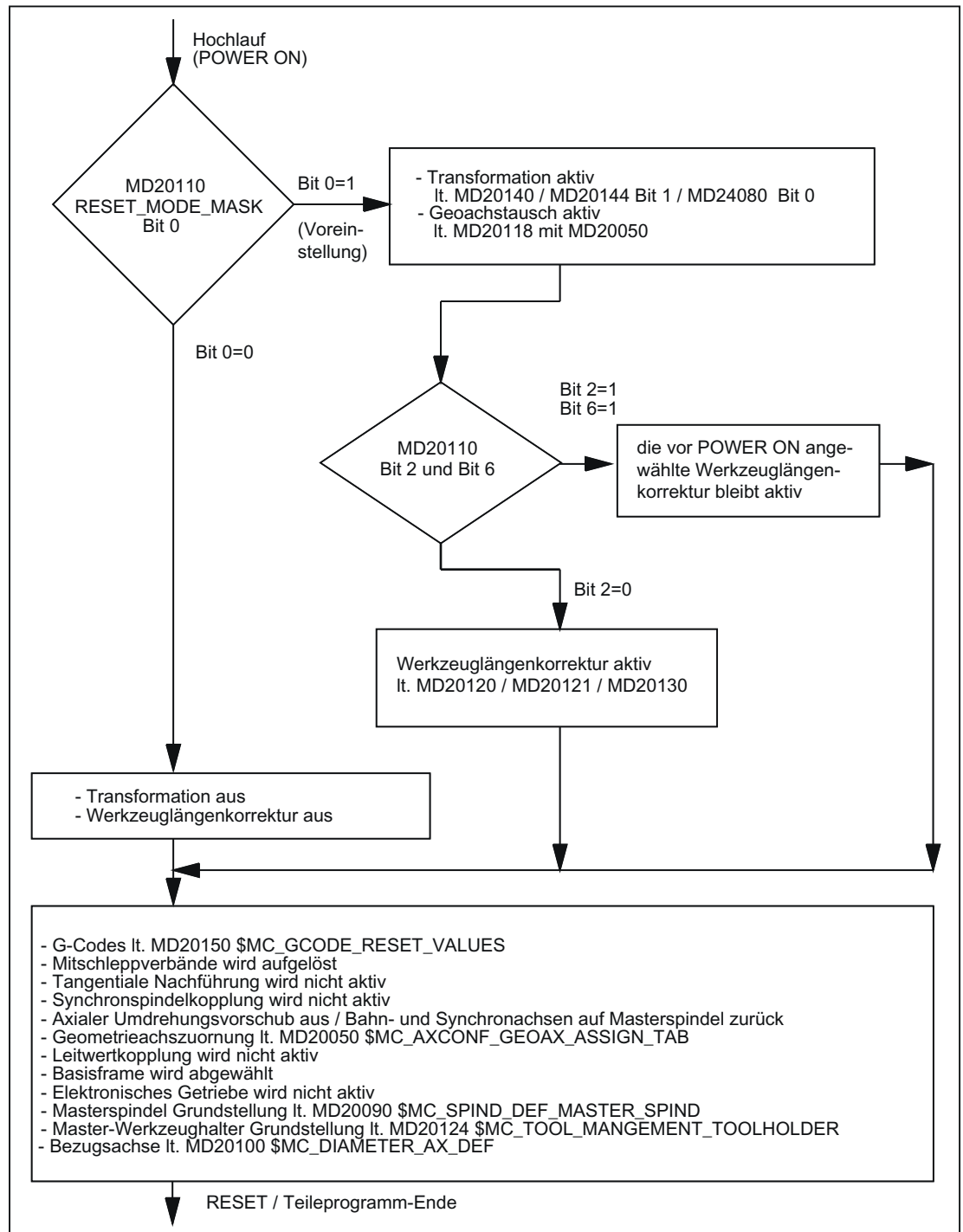


Bild 10-8 Systemeinstellungen nach Hochlauf (Power-On)

Systemeinstellungen nach Reset / Teileprogramm-Ende und Teileprogramm-Start

MD20110 \$MC_RESET_MODE_MASK, Bit 0 = 0 oder 1

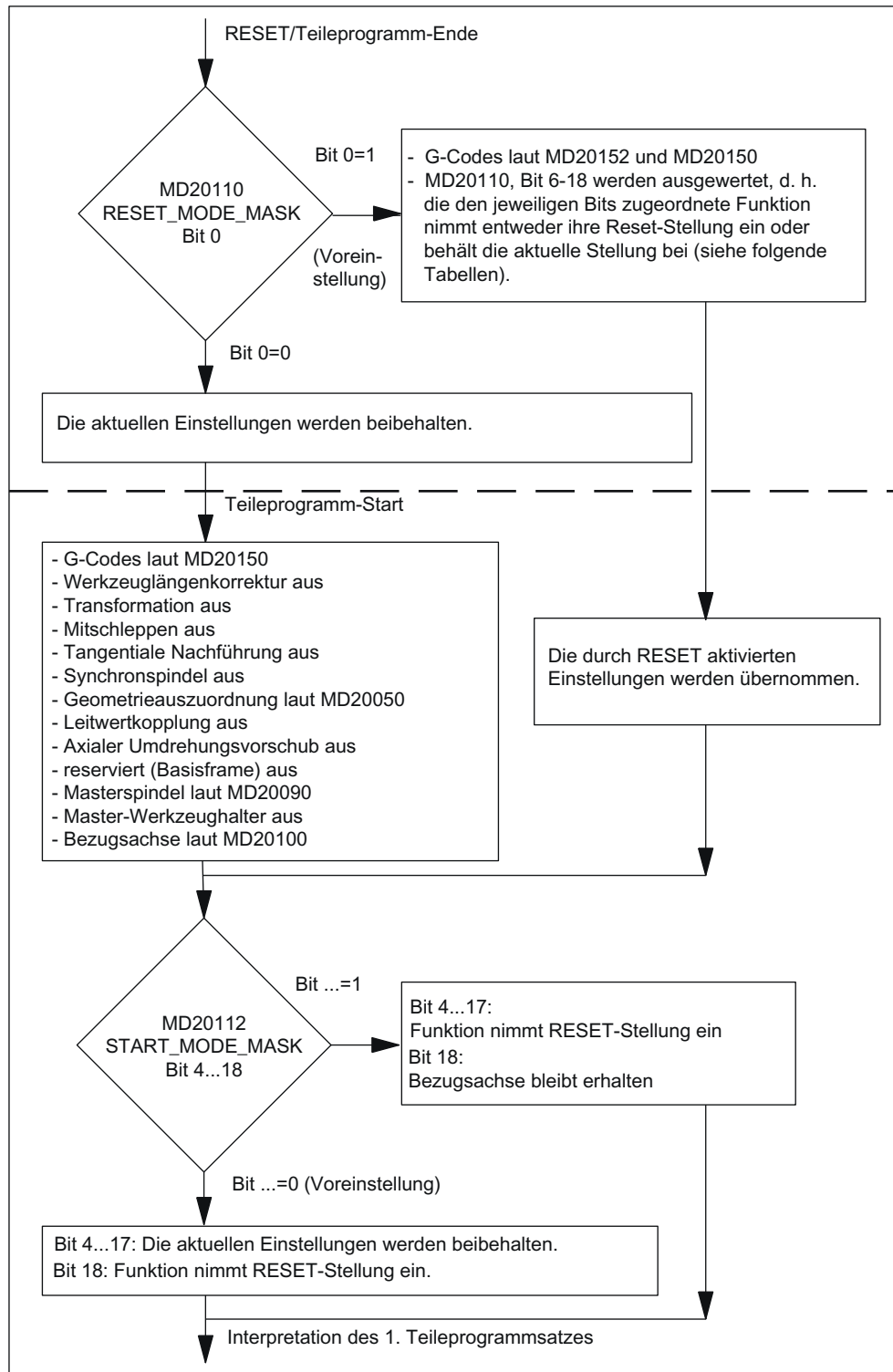


Bild 10-9 Systemeinstellungen nach RESET/Teileprogrammende und Teileprogrammstart

Wirksamer G-Befehl nach Hochlauf und Reset / Teileprogrammende

Die Einstellung des nach Hochlauf (Power-On) und Reset / Teileprogrammende in jeder G-Gruppe wirksamen G-Codes erfolgt über folgende Maschinendaten:

MD20150 \$MC_GCODE_RESET_VALUES[<G-Gruppe>] = <Default-G-Code>

MD20152 \$MC_GCODE_RESET_MODE[<G-Gruppe>] = <Wert>

Wert	Beschreibung: Pro G-Gruppe
0	Der Default-G-Befehl aus MD20150 \$MC_GCODE_RESET_VALUES wird wirksam.
1	Der letzte aktive / aktuelle G-Befehl ist wirksam

Steuerungsgrundstellung nach Hochlauf, Reset / Teileprogrammende und Teileprogrammstart

Die Festlegung der Steuerungsgrundstellung nach Hochlauf (Power-On), Reset / Teileprogrammende und Teileprogrammstart erfolgt über folgende Maschinendaten:

- MD20110 \$MC_RESET_MODE_MASK (Festlegung der Steuerungs-Grundstellung nach **Hochlauf** und **Reset / Teileprogrammende**)
- MD20112 \$MC_START_MODE_MASK (Festlegung der Grundstellung der Steuerung nach **Teileprogrammstart**)

Literatur

Ausführliche Maschinendaten-Beschreibung

Relevante der Maschinendaten

Maschinendatum	Bedeutung
MD20120 \$MC_TOOL_RESET_VALUE	Werkzeug - Längenkorrektur im Hochlauf, Reset / Teileprogrammende
MD20121 \$MC_TOOL_PRESEL_RESET_VALUE	Vorgewähltes Werkzeug bei Reset
MD20130 \$MC_CUTTING_EDGE_RESET_VALUE	Werkzeugschneide - Längenkorrektur im Hochlauf
MD20140 \$MC_TRAFO_RESET_VALUE	Transformationsdatensatz Hochlauf
MD20144 \$MC_TRAFO_MODE_MASK	Funktionsanwahl der kinematischen Transformation
MD20150 \$MC_GCODE_RESET_VALUES	Löschstellung der G-Gruppen
MD20152 \$MC_GCODE_RESET_MODE	Resetverhalten der G-Gruppen
MD21330 \$MC_COUPLE_RESET_MODE_1	Kopplungs-Abbruch-Verhalten
MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB	Zuordnung Geometrieachse zu Kanalachse
MD20118 \$MC_GEOAX_CHANGE_RESET	Automatischen Geometrieachswechsel erlauben

Beispiel**Reset-Stellung bei Reset aktivieren:**

- MD20110, Bit 0 = 1
- MD20112 = 0

Transformation bleibt bei Reset / Teileprogrammstart erhalten:

- MD20110, Bit 0 = 1
- MD20110, Bit 7 = 1
- MD20112 = 0

Werkzeuglängenkorrektur bleibt über Reset / Teileprogrammstart erhalten:

- MD20110, Bit 4 = 1
- MD20110, Bit 6 = 1
- MD20112 = 0

Aktive Ebene (Bit 4) und einstellbarer Frame (Bit 5) bleiben über Reset erhalten und werden bei Teileprogrammstart zurückgesetzt:

- MD20110, Bit 4 = 1
- MD20110, Bit 5 = 1
- MD20112, Bit 4 = 1
- MD20112, Bit 5 = 1

Hinweis

MD20110 / MD20112, Bit 5 und Bit 6

Ist bei Teileprogrammstart in Betriebsart AUTOMATIK bzw. MDA durch entsprechende Parametrierung von MD20110 / MD20112 die Werkzeuglängenkorrektur oder ein Frame aktiv, muss wegen des Verfahrens der Verschiebung, die erste Programmierung der Achsen mit Absolutmaß erfolgen.

Ausnahme: Mit MD42442 / MD42440 wurde das Verfahren der Verschiebung bei G91 unterdrückt.

10.17.1 Werkzeugrückzug nach POWER ON mit Orientierungstransformation

Funktion

Wurde ein Teileprogramm mit einer Bearbeitung mit Werkzeugorientierung durch Spannungsausfall oder Reset abgebrochen, besteht nach dem Hochlauf der Steuerung (Power On) die Möglichkeit, die zuvor aktive Transformation wieder anzuwählen und einen Frame in Richtung der Werkzeugachse zu generieren. Danach kann das Werkzeug in der Betriebsart JOG durch eine Rückzugsbewegung in Richtung der Werkzeugachse freigefahren werden.

Voraussetzung

Für alle an der Transformation beteiligten Maschinenachsen, müssen die aktiven Messsysteme einen Maschinenbezug haben. Siehe Kapitel "Automatisches Wiederherstellen des Maschinenbezugs (Seite 1287)".

Parametrierung

Damit die zuletzt aktive Transformation über POWER ON erhalten bleibt, sind folgende Maschinendaten einzustellen:

- MD20144 \$MC_TRAFO_MODE_MASK, Bit 1 = 1
- MD20110 \$MC_RESET_MODE_MASK, Bit 0 = 1
- MD20110 \$MC_RESET_MODE_MASK, Bit 7 = 1

Siehe auch Kapitel "Systemeinstellungen für Hochlauf, RESET/Teileprogrammende und Teileprogramm-Start (Seite 658)".

Programmierung

Warten auf Maschinenbezug WAITENC

Mit dem Befehl WAITENC wird in einem Programm kanalspezifisch gewartet, bis für alle aktiven Messsysteme der parametrierten Achsen ein gültiger Maschinenbezug vorliegt. Siehe oben Abschnitt "Voraussetzung". Die Parametrierung der Achsen erfolgt über:

MD34800 \$MA_WAIT_ENC_VALID = 1

Anwendung

In dem beim Hochlauf (Voraussetzung: MD20108 Bit 3 = 1) aufzurufenden ereignisgesteuerten Anwenderprogramm (.../_N_CMA_DIR/_N_PROG_EVENT_SPF) muss mit dem Befehl WAITENC gewartet werden, bis gültige Achspositionen zur Verfügung stehen.

Danach kann durch den NC-Sprachbefehl TOROTX/TOROTY/TOROTZ ein Frame generiert werden, der die Werkzeugachse in Richtung der X-, Y- oder Z-Achse legt.

Beispiel

Orientierungstransformation und Orientierungsachsen mit inkrementellen Gebern.

Projektiertung:	Bedeutung:
MD10720 \$MN_OPERATING_MODE_DEFAULT [0] = 6	Hochlauf in Betriebsart JOG
MD30240 \$MA_ENC_TYPE [0, <Achse>] = 1	Inkrementelles Messsystem
MD34210 \$MA_ENC_REFP_STATE [0, <Achse>] = 3	Restaurieren der Achspositionen für Inkrementalgeber freigeben.
MD20108 \$MC_PROG_EVENT_MASK = 'H9'	Ereignisgesteuertes Anwenderprogramm (PROG_EVENT) im Hochlauf und bei Teileprogrammstart aktivieren.
MD20152 \$MC_GCODE_RESET_MODE [52] = 1	TOFRAME über Reset erhalten.
MD20110 \$MC_RESET_MODE_MASK = 'HC1'	Transformation und WZK über Reset erhalten.
MD20144 \$MC_TRAFO_MODE_MASK = 'H02'	Transformation über POWER OFF erhalten.

Ereignisgesteuertes Anwenderprogramm (.../_N_CMA_DIR/_N_PROG_EVENT_SPF)

Programmcode	Kommentar
; Beispiel mit Aktivierung des Frames, der das WKS in Werkzeugrichtung ausrichtet, im Hochlauf und Rücksetzen mit Teileprogrammstart.	

Programmcode	Kommentar
IF \$P_PROG_EVENT == 4	; Hochlauf.
IF \$P_TRAFO <> 0	; Transformation wurde angewählt.
WAITENC	; Warten auf gültige Achspositionen der Orientierungsachsen.
TOROTZ	; Z-Achse des WKS in Richtung der Werkzeugachse drehen.
ENDIF	
M17	
ENDIF	
IF \$P_PROG_EVENT == 1	; Teilprogrammstart.
TOROTOF	; Werkzeugframe zurücksetzen.
RET	
ENDIF	

Der Befehl WAITENC entspricht prinzipiell folgender Programmsequenz (Beispiel für 5-Achs-Maschine mit AB-Kinematik):

Programmcode	Kommentar
WHILE TRUE	; Warten auf Messsystem.
IF ((\$AA_ENC_ACTIVE[X]==TRUE) AND (\$AA_ENC_ACTIVE[Y]==TRUE) AND (\$AA_ENC_ACTIVE[Z]==TRUE) AND (\$AA_ENC_ACTIVE[A]==TRUE) AND (\$AA_ENC_ACTIVE[B]==TRUE)) GOTO GET_LABEL	
ENDIF	
G4 F0.5	; Wartezeit 0,5 s
ENDWHILE	
:Position synchronisieren	
GET_LABEL: GET(X,Y,Z,A,B,)	

Bearbeitung fortsetzen

Betriebsart AUTOMATIK

Zum automatischen Abarbeiten von Programmen in der Betriebsart AUTOMATIK müssen alle Maschinenachsen, deren Istposition des aktive Messsystems restauriert wurde, referenziert werden.

Betriebsart MDA und Überspeichern

In der Betriebsart MDA und für das Überspeichern, kann die Bearbeitung auch ohne Referenzieren der Achsen, mit restaurierten Positionen erfolgen. Dazu muss kanalspezifisch NC-Start mit restaurierten Positionen explizit freigegeben werden:

MD20700 \$MC_REFP_NC_START_LOCK = 2

Randbedingung

Achsen mit inkrementellen Gebern und ohne Istwert-Pufferung

Es wird davon ausgegangen, dass Achsen mit inkrementellen Gebern und **ohne** Istwert-Pufferung bei Spannungsausfall hinreichend schnell geklemmt werden, um ein Wegdriften von der letzten Sollposition zu verhindern.

10.18 Ersetzung von Funktionen durch Unterprogramme

10.18.1 Übersicht

Funktion

Anwenderspezifische Hilfsfunktionen (z. B. M101) lösen keine Systemfunktionen aus. Sie werden lediglich an die NC/PLC-Nahtstelle ausgegeben. Die Funktionalität der Hilfsfunktion ist vom Anwender/Maschinenhersteller im PLC-Anwenderprogramm zu realisieren. Im weiteren Verlauf wird beschrieben, wie statt der standardmäßigen Ausgabe an die NC/PLC-Nahtstelle der Aufruf eines anwenderspezifischen Unterprogramms (Ersetzungsunterprogramm) projiziert wird.

Im Teileprogramm wird dann weiterhin die Funktion M101 programmiert. Bei der Abarbeitung des Teileprogramms erfolgt aber der Aufruf des Ersetzungsunterprogramms. Die Funktion wird somit von der NC durch einen Unterprogrammaufruf ersetzt. Daraus ergeben sich folgende Vorteile:

- Bei Anpassungen an den Fertigungsprozess kann ein bestehendes, getestetes und bewährtes Teileprogramm unverändert weiter verwendet werden. Die Anpassungen werden in anwenderspezifische Unterprogramme ausgelagert.
- Die Realisierung der Funktionalität kann innerhalb des Ersetzungsunterprogramms mit dem vollen Funktionsumfang der NC-Sprache erfolgen.
- Es entfällt der Kommunikations-Overhead zwischen NC und PLC.

Ersetzbare Funktionen

Folgende Funktionen können durch Unterprogramme ersetzt werden:

Hilfsfunktionen	
M	Schaltfunktionen
T	Werkzeuganwahl
TCA	Werkzeuganwahl unabhängig vom Status des Werkzeugs
D	Werkzeugkorrektur
DL	Additive Werkzeugkorrektur

Spindelbezogene Funktionen bei aktiver Synchronspindelkopplung	
M40	Automatischer Getriebestufenwechsel
M41 - M45	Getriebestufenanwahl 1 ... 5
SPOS	Spindelpositionieren
SPOSA	Spindelpositionieren
M19	Spindelpositionieren

10.18.2 Ersetzung von M-, T/TCA- und D/DL-Funktionen

10.18.2.1 Ersetzung von M-Funktionen

Allgemeine Informationen

Für die Ersetzung von M-Funktionen gelten folgende Bedingungen:

- Pro Satz wird nur eine M-Funktion ersetzt.
- Ein Satz in dem eine M-Funktion ersetzt werden soll, darf folgende Elemente **nicht** enthalten:
 - M98
 - Modalen Unterprogramm-Aufruf
 - Unterprogrammrücksprung
 - Teileprogrammende
- M-Funktionen, die Systemfunktionen auslösen, dürfen nicht durch ein Unterprogramm ersetzt werden (siehe Absatz "M-Funktionen, die nicht ersetzt werden").

Parametrierung

M-Funktion und Unterprogramm

Die Parametrierung der M-Funktionen und der Ersetzungsunterprogramme erfolgt in folgenden Maschinendaten:

- MD10715 \$MC_M_NO_FCT_CYCLE[<Index>] = <M-Funktionsnummer>
- MD10716 \$MC_M_NO_FCT_CYCLE_NAME[<Index>] = "<Unterprogrammname>"

Die M-Funktion und das entsprechende Ersetzungsunterprogramm sind durch den gleichen Index verbunden.

Beispiel: M-Funktion M101 wird durch Unterprogramm SUB_M101 und M-Funktion M102 durch SUB_M102 ersetzt:

MD10715 \$MC_M_NO_FCT_CYCLE[0]	= 101
MD10716 \$MC_M_NO_FCT_CYCLE_NAME[0]	= "SUB_M101"
MD10715 \$MC_M_NO_FCT_CYCLE[1]	= 102
MD10716 \$MC_M_NO_FCT_CYCLE_NAME[1]	= "SUB_M102"

Systemvariable zur Informationsweitergabe

Für eine frei wählbare M-Funktion werden über Systemvariable (siehe Kapitel "Systemvariable (Seite 671)") Informationen bezüglich der ersetzten M-Funktion und weiterer Funktionen (T, TCA, D, DL) zur Auswertung im Ersetzungsunterprogramm zur Verfügung gestellt. Die in den Systemvariablen enthaltenen Daten beziehen sich auf den Satz, in dem die zu ersetzende M-Funktion programmiert wurde.

Die Auswahl der M-Funktion erfolgt mit dem Index des Maschinendatums MD10715
\$MC_M_NO_FCT_CYCLE[<Index>], in dem die zu ersetzende M-Funktion parametrier ist:
MD10718 \$MC_M_NO_FCT_CYCLE_PAR = <Index>

Hinweis

Bei einer M-Funktionsersetzung mit Informationsweitergabe über Systemvariable müssen die Adresserweiterung und der Funktionswert der M-Funktion als konstante Werte programmiert werden.

Zulässige Programmierung:

- M<Funktionswert>
- M=<Funktionswert>
- M[<Adresserweiterung>]=<Funktionswert>

Unzulässige Programmierung:

- M=<Variable1>
 - M[<Variable2>]=<Variable1>
-

Programmierung

Regeln zur Ersetzung von M-Funktionen:

- Das Ersetzungsunterprogramm wird am Satzende aufgerufen
- Innerhalb des Ersetzungsunterprogramms werden keine M-Funktionen ersetzt
- In einem ASUP wird die M-Funktion auch dann ersetzt, wenn das ASUP innerhalb des Ersetzungsunterprogramms gestartet wurde.

M-Funktionen, die nicht ersetzt werden

Folgende M-Funktionen lösen als vordefinierte Hilfsfunktionen Systemfunktionen aus und dürfen **nicht** durch ein Unterprogramm ersetzt werden:

- M0 ... M5
- M17, M30
- M19
- M40 ... M45
- M98, M99 (Nur bei MD18800 \$MN_MM_EXTERN_LANGUAGE ≠ 0)

Die über Maschinendaten parametrierten anwenderspezifischen M-Funktionen dürfen ebenfalls nicht durch ein Unterprogramm ersetzt werden, da auch sie Systemfunktionen auslösen.

Maschinendatum	Bedeutung
MD10714 \$MN_M_NO_FCT_EOP	M-Funktion für Spindel aktiv nach RESET
MD10804 \$MN_EXTERN_CHAN_M_NO_SET_INT	M-Funktion für ASUP-Aktivierung (Externmodus)

10.18 Ersetzung von Funktionen durch Unterprogramme

Maschinendatum	Bedeutung
MD10806 \$MN_EXTERN_CHAN_M_NO_DISABLE_INT	M-Funktion für ASUP-Deaktivierung (Externmodus)
MD10814 \$MN_EXTERN_M_NO_MAC_CYCLE	Makroaufruf über M-Funktion
MD20094 \$MC_SPIND_RIGID_TAPPING_M_NR	M-Funktion für Umschalten in gesteuerten Achsbetrieb
MD20095 \$MC_EXTERN_RIGID_TAPPING_M_NR	M-Funktion für Umschalten in gesteuerten Achsbetrieb (Externmodus)
MD22254 \$MC_AUXFU_ASSOC_M0_VALUE	Zusätzliche M-Funktion für Programm-Halt
MD22256 \$MC_AUXFU_ASSOC_M1_VALUE	Zusätzliche M-Funktion für bedingten Halt
MD26008 \$MC_NIBBLE_PUNCH_CODE	Festlegung der M-Funktionen (für Nibbeln -spezifisch)
MD26012 \$MC_PUNCHNIB_ACTIVATION	Aktivierung der Stanz- und Nibbelfunktionen

Hinweis**Ausnahme**

Die über MD22560 \$MC_TOOL_CHANGE_M_CODE (Werkzeugwechsel mit M-Funktion) parametrisierte M-Funktion darf durch ein Unterprogramm ersetzt werden.

10.18.2.2 Ersetzung von T/TCA- und D/DL-Funktionen

Randbedingungen

Für die Ersetzung der Funktionen T, TCA, D und DL gelten folgende Randbedingungen:

- Pro Satz wird maximal eine Funktionsersetzung wirksam.
- Ein Satz mit Funktionsersetzung darf folgende Elemente **nicht** enthalten:
 - M98
 - Modalen Unterprogramm-Aufruf
 - Unterprogrammrücksprung
 - Teileprogrammende
- Erfolgt bei der Multitool-Anwahl mit T = Platznummer die Programmierung der Multitool-Platznummer über die Adresse MTL, so wird mit der T-Ersetzung auch die Adresse MTL ersetzt. Im Ersetzungsunterprogramm kann der programmierte Werte mit den Systemvariablen \$C_MTL_PROG und \$C_MTL abgefragt werden.

Parametrierung: Ersetzungsunterprogramm

Die Angabe des Ersetzungsunterprogramms erfolgt funktionspezifisch im Maschinendatum:

Funktion	Maschinendatum
T	MD10717 \$MN_T_NO_FCT_CYCLE_NAME
TCA	MD15710 \$MN_TCA_CYCLE_NAME
D/DL	MD11717 \$MN_D_NO_FCT_CYCLE_NAME

Hinweis

Es wird empfohlen, für die Ersetzung der T-, TCA- und D/DL-Funktionen dasselbe Unterprogramm zu verwenden.

Parametrierung: Verhalten bezüglich D- bzw. DL-Funktion bei gleichzeitiger T-Funktion

Bei gleichzeitiger Programmierung der Funktionen D bzw. DL und T in einem Satz wird die D- bzw. DL-Nummer entweder als Parameter an das Ersetzungsunterprogramm übergeben oder die D- bzw. DL-Funktion wird vor Aufruf des Ersetzungsunterprogramms ausgeführt. Das Verhalten ist einstellbar über:

MD10719 \$MN_T_NO_FCT_CYCLE_MODE (Parametrierung der T-Funktionsersetzung)

Bit	Wert	Bedeutung
0	0	Die D- bzw. DL-Nummer steht im Unterprogramm per Systemvariable zur Verfügung (Grundstellung).
	1	Die D bzw. DL-Nummer wird direkt im Satz verrechnet. Hinweis: Diese Funktion ist nur aktiv, wenn der Werkzeugwechsel mit M-Funktion projiziert wurde: MD22550 \$MC_TOOL_CHANGE_MODE = 1 Anderweitig werden die D- bzw. DL-Werte immer übergeben.

Systemvariable zur Informationsweitergabe

Dem Ersetzungsunterprogramm werden alle relevanten Informationen zu den im Satz programmierten Funktionen über Systemvariable (siehe Kapitel "Systemvariable (Seite 671)") zur Verfügung gestellt.

Die in den Systemvariablen enthaltenen Daten beziehen sich auf den Satz, in dem die zu ersetzende Funktion programmiert wurde.

Parametrierung: Aufrufzeitpunkt des Ersetzungsunterprogramms

Der Aufrufzeitpunkt des Ersetzungsunterprogramms wird eingestellt über:

MD10719 \$MN_T_NO_FCT_CYCLE_MODE, Bit 1 und Bit 2

Bit 2	Bit 1	Aufrufzeitpunkt des Ersetzungsunterprogramms
0	0	Am Satzende Nach der Bearbeitung des Ersetzungsunterprogramms wird die Interpretation mit der Programmzeile fortgesetzt, die auf diejenige folgt, die den Ersetzungsvorgang ausgelöst hat.
0	1	Am Satzanfang Nach der Bearbeitung des Ersetzungsunterprogramms wird die Programmzeile, die zum Aufruf des Ersetzungsunterprogramms geführt hat, interpretiert. Dabei werden die Adressen T, D bzw. DL und die M-Funktion für den Werkzeugwechsel nicht mehr bearbeitet.
1	-	Am Satzanfang und Satzende Das Ersetzungsunterprogramm wird zweimal aufgerufen.

Systemvariable für den Aufrufzeitpunkt

Über die Systemvariablen \$P_SUB_STAT kann gelesen werden, ob die Ersetzung aktiv ist und wenn ja, wann das Ersetzungsunterprogramm, bezogen auf den Satz, aufgerufen wurde:

Wert	Bedeutung
0	Ersetzung nicht aktiv
1	Ersetzung aktiv, Unterprogrammaufruf am Satzanfang
2	Ersetzung aktiv, Unterprogrammaufruf am Satzende

Beispiel: Ersetzung der T-Funktion

Parametrierung	Bedeutung
MD22550 \$MC_TOOL_CHANGE_MODE = 0	Werkzeugwechsel mit T-Funktion
MD10717 \$MN_T_NO_FCT_CYCLE_NAME = "MY_T_CYCLE"	Name des Unterprogramms zum Ersetzen der T-Funktion
MD10719 \$MN_T_NO_FCT_CYCLE_MODE = 0	Aufrufzeitpunkt: Satzende

Programmierung	Kommentar
N110 D1	; D1
N120 G90 G0 X100 Y100 Z50	; D1 ist aktiv.
N130 D2 X110 Z0 T5	; D1 bleibt aktiv. Die T-Funktion wird am Satzende durch UP-Aufruf MY_T_CYCLE ersetzt. D2 wird MY_T_CYCLE in einer Systemvariable zur Verfügung gestellt.

Ein ausführliches Beispiel zur Ersetzung der T-Funktion findet sich in Kapitel: "Beispiele zur M/T-Funktionsersetzung beim Werkzeugwechsel (Seite 672)".

10.18.2.3 Systemvariable

Allgemeine Informationen

Dem Ersetzungsunterprogramm werden alle relevanten Informationen zu den im Satz programmierten Funktionen (T bzw. TCA, D bzw. DL, M) über Systemvariable zur Verfügung gestellt.


Ausnahme

Keine Weitergabe der D- bzw. DL-Nummer, wenn:

- MD10719 \$MN_T_NO_FCT_CYCLE_MODE, Bit 0 = 1
- MD22550 \$MC_TOOL_CHANGE_MODE = 1

UND

- D bzw. DL sind zusammen mit T- oder M-Funktion in einem Satz programmiert.

 VORSICHT
Werte sind nicht wirksam
Die dem Ersetzungsunterprogramm in den Systemvariablen zur Verfügung gestellten Werte sind noch nicht wirksam. Es liegt in der alleinigen Verantwortung des Anwenders/ Maschinenherstellers dies durch geeignete Programmierung im Ersetzungsunterprogramm "nachzuholen".

Systemvariable

Systemvariable	Bedeutung
\$C_M_PROG	TRUE, wenn M-Funktion programmiert wurde
\$C_M	Enthält bei \$C_M_PROG == TRUE den Wert der Adresse M Dabei sind zwei Fälle zu unterscheiden: <ul style="list-style-type: none"> • Ist für den Werkzeugwechsel mit M-Funktion ein Unterprogramm mit Parameterübergabe projektiert, liefert \$C_M den Wert aus: MD10715 MN_M_NO_FCT_CYCLE • Ist nur ein Unterprogramm für die Adressen T und/oder D/DL projektiert und im Programm ist die M-Funktion für den Werkzeugwechsel zusammen mit einer der zu ersetzenden Adressen programmiert, liefert \$C_M den Wert aus: MD22560 \$MN_TOOL_CHANGE_M_MODE
\$C_AUX_VALUE[0]	Wert der ersetzten M-Funktion
\$C_ME	Enthält bei \$C_M_PROG == TRUE den Wert der Adresserweiterung der M-Funktion
\$C_AUX_EXT[0]	Adresserweiterung der M-Funktion (identisch zu \$C_ME)
\$C_AUX_IS_QUICK[0]	TRUE, wenn M-Funktion mit schneller Ausgabe an PLC programmiert wurde
\$C_T_PROG	TRUE, wenn T-Funktion programmiert wurde
\$C_T	Enthält bei \$C_T_PROG == TRUE den Wert der T-Funktion

Systemvariable	Bedeutung
\$C_TE	Enthält bei: <ul style="list-style-type: none"> • \$C_T_PROG == TRUE • \$C_TS_PROG == TRUE den Wert der Adresserweiterung der T-Funktion
\$C_TS_PROG	TRUE, wenn bei der T- oder TCA-Ersetzung ein Werkzeugname programmiert wurde.
\$C_TS	Enthält bei \$C_TS_PROG == TRUE den bei T- oder TCA-Ersetzung programmierten Werkzeugname
\$C_TCA	TRUE, wenn die TCA-Ersetzung aktiv ist
\$C_DUPLO_PROG	TRUE, wenn die Duplo-Nummer bei der TCA-Ersetzung programmiert wurde
\$C_DUPLO	Enthält bei \$C_DUPLO_PROG == TRUE den Wert der programmierten Duplo-Nummer
\$C_THNO_PROG	TRUE, wenn die Toolholder/Spindel-Nummer bei der TCA-Ersetzung programmiert wurde
\$C_THNO	Enthält bei \$C_THNO_PROG == TRUE den Wert der programmierten Toolholder/Spindel-Nummer
\$C_D_PROG	TRUE, wenn D-Funktion programmiert wurde
\$C_D	Enthält bei \$C_D_PROG == TRUE den Wert der D-Funktion
\$C_DL_PROG	TRUE, wenn DL-Funktion programmiert wurde
\$C_DL	Enthält bei \$C_DL_PROG == TRUE den Wert der DL-Funktion
\$P_SUB_STAT	Satzbezogener Aufrufzeitpunkt des Ersetzungsunterprogramms
\$C_MTL_PROG	TRUE, wenn Adresse MTL programmiert wurde
\$C_MTL	Enthält bei \$C_MTL_PROG == TRUE den Wert der Adresse MTL

10.18.2.4 Beispiel: Ersetzung einer M-Funktion

Beispiel 1

Die Funktion M6 wird durch den Aufruf des Unterprogramms "SUB_M6" ersetzt.

Die für einen Werkzeugwechsel relevanten Informationen sollen über Systemvariable übergeben werden.

Parametrierung

Maschinendatum

MD10715 \$MN_M_NO_FCT_CYCLE[2] = 6

MD10716 \$MN_M_NO_FCT_CYCLE_NAME[2] = "SUB_M6"

MD10718 \$MN_M_NO_FCT_CYCLE_PAR = 2

Bedeutung

Werkzeugwechsel mit M6

Ersetzungsunterprog. für M6

Info. weitergabe über Systemvariable

Hauptprogramm

Programmierung	Kommentar
PROC MAIN	
...	;
N10 T1 D1 M6	; M6 wird durch Unterprogramm "SUB_M6"
	; ersetzt
...	;
N90 M30	

Unterprogramm "SUB_M6"

Programmierung	Kommentar
PROC SUB_M6	
N110 IF \$C_T_PROG==TRUE	; IF Adresse T programmiert
N120 T[\$C_TE]=\$C_T	; T-Anwahl ausführen
N130 ENDIF	; ENDIF
N140 M[\$C_ME]=6	; Werkzeugwechsel ausführen.
N150 IF \$C_D_PROG==TRUE	; IF Adresse D programmiert
N160 D=\$C_D	; D-Anwahl ausführen
N170 ENDIF	; ENDIF
N190 M17	

Beispiele 2

Das neue Werkzeug wird mit der T-Funktion für den Wechsel vorbereitet. Der Werkzeugwechsel erfolgt erst mit der Funktion M6. Die T-Funktion wird durch den Aufruf des Unterprogramms "MY_T_CYCLE " ersetzt. Die D / DL-Nummer wird an das Unterprogramm übergeben.

Parametrierung

Parametrierung	Bedeutung
MD22550 \$MC_TOOL_CHANGE_MODE = 1	Werkzeugwechsel vorbereiten mit T-Funktion
MD10717 \$MN_T_NO_FCT_CYCLE_NAME = "MY_T_CYCLE"	Ersetzungsunterprogramm
MD10719 \$MN_T_NO_FCT_CYCLE_MODE = 0	Übergabe der D/DL-Nummer

Hauptprogramm

Programmcode	Kommentar
N210 D1	;
N220 G90 G0 X100 Y100 Z50	; D1 ist aktiv.
N230 D2 X110 Z0 T5	; D1 bleibt aktiv, programmiertes D2 wird
	; dem Unterprogramm als Variable übergeben
N240 M6	; Werkzeugwechsel ausführen

Beispiele 3

Das neue Werkzeug wird mit der T-Funktion für den Wechsel vorbereitet. Der Werkzeugwechsel erfolgt erst mit der Funktion M6. Die T-Funktion wird durch den Aufruf des Unterprogramms "MY_T_CYCLE " ersetzt. Die D / DL-Nummer wird **nicht** an das Unterprogramm übergeben.

Parametrierung

Parametrierung	Bedeutung
MD22550 \$MC_TOOL_CHANGE_MODE = 1	Werkzeugwechsel vorbereiten mit T-Funktion
MD10717 \$MN_T_NO_FCT_CYCLE_NAME = "MY_T_CYCLE"	Ersetzungsunterprogramm
MD10719 \$MN_T_NO_FCT_CYCLE_MODE = 1	Keine Übergabe der D/DL-Nummer

Hauptprogramm

Programmcode	Kommentar
N310 D1	
N320 G90 G0 X100 Y100 Z50	; D1 ist aktiv.
N330 D2 X110 Z0 T5	; D2 wird aktiv und nicht als Variable an ; das Ersetzungsunterprogramm übergeben.
N340 M6	; Werkzeugwechsel ausführen.

Beispiele 4

Die Funktionen T und M6 werden durch das Unterprogramm "MY_T_CYCLE" ersetzt.

Die Parameter werden bei der Ersetzung von M6 an das Unterprogramm übergeben.

Wird M6 zusammen mit D bzw. DL im Satz programmiert, wird die D- bzw. DL-Nummer auch dann als Parameter an das Unterprogramm übergeben, wenn keine Übergabe der D/DL-Nummer parametrier ist:

MD10719 \$MN_T_NO_FCT_CYCLE_MODE = 1

Parametrierung

Projektiertung	Bedeutung
MD22550 \$MC_TOOL_CHANGE_MODE = 1	Werkzeugwechsel mit M-Funktion
MD22560 \$MC_TOOL_CHANGE_M_CODE = 6	M-Code für Werkzeugwechsel
MD10715 \$MC_M_NO_FCT_CYCLE[3] = 6	Zu ersetzende M-Funktion
MD10716 \$MC_M_NO_FCT_CYCLE_NAME[3] = "MY_T_CYCLE"	Ersetzungsunterprogramm für M-Funktion
MD10717 \$MN_T_NO_FCT_CYCLE_NAME = "MY_T_CYCLE"	Ersetzungsunterprogramm für T-Funktion
MD10718 \$MN_M_NO_FCT_CYCLE_PAR = 3	Parameterübergabe an das Ersetzungsunterprog. für M6
MD10719 \$MN_T_NO_FCT_CYCLE_MODE = 1	Keine Übergabe der D/DL-Nummer

Hauptprogramm

Programmcode	Kommentar
N410 D1	
N420 G90 G0 X100 Y100 Z50	; D1 ist aktiv.
N330 D2 X110 Z0 T5 M6	; D1 bleibt aktiv, D2 und T5 werden dem M6-Ersetzungsunterprogramm als Variablen übergeben.

10.18.2.5 Beispiel: Ersetzung einer T- und D-Funktion

Die Funktionen T und D werden durch den Aufruf des Unterprogramms "D_T_SUB_PROG" ersetzt. Für das Beispiel soll weiter gelten:

- Der Werkzeugwechsel erfolgt mit der Adresse T.
- Der Unterprogrammaufruf erfolgt am Satzanfang.
- Die Werkzeugverwaltung ist nicht aktiv.
- Die Achse B ist eine Teilungsachse mit Hirth-Verzahnung.

Parametrierung

Maschinendatum

MD11717 \$MN_D_NO_FCT_CYCLE_NAME = "D_T_SUB_PROG"

MD10717 \$MN_T_NO_FCT_CYCLE_NAME = "D_T_SUB_PROG"

MD10719 \$MN_T_NO_FCT_CYCLE_MODE = 'H2'

MD22550 \$MC_TOOL_CHANGE_MODE = 0

Bedeutung

Ersetzungs-unterprogramm
für D-Funktion

Ersetzungs-unterprogramm
für M-Funktion

Aufruf am Satzanfang

Werkzeugwechsel mit
T-Funktion

Hauptprogramm

Programmierung	Kommentar
PROC MAIN	
...	;
N10 G01 F1000 X10 T1=5 D1	; T- und D-Funktion durch Aufruf von
	; "D_T_SUB_PROG" am Satzanfang ersetzt
...	;
N90 M30	

Unterprogramm "D_T_SUB_PROG"

Programmierung	Kommentar
N1000 PROC D_T_SUB_PROG DISPLOF SBLOF	

Programmierung	Kommentar
N4100 IF \$C_T_PROG==TRUE	; IF Adresse T programmiert
N4120 POS[B]=CAC(\$C_T)	; Teilungsposition anfahren
N4130 T[\$C_TE]=\$C_T	; Werkzeug anwählen (T-Anwahl)
N4140 ENDIF	; ENDIF
N4300 IF \$C_D_PROG==TRUE	; IF Adresse D programmiert
N4320 D=\$C_D	; Korrektur anwählen (D-Anwahl)
N4330 ENDIF	; ENDIF
N4400 IF \$C_DL_PROG==TRUE	; IF Adresse DL programmiert
N4420 D=\$C_DL	; Einsatzkorrektur anwählen
N4430 ENDIF	; ENDIF
N9999 RET	

10.18.2.6 Verhalten im Konfliktfall

Konfliktfall

Ein Konfliktfall liegt vor, wenn in einem Satz mehrere Funktionen programmiert sind und die Funktionen mit unterschiedlichen Unterprogrammen ersetzt werden sollen:

- Adresse D und DL ersetzen mit Unterprogramm:
MD11717 \$MN_FCT_CYCLE_NAME = "D_SUB_PROG"
- Adresse T ersetzen mit Unterprogramm:
MD10717 \$MN_FCT_CYCLE_NAME = "T_SUB_PROG"
- M-Funktion M6 ersetzen mit Unterprogramm:
MD10715 \$MN_M_NO_FCT_CYCLE[0] = 6
MD10716 \$MN_M_NO_FCT_CYCLE_NAME[0] = "M6_SUB_PROG"
MD10718 \$MN_M_NO_FCT_CYCLE_PAR = 0
MD22550 \$MC_TOOL_CHANGE_MODE = 1
MD22560 \$MC_TOOL_CHANGE_M_CODE = 6

Auflösung

Ein Konfliktfall wird entsprechend der nachfolgenden Tabelle aufgelöst:

In einer Programmzeile sind programmiert:			Aufgerufenes Unterprogramm:
D und/oder DL	T oder TCA	M6	
–	–	x	M6_SUB_PROG
–	x	–	T_SUB_PROG
–	x	x	M6_SUB_PROG
x	–	–	D_SUB_PROG
x	–	x	M6_SUB_PROG

In einer Programmzeile sind programmiert:			Aufgerufenes Unterprogramm:
D und/oder DL	T oder TCA	M6	
x	x	–	T_SUB_PROG
x	x	x	M6_SUB_PROG

10.18.3 Ersetzung von Spindel-Funktionen

10.18.3.1 Allgemeine Informationen

Funktion

Bei einer aktiven Kopplung können für Leitspindeln folgende Spindelfunktionen ersetzt werden:

- M40: Automatischer Getriebestufenwechsel
- M41 ... M45: Programmierter Getriebestufenwechsel
- SPOS, SPOSA und M19: Spindelpositionieren

Randbedingungen

- Für die Ersetzung einer Spindelfunktion müssen folgende Bedingungen erfüllt sein:
 - Die programmierte Spindel muss Leitspindel einer aktiven Kopplung sein.
 - Leit- und Folgespindel müssen sich im gleichen Kanal befinden. Dies wird nur erkannt, wenn sich die Leitspindel in dem Kanal befindet, in dem die Kopplung geschlossen wurde. Wird die Leitspindel in einen anderen Kanal getauscht, führt ein Getriebestufenwechsel bzw. ein Positionieren dieser Spindel nicht zum Aufruf des Ersetzungsunterprogramms.
 - Ein programmierter Getriebestufenwechsel muss einen realen Getriebestufenwechsel bewirken. Dazu müssen sich programmierte und aktive Getriebestufe unterscheiden.
- In einem Satz kann nur eine Spindelfunktion ersetzt werden. Mehrfachersetzungen führen zum Abbruch der Programmbearbeitung. Die Spindelfunktionen, die ersetzt werden sollen, müssen dann auf mehrere Sätze aufgeteilt werden.

Parametrierung

Spindelfunktion

Die Spindelfunktionen, die durch das Unterprogramm ersetzt werden sollen, werden ausgewählt im Maschinendatum:

MD30465 \$MA_AXIS_LANG_SUB_MASK

Bit	Bedeutung	
0	Getriebestufenwechsel automatisch (M40) und direkt (M41-M45)	
	Wert	Bedeutung
	0	Keine Ersetzung
	1	Ersetzung durch das in MD15700 und MD15702 eingestellte Unterprogramm
1	Spindelpositionieren mit SPOS / SPOSA / M19	
	Wert	Bedeutung
	0	Keine Ersetzung
	1	Ersetzung durch das in MD15700 und MD15702 eingestellte Unterprogramm

Unterprogramm: Name

Der Name des Ersetzungsunterprogramms wird eingetragen im Maschinendatum:

MD15700 \$MN_LANG_SUB_NAME = "<Unterprogrammname>"

Unterprogramm: Pfad

Der Pfad des Ersetzungsunterprogramms wird eingestellt im Maschinendatum:

MD15702 \$MN_LANG_SUB_PATH = <Wert>

Wert	Bedeutung
0	Hersteller -Zyklenverzeichnis: /_N_CMA_DIR
1	Anwender -Zyklenverzeichnis: /_N_CUS_DIR
2	Siemens -Zyklenverzeichnis: /_N_CST_DIR

Systemvariable: Aufrufzeitpunkt des Ersetzungsunterprogramms

Der Aufrufzeitpunkt des Ersetzungsunterprogramms kann über die Systemvariable \$P_SUB_STAT gelesen werden:

Wert	Bedeutung
0	Ersetzung nicht aktiv
1	Ersetzung aktiv, Unterprogrammaufruf am Satzanfang
2	Ersetzung aktiv, Unterprogrammaufruf am Satzende

Satzbearbeitung

Wird das Ersetzungsunterprogramm am Satzanfang aufgerufen, wird nach der Bearbeitung des Ersetzungsunterprogramms der Satz, der zum Aufruf geführt hat, abgearbeitet. Die ersetzten Befehle werden nicht mehr bearbeitet.

Wird das Ersetzungsunterprogramm am Satzende aufgerufen, wird zuerst der Satz, der zum Aufruf des Ersetzungsunterprogramms führt, ohne die zu ersetzenden Befehle abgearbeitet. Anschließend wird das Ersetzungsunterprogramm aufgerufen.

10.18.3.2 Ersetzung von M40 - M45 (Getriebestufenwechsel)

Funktion

Die Befehle zum Getriebestufenwechsel (M40, M41 ... M45) einer Leitspindel werden bei aktiver Kopplung durch den Aufruf eines anwenderspezifischen Unterprogramms ersetzt.

Parametrierung

Aktivierung

- MD30465 \$MA_AXIS_LANG_SUB_MASK, Bit 0 = 1

Aufrufzeitpunkt des Unterprogramms

- M40
Der Aufrufzeitpunkt kann nicht eingestellt werden. Das Ersetzungsunterprogramm wird immer am Satzanfang aufgerufen.
- M41 ... M45
Der Aufrufzeitpunkt ist abhängig vom projektierten Ausgabeverhalten der Hilfsfunktion an die PLC (siehe unten MD22080):
 - Ausgabe **vor** oder **während** der Bewegung: Unterprogrammaufruf am **Satzanfang**.
 - Ausgabe **nach** der Bewegung: Unterprogrammaufruf am **Satzende**

MD22080 \$MC_AUXFU_PREDEF_SPEC[12 ... 16] (Ausgabeverhalten für M41 ... M45)

Bit	Wert	Bedeutung
5	1	Ausgabe der Hilfsfunktion vor der Bewegung
6	1	Ausgabe der Hilfsfunktion während der Bewegung
7	1	Ausgabe der Hilfsfunktion nach der Bewegung

Systemvariable zu Informationsweitergabe

Dem Ersetzungsunterprogramm werden alle relevanten Informationen zu den im Satz programmierten Funktionen über Systemvariable (siehe Kapitel "Systemvariable (Seite 680)") zur Verfügung gestellt. Die Daten beziehen sich ausschließlich auf den Satz, in dem die zu ersetzende Funktion programmiert wurde.

10.18.3.3 Ersetzung von SPOS, SPOSA, M19 (Spindelpositionieren)

Funktion

Die Positionierbefehle (SPOS, SPOSA oder M19) einer Leitspindel werden bei aktiver Kopplung durch den Aufruf eines anwenderspezifischen Unterprogramms (Ersetzungsunterprogramm) ersetzt.

Anwendungsbeispiel

Bei der Parallelbearbeitung von Werkstücken an einer Doppelspindel-Maschine sind die Spindeln über einen Kopplungsfaktor ungleich 1 gekoppelt. Zum Werkzeugwechsel müssen sie auf dieselbe Position positioniert werden. Das Ersetzungsunterprogramm schaltet dazu die Kopplung aus, positioniert die Spindeln separat auf die Werkzeugwechselposition und schaltet anschließend die Kopplung wieder ein.

Parametrierung**Aktivierung**

- MD30465 \$MA_AXIS_LANG_SUB_MASK, Bit 1 = 1

Aufrufzeitpunkt des Ersetzungsunterprogramms

- SPOS, SPOSA
Der Aufrufzeitpunkt kann nicht eingestellt werden. Das Ersetzungsunterprogramm wird immer am Satzanfang aufgerufen.
- M19
Der Aufrufzeitpunkt ist abhängig vom projektierten Ausgabeverhalten der Hilfsfunktion an die PLC (siehe unten MD22080):
 - Ausgabe **vor** oder **während** der Bewegung: Unterprogrammaufruf am **Satzanfang**.
 - Ausgabe **nach** der Bewegung: Unterprogrammaufruf am **Satzende**

MD22080 \$MC_AUXFU_PREDEF_SPEC[9]		
Bit	Wert	Bedeutung
5	1	Ausgabe der Hilfsfunktion vor der Bewegung
6	1	Ausgabe der Hilfsfunktion während der Bewegung
7	1	Ausgabe der Hilfsfunktion nach der Bewegung

Systemvariable zur Informationsweitergabe

Dem Ersetzungsunterprogramm werden alle relevanten Informationen zu den im Satz programmierten Funktionen über Systemvariable (siehe Kapitel "Systemvariable (Seite 680)") zur Verfügung gestellt. Die Daten beziehen sich ausschließlich auf den Satz, in dem die zu ersetzende Funktion programmiert wurde.

10.18.3.4 Systemvariable

Systemvariable	Bedeutung
\$P_SUB_AXFCT	TRUE, wenn M40, M41 ... M45 Ersetzung aktiv ist
\$P_SUB_GEAR	Programmierte bzw. errechnete Getriebestufe Außerhalb des Ersetzungsunterprogramms: Getriebestufe der Masterspindel

Systemvariable	Bedeutung	
\$P_SUB_AUTOGEAR	TRUE, wenn im Satz der den Ersetzungsvorgang ausgelöst hat, M40 aktiv war. Außerhalb des Ersetzungsunterprogramms: aktuelle Einstellung im Interpreter	
\$P_SUB_LA	Enthält den Achsnamen der Leitspindel der aktiven Kopplung, die den Ersetzungsvorgang ausgelöst hat. Hinweis Wird die Variable außerhalb des Ersetzungsunterprogramms verwendet, wird die Programmbearbeitung mit Alarm abgebrochen.	
\$P_SUB_CA	Enthält den Achsnamen der Folgespindel der aktiven Kopplung, die den Ersetzungsvorgang ausgelöst hat. Hinweis Wird die Variable außerhalb des Ersetzungsunterprogramms aufgerufen, wird die Programmbearbeitung mit Alarm abgebrochen.	
\$P_SUB_AXFCT	Enthält die aktiven Ersetzungsarten entsprechend MD30465 \$MA_AXIS_LANG_SUB_MASK	
\$P_SUB_SPOS	TRUE, wenn die SPOS-Ersetzung aktiv ist	
\$P_SUB_SPOSA	TRUE, wenn die SPOSA-Ersetzung aktiv ist	
\$P_SUB_M19	TRUE, wenn die M19-Ersetzung aktiv ist	
\$P_SUB_SPOSIT	Enthält die programmierte Spindelposition Hinweis Wird die Variable außerhalb des Ersetzungsunterprogramms aufgerufen, wird die Programmbearbeitung mit Alarm abgebrochen.	
\$P_SUB_SPOSMODE	Enthält den Positionsanfahrmodus für die prog. Spindelposition:	
	Wert	Bedeutung
	0	keine Änderung des Positionsanfahrmodus
	1	AC
	2	IC
	3	DC
	4	ACP
	5	ACN
	6	OC
	7	PC
Hinweis: Wird die Variable außerhalb des Ersetzungsunterprogramms aufgerufen, wird die Programmbearbeitung mit Alarm abgebrochen.		
\$P_SUB_STAT	Satzbezogener Aufrufzeitpunkt des Ersetzungsunterprogramms	

10.18.3.5 Beispiel: Getriebestufenwechsel

Im Unterprogramm werden alle Befehle zum Getriebestufenwechsel M40, M41 ... M45 ersetzt.

Parametrierung

Maschinendatum	Bedeutung
MD15700 \$MN_LANG_SUB_NAME = "LANG_SUB"	Unterprogramm

MD15702 \$MN_LANG_SUB_PATH = 0	Herstellerverzeichnis
MD22080 \$MC_AUXFU_PREDEF_SPEC[12] = 'H21'	M41: Ausgabe vor der Bewegung
MD22080 \$MC_AUXFU_PREDEF_SPEC[13] = 'H21'	M42: Ausgabe vor der Bewegung
MD22080 \$MC_AUXFU_PREDEF_SPEC[13] = 'H21'	M43: Ausgabe vor der Bewegung
MD22080 \$MC_AUXFU_PREDEF_SPEC[15] = 'H21'	M44: Ausgabe vor der Bewegung
MD22080 \$MC_AUXFU_PREDEF_SPEC[16] = 'H21'	M45: Ausgabe vor der Bewegung
MD30465 \$MA_AXIS_LANG_SUB_MASK[AX5] = 'H0001'	Getriebewechselbefehle ersetzen

Hauptprogramm

Programmierung	Kommentar
PROC MAIN	
N110 COUPON(S2,S1)	; Synchronspindelkopplung einschalten
N120 G01 F100 X100 S5000 M3 M43	; Unterprog.Aufruf aufgrund M43
N130 M40	; Automatischer Getriebestufenwechsel einschalten
N140 M3 S1000	; Unterprog.Aufruf aufgrund S1000 ; und dadurch ausgelöstem autom. ; Getriebestufenwechsel
N9999 M30	

Ersetzungsunterprogramm "LANG_SUB", Variante 1

Optimiert auf Einfachheit und Geschwindigkeit durch direkte Adressierung der Spindeln (S1: Leitspindel, S2: Folgespindel).

Programmierung	Kommentar
N1000 PROC LANG_SUB DISPLOF SBLOF	
N1100 IF(\$P_SUB_AXFCT ==1)	; Ersetzung wegen Getriebestufenwechsel
N1140 DELAYFSTON	; Beginn Stopp-Delay-Bereich
N1150 COUPOF(S2,S1)	; Synchronspindelkopplung ausschalten
N1160	;Getriebestufenwechsel getrennt für Leit- und Folgespindel
N1170 M1=\$P_SUB_GEAR M2=\$P_SUB_GEAR	
N1180 DELAYFSTON	; Ende Stopp-Delay-Bereich
N1190 COUPON(S2,S1)	; Synchronspindelkopplung einschalten
N1200 ENDIF	
...	
N9999 RET	

Ersetzungsunterprogramm "LANG_SUB", Variante 2

Flexibilität durch indirekte Adressierung über Systemvariable (Leitspindel: \$P_SUB_LA, Folgespindel: \$P_SUB_CA).

Programmierung	Kommentar
N1000 PROC LANG_SUB DISPLOF SBLOF	
N1010 DEF AXIS _LA	; Merker für Leitachse/-spindel
N1020 DEF AXIS _CA	; Merker für Folgeachse/-spindel
N1030 DEF INT _GEAR	; Merker für Getriebestufe
N1100 IF(\$P_SUB_AXFCT==1)	; Ersetzung wegen Getriebestufenwechsel
N1110 _GEAR=\$P_SUB_GEAR	; zu aktivierende Getriebestufe
N1120 _LA=\$P_SUB_LA	; Achsname der Leitspindel
N1130 _CA=\$P_SUB_CA	; Achsname der Folgespindel
N1140 DELAYFSTON	; Beginn Stopp-Delay-Bereich
N1150 COUPOF(_CA,_LA)	; Synchronspindelkopplung ausschalten
N1160 ;Getriebestufenwechsel für Leit- und Folgespindel	
N1170 M[AXTOSPI(_LA)]=_GEAR M[AXTOSPI(_CA)]=_GEAR	
N1180 DELAYFSTOF	; Ende Stopp-Delay-Bereich
N1190 COUPON(_CA,_LA)	; Synchronspindelkopplung einschalten
N1200 ENDIF	
...	
N9999 RET	

10.18.3.6 Beispiel: Spindelpositionieren

Im Unterprogramm ist nur die Ersetzung der Befehle SPOS und SPOSA explizit ausgeführt. Weitere Ersetzungen sind sinngemäß zu ergänzen.

Parametrierung

Maschinendatum

MD30465 \$MA_AXIS_LANG_SUB_MASK[AX5] = 'H0002'

MD22080 \$MC_AUXFU_PREDEF_SPEC[9] = 'H0021'

Bedeutung

Positionierbefehle ersetzen

Ausgabe von M19 an die PLC vor der Bewegung

Settingdatum

SD43240 \$SA_M19_SPOS[AX5] = 260

SD43250 \$SA_M19_SPOSMODE[AX5] = 4

Bedeutung

Spindelposition bei M19 = 260

Positionsanfahrmodus bei M19:
"Anfahren in positiver Richtung (ACP)"

Hauptprogramm

Programmierung	Kommentar
PROC MAIN	
...	
N210 COUPON(S2,S1)	; Synchronspindelkopplung aktivieren
N220 SPOS[1]=100	; Leitspindel mit SPOS positionieren
...	
N310 G01 F1000 X100 M19	; Leitspindel mit M19 positionieren

Ersetzungsunterprogramm "LANG_SUB", Variante 1

Optimiert auf Einfachheit und Geschwindigkeit durch direkte Adressierung der Spindeln (S1: Leitspindel, S2: Folgespindel).

Programmierung	Kommentar
N1000 PROC LANG_SUB DISPLOF SBLOF	
N2100 IF(\$P_SUB_AXFCT==2)	
N2110 ;Ersetzung von SPOS/SPOSA/M19 bei aktiver Synchronspindelkopplung	
N2185 DELAYFSTON	; Beginn Stopp-Delay-Bereich
N2190 COUPOF(S2,S1)	; Synchronspindelkopplung ausschalten
N2200	; Leit- und Folgespindel positionieren
N2210 IF(\$P_SUB_SPOS==TRUE) OR (\$P_SUB_SPOSA==TRUE)	
N2220 ;SPOS und SPOSA werden auf SPOS abgebildet	
N2230 CASE \$P_SUB_SPOSMODE OF \	
0 GOTOF LABEL1_DC \	
1 GOTOF LABEL1_IC \	
2 GOTOF LABEL1_AC \	
3 GOTOF LABEL1_DC \	
4 GOTOF LABEL1_ACP \	
5 GOTOF LABEL1_ACN \	
DEFAULT GOTOF LABEL_ERR	
LABEL1_DC: SPOS[1]=DC(\$P_SUB_SPOSIT) SPOS[2]=DC(\$P_SUB_SPOSIT)	
GOTOF LABEL1_CONT	
LABEL1_IC: DELAYFSTOF	
SPOS[1]=IC(\$P_SUB_SPOSIT) SPOS[2]=IC(\$P_SUB_SPOSIT)	
DELAYFSTON	
GOTOF LABEL1_CONT	
LABEL1_AC: SPOS[1]=AC(\$P_SUB_SPOSIT) SPOS[2]=AC(\$P_SUB_SPOSIT)	
GOTOF LABEL1_CONT	
LABEL1_ACP: SPOS[1]=ACP(\$P_SUB_SPOSIT) SPOS[2]=ACP(\$P_SUB_SPOSIT)	
GOTOF LABEL1_CONT	
LABEL1_ACN: SPOS[1]=ACN(\$P_SUB_SPOSIT) SPOS[2]=ACN(\$P_SUB_SPOSIT)	
LABEL1_CONT:	
N2250 ELSE	; Spindel mit M19 positionieren
N2270 M1=19 M2=19	; Leit- und Folgespindel

Programmierung	Kommentar
N2280 ENDIF	; Ende Ersetzung SPOS, SPOSA
N2285 DELAYFSTOF	; Ende Stopp-Delay-Bereich
N2290 COUPON(S2,S1)	; Synchronspindelkopplung aktivieren
N2410 ELSE	
N2420 ;ab hier Bearbeitung weiterer Ersetzungen	
...	
N3300 ENDIF	; Ende Ersetzungen
...	
N9999 RET	; Normales Programmende
LABEL_ERR: SETAL(61000)	; Fehler aufgetreten

Ersetzungsunterprogramm "LANG_SUB", Variante 2

Flexibilität durch indirekte Adressierung über Systemvariable (Leitspindel: \$P_SUB_LA, Folgespindel: \$P_SUB_CA).

Programmierung	Kommentar
N1000 PROC LANG_SUB DISPLOF SBLOF	
N1010 DEF AXIS _LA	; Leitachse/-spindel
N1020 DEF AXIS _CA	; Folgeachse/-spindel
N1030 DEF INT _LSPI	; Leitspindelnummer (programmierte ; Spindel)
N1040 DEF INT _CSPI	; Folgespindelnummer
...	
N2100 IF(\$P_SUB_AXFCT==2)	
N2110 ;Ersetzung von SPOS/SPOSA/M19 bei aktiver ; Synchronspindelkopplung	
N2120 _LA=\$P_SUB_LA	; Achsname der Leitspindel
N2130 _CA=\$P_SUB_CA	; Achsname der Folgespindel
N2140 _LSPI=AXTOSPI(_LA)	; Nummer der Leitspindel
N2180 _CSPI=AXTOSPI(_LA)	; Nummer der Folgepindel
N2185 DELAYFSTON	; Beginn Stopp-Delay-Bereich
N2190 COUPOF(_CA,_LA)	; Synchronspindelkopplung deaktivieren
N2200	; Leit- und Folgespindel positionieren:
N2210 IF(\$P_SUB_SPOS==TRUE) OR (\$P_SUB_SPOSA==TRUE)	
N2220 ;SPOS und SPOSA werden auf SPOS abgebildet	
N2230 CASE \$P_SUB_SPOSMODE OF	
0 GOTOF LABEL1_DC \	
1 GOTOF LABEL1_IC \	
2 GOTOF LABEL1_AC \	
3 GOTOF LABEL1_DC \	
4 GOTOF LABEL1_ACP \	
5 GOTOF LABEL1_ACN \	
DEFAULT GOTOF LABEL_ERR	
LABEL1_DC: SPOS[_LSPI]=DC(\$P_SUB_SPOSIT) SPOS[_CSPI]=DC(\$P_SUB_SPOSIT)	
GOTOF LABEL1_CONT	

Programmierung	Kommentar
<pre> LABEL1_IC: DELAYFSTOF SPOS[_LSPI]=IC(\$P_SUB_SPOSIT) SPOS[_CSPI]=IC(\$P_SUB_SPOSIT) DELAYFSTON GOTOF LABEL1_CONT LABEL1_AC: SPOS[_LSPI]=AC(\$P_SUB_SPOSIT) SPOS[_CSPI]=AC(\$P_SUB_SPOSIT) GOTOF LABEL1_CONT LABEL1_ACP: SPOS[_LSPI]=ACP(\$P_SUB_SPOSIT) POS[_CSPI]=ACP(\$P_SUB_SPOSIT) GOTOF LABEL1_CONT LABEL1_ACN: SPOS[_LSPI]=ACN(\$P_SUB_SPOSIT) POS[_CSPI]=ACN(\$P_SUB_SPOSIT) LABEL1_CONT: N2250 ELSE ; Spindeln mit M19 positionieren N2270 M[_LSPI]=19 M[_CSPI]=19 N2280 ENDIF N2285 DELAYFSTOF ; Ende Stopp-Delay-Bereich N2290 COUPON(_CA,_LA) ; Synchronspindelkopplung aktivieren N2410 ELSE N2420 ;ab hier Bearbeitung weiterer Ersetzungen ... N3300 ENDIF ... N9999 RET ; Normales Programmende LABEL_ERR: SETAL(61000) ; Fehler aufgetreten </pre>	

10.18.4 Eigenschaften der Unterprogramme

Allgemeine Regeln

- Das bei der Ersetzung aufgerufene Unterprogramm kann den Befehl `PROC` und die Attribute `SBLOF` und `DISPLOF` enthalten.
- Die Ersetzung wird auch im ISO-Sprachmode ausgeführt. Die Ersetzungsunterprogramme werden aber ausschließlich im Standard-Sprachmode (Siemens) abgearbeitet. Dabei erfolgt eine implizite Umschaltung in den Standard-Sprachmode. Mit Rücksprung aus dem Ersetzungsunterprogramm wird wieder in den ursprünglichen Sprachmode zurückgeschaltet.
- Die Weitergabe von Informationen an das Ersetzungsunterprogramm erfolgt ausschließlich über Systemvariablen. Übergabeparameter sind nicht möglich.
- Das Verhalten bei Einzelsatz und Attribut `SBLOF` ist abhängig von der Einstellung in: MD10702 IGNORE_SINGLEBLOCK_MASK, Bit 14 (Einzelsatz-Stopp verhindern)

Wert	Bedeutung
0	<p>Das Ersetzungsunterprogramm verhält sich wie ein "normales" Unterprogramm:</p> <ul style="list-style-type: none"> Rücksprung mit M17: Stopp am Ende des Unterprogramms <p>Hinweis Die Ausgabe der M-Funktion an die PLC ist abhängig von: MD20800 \$MC_SPF_END_TO_VDI, Bit 0 (Unterprogrammende an PLC)</p> <ul style="list-style-type: none"> Bit 0 = 0: keine Ausgabe Bit 0 = 1: M17 wird an die PLC ausgegeben. <ul style="list-style-type: none"> Rücksprung mit RET: Kein Stopp am Ende des Ersetzungsunterprogramms
1	<p>Im Satz, in dem das Ersetzungsunterprogramm aufgerufen wird, wird nur einmal angehalten. Unabhängig davon ob:</p> <ul style="list-style-type: none"> Das Unterprogramm am Satzanfang und/oder am Satzende aufgerufen wird Im Unterprogramm weitere Unterprogramme aufgerufen werden Das Unterprogramm mit M17 oder RET verlassen wird <p>Der Einzelsatz-Stopp erfolgt für die Ersetzung von M-Funktionen am Ende des Ersetzungsunterprogramms.</p> <p>Für die Ersetzung von T- und D/DL-Funktionen ist der Zeitpunkt des Einzelsatz-Stopps abhängig vom Aufrufzeitpunkt des Unterprogramms:</p> <ul style="list-style-type: none"> Aufruf am Satzanfang: Einzelsatz-Stopp am Ende des Satzes Aufruf am Satzende: Einzelsatz-Stopp am Ende des Ersetzungsunterprogramms

- Bei Ersetzungsunterprogrammen mit dem Attribut `DISPLOC` wird in der Satzanzeige als aktueller Satz die Programmzeile angezeigt, die zum Aufruf des Unterprogramms geführt hat.
- Im Ersetzungsunterprogramm können mit den Befehlen `DELAYFSTON` und `DELAYFSTOF` Bereiche oder das gesamte Ersetzungsunterprogramm gegen Unterbrechungen, wie NC-Stop, Einlesesperre usw., geschützt werden.
- Ersetzungen erfolgen nicht rekursiv. D. h. die Funktion, die zum Aufruf des Ersetzungsunterprogramms geführt hat, wird bei nochmaliger Programmierung im Ersetzungsunterprogramm nicht mehr ersetzt.

Hilfsfunktionsausgabe an PLC

Bei Ersetzungen von Hilfsfunktionen bewirkt der Aufruf des Ersetzungsunterprogramms noch keine Ausgabe der Hilfsfunktion an die PLC. Die Ausgabe der Hilfsfunktion erfolgt erst wenn die Hilfsfunktion im Ersetzungsunterprogramm erneut programmiert wird.

Verhalten bei Satzsuchlauf

Das Ersetzungsunterprogramm wird auch in den Satzsuchlaufmodi "Satzsuchlauf mit Berechnung" und "Satzsuchlauf mit Berechnung im Modus Programmtest" (SERUPRO) aufgerufen. Eventuelle Besonderheiten müssen im Ersetzungsunterprogramm unter Verwendung der Systemvariable: `$P_SEARCH`, `$AC_SERUPRO` realisiert werden.

Bezüglich des Aufsammelns von Aktionen beim "Satzsuchlauf mit Berechnung" verhalten sich Ersetzungsunterprogramme wie "normale" Unterprogramme.

10.18.5 Randbedingungen

- Funktionersetzungen sind unzulässig in:
 - Synchronaktionen
 - Technologiezyklen
- Einem Satz, der am Anfang zu ersetzende Funktionen enthält, dürfen keine satzweisen Synchronaktionen vorangestellt werden. Siehe unten Absatz "Beispiel zu: Satzweise Synchronaktionen".
- Im Ersetzungsunterprogramm dürfen nur die für die jeweiligen Ersetzungen notwendigen Aktionen durchgeführt werden.
- In einem Satz, in dem das Ersetzungsunterprogramm am Satzende aufgerufen wird, ist folgendes zu beachten:
 - Es darf kein modaler Unterprogrammaufruf aktiv sein
 - Es darf kein Unterprogrammrücksprung programmiert sein
 - Es darf kein Programmende programmiert sein

Hinweis

Die Steuerung überwacht nicht, ob die zu ersetzende Funktion im Ersetzungsunterprogramm realisiert wurde.

Beispiel zu: Satzweise Synchronaktionen

MD30465 \$MA_AXIS_LANG_SUB_MASK, Bit 0 = 1 (Getriebestufenwechsel)

Programmcode

```
...  
N1000 WHENEVER $AA_IM[X2] <= $AA_IM[X1] + 0.5 DO $AA_OVR[X1]=0  
N1010 G1 X100 M43  
...
```

Führt in Satz N1010 die Funktion M43 zum Aufruf eines Ersetzungsunterprogramms, wird die Bearbeitung mit abgebrochen und ein Alarm angezeigt.

10.19 Umbenennung/Sperrung von NC-Befehlen

Funktion

Mit der Funktion "Umbenennung/Sperrung von NC-Befehlen" kann der Maschinenhersteller oder Endanwender den Namen bestehender NC-Befehle ändern.

Anwendung

Die Funktion kann für folgende Zwecke genutzt werden:

- Verbesserung der Lesbarkeit von Teileprogrammen
- Sperrung von NC-Befehlen
- Anwenderspezifische Erweiterung von NC-Funktionen

Parametrierung

Die Umbenennung/Sperrung von NC-Befehlen erfolgt über das Maschinendatum:

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[<n>] (Liste umprojektierter NC-Codes)

Ein gerader Index [<n>] enthält den ursprünglichen Namen eines Befehls. Der darauf folgende ungerade Index enthält den neuen Namen des Befehls.

Hinweis

Bei SINUMERIK 828D sind die Einträge in den geraden Indizes (ursprüngliche Befehlsnamen) fest vorgegeben und können nicht verändert werden. Nur die ungeraden Indizes (neue Befehlsnamen) können geschrieben werden.

Ein Leerstring ("") als neuer Name bedeutet, dass es keinen neuen Namen für den Befehl gibt. Der Befehl ist dadurch gesperrt und kann nicht mehr programmiert werden.

Änderungen von MD10712 werden mit Power On wirksam.

Beispiele

Beispiel 1 (840D sl): Befehl G00 umbenennen und G01 sperren

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[0] = "G00"

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[1] = "EILGANG"

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[2] = "G01"

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[3] = ""

Nach Wirksamsetzen durch Power On kann Eilgang nicht mehr mit dem Befehl G00, sondern nur noch mit dem Befehl EILGANG programmiert werden. G01 kann nicht mehr programmiert werden.

Beispiel 2 (828D): Vorbelegung bei SINUMERIK 828D ändern

Zur Anpassung an den ISO-Dialekt-Modus sind bei der SINUMERIK 828D die Indizes [0] bis [7] so vorbelegt, dass die Befehle G505 und G506 in G58 und G59 umbenannt sind.

Ursprünglich sind G58 und G59 programmierbare Frames, G505 und G506 sind einstellbare Frames. Durch die Umbenennung werden G58 und G59 zu einstellbaren Frames.

Sollte diese Vorbelegung nicht gewünscht sein und sollen G58 und G59 weiterhin programmierbare Frames bleiben, muss die Umbenennung folgendermaßen rückgängig gemacht werden:

Index von MD10712	Befehlsname	
	vordefiniert	neu
[0]	"G58"	"G58"
[1]	""	"G58"
[2]	"G59"	"G59"
[3]	""	"G59"
[4]	"G505"	"G505"
[5]	"G58"	"G505"
[6]	"G506"	"G506"
[7]	"G59"	"G506"

Beispiel 3 (828D): Befehl _TCA in TCA umbenennen

Der Befehl `TCA` ist bei SINUMERIK 828D standardmäßig in `_TCA` umbenannt. Für die TCA-Funktion gibt es einen Zyklus. Falls dieser Zyklus nicht benutzt werden soll, muss der Befehl `_TCA` wieder in `TCA` umbenannt werden:

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[8] = "_TCA"

MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[9] = "TCA"

Beispiel 4 (828D): Bestehenden Befehl in einen anwenderspezifischen Zyklus umlenken

Der bestehende Befehl `WAITM` soll durch einen Zyklus ersetzt werden, in dem anwenderspezifische Funktionen realisiert sind.

- Bestehenden Befehl `WAITM` mit MD10712 umbenennen in `_WAITM`:
MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[20] = "WAITM"
MD10712 \$MN_NC_USER_CODE_CONF_NAME_TAB[21] = "_WAITM"
- Teileprogramm mit dem Namen `WAITM` anlegen.
Das Interface von `PROC WAITM` muss identisch sein zu der NC-Funktion `WAITM`. Dieses Teileprogramm ersetzt jetzt die bisherige `WAITM`-NC-Funktion. Jeder `WAITM`-Befehl aus einem Teileprogramm, auch aus den Siemens-Zyklen und Shopmill/Shopturn-Zyklen, ruft das anwenderspezifische Unterprogramm `WAITM` auf. In diesem Unterprogramm `WAITM` werden die anwenderspezifischen Funktionen durchlaufen und, wenn erforderlich, die umbenannte NC-Funktion `_WAITM` aufgerufen.

Programmcode	Kommentar
N5 PROC WAITM(<Marken-Nr.>,<Kanal-Nr.>)	
...	; Anwenderspezifische Funktionen
N100 _WAITM(<Marken-Nr.>, <Kanal-Nr.>)	
...	; Anwenderspezifische Funktionen
M30	

10.20 Programmlaufzeit / Werkstückzähler

Zur Unterstützung des Werkzeugmaschinenbedieners werden Informationen zur Programmlaufzeit und Werkstückzahl bereitgestellt.

Diese Informationen können als Systemvariablen im NC- und/oder PLC-Programm bearbeitet werden. Gleichzeitig stehen diese Informationen für die Anzeige auf der Bedienoberfläche zur Verfügung.

10.20.1 Programmlaufzeit

10.20.1.1 Funktion

Die Funktion "Programmlaufzeit" stellt zur Überwachung technologischer Prozesse verschiedene Timer zur Verfügung, die über Systemvariablen im Teileprogramm und Synchronaktionen gelesen werden können. Es gibt zwei verschiedene Arten von Timern:

1. Standard-Timer
Standard-Timer sind immer aktiv.
2. Aktivierbare Timer
Aktivierbare Timer müssen über Maschinendaten aktiviert werden.

Standard-Timer

Zeit seit dem letzten Steuerungshochlauf

Systemvariable	Bedeutung
\$AN_SETUP_TIME	Zeit seit dem letzten Steuerungshochlauf mit Standardwerten in Minuten Wird bei jedem Steuerungshochlauf mit Standardwerten automatisch auf "0" zurückgesetzt.
\$AN_POWERON_TIME	Zeit seit dem letzten Normalhochlauf der Steuerung ("Warmstart") in Minuten Wird bei jedem Normalhochlauf der Steuerung automatisch auf "0" zurückgesetzt.

Programmlaufzeit

Die Timer zur Messung der Programmlaufzeiten sind nur in der Betriebsart AUTOMATIK verfügbar.

Systemvariable (kanalspezifisch)	Beschreibung
\$AC_ACT_PROG_NET_TIME	<p>Aktuelle Netto-Laufzeit des aktuellen Programms in Sekunden</p> <p>Netto-Laufzeit bedeutet, dass die Zeit, in der das Programm gestoppt war, abgezogen ist.</p> <p>Wird in der Betriebsart AUTOMATIK aus dem Kanalzustand RESET heraus ein Teileprogramm neu gestartet, wird \$AC_ACT_PROG_NET_TIME automatisch auf "0" zurückgesetzt.</p> <p>Weitere Eigenschaften:</p> <ul style="list-style-type: none"> Die Reset-Taste setzt \$AC_ACT_PROG_NET_TIME nicht auf "0" zurück, sondern hält den Timer nur an. Beim Start eines ASUPs wird \$AC_ACT_PROG_NET_TIME auf "0" gesetzt und zählt die Laufzeit des ASUPs mit. Am Ende eines ASUPs verhält es sich wie bei der RESET-Taste: der Timer wird nur angehalten, aber nicht auf "0" gesetzt. Beim Start eines ereignisgesteuerten Programms (PROG_EVENT) wird \$AC_ACT_PROG_NET_TIME nicht zurückgesetzt. Die Programmlaufzeit wird nur weitergezählt, wenn es sich um ein Start-, M30- oder Suchlauf-PROG_EVENT handelt. Das Verhalten von \$AC_ACT_PROG_NET_TIME bei GOTOS und Override = 0% ist über MD27850 parametrierbar (siehe Abschnitt "Parametrierung") <p>Hinweis: Mit \$AC_PROG_NET_TIME_TRIGGER kann \$AC_ACT_PROG_NET_TIME weiter manipuliert werden.</p>
\$AC_OLD_PROG_NET_TIME	<p>Netto-Laufzeit des gerade korrekt beendeten Programms in Sekunden</p> <p>"Korrekt beendet" bedeutet, dass das Programm nicht mit RESET abgebrochen, sondern regulär mit M30 beendet wurde.</p> <p>Wird ein neues Programm gestartet, bleibt \$AC_OLD_PROG_NET_TIME unangetastet, bis M30 erneut erreicht wird.</p> <p>Weitere Eigenschaften:</p> <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME wird auf "0" gesetzt, wenn das aktuell angewählte Programm editiert wird. Am Ende eines ASUPs oder eines ereignisgesteuerten Programms (PROG_EVENT) wird \$AC_OLD_PROG_NET_TIME nicht verändert. <p>Hinweis: Der implizite Kopiervorgang von \$AC_ACT_PROG_NET_TIME nach \$AC_OLD_PROG_NET_TIME findet nur statt, wenn \$AC_PROG_NET_TIME_TRIGGER nicht beschrieben wird.</p>

Systemvariable (kanalspezifisch)	Beschreibung												
\$AC_OLD_PROG_NET_TIME_COUNT	<p>Änderungen auf \$AC_OLD_PROG_NET_TIME</p> <p>Nach POWER ON steht \$AC_OLD_PROG_NET_TIME_COUNT auf "0".</p> <p>\$AC_OLD_PROG_NET_TIME_COUNT wird immer dann erhöht, wenn die Steuerung \$AC_OLD_PROG_NET_TIME neu geschrieben hat.</p> <p>Wenn der Anwender das laufende Programm mit RESET abbricht, bleiben \$AC_OLD_PROG_NET_TIME und \$AC_OLD_PROG_NET_TIME_COUNT unverändert.</p> <p>Mit \$AC_OLD_PROG_NET_TIME_COUNT lässt sich daher feststellen, ob \$AC_OLD_PROG_NET_TIME geschrieben wurde.</p> <p>Beispiel:</p> <p>Wenn zwei hintereinander laufende Programme die identische Laufzeit haben und korrekt beendet wurden, dann kann der Anwender dies über den veränderten Wert in \$AC_OLD_PROG_NET_TIME_COUNT erkennen.</p>												
\$AC_PROG_NET_TIME_TRIGGER	<p>Trigger zur selektiven Messung von Programmabschnitten.</p> <p>Durch das Schreiben des Wertes in die Variable wird die Funktion ausgelöst:</p> <table border="1"> <thead> <tr> <th>Wert</th><th>Funktion</th></tr> </thead> <tbody> <tr> <td>0</td><td>Inaktiv</td></tr> <tr> <td>1</td><td> Beendet die Messung <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME = \$AC_ACT_PROG_NET_TIME \$AC_ACT_PROG_NET_TIME = 0 und läuft anschließend weiter </td></tr> <tr> <td>2</td><td> Startet die Messung <ul style="list-style-type: none"> \$AC_ACT_PROG_NET_TIME = 0 \$AC_OLD_PROG_NET_TIME wird nicht verändert </td></tr> <tr> <td>3</td><td> Stoppt die Messung <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME wird nicht verändert \$AC_ACT_PROG_NET_TIME wird nicht verändert </td></tr> <tr> <td>4</td><td> Fortsetzten der gestoppten Messung <ul style="list-style-type: none"> \$AC_ACT_PROG_NET_TIME läuft weiter \$AC_OLD_PROG_NET_TIME wird nicht verändert </td></tr> </tbody> </table>	Wert	Funktion	0	Inaktiv	1	Beendet die Messung <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME = \$AC_ACT_PROG_NET_TIME \$AC_ACT_PROG_NET_TIME = 0 und läuft anschließend weiter 	2	Startet die Messung <ul style="list-style-type: none"> \$AC_ACT_PROG_NET_TIME = 0 \$AC_OLD_PROG_NET_TIME wird nicht verändert 	3	Stoppt die Messung <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME wird nicht verändert \$AC_ACT_PROG_NET_TIME wird nicht verändert 	4	Fortsetzten der gestoppten Messung <ul style="list-style-type: none"> \$AC_ACT_PROG_NET_TIME läuft weiter \$AC_OLD_PROG_NET_TIME wird nicht verändert
Wert	Funktion												
0	Inaktiv												
1	Beendet die Messung <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME = \$AC_ACT_PROG_NET_TIME \$AC_ACT_PROG_NET_TIME = 0 und läuft anschließend weiter 												
2	Startet die Messung <ul style="list-style-type: none"> \$AC_ACT_PROG_NET_TIME = 0 \$AC_OLD_PROG_NET_TIME wird nicht verändert 												
3	Stoppt die Messung <ul style="list-style-type: none"> \$AC_OLD_PROG_NET_TIME wird nicht verändert \$AC_ACT_PROG_NET_TIME wird nicht verändert 												
4	Fortsetzten der gestoppten Messung <ul style="list-style-type: none"> \$AC_ACT_PROG_NET_TIME läuft weiter \$AC_OLD_PROG_NET_TIME wird nicht verändert 												
Durch POWER ON werden alle Systemvariablen auf "0" zurückgesetzt!													

Hinweis**Restzeit für ein Werkstück**

Wenn nacheinander gleiche Werkstücke produziert werden, kann mit den folgenden Timerwerten die verbleibende Restzeit für ein Werkstück ermittelt werden:

- Bearbeitungszeit für das zuletzt produzierte Werkstück (siehe \$AC_OLD_PROG_NET_TIME)
- Aktuelle Bearbeitungszeit (siehe \$AC_ACT_PROG_NET_TIME)

Die Restzeit wird zusätzlich zur aktuellen Bearbeitungszeit auf der Bedienoberfläche angezeigt.

Hinweis**Verwendung von STOPRE**

Die Systemvariablen \$AC_OLD_PROG_NET_TIME und \$AC_OLD_PROG_NET_TIME_COUNT erzeugen keinen impliziten Vorlaufstopp. Bei der Verwendung im Teileprogramm ist das unkritisch, wenn der Wert der Systemvariablen aus dem vorangegangenen Programmlauf stammt. Wenn aber der Trigger zur Laufzeitmessung (\$AC_PROG_NET_TIME_TRIGGER) hochfrequent geschrieben wird und sich dadurch \$AC_OLD_PROG_NET_TIME sehr oft ändert, dann sollte im Teileprogramm ein explizites STOPRE verwendet werden.

Aktivierbare Timer**Programmlaufzeit**

Die Timer zur Messung der Programmlaufzeiten sind nur in der Betriebsart AUTOMATIK verfügbar.

Systemvariable (kanalspezifisch)	Beschreibung
\$AC_OPERATING_TIME	<p>Gesamtlaufzeit von NC-Programmen in der Betriebsart Automatik in Sekunden</p> <p>Aufsummiert werden in der Betriebsart Automatik die Laufzeiten aller Programme zwischen NC-Start und Programmende / NC-Reset.</p> <p>Bei NC-Stop und Override = 0% wird standardmäßig nicht gezählt. Das Weiterzählen bei Override 0% kann über MD27860 aktiviert werden.</p> <p>Der Wert wird mit jedem Steuerungshochlauf automatisch auf "0" zurückgesetzt.</p>
\$AC_CYCLE_TIME	<p>Laufzeit des angewählten NC-Programms in Sekunden</p> <p>Im angewählten NC-Programm wird die Laufzeit zwischen NC-Start und Programmende / NC-Reset gemessen.</p> <p>Bei NC-Stop und Override = 0% wird standardmäßig nicht gezählt. Das Weiterzählen bei Override 0% kann über MD27860 aktiviert werden.</p> <p>Der Wert wird mit dem Start eines neuen NC-Programms automatisch auf "0" zurückgesetzt. Über MD27860 ist einstellbar, ob diese Löschung auch bei einem Sprung auf den Programmanfang mit GOTOS oder bei Start von ASUPs und PROG_EVENTS erfolgen soll.</p>
\$AC_CUTTING_TIME	<p>Bearbeitungszeit in Sekunden</p> <p>Gemessen wird die Laufzeit der Bahnachsen (mindestens eine ist aktiv) ohne aktiven Eilgang in allen NC-Programmen zwischen NC-Start und Programmende / NC-Reset. Über MD27860 ist einstellbar, ob nur mit aktivem Werkzeug oder werkzeugunabhängig gemessen werden soll.</p> <p>Die Messung wird zusätzlich bei aktiver Verweilzeit unterbrochen.</p> <p>Der Wert wird bei jedem Steuerungshochlauf mit Standardwerten automatisch auf "0" zurückgesetzt.</p>

10.20.1.2 Inbetriebnahme**Aktivierung / Deaktivierung**

Das Ein-/Ausschalten der aktivierbaren Timer erfolgt über das Maschinendatum:

MD27860 \$MC_PROCESSTIMER_MODE, Bit 0 - 2 = <Wert>

Bit	Wert	Bedeutung
0	0	Timer für \$AC_OPERATING_TIME nicht aktiv.
	1	Timer für \$AC_OPERATING_TIME aktiv.
1	0	Timer für \$AC_CYCLE_TIME nicht aktiv.
	1	Timer für \$AC_CYCLE_TIME aktiv.
2	0	Timer für \$AC_CUTTING_TIME nicht aktiv.
	1	Timer für \$AC_CUTTING_TIME aktiv.

Parametrierung

Verhalten der immer aktiven Timer

Das Verhalten der immer aktiven Timer bei GOTOS und Override = 0% wird eingestellt mit dem Maschinendatum:

MD27850 \$MC_PROG_NET_TIMER_MODE

Bit	Wert	Bedeutung
0	0	\$AC_ACT_PROG_NET_TIME wird bei einem Sprung mit GOTOS auf den Programm-anfang nicht auf "0" zurückgesetzt (Grundeinstellung).
	1	\$AC_ACT_PROG_NET_TIME wird bei einem Sprung mit GOTOS auf den Programm-anfang auf "0" zurückgesetzt, der Wert vorher in \$AC_OLD_PROG_NET_TIME gespeichert und der Programnzähler \$AC_OLD_PROG_NET_TIME_COUNT erhöht.
1	0	\$AC_ACT_PROG_NET_TIME wird bei Override = 0% nicht erhöht. D. h. die Programmlaufzeit wird ohne die Zeit gemessen, für die der Override auf "0" eingestellt wurde (Grundeinstellung).
	1	\$AC_ACT_PROG_NET_TIME wird auch bei Override = 0% erhöht. D. h. die Programmlaufzeit wird mit der Zeit gemessen, für die der Override auf "0" eingestellt wurde.

Verhalten der aktivierbaren Timer

Das Verhalten der aktivierbaren Timer bei bestimmten Funktionen (z. B. Probelaufvorschub, Programmtest) wird eingestellt mit dem Maschinendatum:

MD27860 \$MC_PROCESSTIMER_MODE

Bit	Wert	Bedeutung
4	0	Keine Messung bei aktivem Probelaufvorschub.
	1	Messung auch bei aktivem Probelaufvorschub.
5	0	Keine Messung bei Programm-Test.
	1	Messung auch bei Programm-Test.
6	Nur bei Bit 1 = 1 (Timer für \$AC_CYCLE_TIME ist aktiv)	
	0	\$AC_CYCLE_TIME wird auch bei Start durch ASUP und PROG_EVENTS auf "0" zurückgesetzt.
	1	\$AC_CYCLE_TIME wird bei Start durch ASUP und PROG_EVENTS nicht auf "0" zurückgesetzt.

Bit	Wert	Bedeutung
7	Nur bei Bit 2 = 1 (Timer für \$AC_CUTTING_TIME ist aktiv)	
	0	Timer für \$AC_CUTTING_TIME zählt nur bei aktivem Werkzeug.
	1	Timer für \$AC_CUTTING_TIME zählt werkzeugunabhängig.
8	Nur bei Bit 1 = 1 (Timer für \$AC_CYCLE_TIME ist aktiv)	
	0	\$AC_CYCLE_TIME wird bei einem Sprung mit GOTOS auf den Programmanfang nicht auf "0" zurückgesetzt.
	1	\$AC_CYCLE_TIME wird bei einem Sprung mit GOTOS auf den Programmanfang auf "0" zurückgesetzt.
9	Nur bei Bit 0, 1 = 1 (Timer für \$AC_OPERATING_TIME und \$AC_CYCLE_TIME sind aktiv)	
	0	Kein Weiterzählen der Programmlaufzeit bei Override = 0%
	1	Weiterzählen der Programmlaufzeit bei Override = 0%

10.20.1.3 Randbedingungen

Satzsuchlauf

Bei Satzsuchlauf werden keine Programmlaufzeiten ermittelt.

REPOS

Die Zeitdauer eines REPOS-Vorgangs wird der aktuellen Bearbeitungszeit (\$AC_ACT_PROG_NET_TIME) angerechnet.

10.20.1.4 Beispiele

Beispiel 1: Parametrierung der Laufzeitmessung über MD27860

Aktivierung der Laufzeitmessung für das aktive NC-Programm, dabei kein Messen bei aktivem Probelaufvorschub und Programmtest:

MD27860 \$MC_PROCESSTIMER_MODE = 'H2'

Aktivierung der Messung für die Werkzeug-Eingriffszeit, dabei auch Messen bei aktivem Probelaufvorschub und Programmtest:

MD27860 \$MC_PROCESSTIMER_MODE = 'H34'

Aktivierung der Messung für die Gesamtlaufzeit und die Bearbeitungszeit mit aktivem Werkzeug, dabei auch Messen bei Programmtest:

MD27860 \$MC_PROCESSTIMER_MODE = 'H25'

Aktivierung der Messung für die Gesamtlaufzeit und die Bearbeitungszeit (werkzeugunabhängig), dabei auch Messen bei Programmtest:

MD27860 \$MC_PROCESSTIMER_MODE = 'Ha5'

Aktivierung der Messung für die Bearbeitungszeit mit aktivem Werkzeug, dabei auch Messen bei Override = 0% , aber nicht bei aktivem Probelauf-Vorschub:

MD27860 \$MC_PROCESSTIMER_MODE = 'H22'

Beispiel 2: Zeitdauer von "mySubProgrammA" messen

Programmcode

```
...
N50 DO $AC_PROG_NET_TIME_TRIGGER=2
N60 FOR ii= 0 TO 300
N70 mySubProgrammA
N80 DO $AC_PROG_NET_TIME_TRIGGER=1
N95 ENDFOR
N97 mySubProgrammB
N98 M30
```

Nachdem das Programm die Zeile N80 verarbeitet hat, steht in \$AC_OLD_PROG_NET_TIME die Nettolaufzeit von "mySubProgrammA".

Der Wert von \$AC_OLD_PROG_NET_TIME:

- Bleibt über M30 hinaus erhalten
- Wird nach jedem Schleifendurchlauf aktualisiert

Beispiel 3: Zeitdauer von "mySubProgrammA" und "mySubProgrammC" messen

Programmcode

```
N10 DO $AC_PROG_NET_TIME_TRIGGER=2
N20 mySubProgrammA
N30 DO $AC_PROG_NET_TIME_TRIGGER=3
N40 mySubProgrammB
N50 DO $AC_PROG_NET_TIME_TRIGGER=4
N60 mySubProgrammC
N70 DO $AC_PROG_NET_TIME_TRIGGER=1
N80 mySubProgrammD
N90 M30
```

10.20.2 Werkstückzähler

10.20.2.1 Funktion

Mit der Funktion "Werkstückzähler" stehen in Form von kanalspezifische Systemvariablen diverse Zähler mit einem Wertebereich von 0 bis 999.999.999 zur Verfügung. Auf die Systemvariablen kann lesend und schreibend zugegriffen werden.

Über die nachfolgend beschriebenen kanalspezifischen Maschinendaten kann auf die Zähler-Aktivierung, den Zeitpunkt der Nullung und den Zählalgorithmus Einfluss genommen werden.

Systemvariablen zur Werkstückzählung

Systemvariable	Bedeutung
\$AC_REQUIRED_PARTS	Anzahl der zu fertigenden Werkstücke (Soll-Werkstückzahl) In diesem Zähler kann die Anzahl der Werkstücke definiert werden, bei dessen Erreichen die Ist-Werkstückzahl (\$AC_ACTUAL_PARTS) auf "0" zurückgesetzt wird. Über MD27880 kann die Generierung des Anzeige-Alarms: "Kanal %1 Werkstueck-Soll = %2 erreicht" und des kanalspezifischen NC/PLC-Nahtstellensignals: DB21, ... DBX317.1 (Werkstück-Soll erreicht) aktiviert werden.
\$AC_TOTAL_PARTS	Anzahl der insgesamt gefertigten Werkstücke (Ist-Werkstückzahl Gesamt) Dieser Zähler gibt die Anzahl aller ab Startzeitpunkt gefertigten Werkstücke an. Der Wert wird nur bei einem Steuerungshochlauf mit Standardwerten automatisch auf "0" zurückgesetzt.
\$AC_ACTUAL_PARTS	Anzahl der gefertigten Werkstücke (Ist-Werkstückzahl) In diesem Zähler wird die Anzahl aller ab Startzeitpunkt gefertigten Werkstücke registriert. Bei einem Erreichen der Soll-Werkstückzahl (\$AC_REQUIRED_PARTS) wird der Zähler automatisch auf "0" zurückgesetzt (\$AC_REQUIRED_PARTS > 0 vorausgesetzt).
\$AC_SPECIAL_PARTS	Anzahl der vom Anwender gezählten Werkstücke Dieser Zähler erlaubt dem Anwender eine Werkstückzählung nach eigener Definition. Definiert werden kann eine Alarmausgabe bei Erreichen der Soll-Werkstückzahl (\$AC_REQUIRED_PARTS). Eine Nullung des Zählers muss der Anwender selbst vornehmen.

Hinweis

Alle Werkstückzähler werden bei einem Steuerungshochlauf mit Standardwerten, auf den Wert "0" gesetzt und können, unabhängig von ihrer Aktivierung, gelesen und geschrieben werden.

10.20.2.2 Inbetriebnahme

Aktivierung

Die Werkstückzähler werden aktiviert mit dem Maschinendatum:

MD27880 \$MC_PART_COUNTER (Aktivierung der Werkstückzähler)

Bit	Wert	Bedeutung
0	1	\$AC_REQUIRED_PARTS ist aktiv.
1	0	Alarm- / Signalausgabe bei: \$AC_ACTUAL_PARTS = \$AC_REQUIRED_PARTS
	1	Alarm- / Signalausgabe bei: \$AC_SPECIAL_PARTS = \$AC_REQUIRED_PARTS
4	1	\$AC_TOTAL_PARTS ist aktiv.

Bit	Wert	Bedeutung
5	0	\$AC_TOTAL_PARTS wird durch M02 / M30 um den Wert "1" erhöht.
	1	\$AC_TOTAL_PARTS wird durch den mit MD27882[0] definierten M-Befehl um den Wert "1" erhöht.
6	0	\$AC_TOTAL_PARTS ist auch bei Programm-Test / Satzsuchlauf aktiv.
7	1	\$AC_TOTAL_PARTS wird bei einem Rücksprung mit GOTOS um den Wert "1" erhöht.
8	1	\$AC_ACTUAL_PARTS ist aktiv.
9	0	\$AC_ACTUAL_PARTS wird durch M02 / M30 um den Wert "1" erhöht.
	1	\$AC_ACTUAL_PARTS wird durch den mit MD27882[1] definierten M-Befehl um den Wert "1" erhöht.
10	0	\$AC_ACTUAL_PARTS ist auch bei Programm-Test / Satzsuchlauf aktiv.
11	1	\$AC_ACTUAL_PARTS wird bei einem Rücksprung mit GOTOS um den Wert "1" erhöht.
12	1	\$AC_SPECIAL_PARTS ist aktiv.
13	0	\$AC_SPECIAL_PARTS wird durch M02 / M30 um den Wert "1" erhöht.
	1	\$AC_SPECIAL_PARTS wird durch den mit MD27882[2] definierten M-Befehl um den Wert "1" erhöht.
14	0	\$AC_SPECIAL_PARTS ist auch bei Programm-Test / Satzsuchlauf aktiv.
15	1	\$AC_SPECIAL_PARTS wird bei einem Rücksprung mit GOTOS um den Wert "1" erhöht.

Parametrierung

Werkstückzählung mit anwenderdefiniertem M-Befehl

Ist das entsprechende Bit in MD27880 gesetzt, wird der Zählimpuls statt über das Programmende M2/M30 über einen über folgendes Maschinendatum parametrierbaren M-Befehl ausgelöst:

MD27882 \$MC_PART_COUNTER_MCODE[<n>] (Werkstückzählung mit anwenderdefiniertem M-Befehl)

<n>	Bedeutung
0	MD27882[0] definiert den M-Befehl, bei dem \$AC_TOTAL_PARTS hochgezählt wird.
1	MD27882[1] definiert den M-Befehl, bei dem \$AC_ACTUAL_PARTS hochgezählt wird.
2	MD27882[2] definiert den M-Befehl, bei dem \$AC_SPECIAL_PARTS hochgezählt wird.

Bei Aufruf eines anwenderdefinierten M-Befehls wird der betreffende Werkstückzähler um den Wert "1" erhöht.

Schutzstufe für Werkstückzählung

Mit dem folgenden Maschinendatum wird die Schutzstufe für das Ein-/Ausschalten der Werkstückzählung über die Bedienoberfläche eingestellt:

MD51074 \$MN_ACCESS_WRITE_WPC_COUNTER = <Schutzstufe>

10.20.2.3 Randbedingungen

Betriebsartenwechsel / NC-RESET

Durch Betriebsartenwechsel oder NC-RESET werden die Werkstückzähler nicht beeinflusst.

\$AC_REQUIRED_PARTS ≤ 0

Bei $\$AC_REQUIRED_PARTS \leq 0$ und MD27880 $\$MC_PART_COUNTER$.Bit 0 == 1 wird für alle aktiven Zähler die Zählung und der mit MD27880 eingestellte Identitätsvergleich **nicht** durchgeführt.

10.20.2.4 Beispiele

Aktivierung des Werkstückzählers \$AC_REQUIRED_PARTS:

- MD27880 $\$MC_PART_COUNTER$ = 'H3'

$\$AC_REQUIRED_PARTS$ ist aktiv.

Anzeige-Alarm bei: $\$AC_REQUIRED_PARTS == \$AC_SPECIAL_PARTS$

Aktivierung des Werkstückzählers \$AC_TOTAL_PARTS:

- MD27880 $\$MC_PART_COUNTER$ = 'H10'
- MD27882 $\$MC_PART_COUNTER_MCODE[0]$ = 80

$\$AC_TOTAL_PARTS$ ist aktiv, mit jedem M02 wird der Zähler um den Wert "1" erhöht.

$\$MC_PART_COUNTER_MCODE[0]$ hat keine Bedeutung.

Aktivierung des Werkstückzählers \$AC_ACTUAL_PARTS:

- MD27880 $\$MC_PART_COUNTER$ = 'H300'
- MD27882 $\$MC_PART_COUNTER_MCODE[1]$ = 17

$\$AC_TOTAL_PARTS$ ist aktiv, mit jedem M17 wird der Zähler um den Wert "1" erhöht.

Aktivierung des Werkstückzählers \$AC_SPECIAL_PARTS:

- MD27880 $\$MC_PART_COUNTER$ = 'H3000'
- MD27882 $\$MC_PART_COUNTER_MCODE[2]$ = 77

$\$AC_SPECIAL_PARTS$ ist aktiv.

Mit jedem M77 erfolgt: $\$AC_SPECIAL_PARTS + 1$

Abschalten des Werkstückzählers \$AC_ACTUAL_PARTS:

- MD27880 $\$MC_PART_COUNTER$ = 'H200'
- MD27882 $\$MC_PART_COUNTER_MCODE[1]$ = 50

$\$AC_ACTUAL_PARTS$ ist nicht aktiv.

Aktivierung aller Zähler:

- MD27880 \$MC_PART_COUNTER = 'H3313'
- MD27882 \$MC_PART_COUNTER_MCODE[0] = 80
- MD27882 \$MC_PART_COUNTER_MCODE[1] = 17
- MD27882 \$MC_PART_COUNTER_MCODE[2] = 77

\$AC_REQUIRED_PARTS ist aktiv.

Anzeige-Alarm bei: \$AC_REQUIRED_PARTS == \$AC_SPECIAL_PARTS

\$AC_TOTAL_PARTS ist aktiv, mit jedem M02 wird der Zähler um den Wert "1" erhöht.

\$MC_PART_COUNTER_MCODE[0] hat keine Bedeutung.

\$AC_ACTUAL_PARTS ist aktiv, mit jedem M17 wird der Zähler um den Wert "1" erhöht.

\$AC_SPECIAL_PARTS ist aktiv, mit jedem M77 wird der Zähler um den Wert "1" erhöht.

Keine Bearbeitung des Werkstückzählers \$AC_ACTUAL_PARTS bei Programm-Test / Satzsuchlauf:

- MD27880 \$MC_PART_COUNTER = 'H700'
- MD27882 \$MC_PART_COUNTER_MCODE[1] = 75

\$AC_ACTUAL_PARTS ist aktiv, mit jedem M75 wird der Zähler um den Wert "1" erhöht, außer bei Programm-Test und Satzsuchlauf.

Aufhebung der im MD27880 \$MC_PART_COUNTER getroffenen Zählmodi bei Bit 0 = 1:

- MD27882 \$MC_PART_COUNTER_MCODE[0] = 41
- MD27882 \$MC_PART_COUNTER_MCODE[1] = 42
- MD27882 \$MC_PART_COUNTER_MCODE[2] = 43

Programmcode	Kommentar
...	
N100 \$AC_REQUIRED_PARTS=-10	; Wert < 0: Zählung einstellen.
N200 M41 M43	; Keine Zählung.
N300 M42	
...	
N500 \$AC_REQUIRED_PARTS=52	; Wert > 0: Zählung entsprechend MD27880 aktiv.
N501 M43	; Zählung.
N502 M42 M41	; Zählung.
...	

10.21 Datenlisten

10.21.1 Funktion

Mit der Funktion "Werkstückzähler" stehen in Form von kanalspezifische Systemvariablen diverse Zähler mit einem Wertebereich von 0 bis 999.999.999 zur Verfügung. Auf die Systemvariablen kann lesend und schreibend zugegriffen werden.

Über die nachfolgend beschriebenen kanalspezifischen Maschinendaten kann auf die Zähler-Aktivierung, den Zeitpunkt der Nullung und den Zählalgorithmus Einfluss genommen werden.

Systemvariablen zur Werkstückzählung

Systemvariable	Bedeutung
\$AC_REQUIRED_PARTS	Anzahl der zu fertigenden Werkstücke (Soll-Werkstückzahl) In diesem Zähler kann die Anzahl der Werkstücke definiert werden, bei dessen Erreichen die Ist-Werkstückzahl (\$AC_ACTUAL_PARTS) auf "0" zurückgesetzt wird. Über MD27880 kann die Generierung des Anzeige-Alarms: "Kanal %1 Werkstueck-Soll = %2 erreicht" und des kanalspezifischen NC/PLC-Nahtstellensignals: DB21, ... DBX317.1 (Werkstück-Soll erreicht) aktiviert werden.
\$AC_TOTAL_PARTS	Anzahl der insgesamt gefertigten Werkstücke (Ist-Werkstückzahl Gesamt) Dieser Zähler gibt die Anzahl aller ab Startzeitpunkt gefertigten Werkstücke an. Der Wert wird nur bei einem Steuerungshochlauf mit Standardwerten automatisch auf "0" zurückgesetzt.
\$AC_ACTUAL_PARTS	Anzahl der gefertigten Werkstücke (Ist-Werkstückzahl) In diesem Zähler wird die Anzahl aller ab Startzeitpunkt gefertigten Werkstücke registriert. Bei einem Erreichen der Soll-Werkstückzahl (\$AC_REQUIRED_PARTS) wird der Zähler automatisch auf "0" zurückgesetzt (\$AC_REQUIRED_PARTS > 0 vorausgesetzt).
\$AC_SPECIAL_PARTS	Anzahl der vom Anwender gezählten Werkstücke Dieser Zähler erlaubt dem Anwender eine Werkstückzählung nach eigener Definition. Definiert werden kann eine Alarmausgabe bei Erreichen der Soll-Werkstückzahl (\$AC_REQUIRED_PARTS). Eine Nullung des Zählers muss der Anwender selbst vornehmen.

Hinweis

Alle Werkstückzähler werden bei einem Steuerungshochlauf mit Standardwerten, auf den Wert "0" gesetzt und können, unabhängig von ihrer Aktivierung, gelesen und geschrieben werden.

10.21.2 Maschinendaten

10.21.2.1 Allgemeine Maschinendaten

Anzeige-Maschinendaten

Nummer	Bezeichner: \$MM_	Beschreibung
SINUMERIK Operate		
9421	MA_AXES_SHOW_GEO_FIRST	Geo-Achsen des Kanals zu erst anzeigen
9422	MA_PRESET_MODE	PRESET / Basisverschiebung in JOG
9423	MA_MAX_SKP_LEVEL	Maximale Anzahl von Ausblendebenen

NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10010	ASSIGN_CHAN_TO_MODE_GROUP	Kanal gültig in BAG
10125	EES_NC_NAME	NCU-Name für die Generierung eindeutiger NC-Programmnamen im EES-Betrieb
10280	PROG_FUNCTION_MASK	Vergleichsbefehle ">" und "<"
10700	PREPROCESSING_LEVEL	Programmvorverarbeitungsstufe
10702	IGNORE_SINGLEBLOCK_MASK	Einzelatz-Stopp verhindern
10707	PROG_TEST_MASK	Programmtest Modi
10708	SERUPRO_MASK	Satzsuchlauf Modi
10710	PROG_SD_RESET_SAVE_TAB	Zu aktualisierende Settingdaten
10711	NC_LANGUAGE_CONFIGURATION	Art und Weise, wie mit Sprachbefehlen verfahren wird, deren zugehörige Option bzw. Funktion nicht aktiviert ist.
10713	M_NO_FCT_STOPRE	M-Funktion mit Vorlaufstopp
10715	M_NO_FCT_CYCLE	Durch UP zu ersetzende M-Funktion
10716	M_NO_FCT_CYCLE_NAME	UP-Name für M-Funktions-Ersetzung
10717	T_NO_FCT_CYCLE_NAME	Name des Werkzeugwechselzyklus für T-Funktionsersetzung
10718	M_NO_FCT_CYCLE_PAR	M-Funktionsersetzung mit Parametern
10719	T_NO_FCT_CYCLE_MODE	Parametrierung der T-Funktionsersetzung
11450	SEARCH_RUN_MODE	Suchlauf Parametrierung
11470	REPOS_MODE_MASK	Repositioniereigenschaften
11600	BAG_MASK	BAG-Verhalten bezüglich ASUP
11602	ASUP_START_MASK	Stoppgründe für ASUP ignorieren
11604	ASUP_START_PRIO_LEVEL	Prioritäten, ab denen \$MN_ASUP_START_MASK wirksam ist
11610	ASUP_EDITABLE	Aktivierung eines anwenderspezifischen ASUP Programmes

10.21 Datenlisten

Nummer	Bezeichner: \$MN_	Beschreibung
11612	ASUP_EDIT_PROTECTION_LEVEL	Schutzstufe des anwenderspezifischen ASUP Programms
11620	PROG_EVENT_NAME	Programmname für PROG_EVENT
11625	FILE_ONLY_WITH_EXTENSION	Beim Programmaufruf nur nach Dateien mit einer Dateikennung suchen
11626	CYCLES_ONLY_IN_CYCDIR	Programme mit Interface nur in den Zyklenverzeichnissen suchen
11717	D_NO_FCT_CYCLE_NAME	Unterprogrammname für D-Funktionsersetzung
15700	LANG_SUB_NAME	Name für Substitutionsunterprogramm
15702	LANG_SUB_PATH	Aufrufpfad für Substitutionsunterprogramm
17200	GMMC_INFO_NO_UNIT	globale HMI Info (ohne physikalische Einheit)
17201	GMMC_INFO_NO_UNIT_STATUS	globale HMI Statusinfo (ohne physik. Einheit)
18045	EES_MODE_INFO	Modus, in dem die Funktion EES arbeitet
18360	MM_EXT_PROG_BUFFER_SIZE	FIFO-Buffer Größe für Abarbeiten von Extern (DRAM)
18362	MM_EXT_PROG_NUM	Anzahl der gleichzeitig von 'Extern abarbeitbaren Programmebenen

10.21.2.2 Kanal-spezifische Maschinendaten

Grundmaschinenendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20000	CHAN_NAME	Kanalname
20050	AXCONF_GEOAX_ASSIGN_TAB	Zuordnung Geometrieachse zu Kanalachse
20060	AXCONF_GEOAX_NAME_TAB	Geometrieachsname im Kanal
20070	AXCONF_MACHAX_USED	Maschinenachsnummer gültig im Kanal
20080	AXCONF_CHANAX_NAME_TAB	Kanalachsname im Kanal [Kanalachsnr.]: 0...7
20090	SPIND_DEF_MASTER_SPIND	Löschstellung der Masterspindel im Kanal
20100	DIAMETER_AX_DEF	Geometrieachse mit Planachsfunktion
20106	PROG_EVENT_IGN_SINGLEBLOCK	Prog-Events ignorieren den Einzelsatz
20107	PROG_EVENT_IGN_INHIBIT	Prog-Events ignorieren die Einlesesperre
20108	PROG_EVENT_MASK	Ereignisgesteuerte Programmaufrufe
20109	PROG_EVENT_MASK_PROPERTIES	Eigenschaften Prog-Events
20114	MODESWITCH_MASK	Einstellung für Repos
20116	IGNORE_INHIBIT_ASUP	Anwender-ASUPs trotz Einlesesperre komplett abarbeiten
20117	IGNORE_SINGLEBLOCK_ASUP	Anwender-ASUPs trotz Einzelsatzbearbeitung komplett abarbeiten
20160	CUBIC_SPLINE_BLOCKS	Anzahl der Sätze beim C-Spline
20170	COMPRESS_BLOCK_PATH_LIMIT	Maximale Verfahrenlänge eines NC-Satzes bei Kompression

Nummer	Bezeichner: \$MC_	Beschreibung
20191	IGN_PROG_STATE_ASUP	Ausführung der Interruptroutine auf der Bedienoberfläche nicht anzeigen
20192	PROG_EVENT_IGN_PROG_STATE	Ausführung des Prog-Events auf der Bedienoberfläche nicht anzeigen
20193	PROG_EVENT_IGN_STOP	Prog-Events ignorieren die Stopp-Taste
20194	IGNORE_NONCSTART_ASUP	ASUP-Start trotz anstehender Alarmreaktion "Verriegelung NC-Start" bei bestimmten Anwenderalarmen zulässig
20210	CUTCOM_CORNER_LIMIT	Maximalwinkel für Ausgleichssätze bei WRK
20220	CUTCOM_MAX_DISC	Maximaler Wert für DISC
20230	CUTCOM_CURVE_INSERT_LIMIT	Maximalwinkel für Schnittpunktberechnung bei WRK
20240	CUTCOM_MAXNUM_CHECK_BLOCKS	Sätze für vorausschauende Konturberechnung bei WRK
20250	CUTCOM_MAXNUM_DUMMY_BLOCKS	Maximale Satzanzahl ohne Verfahrbewegung bei WRK
20270	CUTTING_EDGE_DEFAULT	Grundstellung der Werkzeugschneide ohne Programmierung
20400	LOOKAH_USE_VELO_NEXT_BLOCK	Lookahead auf programmierte Folgesatzgeschwindigkeit
20430	LOOKAH_NUM_OVR_POINTS	Anzahl der Override-Eckwerte bei Lookahead
20440	LOOKAH_OVR_POINTS	Override-Schalter-Eckwerte bei Lookahead
20500	CONST_VELO_MIN_TIME	Minimale Zeit mit konstanter Geschwindigkeit
20600	MAX_PATH_JERK	Bahnbezogener Maximalruck
20610	ADD_MOVE_ACCEL_RESERVE	Beschleunigungsreserve für überlagerte Bewegungen
20700	REFP_NC_START_LOCK	NC-Startsperre ohne Referenzpunkt
20750	ALLOW_GO_IN_G96	G0-Logik bei G96, G961
20800	SPF_END_TO_VDI	Unterprogrammende an PLC
21000	CIRCLE_ERROR_CONST	Kreisendpunktüberwachung Konstante
21010	CIRCLE_ERROR_FACTOR	Kreisendpunktüberwachung Faktor
21100	ORIENTATION_IS_EULER	Winkeldefinition bei Orientierungsprogrammierung
21110	X_AXIS_IN_OLD_X_Z_PLANE	Koordinatensystem bei automatischer Framedefinition
21200	LIFTFAST_DIST	Verfahrstrecke bei Schnellabheben von der Kontur
21210	SETINT_ASSIGN_FASTIN	HW-Zuordnung des ext. NC-Eingangsbytes für NC-Programm-Interrupt
21202	LIFTFAST_WITH_MIRROR	Schnellabheben mit Spiegeln

Satzsuchlauf

Nummer	Bezeichner: \$MC_	Beschreibung
20128	COLLECT_TOOL_CHANGE	Werkzeugwechselbefehle an PLC nach Satzsuchlauf
22600	SERUPRO_SPEED_MODE	Geschwindigkeit bei Satzsuchlauf-Typ 5
22601	SERUPRO_SPEED_FACTOR	Geschwindigkeitsfaktor bei Satzsuchlauftyp 5
22621	ENABLE_START_MODE_MASK_PRT	Schaltet MD22620: START_MODE_MASK_PRT bei Satzsuchlauf SERUPRO frei

Nummer	Bezeichner: \$MC_	Beschreibung
22622	DISABLE_PLC_START	Teileprogrammstart via PLC erlauben
22680	AUTO_IPTR_LOCK	Unterbrechungszeiger sperren

Reset-Verhalten

Nummer	Bezeichner: \$MC_	Beschreibung
20110	RESET_MODE_MASK	Grundstellung nach RESET/Teileprogrammende
20112	START_MODE_MASK	Grundstellung bei NC-Start nach Teileprogrammstart
20118	GEOAX_CHANGE_RESET	Automatischen Geometrieachswechsel erlauben
20120	TOOL_RESET_VALUE	Werkzeug-Längenkorrektur im Hochlauf (RESET/Teileprogrammende)
20121	TOOL_PRESEL_RESET_VALUE	Vorgewähltes Werkzeug bei RESET
20130	CUTTING_EDGE_RESET_VALUE	Werkzeugschneiden-Längenkorrektur im Hochlauf (RESET/Teileprogrammende)
20140	TRAFO_RESET_VALUE	Transformationsdatensatz im Hochlauf (RESET/Teileprogrammende)
20150	GCODE_RESET_VALUES	Löschstellung der G-Gruppen
20152	GCODE_RESET_MODE	Resetverhalten der G-Gruppen
20156	MAXNUM_GCODES_EXT	Resetverhalten der externen G-Gruppen
22620	START_MODE_MASK_PRT	Grundstellung bei speziellem NC-Start nach Hochlauf und bei RESET

Hilfsfunktionseinstellungen

Nummer	Bezeichner: \$MC_	Beschreibung
22000	AUXFU_ASSIGN_GROUP	Hilfsfunktionsgruppe
22010	AUXFU_ASSIGN_TYPE	Hilfsfunktionsart
22020	AUXFU_ASSIGN_EXTENSION	Hilfsfunktionserweiterung
22030	AUXFU_ASSIGN_VALUE	Hilfsfunktionswert
22200	AUXFU_M_SYNC_TYPE	Ausgabezeitpunkt der M-Funktionen
22210	AUXFU_S_SYNC_TYPE	Ausgabezeitpunkt der S-Funktionen
22220	AUXFU_T_SYNC_TYPE	Ausgabezeitpunkt der T-Funktionen
22230	AUXFU_H_SYNC_TYPE	Ausgabezeitpunkt der H-Funktionen
22240	AUXFU_F_SYNC_TYPE	Ausgabezeitpunkt der F-Funktionen
22250	AUXFU_D_SYNC_TYPE	Ausgabezeitpunkt der D-Funktionen
22400	S_VALUES_ACTIVE_AFTER_RESET	S-Funktion über RESET hinaus wirksam
22410	F_VALUES_ACTIVE_AFTER_RESET	F-Funktion über RESET hinaus wirksam
22510	GCODE_GROUPS_TO_PLC	G-Befehle, die bei Satzwechsel/Reset an die Nahtstelle NC/PLC ausgegeben werden
22550	TOOL_CHANGE_MODE	Neue Werkzeugkorrektur bei M-Funktion
22560	TOOL_CHANGE_M_CODE	M-Funktion für Werkzeugwechsel

Speichereinstellungen

Nummer	Bezeichner: \$MC_	Beschreibung
27900	REORG_LOG_LIMIT	Prozentsatz des IPO-Puffers für Freigabe des Logfiles
28000	MM_REORG_LOG_FILE_MEM	Speichergöße für REORG (DRAM)
28010	MM_NUM_REORG_LUD_MODULES	Anzahl der Bausteine für lokale Anwendervariablen bei REORG
28020	MM_NUM_LUD_NAMES_TOTAL	Anzahl der lokalen Anwendervariablen (DRAM)
28040	MM_LUD_VALUES_MEM	Speichergöße für lokale Anwendervariablen (DRAM)
28050	MM_NUM_R_PARAM	Anzahl der kanalspezifischen R-Parameter (SRAM)
28060	MM_IPO_BUFFER_SIZE	Anzahl der NC-Sätze im IPO-Puffer (DRAM)
28070	MM_NUM_BLOCKS_IN_PREP	Anzahl der Sätze für die Satzaufbereitung. (DRAM)
28080	MM_NUM_USER_FRAMES	Anzahl der einstellbaren Frames (SRAM)
28090	MM_NUM_CC_BLOCK_ELEMENTS	Anzahl Satzelemente für Compile-Zyklen (DRAM)
28100	MM_NUM_CC_BLOCK_USER_MEM	Größe des Satzspeichers für Compile-Zyklen (DRAM)
28400	MM_ABSBLOCK	Basissätze mit Absolutwerten aktivieren
28402	MM_ABSBLOCK_BUFFER_CONF	Größe des Upload-Buffers dimensionieren
28500	MM_PREP_TASK_STACK_SIZE	Stackgröße der Präparationtask (DRAM)

Programmlaufzeit und Werkstückzähler

Nummer	Bezeichner: \$MC_	Beschreibung
27860	PROCESSTIMER_MODE	Aktivierung der Laufzeit-Messung
27880	PART_COUNTER	Aktivierung der Werkstück-Zähler
27882	PART_COUNTER_MCODE[]	Werkstückzählung über einen M-Befehl

10.21.2.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30465	AXIS_LANG_SUB_MASK	Substituierung von NC-Sprachbefehlen
30550	AXCONF_ASSIGN_MASTER_CHAN	Löschstellung des Kanals für Achswechsel
30600	FIX_POINT_POS	Festwertpositionen der Achsen bei G75
33100	COMPRESS_POS_TOL	Maximale Abweichung bei Kompression

10.21.3 Settingdaten

10.21.3.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42000	THREAD_START_ANGLE	Startwinkel bei Gewinde
42010	THREAD_RAMP_DISP	Beschleunigungsverhalten der Achse beim Gewinde-schneiden
42100	DRY_RUN_FEED	Probelaufvorschub
42200	SINGLEBLOCK2_STOPRE	Debugmode für SBL2 aktivieren
42444	TARGET_BLOCK_INCR_PROG	Aufsetzmodus nach Satzsuchlauf mit Berechnung
42700	EXT_PROG_PATH	Programmpfades für externen Unterprogrammaufruf EXTCALL
42750	ABSBLOCK_ENABLE	Basis-Satzanzeige freigeben
42990	MAX_BLOCKS_IN_IPOBUFFER	Maximale Anzahl der Sätze im Interpolationspuffer

10.21.4 Signale

10.21.4.1 Signale an NC

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Not-Halt	DB10.DBX56.1	DB2600.DBX0.1

10.21.4.2 Signale an BAG

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Betriebsart AUTOMATIK	DB11.DBX0.0	DB3000.DBX0.0
Betriebsart MDA	DB11.DBX0.1	DB3000.DBX0.1
Betriebsart JOG	DB11.DBX0.2	DB3000.DBX0.2
Betriebsart-Wechselsperre	DB11.DBX0.4	DB3000.DBX0.4
BAG-Stop	DB11.DBX0.5	DB3000.DBX0.5
BAG-Stop Achsen plus Spindel	DB11.DBX0.6	DB3000.DBX0.6
BAG-Reset	DB11.DBX0.7	DB3000.DBX0.7
Maschinenfunktion Teach In	DB11.DBX1.0	DB3000.DBX1.0
Maschinenfunktion REPOS	DB11.DBX1.1	-
Maschinenfunktion REF	DB11.DBX1.2	DB3000.DBX1.2

10.21.4.3 Signale von BAG

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
angewählte Betriebsart AUTOMATIK	DB11.DBX4.0	-
angewählte Betriebsart MDA	DB11.DBX4.1	-
angewählte Betriebsart JOG	DB11.DBX4.2	-
angewählte Maschinenfunktion Teach In	DB11.DBX5.0	-
angewählte Maschinenfunktion REPOS	DB11.DBX5.1	-
angewählte Maschinenfunktion REF	DB11.DBX5.2	-
aktive Betriebsart AUTOMATIK	DB11.DBX6.0	DB3100.DBX0.0
aktive Betriebsart MDA	DB11.DBX6.1	DB3100.DBX0.1
aktive Betriebsart JOG	DB11.DBX6.2	DB3100.DBX0.2
BAG betriebsbereit	DB11.DBX6.3	DB3100.DBX0.3
BAG-Reset ausgeführt	DB11.DBX6.4	DB3100.DBX0.4
NC Intern JOG aktiv	DB11.DBX6.5	-
alle Kanäle im Reset-Zustand	DB11.DBX6.7	DB3100.DBX0.7
aktive Maschinenfunktion Teach In	DB11.DBX7.0	DB3100.DBX1.0
aktive Maschinenfunktion REPOS	DB11.DBX7.1	-
aktive Maschinenfunktion REF	DB11.DBX7.2	DB3100.DBX1.2

10.21.4.4 Signale an Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
DRF aktivieren	DB21,DBX0.3	DB320x.DBX0.3
Einzelsatz aktivieren	DB21,DBX0.4	DB320x.DBX0.4
M01 aktivieren	DB21,DBX0.5	DB320x.DBX0.5
Probelaufvorschub aktivieren	DB21,DBX0.6	DB320x.DBX0.6
PLC-Aktion beendet	DB21,DBX1.6	-
Programmtest aktivieren	DB21,DBX1.7	DB320x.DBX1.7
Satz ausblenden Ebenen: /0 bis /7	DB21,DBX2.0-7	DB320x.DBX2.0-7
Einlesesperre	DB21,DBX6.1	DB320x.DBX6.1
Programmebenen-Abbruch	DB21,DBX6.4	DB320x.DBX6.4
NC-Startsperre	DB21,DBX7.0	DB320x.DBX7.0
NC-Start	DB21,DBX7.1	DB320x.DBX7.1
NC-Stop an Satzgrenze	DB21,DBX7.2	DB320x.DBX7.2
NC-Stop	DB21,DBX7.3	DB320x.DBX7.3
NC-Stop Achsen plus Spindeln	DB21,DBX7.4	DB320x.DBX7.4
Reset	DB21,DBX7.7	DB330x.DBX3.7
REPOSPATHMODE	DB21,DBX31.0-2	-
REPOSMODEEDGE	DB21,DBX31.4	-

10.21.4.5 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
DRF angewählt	DB21,DBX24.3	DB170x.DBX0.3
NC assoziiertes M01 auswählen	DB21,DBX24.4	-
M01 angewählt	DB21,DBX24.5	DB170x.DBX0.5
Probelaufvorschub angewählt	DB21,DBX24.6	DB170x.DBX0.6
REPOSPATHMODE 0 - 2	DB21,DBX25.0-2	-
Vorschub-Override für Eilgang angewählt	DB21,DBX25.3	DB170x.DBX1.3
REPOS MODE EDGE	DB21,DBX25.4	-
Programmtest angewählt	DB21,DBX25.7	DB170x.DBX1.7
Satzausblenden angewählt: /0 - /7	DB21,DBX26.0-7	DB170x.DBX2.0-7
Satzausblenden angewählt: /8	DB21,DBX27.0	DB170x.DBX3.0
Satzausblenden angewählt: /9	DB21,DBX27.1	DB170x.DBX3.1
REPOSPATHMODE: 0 - 2	DB21,DBX31.0-2	-
REPOS MODE EDGE	DB21,DBX31.4	-
Satzausblenden aktiv /8	DB21,DBX31.6	DB320x.DBX15.6
Satzausblenden aktiv /9	DB21,DBX31.7	DB320x.DBX15.7
Abarbeiten von extern aktiv	DB21,DBX32.0	DB330x.DBX0.0
Aktionssatz aktiv	DB21,DBX32.3	DB330x.DBX0.3
Anfahrsatz aktiv	DB21,DBX32.4	DB330x.DBX0.4
M0/M1 aktiv	DB21,DBX32.5	DB330x.DBX0.5
letzter Aktionssatz aktiv	DB21,DBX32.6	DB330x.DBX0.6
Satzsuchlauf aktiv	DB21,DBX33.4	DB330x.DBX1.4
M02/M30 aktiv	DB21,DBX33.5	DB330x.DBX1.5
Transformation aktiv	DB21,DBX33.6	DB330x.DBX1.6
Programmtest aktiv	DB21,DBX33.7	DB330x.DBX1.7
Programm Zustand: läuft	DB21,DBX35.0	DB330x.DBX3.0
Programm Zustand: warten	DB21,DBX35.1	DB330x.DBX3.1
Programm Zustand: angehalten	DB21,DBX35.2	DB330x.DBX3.2
Programm Zustand: unterbrochen	DB21,DBX35.3	DB330x.DBX3.3
Programm Zustand: abgebrochen	DB21,DBX35.4	DB330x.DBX3.4
Kanal Zustand: aktiv	DB21,DBX35.5	DB330x.DBX3.5
Kanal Zustand: unterbrochen	DB21,DBX35.6	DB330x.DBX3.6
Kanal Zustand: Reset	DB21,DBX35.7	DB330x.DBX3.7
Interrupt-Behandlung aktiv	DB21,DBX36.4	-
Kanal betriebsbereit	DB21,DBX36.5	DB330x.DBX4.5
Einlesefreigabe wird ignoriert	DB21,DBX37.6	-
Stoppen am Blockende bei SBL wird unterdrückt	DB21,DBX37.7	-
Nr. der aktiven G-Befehl der G-Gruppe 1 – n (8 Bit-Int)	DB21,DBB208-271	DB350x.DBB0-63
Werkstück-Soll erreicht	DB21,DBX317.1	DB330x.DBX4001.1
ASUP ist angehalten	DB21,DBX318.0	DB330x.DBX4002.0

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Satzsuchlauf via Programmtest ist aktiv	DB21,DBX318.1	-
REPOS MODE EDGEACKN	DB21,DBX319.0	-
Repos Path Mode Quitt: 0 - 2	DB21,DBX319.1-3	-
Repos DEFERAL Chan	DB21,DBX319.5	-
Anzeige des auslösenden Ereignisses beim ereignisgesteuerten Programmaufruf	DB21,DBX376.0-7	DB330x.DBB4004
ASUP ist aktiv	DB21,DBX378.0	DB330x.DBB4006.0
ASUP mit unterdrückter Anzeigenaktualisierung ist aktiv	DB21,DBX378.1	DB330x.DBB4006.1

10.21.4.6 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
REPOSDELAY	DB31,DBX10.0	-

10.21.4.7 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
REPOS Verschiebung	DB31,DBX70.0	-
REPOS Verschiebung gültig	DB31,DBX70.1	-
REPOS Delay Quitt	DB31,DBX70.2	-
REPOSDELAY	DB31,DBX72.0	-
Bahnachse	DB31,DBX76.4	DB390x.DBX1002.4

K2: Achsen, Koordinatensysteme, Frames

11.1 Kurzbeschreibung

11.1.1 Achsen

Maschinenachsen

Maschinenachsen sind die real an der (Werkzeug-)Maschine vorhandenen Achsen.

Kanalachsen

Jede Geometrieachse und jede Zusatzachse wird einem Kanal und somit einer Kanalachse zugewiesen. Geometrieachsen und Zusatzachsen werden immer in "ihrem" Kanal verfahren.

Geometrieachsen

Die drei Geometrieachsen bilden immer ein fiktives rechtwinkliges Koordinatensystem, das Basiskoordinatensystem (BKS).

Durch Verwendung von FRAMES (Verschiebung, Drehung, Skalierung, Spiegelung) können Geometrieachsen des Werkstückkoordinatensystems (WKS) auf das BKS abgebildet werden.

Zusatzachsen

Bei Zusatzachsen ist, im Gegensatz zu Geometrieachsen, kein geometrischer Zusammenhang zwischen den Achsen definiert.

Bahnachsen

Bahnachsen zeichnen sich dadurch aus, dass sie gemeinsam interpoliert werden (alle Bahnachsen eines Kanals haben gemeinsam einen Bahninterpolator).

Alle Bahnachsen eines Kanals haben eine gemeinsame Beschleunigungsphase, eine Konstantfahrphase und eine Verzögerungsphase.

Positionierachsen

Positionierachsen zeichnen sich dadurch aus, dass sie getrennt interpoliert werden (jede Positionierachse hat einen eigenen Achsinterpolator). Jede Positionierachse hat einen eigenen Vorschub und eine eigene Beschleunigungskennlinie.

Synchronachsen

Synchronachsen werden gemeinsam mit Bahnachsen interpoliert (alle Bahnachsen und Synchronachsen eines Kanals haben einen gemeinsamen Bahninterpolator).

Alle Bahnachsen und alle Synchronachsen eines Kanals haben eine gemeinsame Beschleunigungsphase, eine Konstantfahrphase und eine Verzögerungsphase.

Achskonfiguration

Die Zuordnung zwischen den Geometrieachsen, Zusatzachsen, Kanalachsen und Maschinenachsen, sowie die Festlegung der Namen der einzelnen Achstypen wird über folgende Maschinendaten getroffen:

MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB (Zuordnung Geometrieachse zu Kanalachse)

MD20060 \$MC_AXCONF_GEOAX_NAME_TAB (Geometrieachsname im Kanal)

MD20070 \$MC_AXCONF_MACHAX_USED (Maschinenachsnummer gültig im Kanal)

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB (Kanalachsname im Kanal)

MD10000 \$MN_AXCONF_MACHAX_NAME_TAB (Maschinenachsname)

MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX (Zuordnung Spindel zu Maschinenachse)

Umschaltbare Geometrieachsen

Mit der Funktion "Umschaltbare Geometrieachsen" kann aus dem Teileprogramm heraus der Verbund der Geometrieachsen aus anderen Kanalachsen zusammengesetzt werden.

Achsen, die in einem Kanal zunächst als synchrone Zusatzachsen projiziert sind, können durch einen Programmbefehl eine beliebige Geometrieachse ersetzen.

Link-Achse

Link-Achsen sind Achsen, die an einer anderen NCU physikalisch angeschlossen sind und deren Lageregelung unterliegen. Link-Achsen können dynamisch Kanälen einer **anderen** NCU zugeordnet werden. Link-Achsen sind aus Sicht einer bestimmten NCU nicht-lokale Achsen.

Der dynamischen Änderung der Zuordnung zu einer NCU dient das Konzept der **Achscontainer**. Achstausch mit `GET` und `RELEASE` aus dem Teileprogramm ist für Link-Achsen über NCU-Grenzen hinweg nicht verfügbar.

Die Link-Achsen sind beschrieben in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Mehrere Bedientafeln an mehreren NCUs, Dezentrale Systeme (B3)

Achscontainer

Ein Achscontainer ist eine Ringpuffer-Datenstruktur, in der die Zuordnung von lokalen Achsen und/oder Link-Achsen zu Kanälen erfolgt. Die Einträge im Ringpuffer sind **zyklisch verschiebbar**.

Die Link-Achsen Konfiguration lässt im logischen Maschinenachs-Abbild neben dem direkten Verweis auf lokale Achsen oder Link-Achsen den Verweis auf Achscontainer zu.

Ein solcher Verweis besteht aus:

- Achscontainer-Nummer
- Slot (Ringpuffer-Platz innerhalb des entsprechenden Achscontainers)

Als Eintrag in einem Ringpuffer-Platz steht:

- Eine lokale Achse
oder
- Eine Link-Achse

Die Funktion "Achscontainer" ist beschrieben in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Mehrere Bedientafeln an mehreren NCUs, Dezentrale Systeme (B3)

11.1.2 Koordinatensysteme

MKS

Das Maschinenkoordinatensystem (MKS) zeichnet sich durch folgende Eigenschaften aus:

- Es wird durch die Maschinenachsen gebildet.
- Die Maschinenachsen können rechtwinkelig kartesisch oder beliebig zueinander angeordnet sein.
- Die Maschinenachsen haben einstellbare Namen.
- Die Maschinenachsen können Linearachsen oder Rundachsen sein.

BKS

Das Basiskoordinatensystem (BKS) zeichnet sich durch folgende Eigenschaften aus:

- Die Geometrieachsen bilden ein rechtwinkeliges kartesisches Koordinatensystem.
- Das BKS geht durch kinematische Transformation aus dem MKS hervor.

BNS

Das Basis-Nullpunktsystem (BNS) ist das Basis-Koordinatensystem mit Basis-Verschiebung.

ENS

Das Einstellbare Nullpunktsystem (ENS) ist das Werkstückkoordinatensystem mit programmierbarem Frame aus Sicht vom WKS. Der Werkstücknullpunkt wird durch die einstellbaren Frames G54 ... G599 festgelegt.

WKS

Das Werkstückkoordinatensystem (WKS) zeichnet sich durch folgende Eigenschaften aus:

- Im WKS werden alle Koordinaten der Achsen programmiert (Teileprogramm).
- Es wird durch Geometrieachsen und Zusatzachsen gebildet.
- Geometrieachsen bilden immer ein rechtwinkeliges kartesisches Koordinatensystem
- Zusatzachsen bilden ein Koordinatensystem ohne geometrischen Zusammenhang zwischen den Zusatzachsen.
- Geometrieachsen und Zusatzachsen haben einstellbare Namen.
- Durch FRAMES kann das WKS verschoben, gedreht, skaliert oder gespiegelt werden (TRANS, ROT, SCALE, MIRROR).
Auch Mehrfachverschiebungen, Mehrfachdrehungen, etc. sind möglich.

Externe Nullpunktverschiebung

Die externe Nullpunktverschiebung hat folgende Eigenschaften:

- Zu einem von PLC bestimmten Zeitpunkt wird eine zuvor festgelegte zusätzliche Nullpunktverschiebung zwischen Basis- und Werkstück-Koordinatensystem aktiviert.
- Die Beträge der Verschiebungen können pro beteiligte Achse gesetzt werden durch:
 - PLC
 - Bedientafel
 - Teileprogramm
- Aktivierte Verschiebungen wirken sich ab dem Abarbeiten des ersten Bewegungssatzes dieser Achsen nach der Aktivierung aus. Die Verschiebungen werden dem programmierten Weg überlagert (keine Interpolation).
Die Geschwindigkeit, mit der die externe Nullpunktverschiebung herausgefahren wird, beträgt:
Programmierter F-Wert + 1/2 JOG-Geschwindigkeit
In G0-Sätzen wird die externe Nullpunktverschiebung am Satzende gefahren.
- Die aktivierten Verschiebungen werden bei RESET und Programmende beibehalten.
- Nach Power On ist die zuletzt aktive Verschiebung weiterhin in der Steuerung gespeichert, muss aber über die PLC wieder aktiviert werden.

11.1.3 Frames

Ein Frame stellt eine geschlossene Rechenvorschrift dar, die kartesische Koordinatensysteme ineinander überführt.

Frame-Komponenten

Ein Frame setzt sich aus folgenden Komponenten zusammen:

Frame-Komponente		Programmierbar mit:
Verschiebung	Grobverschiebung	TRANS ATRANS (additiver Translationsanteil) CTrans (Nullpunktverschiebung für mehrere Achsen) G58 (axiale Nullpunktverschiebung)
	Feinverschiebung	CFINE G59 (axiale Nullpunktverschiebung)
Drehung		ROT / ROTs AROT / AROTs CROTs
Skalierung		SCALE ASCALE
Spiegelung		MIRROR AMIRROR

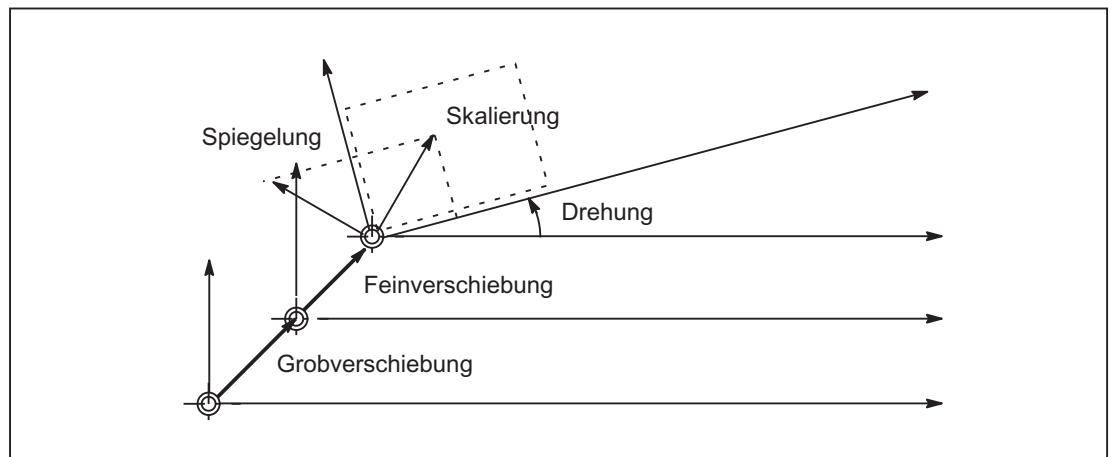


Bild 11-1 Frame-Komponenten

Grob- und Feinverschiebung

Da die Zuordnung von Maschinenachsen zu Kanalachsen und speziell zu Geometrieachsen in allen Kanälen unterschiedlich sein kann, gibt es demzufolge keine eindeutigen kanalübergreifenden geometrischen Zusammenhang zwischen den Kanalachsen. Bei NCU-globalen Frames sind daher nur Verschiebung, Skalierung und Spiegelung möglich. Drehungen sind nicht möglich.

G58: Absolute achsspezifische programmierbare Nullpunktverschiebung (Grobverschiebung)

Mit G58 wird achsspezifisch der absolute Anteil der translatorischen Verschiebung (Grobverschiebung) programmiert. Der additive Anteil der translatorischen Verschiebung (Feinverschiebung) bleibt erhalten.

G59: Additive achsspezifische programmierbare Nullpunktverschiebung (Feinverschiebung)

Mit G59 wird achsspezifisch der additive Anteil der translatorischen Verschiebung (Feinverschiebung) programmiert. Der absolute Anteil der translatorischen Verschiebung (Grobverschiebung) bleibt erhalten.

G59 kann nur verwendet werden, wenn die Feinverschiebung freigegeben wurde:

- MD18600 \$MN_MM_FRAME_FINE_TRANS = TRUE
- MD24000 \$MC_FRAME_ADD_COMPONENTS = TRUE

Drehung

Orientierungen im Raum werden über Frame-Drehungen wie folgt festgelegt:

- Drehung mit ROT legt die Einzeldrehungen für alle Geometrieachsen fest.
- Raumwinkel mit ROTS, AOTS, COTS legen die Orientierung einer Ebene im Raum fest.
- Framedrehung mit TOFRAME definiert einen Frame, dessen Z-Achse in Werkzeugrichtung zeigt.

Skalierung

Mit SCALE werden die programmierbaren Skalierungen (Maßstabfaktor) für alle Geometrieachsen und Zusatzachsen programmiert.

Soll eine neue Skalierung auf einer anderen Skalierung, Drehung, Verschiebung oder Spiegelung aufbauen, muss ASCALE programmiert werden.

Spiegeln

Mit folgendem Maschinendatum wird eingestellt, wie das Spiegeln ausgeführt wird:

MD10610 \$MN_MIRROR_REF_AX

Verkettung

Frames und Frame-Komponenten lassen sich über den Verkettungsoperator ":" verketteten. Der kanalspezifische Aktive Frame \$P_ACTFRAME ergibt sich z.B. aus der Verkettung aller aktiven Frames des Kanals:

```
$P_ACTFRAME = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME :  
               $P_ISO1FRAME : $P_ISO2FRAME : $P_ISO3FRAME :  
               $P_ACTBFRAME : $P_IFRAME : $P_GFRAME :  
               $P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME :  
               $P_PFRAME : $P_ISO4FRAME : $P_CYCFRAME
```

Randbedingungen

Kettenmaßangabe G91

Kettenmaß-Programmierung mit G91 ist so definiert, dass bei Anwahl einer Nullpunktverschiebung der Korrekturwert additiv zum inkrementell programmierten Wert gefahren wird.

Das Verhalten ist abhängig von der Einstellung im Settingdatum:

SD42440 \$SC_FRAME_OFFSET_INCR_PROG (Nullpunktverschiebungen in Frames)

Wert	Bedeutung
1	Nullpunktverschiebung wird bei FRAME und inkrementeller Programmierung einer Achse herausgefahren (= Standardeinstellung).
0	Nur der programmierte Weg wird gefahren.

Konsistenz

Es liegt in der alleinigen Verantwortung des Anwenders beim Schreiben, Lesen und Aktivieren von Frames z. B. mittels Kanalkoordinierung, ein konsistentes Verhalten innerhalb der Kanäle zu erzielen. Eine kanalübergreifende Aktivierung von Frames ist nicht möglich.

11.2 Achsen

11.2.1 Übersicht

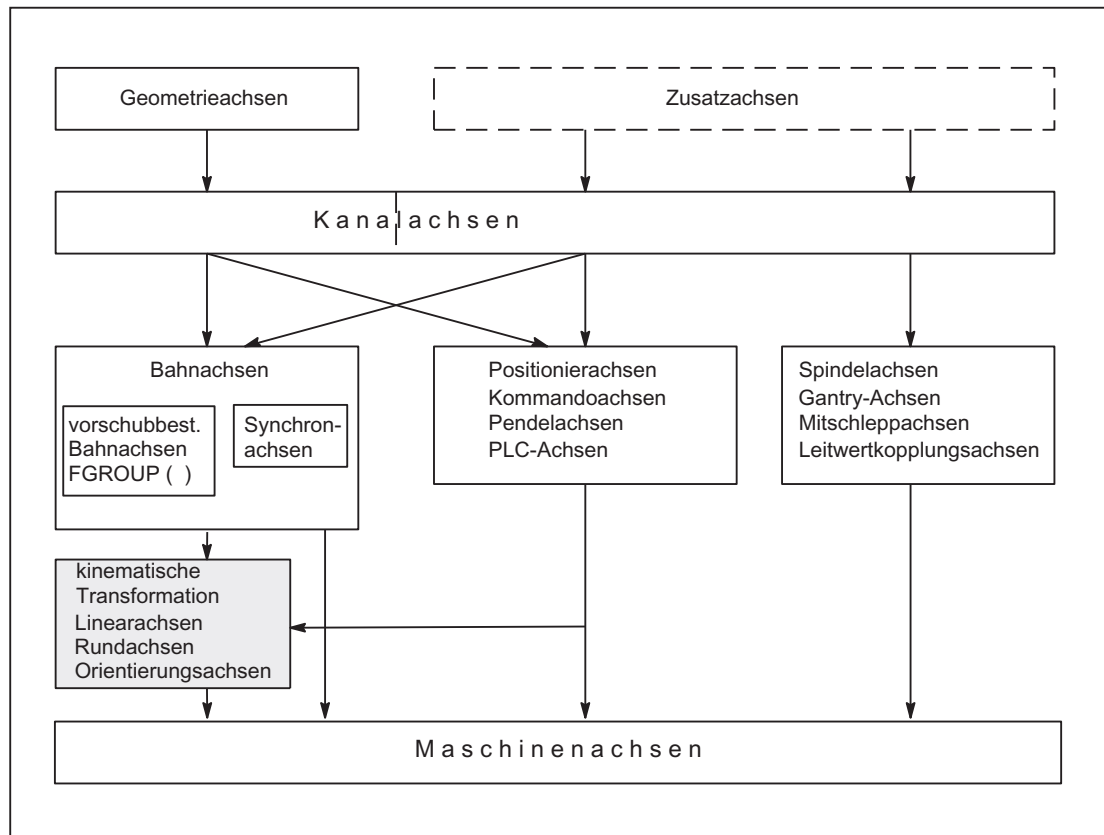


Bild 11-2 Zusammenhang zwischen Geometrie-, Zusatz- und Maschinenachsen

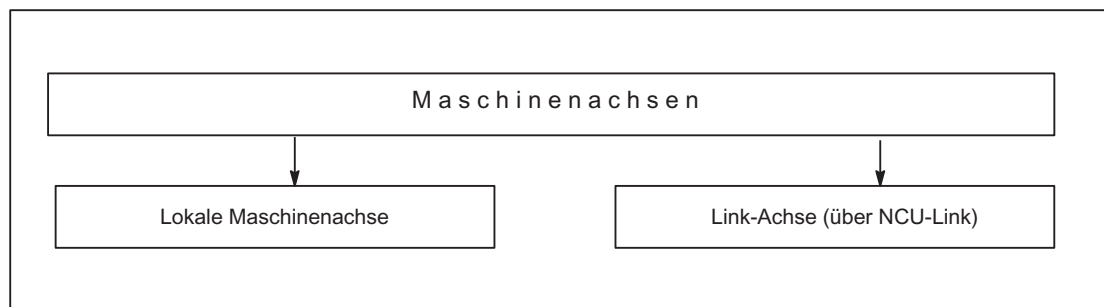


Bild 11-3 Lokale und externe Maschinenachsen (Link-Achsen)

11.2.2 Maschinenachsen

Bedeutung

Maschinenachsen sind die real an der (Werkzeug-)Maschine vorhandenen Achsen.

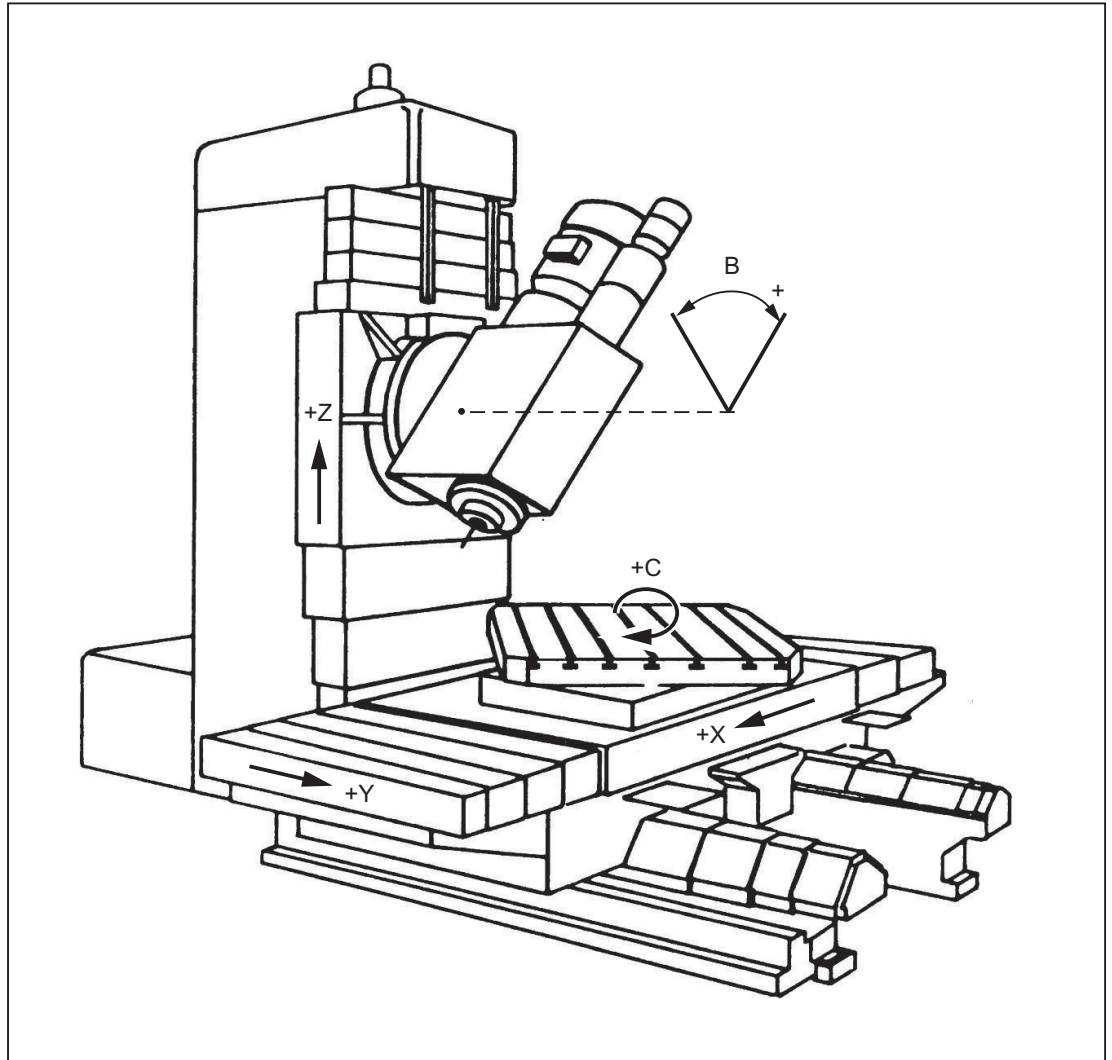


Bild 11-4 Maschinenachsen X, Y, Z, B, S einer kartesischen Maschine

Anwendung

Maschinenachsen können sein:

- Geometrieachsen X, Y, Z
- Orientierungsachsen A, B, C
- Laderachsen
- Werkzeugrevolver

11.2 Achsen

- Achsen für Werkzeugmagazin
- Achsen für Werkzeugwechsler
- Pinole
- Achsen für Palettenwechsler
- etc.

11.2.3 Kanalachsen

Bedeutung

Jede Geometrieachse und jede Zusatzachse wird einem Kanal zugewiesen. Geometrieachsen und Zusatzachsen werden immer in "ihrem" Kanal verfahren.

11.2.4 Geometrieachsen

Bedeutung

Die drei Geometrieachsen bilden immer ein fiktives rechtwinkliges Koordinatensystem.

Durch Verwendung von FRAMES (Verschiebung, Drehung, Skalierung, Spiegelung) können Geometrieachsen des Werkstückkoordinatensystems (WKS) auf das BKS abgebildet werden.

Anwendung

Geometrieachsen werden für Programmierung der Werkstückgeometrie (der Kontur) verwendet.

Die Ebenenanwahl G17, G18 und G19 (DIN 66217) bezieht sich immer auf die drei Geometrieachsen. Deshalb ist es vorteilhaft, die drei Geometrieachsen X, Y und Z zu nennen.

11.2.5 Zusatzachsen

Bedeutung

Bei Zusatzachsen ist, im Gegensatz zu Geometrieachsen, kein geometrischer Zusammenhang zwischen den Achsen definiert.

Hinweis

Geometrieachsen haben einen exakt definierten Zusammenhang in Form eines rechtwinkligen Koordinatensystems.

Zusatzachsen sind Bestandteile des Basiskoordinatensystems (BKS). Durch Verwendung von FRAMES (Verschiebung, Skalierung, Spiegelung) können Zusatzachsen des Werkstückkoordinatensystems (WKS) auf das BKS abgebildet werden.

Anwendung

Typische Zusatzachsen sind:

- Rundachsen
- Werkzeugmagazinachsen
- Werkzeugrevolverachsen
- Laderachsen

11.2.6 Bahnachsen

Bedeutung

Bahnachsen zeichnen sich dadurch aus, dass sie gemeinsam interpoliert werden (alle Bahnachsen eines Kanals haben gemeinsam einen Bahninterpolator).

Alle Bahnachsen eines Kanals haben eine gemeinsame Beschleunigungsphase, eine Konstantfahrphase und eine Verzögerungsphase.

Der unter der Adresse `F` programmierte Vorschub (Bahnvorschub) gilt für alle im Satz programmierten Bahnachsen, mit folgenden Ausnahmen:

- Es wurde eine Achse programmiert, die mit der Anweisung `FGROUP` als nicht bahngeschwindigkeitsbestimmend festgelegt wurde.
- Achsen, die mit den Anweisungen `POS` oder `POSA` programmiert wurden, besitzen einen eigenen Vorschub (Achsinterpolator).

Anwendung

Bahnachsen dienen zum Bearbeiten des Werkstücks mit der programmierten Kontur.

11.2.7 Positionierachsen

Bedeutung

Positionierachsen zeichnen sich dadurch aus, dass sie getrennt interpoliert werden (jede Positionierachse hat einen eigenen Achsinterpolator). Jede Positionierachse hat einen eigenen Vorschub und eine eigene Beschleunigungskennlinie. Positionierachsen können zusätzlich zu Bahnachsen (auch im gleichen Satz) programmiert werden. Die Interpolation der Bahnachsen (Bahninterpolator) wird durch Positionierachsen nicht beeinflusst. Bahnachsen und die einzelnen Positionierachsen erreichen nicht notwendigerweise gleichzeitig ihre Satzendpunkte.

Durch die Anweisungen `POS` und `POSA` werden Positionierachsen programmiert und das Satzwechselkriterium festgelegt:

- `POS`
Der Satzwechsel erfolgt, wenn Bahnachsen und Positionierachsen ihre Satzendpunkte erreicht haben.
- `POSA`
Der Satzwechsel erfolgt, wenn Bahnachsen ihre Satzendpunkte erreicht haben.
Positionierachsen laufen über Satzgrenzen hinweg bis zu ihrem Satzendpunkt weiter.

Konkurrierende Positionierachsen unterscheiden sich von Positionierachsen durch folgende Punkte:

- Sie bekommen ihre Satzendpunkte ausschließlich von der PLC.
- Sie können zu jedem beliebigen Zeitpunkt (nicht an Satzgrenzen) gestartet werden.
- Sie beeinflussen das laufende Teileprogramm in ihrer Bearbeitung nicht.

Anwendung

Typische Positionierachsen sind:

- Lader für Werkstücktransport
- Werkzeugmagazin / -revolver

Verweis

Für weitere Informationen siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)" und "S1: Spindeln (Seite 1295)".

Literatur:

- Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2)
- Funktionshandbuch Sonderfunktionen; Gantry-Achsen (G1)
- Funktionshandbuch Sonderfunktionen; Achskopplungen und ESR (M3)
- Funktionshandbuch Erweiterungsfunktionen; Pendeln (P5)
- Funktionshandbuch Synchronaktionen

11.2.8 Hauptlaufachsen

Bedeutung

Als Hauptlaufachse wird eine Achse bezeichnet, die vom Hauptlauf interpoliert wird.

Diese Interpolation kann folgendermaßen gestartet werden:

- Aus Synchronaktionen
(als Kommandoachsen auf Grund eines Ereignisses über satzbezogene, modale oder statische Synchronaktionen)
- Vom PLC über spezielle Funktionsbausteine im PLC-Grundprogramm
(als konkurrierende Positionierachse oder auch PLC-Achse genannt)
- Durch Settingdatum oder aus dem Teileprogramm
(als asynchrone oder satzsynchrone Pendelachse)

Beeinflussung

Eine vom Hauptlauf interpolierte Achse reagiert bezüglich:

- NC-Stop
- Alarmbehandlung
- Programmbeeinflussungen
- Programmende
- RESET

Hinweis

Das Verhalten am Programmende ist unterschiedlich. Die Achsbewegung muss nicht immer am Programmende abgeschlossen sein und kann damit auch über das Programmende erfolgen.

Anwendung

Bestimmte Achsen können im Hauptlauf von dem durch den NC-Programmablauf getriggerten Kanalverhalten entkoppelt und vom PLC aus kontrolliert werden. Auch diese Achsen werden im Hauptlauf interpoliert und verhalten sich unabhängig vom Kanal- und Programmablauf.

Eine vom PLC kontrollierte Achse kann dann autark von der NC beeinflusst werden. Dies betrifft folgende Aktionen:

- Ablauf der Achse abbrechen (entspricht Restweg löschen)
- Achse stoppen bzw. unterbrechen
- Achse weiterfahren (Bewegungsablauf fortsetzen)
- Achse in den Grundzustand zurücksetzen

11.2.9 Synchronachsen

Bedeutung

Synchronachsen sind Teil der Bahnachsen, die nicht zur Berechnung der Bahngeschwindigkeit herangezogen werden. Sie werden gemeinsam mit Bahnachsen interpoliert (alle Bahnachsen und Synchronachsen eines Kanals haben einen gemeinsamen Bahninterpolator).

Alle Bahnachsen und alle Synchronachsen eines Kanals haben eine gemeinsame Beschleunigungsphase, eine Konstantfahrphase und eine Verzögerungsphase.

Der unter der Adresse **F** programmierte Vorschub (Bahnvorschub) gilt für alle im Satz programmierten Bahnachsen, jedoch nicht für die Synchronachsen.

Synchronachsen benötigen für ihren programmierten Weg die gleiche Zeit wie die Bahnachsen.

Befehl FGROUP

Mit dem Befehl **FGROUP** wird festgelegt, ob die Achse eine vorschubbestimmende **Bahnachse** (wird zur Berechnung der Bahngeschwindigkeit herangezogen) oder eine **Synchronachse** (wird zur Berechnung der Bahngeschwindigkeit nicht herangezogen) ist.

Beispiel

Programmcode	Kommentar
N05 G00 G94 G90 M3 S1000 X0 Y0 Z0	;
N10 FGROUP (X,Y)	; Achsen X und Y sind Bahnachsen Achse Z ist Synchronachse
N20 G01 X100 Y100 F1000	; progr. Vorschub 1000 mm/min Vorschub der Achse X = 707 mm/min Vorschub der Achse Y = 707 mm/min
N30 FGROUP (X)	; Achse X ist eine Bahnachse Achse Y ist eine Synchronachse
N20 X200 Y150	; progr. Vorschub 1000 mm/min Vorschub der Achse X = 1000 mm/min Vorschub der Achse Y stellt sich auf 500 mm/min ein, da nur der halbe Weg zu fahren ist.

Hinweis

Beim Befehl **FGROUP** ist der Kanalachsname zu verwenden.

Dies wird definiert durch das Maschinendatum:

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB (Kanalachsname im Kanal)

Anwendung

Bei Schraubenlinieninterpolation (Helixinterpolation) kann durch `FGROUP` wahlweise eingestellt werden:

- Ob der programmierte Vorschub auf der Bahn gelten soll
(alle 3 programmierten Achsen sind Bahnachsen)
- Ob der programmierte Vorschub auf dem Kreis gelten soll
(2 Achsen sind Bahnachsen und die Zustellachse ist eine Synchronachse)

11.2.10 Achskonfiguration

Zuordnung von Geometrie-, Zusatz-, Kanal- und Maschinenachsen und Antrieben

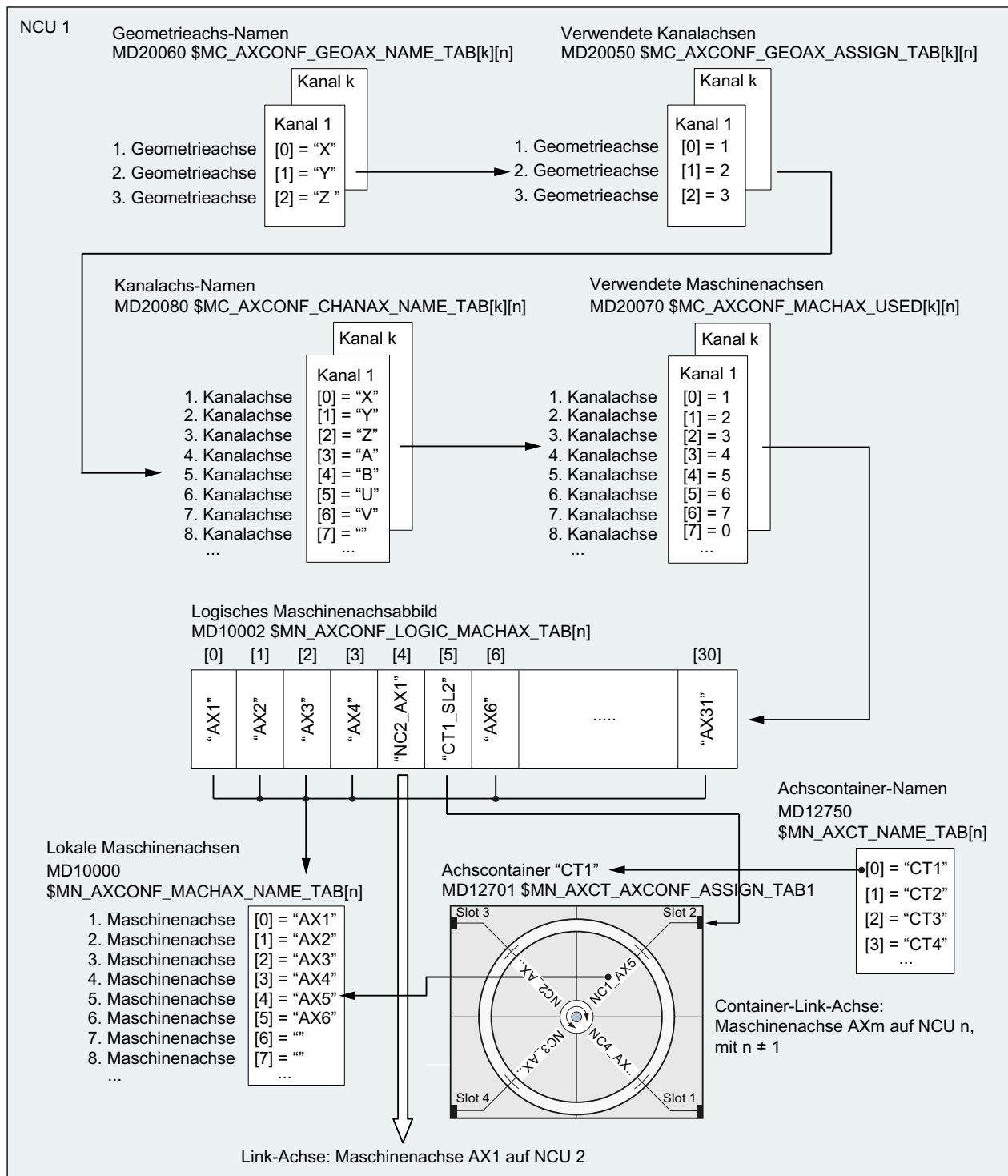


Bild 11-5 Achszuordnung

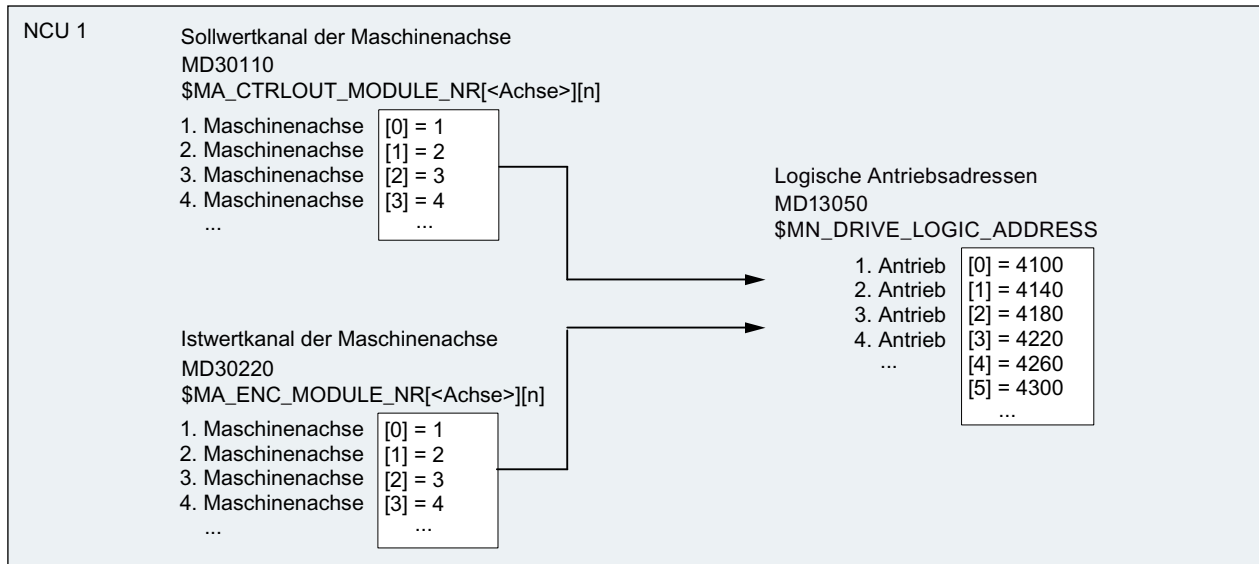


Bild 11-6 Antriebszuordnung

Randbedingungen

- Führende Nullen bei anwenderdefinierten Achsnamen werden ignoriert:
MD10000 `$MN_AXCONF_MACHAX_NAME_TAB[0] = X01` entspricht X1
- Die Zuordnung der Geometrieachsen zu den Kanalachsen muss lückenlos und in aufsteigender Reihenfolge erfolgen.
- Alle Kanalachsen, die keine Geometrieachsen sind, sind Zusatzachsen.

Kanalachslücken

Im Normalfall muss jeder im Maschinendatum MD20080 `$MC_AXCONF_CHANAX_NAME_TAB` definierten Kanalachse über MD20070 `$MC_AXCONF_MACHAX_USED` eine Maschinenachse zugeordnet werden.

Zur Vereinfachung der Inbetriebnahme von Maschinenbaureihen mit unterschiedlicher Anzahl von Maschinenachsen, dürfen auch Kanalachsen definiert sein, denen keine Maschinenachse zugeordnet ist. In der Reihenfolge der Aufzählung der Kanalachsen können dadurch Lücken entstehen.

Die Zulässigkeit von Kanalachslücken muss explizit freigegeben werden:

MD11640 `$MN_ENABLE_CHAN_AX_GAP = 1`

Ohne Freigabe beendet ein Wert von 0 im Maschinendatum MD20070 `$MC_AXCONF_MACHAX_USED` die Zuweisung weiterer Kanalachsen zu Maschinenachsen.

Beispiel

Der Kanalachse B ist keine Maschinenachse zugeordnet.

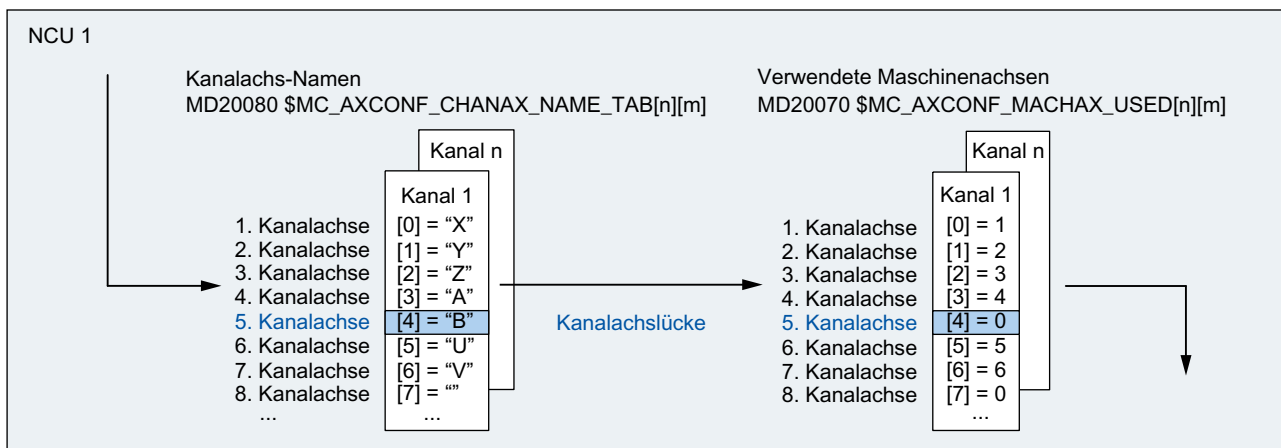


Bild 11-7 Achskonfiguration mit Kanalachslücke (Ausschnitt)

Randbedingungen

- Kanalachsen ohne zugeordnete Maschinenachsen (Kanalachslücken) werden bezüglich Anzahl und Indizierung der Kanalachsen wie normale Kanalachsen mit zugeordneten Maschinenachsen behandelt.
- Wird eine Kanalachse ohne zugeordnete Maschinenachse (Kanalachslücke) als Geometrieachse definiert, wird dies ohne Alarm abgewiesen.

11.2.11 Link-Achsen**Bedeutung**

Eine Link-Achse ist eine Maschinenachse, die nicht auf der NCU liegt von der aus sie verfahren wird. Im Maschinendatum für das logische Maschinenachsabbild der verfahrenen NCU wird dazu nicht der Name einer lokalen Maschinenachse eingetragen, sondern der NCU- und der Maschinenachsname der NCU an der sie physikalisch angeschlossen ist.

Beispielhaft soll die Maschinenachse AX1 der NCU2 von NCU1 aus verfahren werden:

- NCU1: MD10002 \$MN_AXCONF_LOGIC_MACHAX_TAB[n] = NC2_AX1

Voraussetzung

Als Voraussetzung für die Verwendung von Link-Achsen müssen die beteiligten NCUs per Link-Kommunikation verbunden sein. Die Funktionen Link-Achsen und Link-Kommunikation sind ausführlich beschrieben in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Mehrere Bedientafeln an mehreren NCUs, Dezentrale Systeme (B3)





11.3 Nullpunkte und Referenzpunkte

11.3.1 Bezugspunkte im Arbeitsraum

Nullpunkte und Referenzpunkte

Aus den Koordinatenachsen und den konstruktiven Merkmalen der Maschine ergibt sich deren Nullstellung. Den Nullpunkt des Koordinatensystems erhält man durch Festlegung eines zweckmäßigen Bezugspunktes an der Maschine in ihrer Nullstellung.

Die Lage der Koordinatensysteme (MKS, BKS, BNS, ENS, WKS) wird durch Nullpunkte festgelegt.

Nullpunkte		Referenzpunkte	
	M = Maschinennullpunkt		R = Referenzpunkt
	W = Werkstücknullpunkt		T = Werkzeugträgerbezugspunkt

Maschinennullpunkt M

Mit dem Maschinennullpunkt M wird das Maschinen-Koordinatensystem MKS festgelegt. Auf den Maschinennullpunkt beziehen sich alle anderen Bezugspunkte.

Werkstücknullpunkt W

Der Werkstücknullpunkt W legt das Werkstück-Koordinatensystem in Bezug auf den Maschinennullpunkt M fest. Im Werkstück-Koordinatensystem WKS werden die programmierten Teileprogrammsätze abgefahren.

Referenzpunkt R

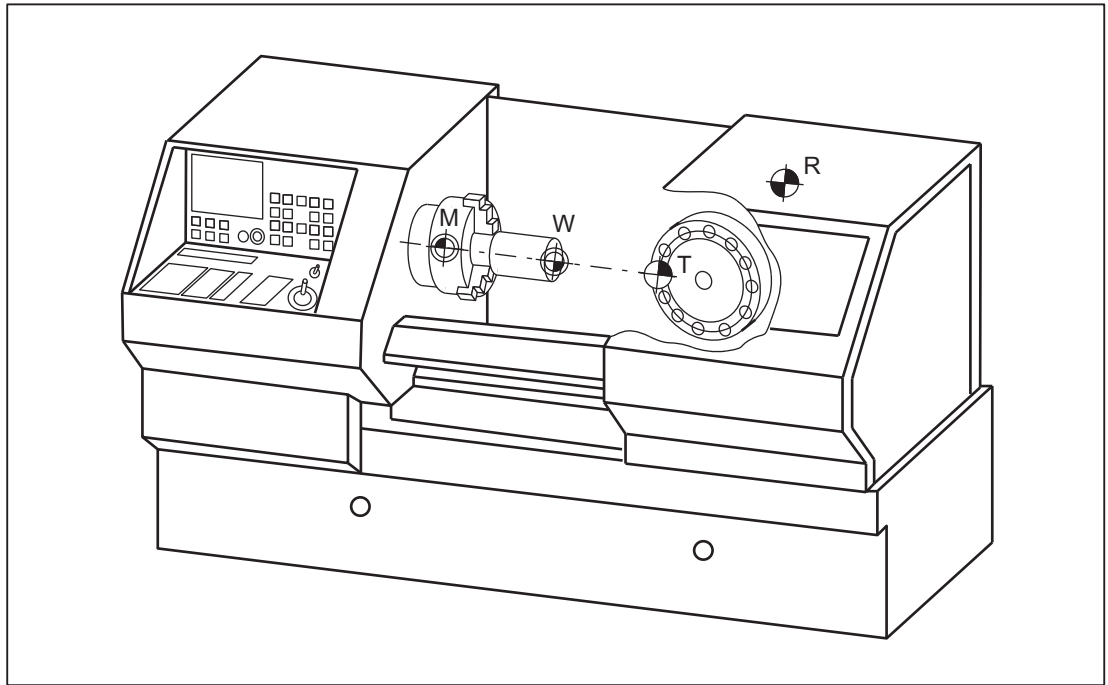
Die Position des Referenzpunktes R wird durch Nockenschalter vorgegeben. Durch ihn wird das Wegmesssystem geeicht.

Bei inkrementellen Messgebern muss der Referenzpunkt nach jedem Einschalten der Steuerung angefahren werden. Erst dann kann die Steuerung mit dem Messsystem arbeiten und alle Positionswerte auf die Koordinatensysteme übertragen.

Werkzeugträgerbezugspunkt T

Der Werkzeugträgerbezugspunkt T befindet sich an der Werkzeughalteraufnahme. Durch Eingabe der Werkzeuglängen berechnet die Steuerung den Abstand der Werkzeugspitze (TCP-Tool Center Position) vom Werkzeugträgerbezugspunkt.

Beispiel: Nullpunkte und Referenzpunkte bei einer Drehmaschine



11.3.2 Lage der Koordinatensysteme und Referenzpunkte

Einschalten der Steuerung

Bei inkrementellen Messgebern muss der Referenzpunkt nach jedem Einschalten der Steuerung angefahren werden, damit die Steuerung alle Positionswerte auf das Koordinatensystem übertragen kann.

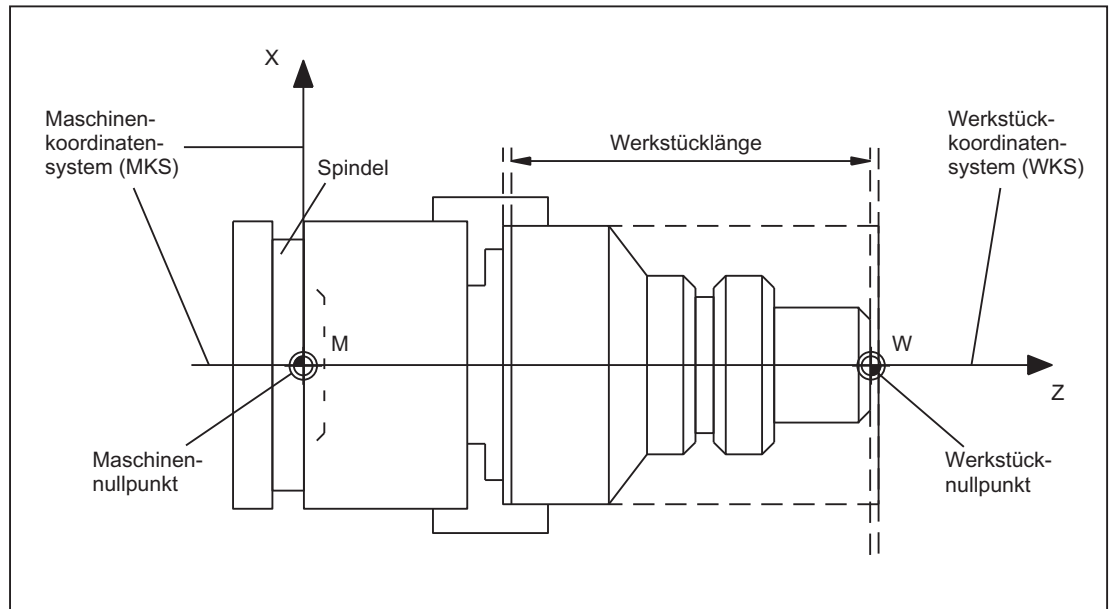


Bild 11-8 Lage der Koordinatensysteme durch Maschinennullpunkt M und Werkstücknullpunkt W

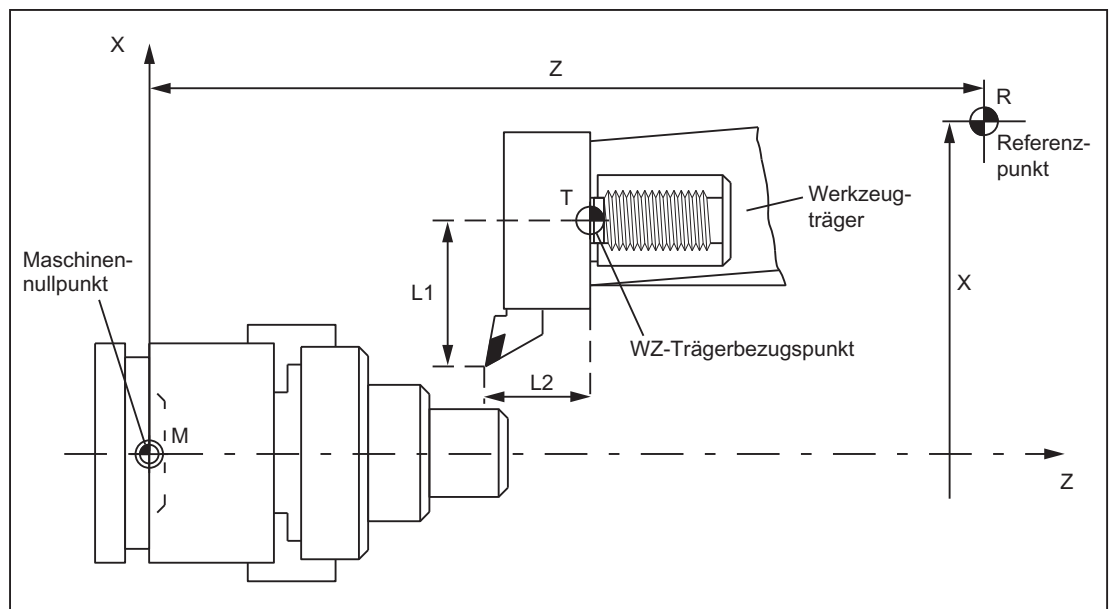


Bild 11-9 Lage des Referenzpunktes zum Maschinennullpunkt

11.4 Koordinatensysteme

11.4.1 Übersicht

Definitionen

Nach DIN 66217 werden bei der Programmierung von Werkzeugmaschinen rechtwinkelige (kartesische) Koordinatensysteme benutzt. Mit der "Rechten-Hand-Regel" können die positiven Richtungen der Koordinatenachsen bestimmt werden.

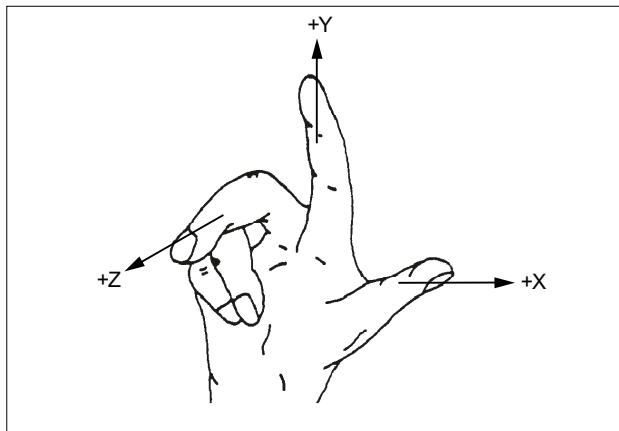


Bild 11-10 Rechte_Hand_Regel

Das Koordinatensystem in dem programmiert wird, wird auf das Werkstück bezogen. Die Programmierung erfolgt unabhängig davon, ob das Werkzeug oder das Werkstück bewegt wird. Bei der Programmierung wird immer davon ausgegangen, dass sich das Werkzeug relativ zum Koordinatensystem des stillstehend gedachten Werkstückes bewegt.

Mit der "Rechten-Hand-Regel" kann auch die positive (rechtsdrehende) Drehrichtung der Rundachsen bestimmt werden. Zeigt der Daumen der rechten Hand in Richtung der positiven Richtung einer Koordinatenachse (Linearachse), zeigen die Finger in Richtung der positiven Drehrichtung der zugehörigen Rundachse.

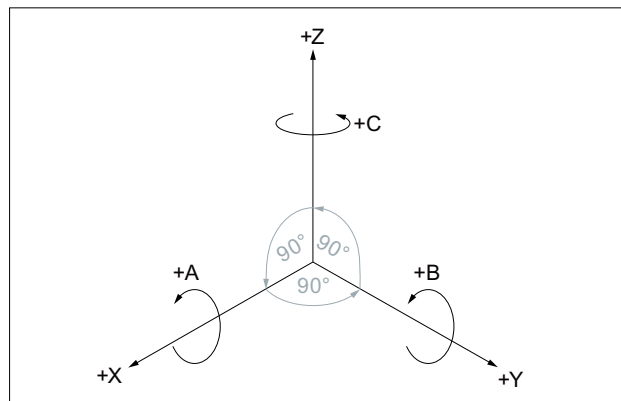


Bild 11-11 Rechtsdrehendes, rechtwinkliges kartesisches Koordinatensystem

Koordinatensysteme

Für eine Werkzeugmaschine sind folgende Koordinatensysteme definiert:

Koordinatensystem	Abkürzung	Bemerkung
Werkstück-KoordinatenSystem	WKS	Im WKS erfolgt die Programmierung der Verfahrensbewegungen der Geometrieachsen für die Bearbeitung des Werkstücks.
Einstellbares NullpunktSystem	ENS	Koordinatentransformation über Frames: WKS \Rightarrow ENS
Basis-NullpunktSystem	BNS	Koordinatentransformation über Frames: ENS \Rightarrow BNS
Basis-KoordinatenSystem	BKS	Koordinatentransformation über Frames: BNS \Rightarrow BKS
Maschinen-KoordinatenSystem	MKS	Koordinatentransformation über Frames und kinematische Transformation: BKS \Rightarrow MKS Die programmierten Verfahrensbewegungen der Geometrieachsen im WKS werden über Frames und kinematische Transformationen auf die Maschinenachsen des MKS abgebildet.

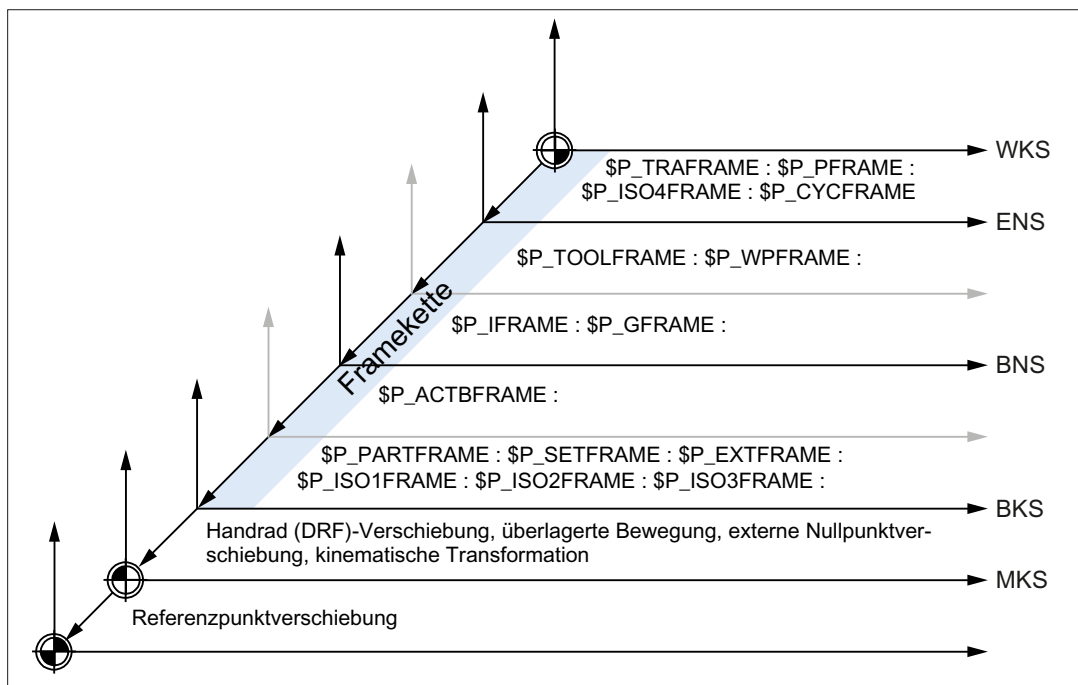


Bild 11-12 Koordinatensysteme, Frames und kinematische Transformationen

11.4.2 Maschinenkoordinatensystem (MKS)

Maschinenkoordinatensystem (MKS)

Das Maschinenkoordinatensystem (MKS) wird aus allen physikalisch vorhandenen Maschinenachsen gebildet.

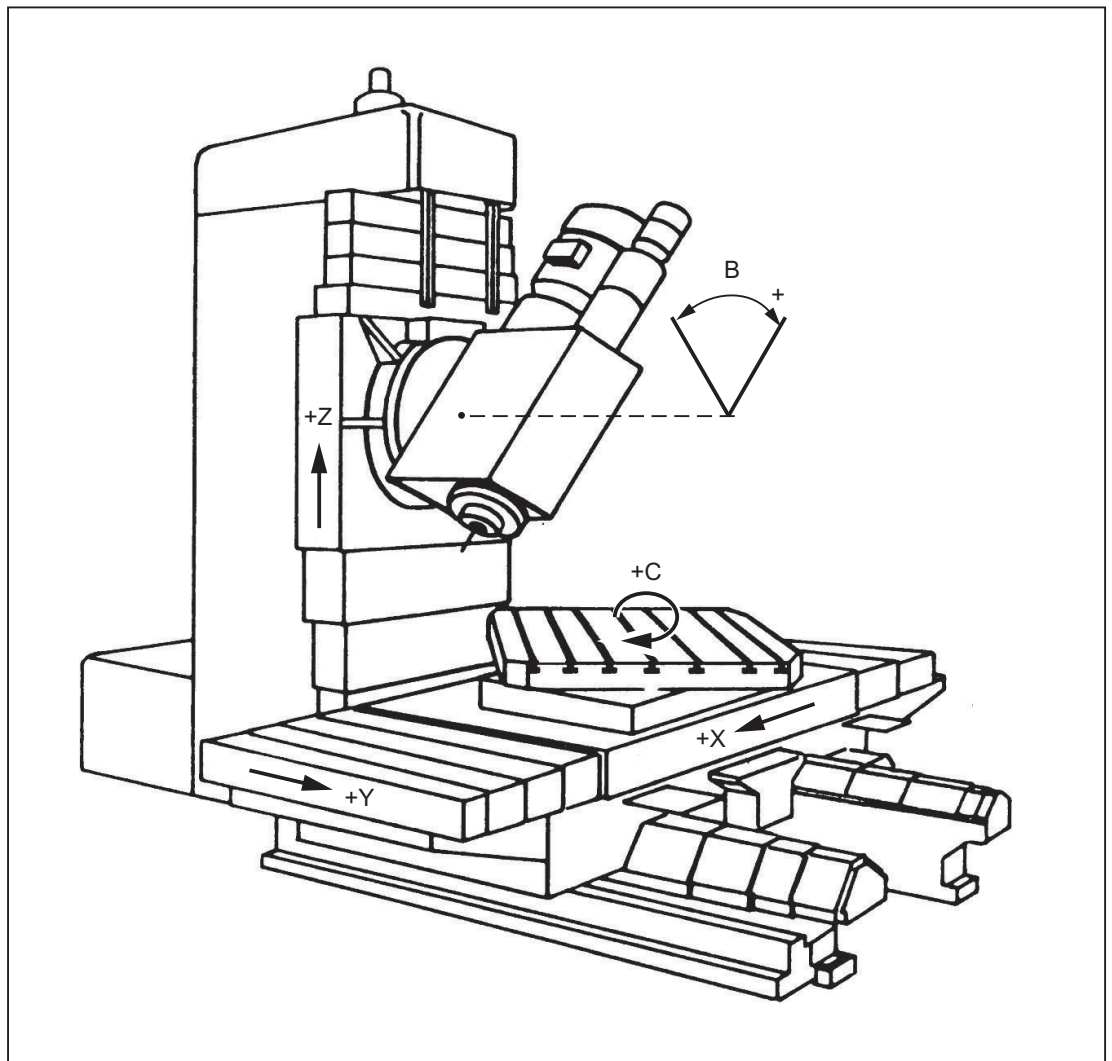


Bild 11-13 MKS mit den Maschinenachsen X, Y, Z, B, C (5-Achs-Fräsmaschine)

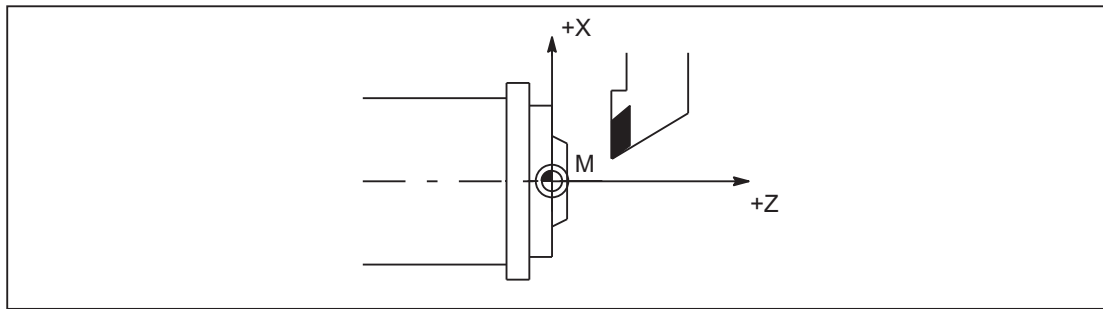


Bild 11-14 MKS mit den Maschinenachsen X, Z (Drehmaschine)

Axiale Preset-Verschiebung

Über die Funktion "Preset-Verschiebung (PRESETON)" kann der Bezugspunkt der Steuerung im Maschinenkoordinatensystem (Maschinennullpunkt) neu gesetzt werden.

VORSICHT

Verlust der Geber-Justage

Nach einer Preset-Verschiebung ist die entsprechende Maschinenachse im Status "nicht referenziert"! Das bedeutet bei Verwendung von Absolutgebern, dass die Geber-Justage verlorengegangen und neu durchgeführt werden muss (z. B. durch Einmessen mit einem Laser-Interferometer). Die Verwendung von PRESETON in Kombination mit Absolutgebern ist deshalb nicht zu empfehlen.

Hinweis

Es wird empfohlen, die Funktion **nur** für Maschinenachsen ohne Referenzpunktpflicht zu verwenden.

Zum Wiederherstellen des ursprünglichen Maschinenkoordinatensystems muss die Maschinenachse z. B. mit G74 (Referenzpunktfahren) erneut referenziert werden.

Bei der Preset-Verschiebung findet keine Bewegung der Maschinenachsen statt.

Literatur

- Programmieranleitung Grundlagen
Kapitel: "Ergänzende Befehle" > "Referenzpunktfahren (G74)"
- Programmierhandbuch Arbeitsvorbereitung
Kapitel: "Koordinatentransformationen (FRAMES)" > "Preset-Verschiebung (PRESETON)"

11.4.2.1 Istwertsetzen mit Verlust des Referenzierstatus (PRESETON)

Funktion

Die Prozedur `PRESETON()` setzt für eine oder mehrere Achsen einen neuen Istwert im Maschinenkoordinatensystem (MKS). Dies entspricht einer Nullpunktverschiebung des MKS der Achse. Die Achse wird dadurch nicht verfahren.

Durch `PRESETON` wird ein Vorlaufstopp mit Synchronisation ausgelöst. Die Istposition wird der Achse erst im Stillstand zugewiesen.

Ist die Achse bei `PRESETON` dem Kanal nicht zugeordnet, ist das weitere Vorgehen abhängig von dem in folgenden Maschinendatum parametrisierten Achstauschverhalten:

MD30552 \$MA_AUTO_GET_TYPE

Referenzierstatus

Durch das Setzen eines neuen Istwertes im Maschinenkoordinatensystem wird der Referenzierstatus der Maschinenachse zurückgesetzt:

DB31, ... DBX60.4 / .5 = 0 (Referenziert / Synchronisiert Messsystem 1 / 2)

Es wird empfohlen, `PRESETON` nur bei Achsen ohne Referenzpunktspflicht zu verwenden.

Zum Wiederherstellen des ursprünglichen Maschinenkoordinatensystems muss das Messsystem der Maschinenachse, z.B. durch Referenzpunktfahren aus dem Teileprogramm (`G74`), erneut referenziert werden.

VORSICHT

Verlust des Referenzierstatus

Durch das Setzen eines neuen Istwertes im Maschinenkoordinatensystem mit `PRESETON` wird der Referenzierstatus der Maschinenachse auf "nicht referenziert / synchronisiert" zurückgesetzt.

Programmierung

Syntax

`PRESETON(<Achse_1>, <Wert_1> [, <Achse_2>, <Wert_2>, ... <Achse_8>, <Wert_8>])`

Bedeutung

<code>PRESETON:</code>	Istwertsetzen mit Verlust des Referenzierstatus
	Vorlaufstopp: ja
	Alleine im Satz: ja
<code><Achse_x>:</code>	Maschinenachsname
	Typ: AXIS
	Wertebereich: im Kanal definierte Maschinenachsnamen
<code><Wert_x>:</code>	Neuer Istwert der Maschinenachse im Maschinenkoordinatensystem (MKS)
	Die Eingabe erfolgt im aktuell gültigen Maßsystem (inch / metrisch)
	Eine aktive Durchmesserprogrammierung (<code>DIAMON</code>) wird berücksichtigt
	Typ: REAL

Systemvariable

\$AC_PRESET

Die achsspezifische Systemvariable \$AC_PRESET liefert den Vektor vom Nullpunkt des aktuellen verschobenen MKS' zum Nullpunkt des ursprünglichen MKS₀ nach dem Referenzieren der Maschinenachse.

$\$AC_PRESET<Achse> = \$AC_PRESET<Achse> + \text{"aktuelle Istposition der Achse im MKS"} - \text{"PRESETON-Istposition"}$

Mit Hilfe der Systemvariablen kann die Nullpunktverschiebung wieder rückgängig gemacht werden:

```
PRESETON(<Achse>, $VA_IM + $AC_PRESET[<Achse>]) ; "Aktueller Istwert  
der Achse im MKS'" + "Verschiebungen"
```

Beispiel

Programmcode

```
N10 G1 X10 F5000
N20 PRESETON(X, $AA_IM[X]+70) ; Istwert = 10 + 70 = 80 =>
                               ; $AC_PRESET = $AC_PRESET - 70
```

Randbedingungen

Achsen bei denen PRESETON nicht angewandt werden darf

- Fahrende Bahnachsen
- Fahrende Positionierachsen
- Fahrende Kommandoachsen im Spindelbetrieb
- Fahrende konkurrierende Positionierachsen (FC18)
- Achsen, die an einer Transformation beteiligt sind
- Achsen, bei denen eine oder mehrere der folgenden Safety Integrated Funktionen aktiv ist:
 - Sichere Software-Endschalter (SE):
MD36901 \$MA_SAFE_FUNCTION_ENABLE[<SafeAxis>], Bit 1 = 1
 - Sichere Software-Nocken, Sichere Nockenspur (SN):
MD36901 \$MA_SAFE_FUNCTION_ENABLE[<SafeAxis>], Bit 8 ... 15 = 1
- Pendelachsen
- Hirth-Achsen
- Gleichlaufachsen eines Gantry-Verbunds
- Achsen für die Referenzpunktfahren aus dem Teileprogramm (G74) aktiv ist
- Slave-Achse einer Drehzahl-/Drehmomentkopplung (Master-Slave)

Geometrieachsen

- PRESETON kann auf eine stehende Geometrieachse angewandt werden, wenn im Kanal nicht gleichzeitig eine weitere Geometrieachse verfährt.
- PRESETON kann auf eine stehende Geometrieachse angewandt werden, auch wenn im Kanal gleichzeitig eine weitere Geometrieachse verfährt, sich diese aber im Zustand "Neutrale Achse" befindet oder als Kommandoachse verfährt.

Programmcode	Kommentar
N10 G0 X0 Y0	; X, Y: Geometrieachsen
N15 RELEASE(Y)	; Neutrale Achse
N20 PRESETON(Y,20)	; Istposition Y im MKS = 20
N30 G0 X40	; Geometrieachse X verfährt
N40 M30	

Beispiel: Eine andere Geometrieachse (X) verfährt gleichzeitig im Zustand "Neutrale Achse"

Programmcode	Kommentar
N10 G0 X0 Y0	; X, Y: Geometrieachsen
N20 POS[X]=40 FA[X]=1000	; Kommandoachse X
N30 DO PRESETON(Y,20)	; Istposition Y im MKS = 20
N40 M30	

Beispiel: Geometrieachse (X) verfährt gleichzeitig als **Kommandoachse**

PLC-kontrollierte Achsen

PRESETON kann auf PLC-kontrollierte Achsen entsprechend ihres aktuellen Typs angewandt werden.

Spindelzustände

In der nachfolgenden Tabelle sind die Reaktionen aufgezeigt, die erfolgen, wenn PRESETON in einer Synchronaktion auf eine Spindel angewandt wird:

PRESETON im NC-Programm			
Spindelbetriebsart	Verfahrstatus	dem NC-Programm zugeordnet	Hauptlaufachse
Drehzahlsteuerbetrieb	in Bewegung	Alarm 22324	Alarm 22324
	steht	+	+
Positionierbetrieb SPOS	Bewegung	-	+
	steht	+	+
Positionieren über Satzgrenzen SPOSA	in Bewegung	Alarm 10610	-
Achsbetrieb	in Bewegung	-	+
	steht	+	+
+: möglich			
-: nicht möglich			

Achskopplungen

- Leitachsen: Die durch `PRESETON` verursachte sprungförmige Änderung der Leitachseposition wird in den Folgeachsen nicht herausgefahren. Die Kopplung bleibt unverändert erhalten.
- Folgeachsen: Durch `PRESETON` wird nur der überlagerte Positionsanteil der Folgeachse beeinflusst.

Gantry-Verbund

Wird `PRESETON` auf die Führungsachse eines Gantry-Verbunds angewandt, wird die Nullpunktverschiebung auch in allen Gleichlaufachsen des Gantry-Verbunds durchgeführt.

Teilungsachsen

`PRESETON` kann auf Teilungsachsen angewandt werden.

Software-Endschalter, Arbeitsfeldbegrenzung, Schutzbereiche

Liegt nach einer Nullpunktverschiebung durch `PRESETON` die Achsposition außerhalb einer der genannten Begrenzungen, wird erst dann ein Alarm angezeigt, wenn versucht wird die Achse zu verfahren.

Satzsuchlauf mit Berechnung

`PRESETON`-Befehle werden während des Satzsuchlaufs aufgesammelt und mit NC-Start zum Fortsetzen des NC-Programms ausgeführt.

Positionsabhängige NC/PLC-Nahtstellensignale

Der Status der positionsabhängigen NC/PLC-Nahtstellensignale wird aufgrund der neuen Istposition neu bestimmt.

Beispiel: Festpunktpositionen

- Parametrisierte Festpunktpositionen: MD30600 \$MA_FIX_POINT_POS[0...3] = <Festpunktposition 1...4>
- NC/PLC-Nahtstellensignale: DB31, ... DBX75.3 ... 5 (JOG Festpunkt anfahren: erreicht)

Steht die Achse innerhalb der Genauhalttoleranz auf einer Festpunktposition, wird das zugehörige NC/PLC-Nahtstellensignal gesetzt. Mit dem Setzen des Istwertes durch `PRESETON` auf einen anderen Wert ausserhalb der Genauhalttoleranz um die Festpunktposition, wird das Nahtstellensignal zurückgesetzt.

DRF-Verschiebung

Durch `PRESETON` wird eine DRF-Verschiebung der Achse gelöscht.

Überlagerte Bewegung \$AA_OFF

Eine überlagerte Bewegung aus einer Synchronaktion mit \$AA_OFF wird durch `PRESETON` nicht beeinflusst.

Online-Werkzeugkorrektur FTOC

Eine aktive Online-Werkzeugkorrektur aus einer Synchronaktion mit FTOC bleibt auch nach `PRESETON` weiter aktiv.

Achsspezifische Kompensationen

Achsspezifische Kompensationen bleiben nach `PRESETON` weiter aktiv.

Betriebsart JOG

`PRESETON` darf nur auf eine stehende Achsen angewandt werden.

Betriebsart JOG, Maschinenfunktion REF

`PRESETON` darf **nicht** angewandt werden..

11.4.2.2 Istwertsetzen ohne Verlust des Referenzierstatus (PRESETONS)**Funktion**

Die Prozedur `PRESETONS ()` setzt für eine oder mehrere Achsen einen neuen Istwert im Maschinenkoordinatensystem (MKS). Dies entspricht einer Nullpunktverschiebung des MKS der Achse. Die Achse wird dadurch nicht verfahren.

Durch `PRESETONS` wird ein Vorlaufstop mit Synchronisation ausgelöst. Die Istposition wird der Achse erst im Stillstand zugewiesen.

Ist die Achse bei `PRESETONS` dem Kanal nicht zugeordnet, ist das weitere Vorgehen abhängig von der achsspezifischen Projektierung des Achstauschverhaltens:

MD30552 \$MA_AUTO_GET_TYPE

Referenzierstatus

Durch das Setzen eines neuen Istwertes im Maschinenkoordinatensystem (MKS) mit `PRESETONS` wird der Referenzierstatus der Maschinenachse **nicht** verändert.

Voraussetzungen

- **Gebertyp**

`PRESETONS` ist nur bei folgenden Gebertypen des aktiven Messsystems möglich:

- MD30240 \$MA_ENC_TYPE[<Messsystem>] = 0 (simulierter Geber)
- MD30240 \$MA_ENC_TYPE[<Messsystem>] = 1 (Rohsignalgeber)

- **Referenzier-Mode**

`PRESETONS` ist nur bei folgenden Referenzier-Modi des aktiven Messsystems möglich:

- MD34200 \$MA_ENC_REFP_MODE[<Messsytem>] = 0 (kein Referenzpunktfahren möglich)
- MD34200 \$MA_ENC_REFP_MODE[<Messsytem>] = 1 (Referenzieren von inkrementellen, rotatorischen oder linearen Messsystemen: Nullimpuls auf der Geberspur)

Inbetriebnahme**Achsspezifische Maschinendaten**

Istwertsetzen ohne Verlust des Referenzierstatus (`PRESETONS`) muss achsspezifisch aktiviert werden:

MD30455 \$MA_MISC_FUNCTION_MASK, Bit 9 = 1

Hinweis

PRESETON deaktiviert

Mit dem Aktivieren der Funktion "Istwertsetzens **ohne** Verlust des Referenzierstatus PRESETONS" wird die Funktion "Istwertsetzen **mit** Verlust des Referenzierstatus PRESETON" deaktiviert. Beide Funktionen schließen sich gegenseitig aus.

Programmierung

Syntax

```
PRESETONS(<Achse_1>, <Wert_1> [, <Achse_2>, <Wert_2>, ... <Achse_8>, <Wert_8>])
```

Bedeutung

PRESETONS:	Istwertsetzen ohne Verlust des Referenzierstatus
	Vorlaufstopp: ja
	Alleine im Satz: ja
<Achse_x>:	Maschinenachsname
	Typ: AXIS
	Wertebereich: im Kanal definierte Maschinenachsnamen
<Wert_x>:	Neuer aktueller Istwert der Maschinenachse im Maschinenkoordinatensystem (MKS)
	Der Wert bezieht sich auf das aktive Maßsystem (inch / metrisch)
	Eine aktive Durchmesserprogrammierung (DIAMON) wird berücksichtigt
	Typ: REAL

Systemvariable

\$AC_PRESET

Die achsspezifische Systemvariable \$AC_PRESET liefert den Vektor vom Nullpunkt des aktuellen verschobenen MKS' zum Nullpunkt des ursprünglichen MKS₀ nach dem Referenzieren der Maschinenachse.

$\$AC_PRESET\langle Achse \rangle = \$AC_PRESET\langle Achse \rangle + \text{"aktuelle Istposition der Achse im MKS"} - \text{"PRESETONS-Istposition"}$

Mit Hilfe der Systemvariablen kann die Nullpunktverschiebung wieder rückgängig gemacht werden:

```
PRESETONS(<Achse>, $VA_IM + $AC_PRESET[<Achse>]) ; "Aktueller Istwert der Achse im MKS'" + "Verschiebungen"
```

Beispiel

Nullpunktverschiebung des MKS der Achse X um 70 Einheiten.

Die programmierte Endposition der Achse X (Kommandoachse) wird mit PRESETONS in das neue MKS transformiert.

Programmcode

```
N10 G1 X10 F5000
N20 PRESETONS(X, $AA_IM[X]+70) ; Istwert = 10 + 70 = 80 =>
                               ; $AC_PRESET = $AC_PRESET - 70
```

Randbedingungen

Achsen bei denen PRESETONS nicht angewandt werden darf

- Fahrende Positionierachsen
- Fahrende Kommandoachsen im Spindelbetrieb
- Fahrende konkurrierende Positionierachsen (FC18)
- Achsen, die an einer Transformation beteiligt sind
- Fahrende Bahnachsen
- Pendelachsen
- Achsen, bei denen eine oder mehrere der folgenden Safety Integrated Funktionen aktiv ist:
 - Sichere Software-Endschalter (SE):
MD36901 \$MA_SAFE_FUNCTION_ENABLE[<SafeAxis>], Bit 1 = 1
 - Sichere Software-Nocken, Sichere Nockenspur (SN):
MD36901 \$MA_SAFE_FUNCTION_ENABLE[<SafeAxis>], Bit 8 ... 15 = 1
- Hirth-Achsen
- Gleichlaufachsen eines Gantry-Verbunds
- Achsen für die Referenzpunktfahren aus dem Teileprogramm (G74) aktiv ist
- Slave-Achse einer Drehzahl-/Drehmomentkopplung (Master-Slave)

Geometrieachsen

- PRESETONS kann auf eine stehende Geometrieachse angewandt werden, wenn im Kanal nicht gleichzeitig eine weitere Geometrieachse verfährt.
- PRESETONS kann auf eine stehende Geometrieachse angewandt werden, auch wenn im Kanal gleichzeitig eine weitere Geometrieachse verfährt, sich diese aber im Zustand "Neutrale Achse" befindet oder als Kommandoachse verfährt.
Beispiel: Eine andere Geometrieachse (X) verfährt gleichzeitig im Zustand "**Neutrale Achse**"

Programmcode	Kommentar
N10 G0 X0 Y0	; X, Y: Geometrieachsen
N15 RELEASE(Y) ¹⁾	; Neutrale Achse
N20 PRESETONS(Y,20)	; Istposition Y im MKS = 20
N30 G0 X40	; Geometrieachse X verfährt
N40 M30	
1) Hinweis Bei einer Freigabe der Achse im Aktionsteil der Synchronaktion ist nicht gewährleistet, dass die Freigabe rechtzeitig erfolgt. N20 ID=1 WHEN 20.0 < \$AA_IM[X] DO RELEASE(Y) PRESETONS(Y,20) ; NICHT empfohlen!	

Beispiel: Eine andere Geometrieachse (X) verfährt gleichzeitig als **Kommandoachse**

Programmcode	Kommentar
N10 G0 X0 Y0	; X, Y: Geometrieachsen
N20 POS[X]=40 FA[X]=1000	; Kommandoachse X
N30 PRESETONS(Y,20)	; Istposition Y im MKS = 20
N40 M30	

PLC-kontrollierte Achsen

PRESETONS kann auf PLC-kontrollierte Achsen entsprechend ihres aktuellen Typs angewandt werden.

Spindelzustände

In der nachfolgenden Tabelle sind die Reaktionen aufgezeigt, die erfolgen, wenn PRESETONS in einer Synchronaktion auf eine Spindel angewandt wird:

PRESETONS im NC-Programm			
Spindelbetriebsart	Verfahrstatus	dem NC-Programm zugeordnet	Hauptlaufachse
Drehzahlsteuerbetrieb	in Bewegung	Alarm 22324	Alarm 22324
	steht	+	+
Positionierbetrieb SPOS	Bewegung	-	+
	steht	+	+
Positionieren über Satzgrenzen SPOSA	in Bewegung	Alarm 10610	-

PRESETONS im NC-Programm			
Spindelbetriebsart	Verfahrstatus	dem NC-Programm zugeordnet	Hauptlaufachse
Achsbetrieb	in Bewegung	-	+
	steht	+	+
+: möglich -: nicht möglich			

Achskopplungen

- Leitachsen: Die durch PRESETONS verursachte sprungförmige Änderung der Leitachsposition wird in den Folgeachsen nicht herausgefahren. Die Kopplung bleibt unverändert erhalten.
- Folgeachsen: Durch PRESETONS wird nur der überlagerte Positionsanteil der Folgeachse beeinflusst.

Gantry-Verbund

Wird PRESETONS auf die Führungssachse eines Gantry-Verbunds angewandt, wird die Nullpunktverschiebung auch in allen Gleichlaufachsen des Gantry-Verbunds durchgeführt.

Teilungsachsen

PRESETONS kann auf Teilungsachsen angewandt werden.

Software-Endschalter, Arbeitsfeldbegrenzung, Schutzbereiche

Liegt nach einer Nullpunktverschiebung durch PRESETONS die Achsposition außerhalb einer der genannten Begrenzungen, wird erst dann ein Alarm angezeigt, wenn versucht wird die Achse zu verfahren.

Satzsuchlauf mit Berechnung

PRESETONS-Befehle werden während des Suchlaufs aufgesammelt und mit NC-Start zum Fortsetzen des NC-Programms ausgeführt.

Positionsabhängige NC/PLC-Nahtstellensignale

Der Status der positionsabhängigen NC/PLC-Nahtstellensignale wird aufgrund der neuen Istposition neu bestimmt.

Beispiel: Festpunktpositionen

- Parametrierte Festpunktpositionen: MD30600 \$MA_FIX_POINT_POS[0...3] = <Festpunktposition 1...4>
- NC/PLC-Nahtstellensignale: DB31, ... DBX75.3 ... 5 (JOG Festpunkt anfahren: erreicht)

Steht die Achse innerhalb der Genauhalttoleranz auf einer Festpunktposition, wird das zugehörige NC/PLC-Nahtstellensignal gesetzt. Mit dem Setzen des Istwertes durch PRESETONS auf einen anderen Wert außerhalb der Genauhalttoleranz um die Festpunktposition, wird das Nahtstellensignal zurückgesetzt.

DRF-Verschiebung

Durch PRESETONS wird eine DRF-Verschiebung der Achse gelöscht.

Überlagerte Bewegung \$AA_OFF

Eine überlagerte Bewegung aus einer Synchronaktion mit \$AA_OFF wird durch PRESETONS nicht beeinflusst.

Online-Werkzeugkorrektur FTOC

Eine aktive Online-Werkzeugkorrektur aus einer Synchronaktion mit FTOC bleibt auch nach PRESETONS weiter aktiv.

Achsspezifische Kompensationen

Achsspezifische Kompensationen bleiben nach PRESETONS weiter aktiv.

Betriebsart JOG

PRESETONS darf nur auf eine stehende Achsen angewandt werden.

Betriebsart JOG, Maschinenfunktion REF

PRESETONS darf **nicht** angewandt werden..

Synchronaktionen

Sind in einer Synchronaktion für eine Achse mehrere PRESETONS-Anweisungen enthalten, wird nur die letzte in der Reihenfolge von links nach rechts stehende Anweisung ausgeführt.

Beispiel:

Programmcode

```
N10 ID=1 WHEN TRUE DO PRESETONS(X, 40) PRESETONS(X, 39) PRESETONS(X, 38)
; Gleichbedeutend mit:
N10 ID=1 WHEN TRUE DO PRESETONS(X, 38)
```

11.4.3 Basiskoordinatensystem (BKS)

Basiskoordinatensystem (BKS)

Das Basiskoordinatensystem (BKS) besteht aus drei rechtwinklig angeordneten Achsen (Geometrieachsen), sowie aus weiteren Achsen (Zusatzachsen) ohne geometrischen Zusammenhang.

WZ-Maschinen ohne kinematische Transformation

Das BKS und das MKS fallen immer dann zusammen, wenn das BKS ohne kinematische Transformation (z. B. TRANSMIT / Stirnflächentransformation, 5-Achstransformation und max. drei Maschinenachsen) auf das MKS abgebildet werden kann.

Bei diesen Maschinen können Maschinenachsen und Geometrieachsen den gleichen Namen haben.

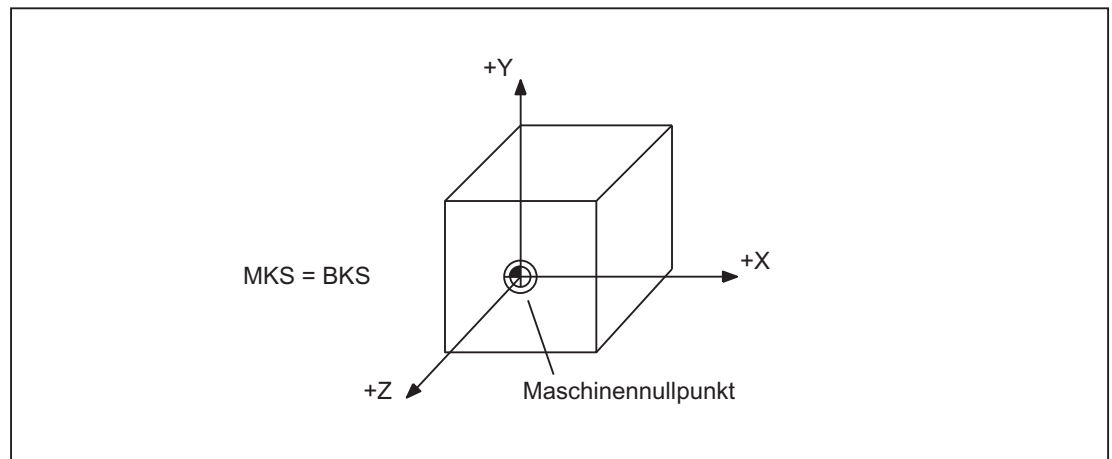


Bild 11-15 MKS=BKS ohne kinematische Transformation

WZ-Maschinen mit kinematischer Transformation

Das BKS und das MKS fallen nicht zusammen, wenn das BKS mit kinematischer Transformation (z. B. TRANSMIT / Stirnflächen-Transformation, 5-Achstransformation oder mehr als drei Achsen) auf das MKS abgebildet wird.

Bei diesen Maschinen müssen Maschinenachsen und Geometrieachsen unterschiedliche Namen haben.

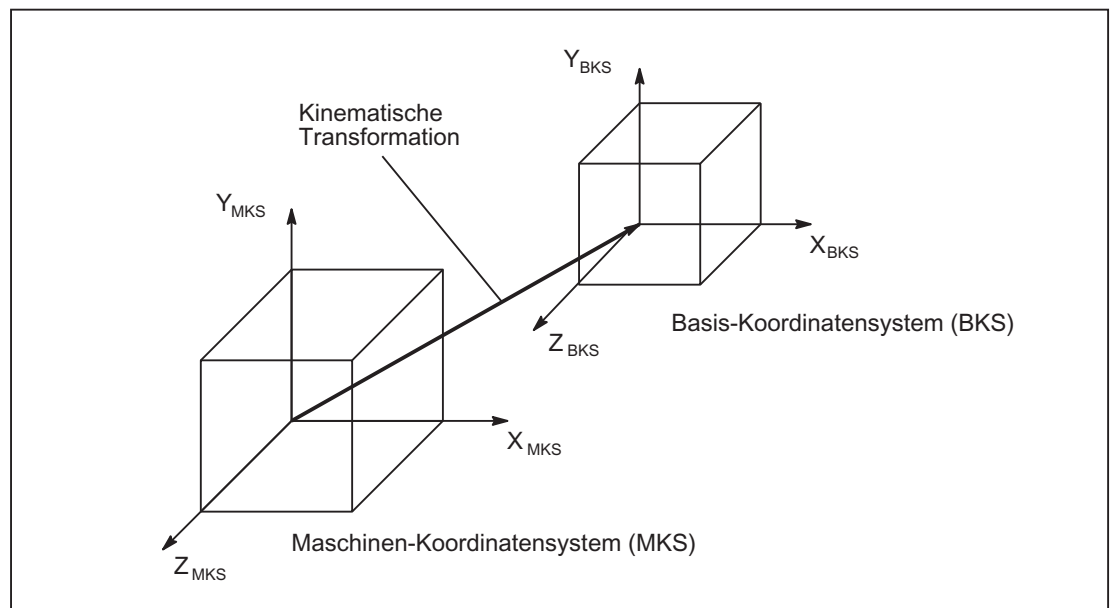


Bild 11-16 Kinematische Transformation zwischen MKS und BKS

Maschinenkinematik

Das Werkstück wird immer in einem zwei- oder dreidimensionalen rechtwinkligen Koordinatensystem (WKS) programmiert. Zur Fertigung dieser Werkstücke werden aber immer häufiger Werkzeugmaschinen mit Rundachsen oder nicht rechtwinklig angeordneten Linearachsen eingesetzt. Zur Abbildung der im WKS programmierten Koordinaten (rechtwinklig) in reale Maschinenachsbewegungen dient die kinematische Transformation.

Literatur:

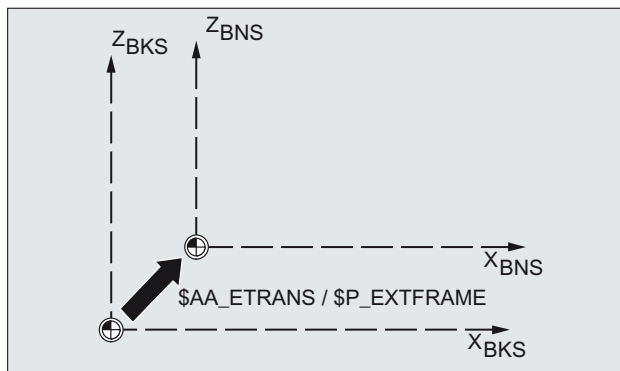
Funktionshandbuch Sonderfunktionen; 3- bis 5-Achs-Transformation (F2)

Funktionshandbuch Erweiterungsfunktionen; Kinematische Transformation (M1)

11.4.4 Additive Korrekturen

11.4.4.1 Externe Nullpunktverschiebungen

Die Externe Nullpunktverschiebung ist eine lineare Verschiebung zwischen Basiskoordinatensystem (BKS) und Basisnullpunktsystem (BNS).



Die Externe Nullpunktverschiebung mittels `$AA_ETRANS` wirkt, abhängig von der Maschinendaten-Parametrierung, auf zwei Arten:

1. Die Systemvariablen `$AA_ETRANS` wirkt nach Aktivierung durch das NC/PLC-Nahtstellensignal direkt als Verschiebungswert
2. Der Wert der Systemvariablen `$AA_ETRANS` wird nach Aktivierung durch das NC/PLC-Nahtstellensignal in die aktiven Systemframes `$P:EXTFRAME` und den Datenhaltungsframe `$P_EXTFR` übernommen. Anschließend wird der aktive Gesamtframe `$P_ACTFRAME` neu berechnet.

Maschinendaten

Im Zusammenhang mit der Systemvariablen `$AA_ETRANS` sind zwei Vorgehensweisen zu unterscheiden, die über das folgende Maschinendatum ausgewählt werden:

MD28082 \$MC_MM_SYSTEM_FRAME_MASK, Bit1 = <Wert>

<Wert>	Bedeutung
0	Funktion: Direktes Schreiben von \$AA_ETRANS[<Achse>] durch PLC, HMI oder NC-Programm. Freigabe zum Herausfahren der Nullpunktverschiebung von \$AA_ETRANS[<Achse>] im nächstmöglichen Verfahrssatz: DB31, ... DBX3.0
1	Funktion: Aktivierung des aktiven Systemframes \$P_EXTFRAME und des Datenhaltungsframes \$P_EXTFR Freigabe zum Herausfahren der Nullpunktverschiebung von \$AA_ETRANS[<Achse>] durch: DB31, ... DBX3.0. Daraufhin erfolgt im Kanal: <ul style="list-style-type: none"> • Stop aller Verfahrbewegungen im Kanal (außer Kommando- und PLC-Achsen) • Vorlaufstop mit anschließendem Reorganisieren (STOPRE) • Grobverschiebung aktiver Frame \$P_EXTFRAME[<Achse>] = \$AA_ETRANS[<Achse>] • Grobverschiebung Datenhaltungsframe \$P_EXTFR[<Achse>] = \$AA_ETRANS[<Achse>] • Neuberechnung des aktiven Gesamtframes \$P_ACTFRAME • Herausfahren der Verschiebung in den programmierten Achsen. • Fortsetzen der unterbrochenen Verfahrbewegung bzw. des NC-Programms

Programmierung

Syntax

\$AA_ETRANS[<Achse>] = <Wert>

Bedeutung

\$AA_ETRANS:	Systemvariable zum Zwischenspeichern der externen Nullpunktverschiebung
<Achse>:	Kanalachse
<Wert>:	Verschiebungswert

NC/PLC-Nahtstellensignal

Aktivierung der Externe Nullpunktverschiebung:

DB31, ... DBX3.0 = 0 → 1 ⇒ \$P_EXTFRAME[<Achse>] = \$P_EXTFR[<Achse>] = \$AA_ETRANS[<Achse>]

Unterdrückung: Externe Nullpunktverschiebung

- Die Befehl SUPA unterdrückt die Externe Nullpunktverschiebung für die Dauer der Bearbeitung des Satzes.
- Aktive Externe Nullpunktverschiebungen werden für die Dauer des Referenzpunktfahrens durch Befehl G74 (Referenzpunktfahren), sowie die entsprechende Bedienung in der Betriebsart Referenzpunktfahren, unterdrücken die Externe Nullpunktverschiebung für die Dauer des Referenzpunktfahrens.

- Bei G74, d. h. Betriebsart "Automatik" oder "MDA", wird die zuvor aktive Externe Nullpunktverschiebung mit der nächsten Verfahrbewegung im Satz wieder automatisch aktiv.
- Nach einem Betriebsartenwechsel aus der Betriebsart Referenzpunktfahren muss für die referenzierten Achsen das NC/PLC-Nahtstellensignal zur erneuten Aktivierung gesetzt werden.

11.4.4.2 DRF-Verschiebung

Die DRF-Verschiebung ermöglicht das Einstellen einer additiven inkrementellen Nullpunktverschiebung für Geometrie- und Zusatzachsen im Basiskoordinatensystem **über Handrad**.

Systemvariable

Die DRF-Verschiebung kann über die achsspezifische Systemvariable gelesen werden:

\$AC_DRF[<Achse>]

Literatur

Funktionshandbuch Erweiterungsfunktionen; manuelles Verfahren und manuelles Verfahren per Handrad (H1); Kapitel: DRF-Verschiebung

11.4.4.3 Überlagerte Bewegungen

Maschinendaten

- Über das achsspezifische Maschinendatum werden spezifische Einstellung für die überlagerte Bewegung einer Achse vorgenommen:
36750 \$MA_AA_OFF_MODE, Bit<n> = <Wert>

Bit	Wert	Bedeutung
0	0	Programmierung von \$AA_OFF mit absolute Position
	1	Programmierung von \$AA_OFF mit inkrementellem Weg
1	0	Die überlagerte Bewegung wird bei Reset abgewählt.
	1	Die überlagerte Bewegung bleibt über Reset hinaus erhalten.
2	0	In der Betriebsart JOG wird eine überlagerte Bewegung aufgrund von \$AA_OFF nicht herausgefahren.
	1	In der Betriebsart JOG wird eine überlagerte Bewegung aufgrund von \$AA_OFF herausgefahren.
3	0	Eine überlagerte Bewegung wird bei NC-Stopp unterbrochen.
	1	Eine überlagerte Bewegung wird bei NC-Stopp nicht unterbrochen.

Systemvariable

- Über die achsspezifische Systemvariable wird einer Achse eine absolute Position bzw. ein inkrementeller Weg vorgegeben. Die absolute Position bzw. der inkrementelle Weg wird dann parallel zu einer eventuell aktiven Verfahrbewegung angefahren bzw. abgefahren.
\$AA_OFF[<Achse>] = <Wert>
- Über die achsspezifische Systemvariable kann der integrierten Wert der überlagerten Bewegung einer Achse gelesen werden.
\$AA_OFF_VAL[<Achse>]

11.4.4.4**Reset-Verhalten**

Das Reset- und Power On-Verhalten des aktiven Frames \$P_EXTFRAME und des Datenhaltungsframes \$P_EXTFR kann über Maschinendaten eingestellt werden:

Maschinendaten

- Das Reset-Verhalten bezüglich des im Kanal aktiven Systemframes der Externen Nullpunktverschiebung \$P_EXTFRAME wird über folgendes Maschinendatum eingestellt:
- MD24006 \$MC_CHSFRAME_RESET_MASK, Bit1 = <Wert>

Wert	Bedeutung
	Das aktive Systemframe der Externen Nullpunktverschiebung ist nach Kanal- / Programmende-Reset:
0	nicht aktiv
1	aktiv

- Das Power On-Verhalten bezüglich des kanalspezifischen Systemframes der Externen Nullpunktverschiebung \$P_EXTFR der Datenhaltung wird über folgendes Maschinendatum eingestellt:
MD24008 \$MC_CHSFRAME_POWERON_MASK, Bit1 = <Wert>

Wert	Bedeutung
	Das Systemframe \$P_EXTFR der Externen Nullpunktverschiebung wird bei Power On
0	nicht gelöscht
1	gelöscht

11.4.5 Basis-Nullpunktsystem (BNS)

Basis-Nullpunktsystem (BNS)

Das Basis-Nullpunktsystem (BNS) ergibt sich aus dem Basis-Koordinatensystem durch die Basisverschiebung.

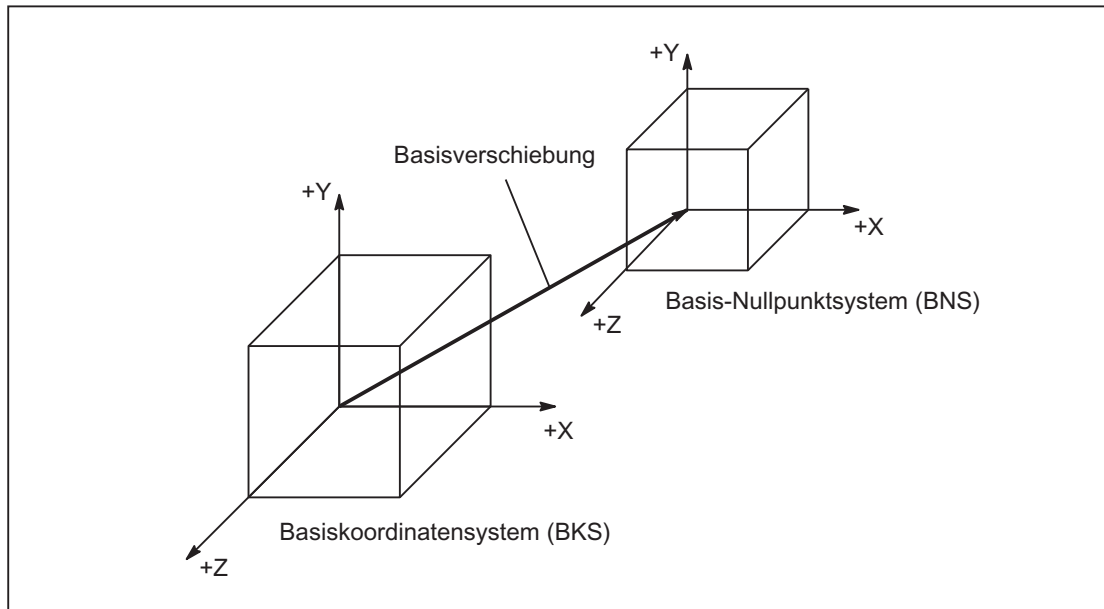


Bild 11-17 Basisverschiebung zwischen BKS und BNS

Basisverschiebung

Die Basisverschiebung beschreibt die Koordinatentransformation zwischen dem BKS und BNS. Mit ihr kann z. B. der Paletten-Nullpunkt festgelegt werden.

Die Basisverschiebung setzt sich zusammen aus:

- Externe Nullpunktverschiebung
- DRF-Verschiebung
- Überlagerte Bewegung

- Verkettete Systemframes
- Verkettete Basisframes

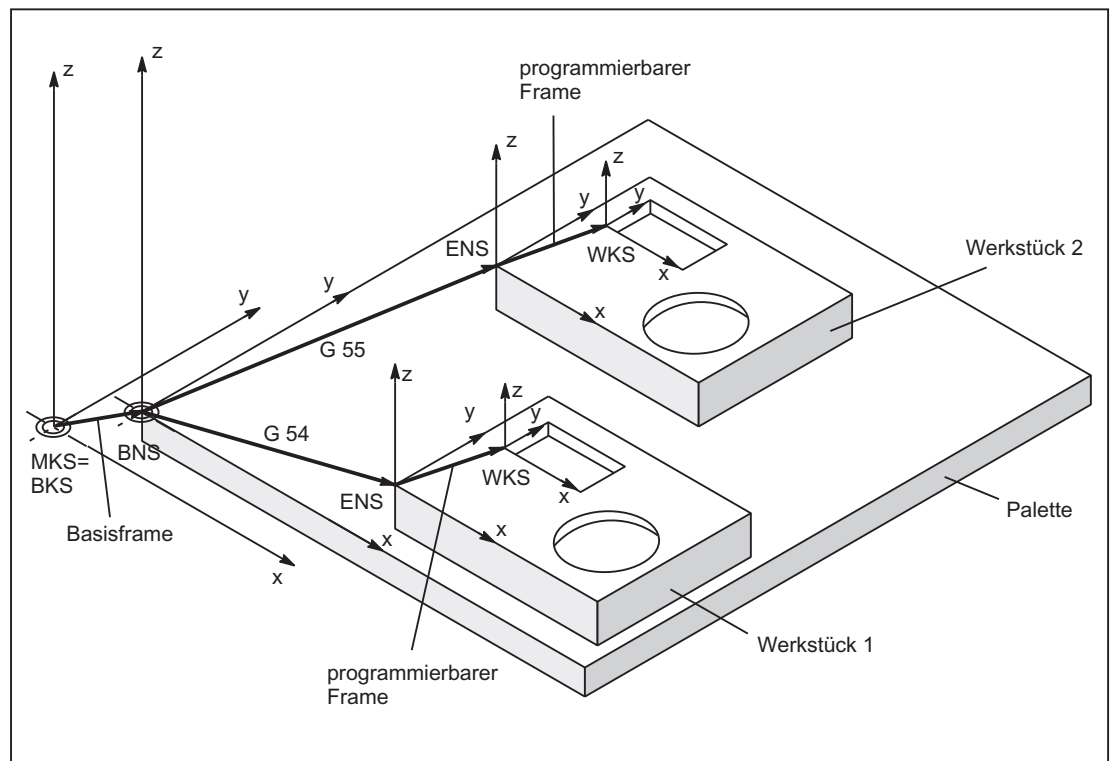


Bild 11-18 Beispiel für Anwendung der Basisverschiebung

Es gilt:

- Der Anwender kann die Basisverschiebung aus dem Teileprogramm, der Bedienung und von der PLC verändern.
- Soll die Basisverschiebung sofort wirksam werden, so kann über PLC mit FC9 ein ASUP gestartet werden, der den entsprechenden G-Befehl ausführt.

Hinweis

Empfehlung an den Maschinenhersteller

Verwenden Sie für eigene Anwendungen Verschiebungen ab der 3. Basisverschiebung.

Die 1. und 2. Basisverschiebung sind für das "Istwertsetzen" und die "Externe Nullpunktverschiebung" vorgesehen.

11.4.6 Einstellbares Nullpunktsystem (ENS)

Einstellbare Nullpunktsystem (ENS)

Das "Einstellbare Nullpunktsystem" (ENS) ist das Werkstückkoordinatensystem WKS mit programmierbarem FRAME (gesehen aus der Perspektive WKS). Der Werkstücknullpunkt wird durch die einstellbaren FRAMES G54...G599 festgelegt.

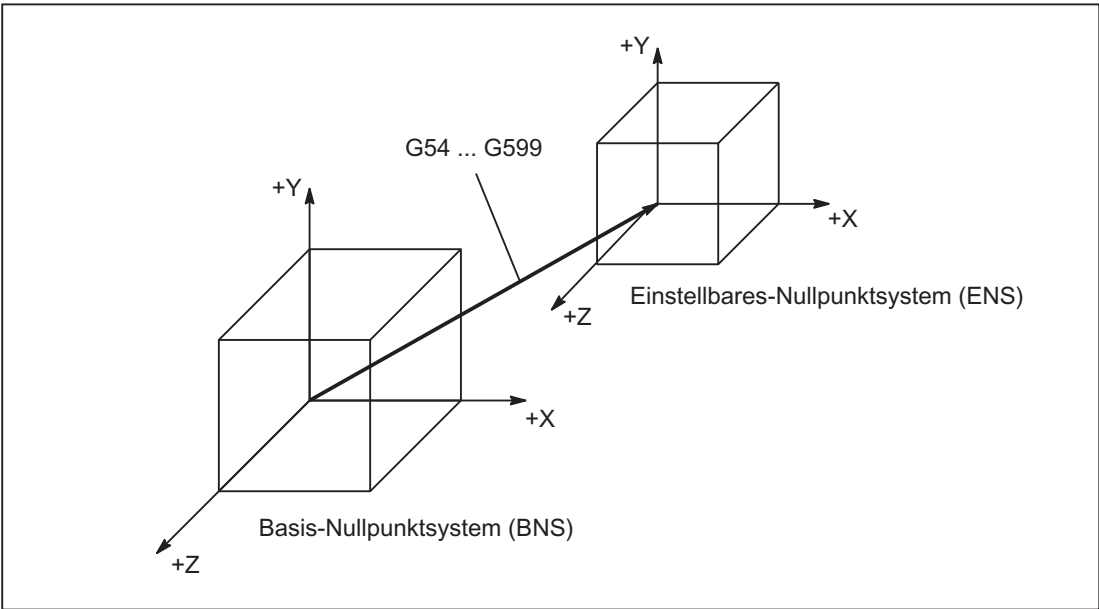


Bild 11-19 Einstellbarer FRAME G54 ... G599 zwischen BNS und ENS

Vom "Einstellbaren-Nullpunktsystem" aus wirken programmierbare Verschiebungen. Alle programmierbaren Verschiebungen beziehen sich auf das "Einstellbare-Nullpunktsystem".

WKS-Istwertanzeige im WKS oder ENS

Auf der HMI-Bedienoberfläche können die Istwerte der Achsen im Maschinenkoordinatensystem (MKS) oder im WKS angezeigt werden. Bei Anzeige im WKS können die Istwerte auch bezogen auf das ENS angezeigt werden. Die entsprechende Parametrierung erfolgt über das Maschinendatum:

MD9424 \$MM_MA_COORDINATE_SYSTEM (Koordinatensystem für Istwertanzeige)

Wert	Bedeutung
0	Istwertanzeige bezogen auf das WKS
1	Istwertanzeige bezogen auf das ENS

Hinweis**Anzeige des aktuellen Koordinatensystems**

Auf der HMI-Bedienoberfläche wird auch bei aktiver "Istwertanzeige bezogen auf das ENS" das WKS als Koordinatensystem angezeigt, auf das sich die Istwertanzeige bezieht.

Beispiel

Istwertanzeige bezogen auf das WKS bzw. ENS

Code (Ausschnitt)	Istwertanzeige: Achse X (WKS)	Istwertanzeige: Achse X (ENS)
N10 X100	100	100
N20 X0	0	0
N30 \$P_PFRAME = CTRANS(X,10)	0	10
N40 X100	100	110

11.4.7 Werkstückkoordinatensystem (WKS)**Werkstückkoordinatensystem WKS**

Das Werkstückkoordinatensystem (WKS) ist die Basis für die Programmierung.

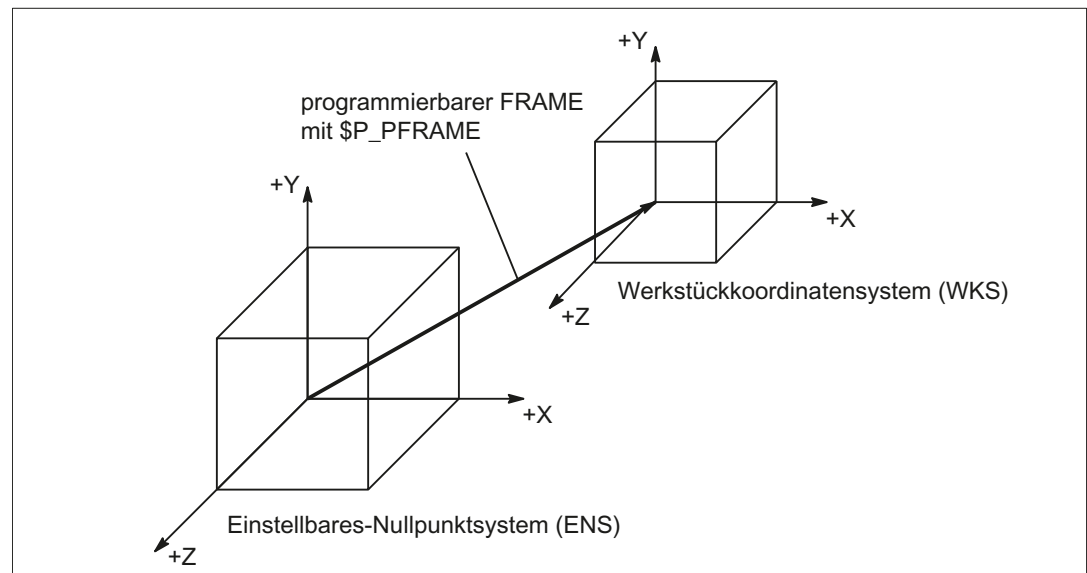


Bild 11-20 Programmierbarer FRAME zwischen ENS und WKS

11.5 Frames

11.5.1 Frame-Arten

Ein Frame ist eine Datenstruktur die Werte für Verschiebung (**TRANS**), Feinverschiebung (**FINE**), Drehung (**ROT**), Spiegelung (**MIRROR**) und Skalierung (**SCALE**) für Achsen enthält.

Bei Aktivierung des Frames wird anhand der Frame-Werte über eine definierte Rechenvorschrift eine statische Koordinatentransformation für die im Frame enthaltenen Achsen durchgeführt.

Achsspezifischer Frame

Ein achsspezifischer Frame enthält die Frame-Werte einer Achse.

Beispielhafte Datenstruktur eines achsspezifischen Frames für Geometrieachse X:

Achse	TRANS	FINE	ROT	MIRROR	SCALE
X	10.0	0.1	0.0	0	1

Kanalspezifischer Frame

Ein kanalspezifischer Frame enthält die Frame-Werte für alle Kanalachsen (Geometrie-, Zusatz- und Maschinenachsen).

Drehungen (**ROT**) werden nur bei Geometrieachsen eingerechnet.

Ein kanalspezifischer Frame wirkt nur in dem Kanal in dem der Frame definiert ist.

Beispielhafte Datenstruktur eines kanalspezifischen Frames:

- Geometrieachsen: X, Y, Z
- Zusatzachsen: A
- Maschinenachsen: AX1

Achse	TRANS	FINE	ROT	MIRROR	SCALE
X	10.0	0.1	0.0	0	1
Y	0.0	0.0	0.0	1	1
Z	0.0	0.0	45.0	0	1
A	2.0	0.1	0.0	0	2
AX1	0.0	0.0	0.0	0	0

Globaler Frame

Ein globaler Frame enthält die Frame-Werte für alle Maschinenachsen.

Ein globaler Frame wirkt in allen Kanälen der NC.

Beispielhafte Datenstruktur eines globalen Frames:

- Maschinenachsen: AX1, ... AX5

Achse	TRANS	FINE	ROT	MIRROR	SCALE
AX1	10.0	0.1	-	0	1
AX2	0.0	0.0	-	1	1
AX3	0.0	0.0	-	0	1
AX4	2.0	0.1	-	0	2
AX5	0.0	0.0	-	1	1

11.5.2 Frame-Komponenten

11.5.2.1 Translation

Programmierung

Die Programmierung der Translation bzw. Grobverschiebung kann über folgende Befehle erfolgen:

- Beispiel Datenhaltungsframes \$P_UIFR
 - Gesamtframe: \$P_UIFR[<n>] = CTRANS (<K1>, <V1>[, <K2>, <V2>] [, <K3>, <V3>])
mit Km = Koordinate x, y oder z und Vm = Verschiebung m
 - Frame-Komponente: \$P_UIFR[<n>, <K>, TR] = <V>
mit K = Koordinate x, y oder z und V = Verschiebung
- Beispiel Programmierbarer Frame
 - TRANS <K1> <V1> [, <K2> <V2>] [, <K3> <V3>]
mit Km = Koordinate x, y oder z und Vm = Verschiebung m

Programmbeispiele:

Programmcode	Bemerkung
\$P_UIFR[1] = CTRANS (X,10,Y,10)	Gesamtframe
\$P_UIFR[1,X,TR] = 10	Frame-Komponente
TRANS X=10 Y=10	Programmierbarer Frame

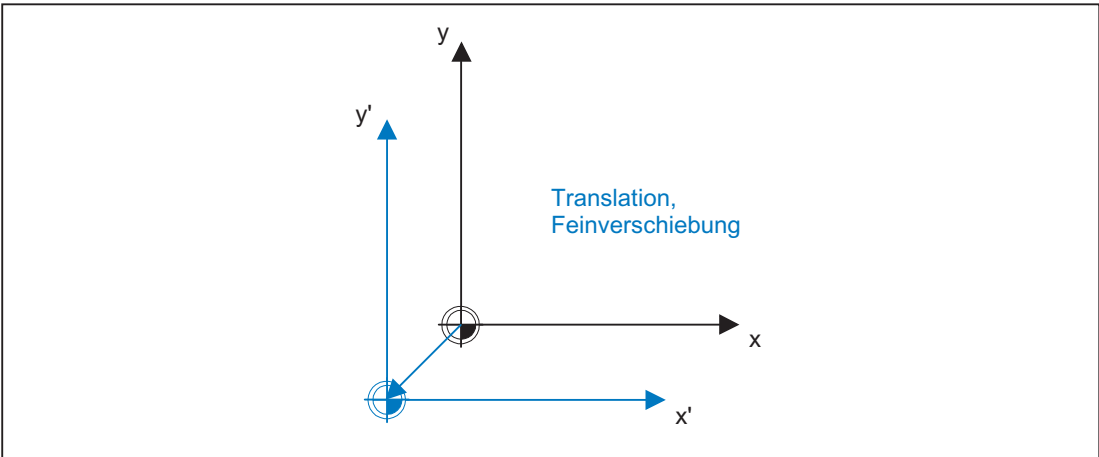


Bild 11-21 Verschiebung in Z-Richtung

11.5.2.2 Feinverschiebung

Parametrierung

Die Freigabe der Feinverschiebung erfolgt über das Maschinendatum:

MD18600 \$MN_MM_FRAME_FINE_TRANS = <Wert>

Wert	Bedeutung
0	Feinverschiebung kann nicht eingegeben bzw. nicht programmiert werden.
1	Feinverschiebung für einstellbare Frames, Basisframes und das Programmierbare Frame ist von der Bedienung oder über Programm möglich.

Programmierung

Die Programmierung der Translation bzw. Grobverschiebung kann über folgende Befehle erfolgen:

- Beispiel Datenhaltungsframes \$P_UIFR
 - Gesamtframe: \$P_UIFR[<n>] = CFINE(<K1>,<V1>[,<K2>,<V2>][,<K3>,<V3>])
mit Km = Koordinate x, y oder z und Vm = Verschiebung m
 - Frame-Komponente: \$P_UIFR[<n>,<K>,<FI>] = <V>
mit K = Koordinate x, y oder z und V = Verschiebung
- Beispiel Programmierbarer Frame
 - TRANS <K1> <V1> [<K2> <V2>][<K3> <V3>]
mit Km = Koordinate x, y oder z und Vm = Verschiebung m

Programmierbeispiele:

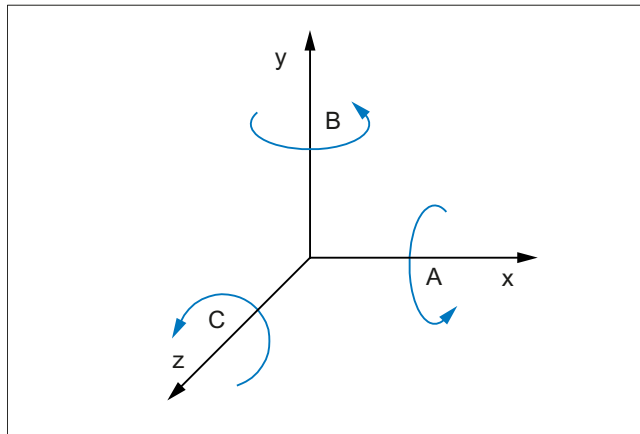
Programmcode	Bemerkung
\$P_UIFR[1] = CTRANS(X,10,Y,10)	Gesamtframe

Programmcode	Bemerkung
<code>\$P_UIFR[1,X,TR] = 10</code>	Frame-Komponente
<code>TRANS X=10 Y=10</code>	Programmierbarer Frame

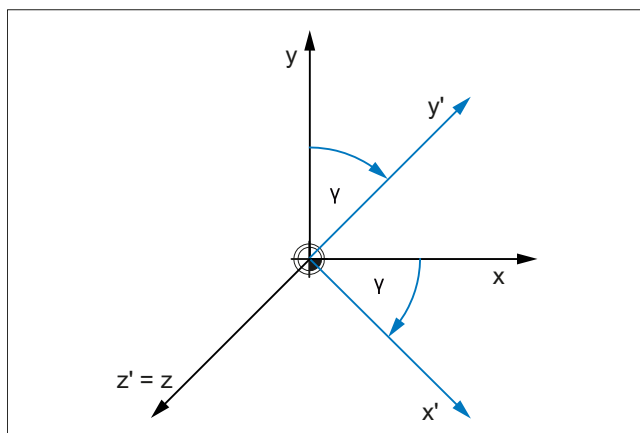
11.5.2.3 Drehung: Übersicht (nur Geometrieachsen)

Funktion

Die Drehrichtung um die Koordinatenachsen wird durch ein rechtshändiges, rechtwinkliges Koordinatensystem mit den Achsen x , y und z bestimmt. Der Drehsinn der Drehung ist positiv, wenn die Drehbewegung bei Blick in die positive Richtung der Koordinatenachse im Uhrzeigersinn erfolgt. A, B und C bezeichnen Drehungen, deren Achsen parallel zu den Koordinatenachsen sind.



Das folgende Bild zeigt die neue Lage des Koordinatensystems x' , y' und z' nach der Drehung um z mit $\gamma = -45^\circ$



Parametrierung der Drehreihenfolge

Über das folgende Maschinendatum wird eingestellt, um welche Koordinatenachsen und in welcher Reihenfolge die Drehungen ausgeführt werden, wenn mehr als ein Drehwinkel programmiert ist:

MD10600 \$MN_FRAME_ANGLE_INPUT_MODE = <Wert>

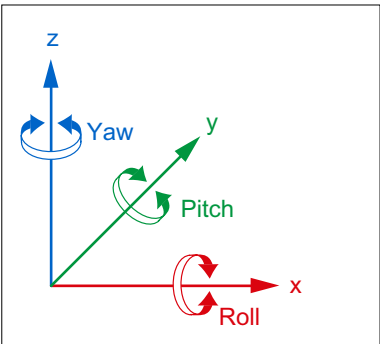
Wert	Bedeutung
1	Euler-Winkel in zy'x"-Konvention (RPY-Winkel)
2	Euler-Winkel in zx'z"-Konvention

Hinweis

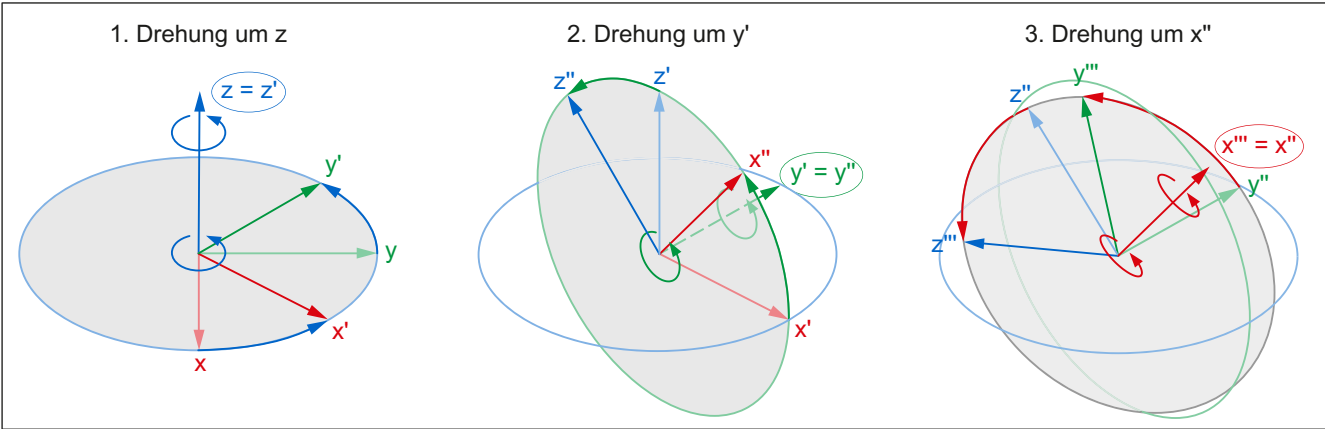
Aus historischen Gründen ist die Möglichkeit der Verwendung von Euler-Winkeln in zx'z"-Konvention vorhanden. Es wird aber dringend empfohlen ausschließlich Euler-Winkel in zy'x"-Konvention (RPY-Winkel) (siehe Kapitel "Drehung mit Euler-Winkeln: ZY'X"-Konvention (RPY-Winkel) (Seite 762)") zu verwenden.

11.5.2.4 Drehung mit Euler-Winkeln: ZY'X"-Konvention (RPY-Winkel)

Euler-Winkel in ZY'X"-Konvention werden auch als RPY-Winkel bezeichnet. RPY ist abgeleitet aus dem Englischen Roll (Rollen), Pitch (Nicken), und Yaw (Gieren):



Bei ZY'X"-Konvention erfolgen die Drehungen in folgender Reihenfolge:



Wertebereich

Bei RPY-Winkeln können programmierte Werte nur innerhalb folgender Wertebereiche eindeutig zurückgerechnet werden:

$$\begin{array}{rclclcl} -180 & \leq & x & \leq & 180 \\ -90 & < & y & < & 90 \\ -180 & \leq & z & \leq & 180 \end{array}$$

Programmierung: Schreiben aller Drehkomponenten

Bei der Programmierung der Drehkomponenten eines Frames mittels CROT, ROT oder AROT werden immer alle Drehkomponenten geschrieben. Nicht explizit programmierte Drehkomponenten werden mit dem Wert 0° belegt.

Syntax

```
<Frame> = CROT([<1. GAx>,<Winkel>],[<2. GAx>,<Winkel>],[<3. GAx>,<Winkel>])
ROT [<1. GAx> <Winkel>] [<2. GAx> <Winkel>] [<3. GAx> <Winkel>]
AROT [<1. GAx> <Winkel>] [<2. GAx> <Winkel>] [<3. GAx> <Winkel>]
```

Bedeutung

CROT:	Drehung absolut	
<Frame>:	Beliebiger aktiver oder Datenhaltungsframe	
ROT:	Drehung absolut Bezugsframe: Programmierbarer Frame \$P_PFRAME, Bezugspunkt: Nullpunkt des aktuellen mit G54 ... G57, G505 ... G599 ein- gestellten Werkstückkoordinatensystems	
AROT:	Drehung additiv Bezugsframe: Programmierbarer Frame \$P_PFRAME, Bezugspunkt: Nullpunkt des aktuellen mit G54 ... G57, G505 ... G599 ein- gestellten Werkstückkoordinatensystems	
<n-te GAx>:	Name der n-ten Geometrieachse um die um den angegebenen Winkel gedreht werden soll. Für nicht programmierte Geometrieachse wird als Drehwinkel implizit der Wert 0° gesetzt. Zuordnung von Geometrieachse zu Drehachse:	
	Geometrieachse	Drehachse
	1. Geometrieachse	x''
	2. Geometrieachse	y'
	3. Geometrieachse	z
<Winkel>:	Winkelangabe in Grad	
[...]:	Die Angaben in eckigen Klammern sind optional.	

Programmierung: Schreiben einer Drehkomponente

Bei der expliziten Programmierung einer Drehkomponente eines Frames wird nur die programmierte Drehkomponente geschrieben. Nicht programmierte Drehkomponenten bleiben unverändert.

Syntax

<Frame> [<Index>, <GAx>, RT] = <Winkel>

Bedeutung

<Frame>:	Beliebiger aktiver Frame oder Datenhaltungsframe
<Index>:	Feldindex des Frames, z.B. \$P_UIFR[0 ... n]
<GAx>:	Name der Geometrieachse um die um den angegebenen Winkel gedreht werden soll.
RT:	Schlüsselwort für Drehung "RoTation"
<Winkel>	Winkelangabe in Grad

Zurücklesen der Drehkomponenten

Im Allgemeinen erhält man beim Zurücklesen der Drehkomponenten eines Frames die gleichen Werte wie die, welche programmiert wurden:

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
<Frame> = CROT(X, 45, Y, 30, Z, -20)	45	30	-20

Werte außerhalb des Wertebereichs

Programmierte Werte außerhalb eines Wertebereichs werden auf die Bereichsgrenzen abgebildet:

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
<Frame> = CROT(X, 190, Y, 0, Z, -200)	-170	0	160

Hinweis

Es wird empfohlen, beim Schreiben der Drehkomponenten des Frames die angegebenen Wertebereiche einzuhalten, um beim Rücklesen der Drehkomponenten wieder die gleichen Werte zu erhalten.

Kardanische Blockade (engl. Gimbal-Lock)

Kardanische Blockade (engl. Gimbal-Lock) bezeichnet ein geometrisches Problem, bei dem die Drehkomponenten nicht mehr eindeutig aus dem Ortsvektor zurückgerechnet werden können. Die Kardanische Blockade tritt bei RPY-Winkeln bei einer Winkelstellung der Drehkomponente $y = 90^\circ$ auf. In diesem Fall werden von der Steuerung die Drehkomponenten nach dem Schreiben so umgerechnet, dass gilt:

- Drehkomponente z = Drehkomponente z - Drehkomponente x
- Drehkomponente x = 0°
- Drehkomponente y = 90°

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
<Frame> = CROT(X, 30, Y, 90, Z, 40)	0	90	40 - 30 = 10

**VORSICHT****Unterschiedliche Werte beim Zurücklesen der Drehkomponente z**

Aufgrund des unterschiedlichen Umrechnungszeitpunktes können nach dem Schreiben des **Gesamtframes** bzw. dem Schreiben **einzelner Drehkomponenten** eines **Datenhaltungsframes** und dem Schreiben **einzelner Drehkomponenten** eines **aktiven Frames** für die Drehkomponente z unterschiedliche Werte zurückgelesen werden.

Unterschiede beim Schreiben von Gesamtframe und Framekomponenten

Beim Schreiben der Drehkomponenten eines Frames sind zwei Fälle zu unterscheiden:

1. Schreiben des Gesamtframes: <Frame> = CROT(X, a, Y, b, Z, c)
Beim Schreiben des Gesamtframes, erfolgt die Umrechnung sofort zum Zeitpunkt des Schreibens.
2. Schreiben einzelner Drehkomponenten z.B. Drehung um X: <Frame>[0, X, RT] = <a>
Beim Schreiben einzelner Drehkomponenten, erfolgt die Umrechnung abhängig vom Speicherort des Frames:
 - Datenhaltungsframes
Bei Datenhaltungsframes erfolgt die Umrechnung zum Zeitpunkt des Aktivierens des Frames aufgrund der bis dahin geschriebenen Drehkomponenten. Bezüglich der Umrechnung der Drehkomponenten verhält sich somit ein Datenhaltungsframe nach dem Schreiben einzelner Drehkomponenten wie beim Schreiben des Gesamtframes.
 - Aktive Frames
Bei aktiven Frames erfolgt die Umrechnung sofort zum Zeitpunkt des Schreibens der Drehkomponente.

Beispiel

- Schreiben des Gesamtframes
Die **Umrechnung** erfolgt in **jedem Satz** nach dem Schreiben des **Gesamtframes**.

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
N10 <Frame> = CROT(X, 0, Y, 90, Z, 90)	0	90	90
N20 <Frame> = CROT(X, 90, Y, 90)	0	90	-90 ¹⁾
N30 <Frame> = CROT(X, 90, Y, 90, Z, 90)	0	90	0 ¹⁾
1) unterschiedliche Werte gegenüber dem Schreiben einzelner Drehkomponenten eines aktiven Frames			

- Schreiben einzelner Drehkomponenten eines **Datenhaltungsframes**
Die **Umrechnung** erfolgt **bei Aktivierung des Datenhaltungsframes**. Im Beispiel zu einem beliebigen Zeitpunkt nach N30.

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
N10 <Datenhaltungsframe>[0, X, RT] = 0 N20 <Datenhaltungsframe>[0, Y, RT] = 90 N30 <Datenhaltungsframe>[0, Z, RT] = 90	0	90	90
N10 <Datenhaltungsframe>[0, X, RT] = 90 N20 <Datenhaltungsframe>[0, Y, RT] = 90 N30 <Datenhaltungsframe>[0, Z, RT] = 0	0	90	-90 ¹⁾
N10 <Datenhaltungsframe>[0, X, RT] = 90 N20 <Datenhaltungsframe>[0, Y, RT] = 90 N30 <Datenhaltungsframe>[0, Z, RT] = 90	0	90	0 ¹⁾
1) unterschiedliche Werte gegenüber dem Schreiben einzelner Drehkomponenten eines aktiven Frames			

- Schreiben einzelner Drehkomponenten eines **aktiven Frames**
Eine eventuell notwendige **Umrechnung** erfolgt **sofort beim Schreiben der jeweiligen Drehkomponente**.
Die gespeicherten Ausgangswerte des aktiven Frames seien: x = 0, y = 0, z = 0.

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
N10 <aktiver Frame>[0, X, RT] = 0 N20 <aktiver Frame>[0, Y, RT] = 90 N30 <aktiver Frame>[0, Z, RT] = 90	0	90	90
N10 <aktiver Frame>[0, X, RT] = 90 N20 <aktiver Frame>[0, Y, RT] = 90 N30 <aktiver Frame>[0, Z, RT] = 0	0	90	0 ¹⁾

Programmiert	Werte beim Zurücklesen		
	x, RT	y, RT	z, RT
N10 <aktiver Frame>[0,X,RT] = 90	0	90	90 ¹⁾
N20 <aktiver Frame>[0,Y,RT] = 90			
N30 <aktiver Frame>[0,Z,RT] = 90			
1) unterschiedliche Werte gegenüber dem Schreiben des Gesamtframes bzw. dem Schreiben einzelner Drehkomponenten eines Datenhaltungsframes			

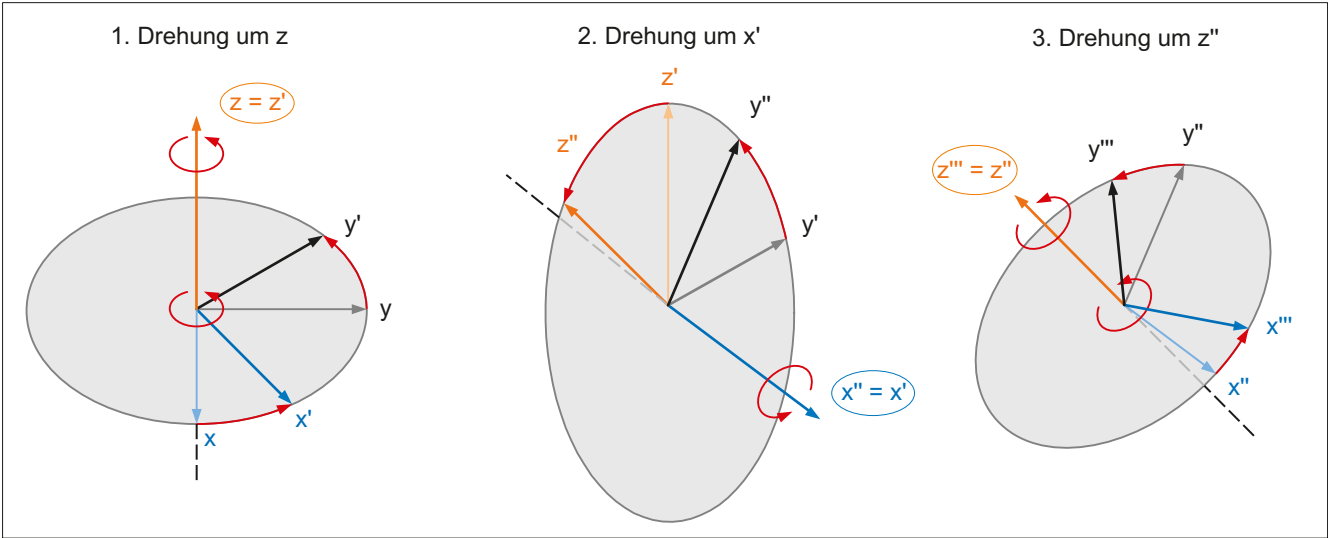
11.5.2.5 Drehung mit Euler-Winkeln: ZX'Z''-Konvention

Mit Euler-Winkeln erfolgen die Drehungen in der Reihenfolge z, x', z''.

Hinweis

Verwendungsempfehlung

Aus historischen Gründen ist die Möglichkeit der Verwendung von Euler-Winkeln in zx'z''-Konvention vorhanden. Es wird aber dringend empfohlen ausschließlich Euler-Winkel in zy'x''-Konvention (RPY-Winkel) (siehe Kapitel "Drehung mit Euler-Winkeln: ZY'X''-Konvention (RPY-Winkel) (Seite 762)") zu verwenden.



Zuordnung Drehachse zu Geometrieachse

Drehachse	Geometrieachse im Kanal
z	3. Geometrieachse
x'	1. Geometrieachse
z''	3. Geometrieachse

Wertebereich

Angaben von Euler-Winkeln können nur innerhalb folgender Wertebereiche eindeutig rückgerechnet werden:

0	<=	x	<	180
-180	<=	y	<=	180
-180	<=	z	<=	180

Angaben außerhalb der angegebenen Wertebereiche werden modulo der Bereichsgrenzen gerechnet.

Hinweis

Es wird empfohlen, beim Schreiben der Drehkomponenten des Frames die angegebenen Wertebereiche einzuhalten, um beim Zurücklesen der Drehkomponenten wieder die gleichen Werte zu erhalten.

11.5.2.6 Drehung in beliebiger Ebene**CRPL - Constant Rotation Plane**

Mit der vordefinierten Funktion "**Constant Rotation Plane**" kann für einen Frame eine Drehung in einer beliebigen Ebene (G17, G18, G19), ohne Angabe des Namens einer Geometrieachse, programmiert werden. Somit können auch Drehungen in der dritten Ebene programmiert werden, wenn aufgrund der spezifischen Maschinenkonstellation nur zwei Geometrieachsen im Kanal vorhanden sind.

Syntax

CRPL(<Drehachse>,<Drehwinkel>)

Bedeutung

CRPL: Drehung in beliebiger Ebene

<Drehachse>: Achse um die gedreht wird

Typ: INT

Wert	Bedeutung
0	Drehung in der aktiven Ebene
1	Drehung um Z
2	Drehung um Y
3	Drehung um X

<Drehwinkel>: Winkel in Grad um den gedreht wird

Typ: REAL

Es wird dringend empfohlen die angegebenen Winkelbereiche einzuhalten. Werden die Grenzen nicht eingehalten, ist keine eindeutige Rückrechnung möglich. Winkelangaben außerhalb der Grenzen werden nicht abgelehnt.

RPY-Winkel:	X	-180 <= <Drehwinkel> <= 180
	Y	-90 <= <Drehwinkel> <= 90

	Z	-180 <= <Drehwinkel> <= 180
ZX'Z"-Konvention:	X	-180 <= <Drehwinkel> <= 180
	Y	0 <= <Drehwinkel> <= 180
	Z	-180 <= <Drehwinkel> <= 180

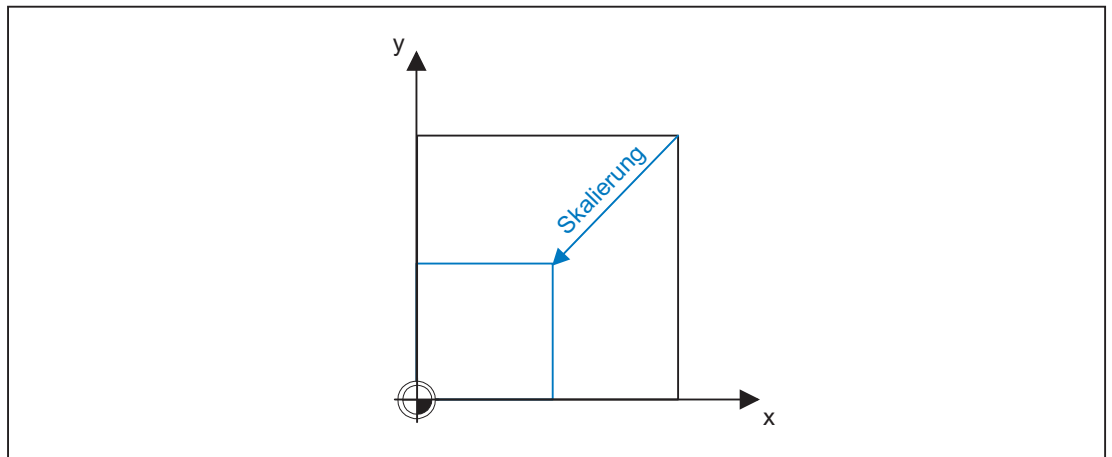
Verkettung mit Frames

CRPL() kann mit Frames und Frame-Funktionen wie CTRANS(), CROT(), CMIRROR(), CSCALE(), CFINE() etc. verkettet werden.

Beispiele:

```
$P_PFRAME = $P_PFRAME : CRPL(0,30.0)
$P_PFRAME = CTRANS(X,10) : CRPL(1,30.0)
$P_PFRAME = CROT(X,10) : CRPL(2,30.0)
$P_PFRAME = CRPL(3,30.0) : CMIRROR(Y)
```

11.5.2.7 Skalierung

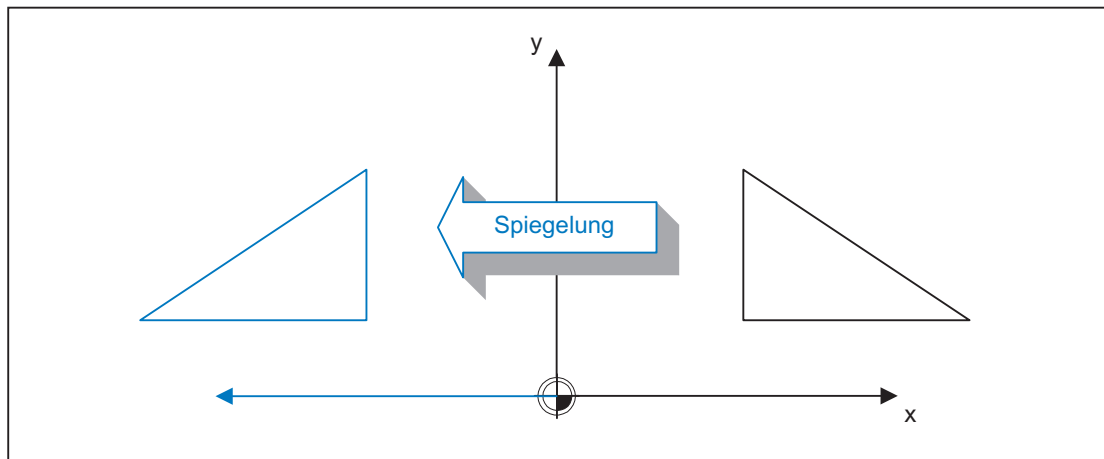


Programmierung

Die Programmierung einer Skalierung erfolgt über folgende Programmbefehle:

```
$P_UIFR[1] = CSCALE(x,1,y,1)
SCALE x = 1 y = 1
$P_UIFR[1,x,sc] = 1
```

11.5.2.8 Spiegelung



Programmierung

Die Programmierung einer Spiegelung erfolgt über folgende Programmbefehle:

```
$P_UIFR[1] = CMIRROR(x,1,y,1)
```

```
MIRROR x = 1 y = 1
```

```
$P_UIFR[1,x,mi] = 1
```

11.5.2.9 Verkettungsoperator

Framekomponenten oder gesamte Frames lassen sich über den Verkettungsoperator (:) zu einem Gesamtframe zusammenfassen.

11.5.2.10 Programmierbare Achsname

In den Frame-Befehlen können Geo-, Kanal- und Maschinen-Achsname verwendet werden. Die programmierte Achse muss bei den kanalspezifischen Frames im Kanal bekannt sein.

SPI

Bei der Programmierung von Frame-Anweisungen kann an Stelle eines Achsnamens auch die Achsfunktion `SPI(<spindelnummer>)` verwendet werden.

`SPI(<spindelnummer>)` stellt dabei den Bezug der Spindel zu der Kanalachse.
→ siehe MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX[] (Zuordnung Spindel zur Maschinenachse)

Folgende Frame-Anweisungen können mit `SPI(spino)` programmiert werden:

```
CTRANS ( )
```

```
CFINE ( )
```

```
CMIRROR ( )
```

CSCALE ()

Eine Spindel kann immer nur einer Rundachse zugewiesen werden. Deshalb kann die Funktion CROT (. .) nicht mit SPI () programmiert werden, da für CROT () nur Geometrie-Achsen erlaubt sind.

Bei der Rückübersetzung von Frames wird immer der Kanalachsname bzw. der Maschinenachsname der zur Spindel gehörenden Achse ausgegeben, auch wenn im Teileprogramm Achsname mit SPI (. .) programmiert worden sind.

Ist die Spindel z. B. der Kanalachse "A" zugeordnet, dann wird aus der Programmierung:

```
N10 $P_UIFR[1] = CTRANS (SPI (1) , 33.33, X, 1) : CSCALE (SPI (1) ,
33.33) : CMIRROR (SPI (1) )
```

bei der Rückübersetzung:

```
$P_UIFR[1]=CTrans (X, 1, A, 33.33) : CSCALE (A, 33.33) : CMIRROR (A)
```

Wird in einer Frame-Anweisung eine Spindel und die zugeordnete Achse programmiert, wird der Alarm 16420 "Achse % mehrfach programmiert" ausgegeben.

Beispiel:

```
$P_UIFR[1] = CTRANS (SPI (1) , 33.33, X, 1, A, 44)
```

(Die Spindel ist der Achse A zugeordnet.)

Programmierbeispiele

```
$P_PFRAME [SPI (1) , TR]=22.22
```

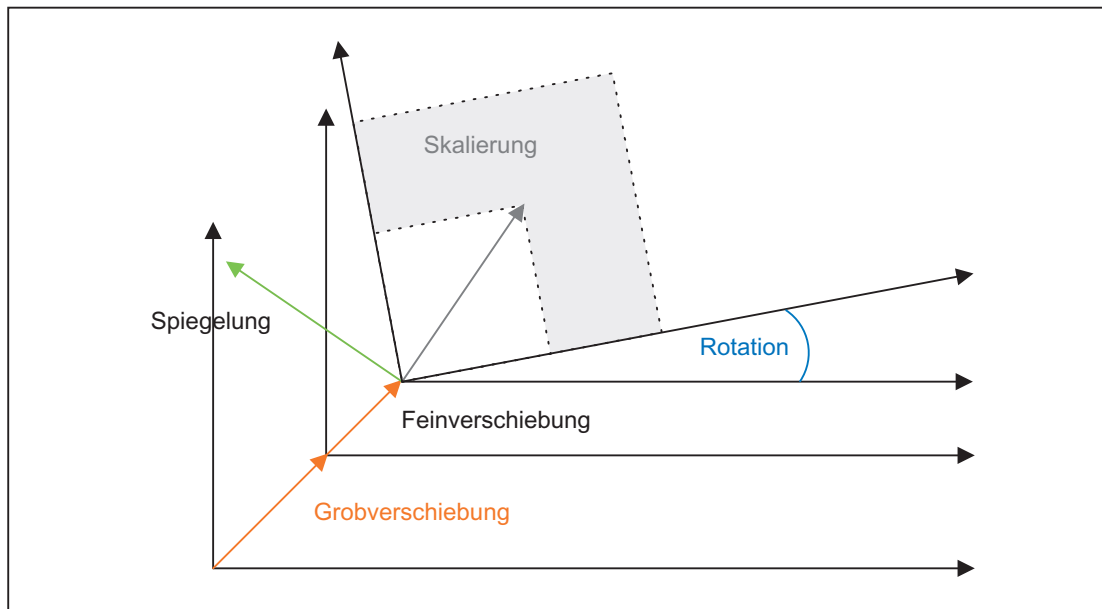
```
$P_PFRAME=CTrans (X, Achswert, Y, Achswert, SPI (1) , Achswert)
```

```
$P_PFRAME=CSCALE (X, Maßstab, Y, Maßstab, SPI (2) , Maßstab)
```

```
$P_PFRAME=CMIRROR (S1, Y, Z)
```

```
$P_UBFR=CTrans (A, 10) : CFINE (SPI (1) , 0.1)
```

11.5.2.11 Koordinatentransformation



Die Koordinatentransformation für Geometrieachsen ergibt sich anhand folgender Formeln:

WKS → BKS

$$\vec{v} = R * \underline{S} * \underline{M} * \vec{v}' + \vec{t}$$

BKS → WKS

$$\vec{v}' = \text{inv}(\underline{M}) * \text{inv}(\underline{S}) * \text{inv}(\underline{R}) + (\vec{v} - \vec{t})$$

V: Positionsvektor im BKS

V': Positionsvektor im WKS

11.5.3 Datenhaltungs-Frames und aktive Frames

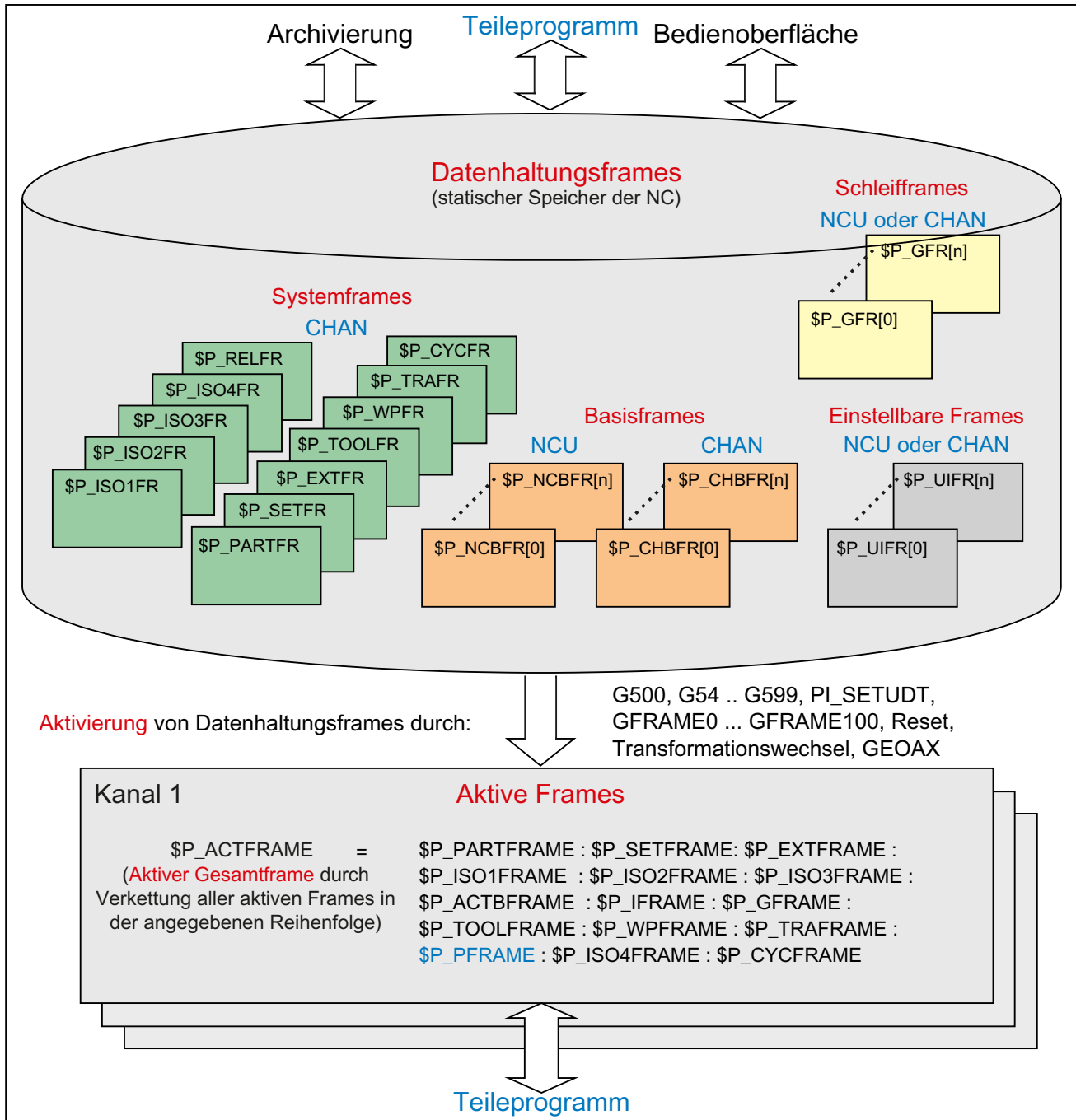
11.5.3.1 Übersicht

Frame-Typen

Es gibt folgende Frames-Typen:

- Systemframes (\$P_PARTFR, ... siehe Bild)
- Basisframes (\$P_NCBFR[<n>], \$P_CHBFR[<n>])
- Schleifframes (\$P_GFR[<n>])
- Einstellbare Frames (\$P_UIFR[<n>])
- Programmierbarer Frame (\$P_PFRAME)

Bei allen Frame-Typen, außer dem Programmierbaren Frame, existiert neben dem im Kanal aktiven Frame ein oder mehrere Frames in der Datenhaltung (Datenhaltungsframes). Beim Programmierbaren Frame existiert nur der im Kanal aktive Frame.



Schreiben von Frames

Aus dem Teileprogramm heraus können Datenhaltungsframes und aktive Frames geschrieben werden. Über die Bedienoberfläche können nur Datenhaltungsframes geschrieben werden.

Archivieren von Frames

Es können nur Datenhaltungsframes archiviert werden.

11.5.3.2 Aktivierung von Datenhaltungsframes

Datenhaltungsframes werden zu aktiven Frames durch folgende Aktionen:

- G-Gruppe "Einstellbare Frames": G54 ... G57, G500, G505 ... G599
- G-Gruppe "Schleifframes": GFRAME0 ... GFRAME100
- RESET und MD20110 \$MC_RESET_MODE_MASK, Bit14 == 1 (Die aktuelle Einstellung des Basisframes bleibt erhalten)
- Transformationswechsel
- Änderung der Geometrieachszuordnung GEOAX
- Von HMI aus mit PI-Dienst "_N_SETUDT"

Aktivierung von HMI aus

Die Aktivierung eines Datenhaltungsframes von HMI aus mit dem PI-Dienst "_N_SETUDT" wirkt im Kanal erst nach einem Fortsetzstart für das aktuelle NC-Programm.

Die Aktivierung wirkt im Reset-Zustand sofort, wenn folgendes Maschinendatum gesetzt ist:
MD9440 \$MM_ACTIVATE_SEL_USER_DATA (Aktive Verschiebung sofort wirksam setzen)

Aktivierung von Systemframes

Die Aktivierung von Systemframes erfolgt durch:

- Programmierung der entsprechenden Systemfunktion im Teileprogramm
- Bedienhandlung an SINUMERIK Operate

Hinweis

Modifizieren von Systemframes der Datenhaltung

Systemframes der Datenhaltung können zwar prinzipiell durch den Zyklen-Programmierer modifiziert und über eine G500, G54...G599-Anweisung aktiviert werden. Diese Möglichkeit sollte aber nur unter Vorbehalt verwendet werden.

Aktivierung von Datenhaltungsframes

Das Verhalten bei Aktivierung von Datenhaltungsframes wird über folgendes Maschinendatum eingestellt:

MD24050 \$MC_FRAME_SAA_MODE (Speichern und aktivieren von Datenhaltungsframes)

Bit	Wert	Bedeutung
0	0	@@@
	1	<ul style="list-style-type: none"> Datenhaltungsframes werden nur durch Programmierung der Systemvariablen \$P_CHBFRMASK, \$P_NCBFRMASK und \$P_CHSFRMASK aktiv. G500...G599 aktiviert nur das entsprechende Einstellbare Frame.
1	0	Datenhaltungsframes werden durch Funktionen wie TOROT, PAROT, externe Nullpunktverschiebung, Transformationen, implizit beschrieben.
	1	Datenhaltungsframes werden durch Funktionen wie TOROT, PAROT, externe Nullpunktverschiebung, Transformationen nicht implizit beschrieben.

Aktivierung von Systemframes über Systemvariable \$P_CHSFRMASK

Die Systemframes der Datenhaltung können über die Systemvariable \$P_CHSFRMASK aktiviert werden. Der Wert der Variablen wird bitcodiert angegeben, entsprechend dem Maschinendatum:

MD28082 \$MC_MM_SYSTEM_FRAME_MASK (Systemframes der Datenhaltung)

Durch Setzen eines Bits der Systemvariable \$P_CHSFRMASK auf den Wert 1, wird der entsprechende Systemframe der Datenhaltung im Kanal aktiv. Bei einem Wert 0 bleibt der im Kanal aktuell aktive Systemframe weiterhin aktiv.

Aktivierung von Systemframes nach RESET

Nach RESET werden im Kanal die Systemframes aktiviert, deren Bits im folgenden Maschinendatum gesetzt sind:

MD24006 \$MC_CHSFRAME_RESET_MASK (Aktive Systemframes nach Reset)

Aktivierung der Systemframes für TCARR, PAROT und TOROT, TOFRAME

Die Systemframes für TCARR, PAROT und TOROT, TOFRAME werden entsprechend der Einstellung im folgenden Maschinendatum aktiviert:

MD20150 \$MC_GCODE_RESET_VALUES (Löschstellung der G-Gruppen)

Beim Umschalten von Geometrieachsen durch Transformations-An/Abwahl oder Befehl GEOAX, wird das aktuelle Gesamtframe \$P_ACTFRAME entweder gelöscht oder anhand der neuen Geometrieachs-Konstellation neu berechnet und aktiviert. Die Systemframes werden, ebenso wie alle anderen Frames, bezüglich der Geometrieachsen neu aufbereitet.

11.5.3.3 NCU-lobale und kanalspezifische Frames

- Einstellbare Frames und Schleifframes können nur alternativ NCU-global **oder** kanalspezifisch projiziert werden.
- Basisframes können NCU-global **und** kanalspezifisch projiziert werden.
- Ein NCU-globaler Frame wirkt in allen Kanäle einer NCU gleich.

- Alle Kanäle einer NCU können NCU-globale Frames gleichberechtigt lesen und schreiben.
- Da die Zuordnung von Maschinenachsen zu Kanalachsen und speziell zu Geometrieachsen in allen Kanälen unterschiedlich sein kann, gibt es demzufolge keine eindeutigen kanalübergreifenden geometrischen Zusammenhang zwischen den Kanalachsen. Bei NCU-globalen Frames sind daher nur Verschiebung, Skalierung und Spiegelung möglich. Drehungen sind nicht möglich.

Hinweis

Programmkoordinierung

Die Koordination von kanalspezifischen Zugriffen auf NCU-globale Frames steht allein in der Verantwortung des Anwenders. Es wird dazu die Verwendung der Befehle zur Programmkoordinierung empfohlen.

Literatur

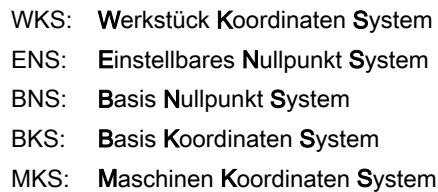
Programmierhandbuch Arbeitsvorbereitung; Kapitel "Flexible NC-Programmierung" > "Programmkoordinierung (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)"

11.5.4 Framekette und Koordinatensysteme

11.5.4.1 Übersicht

Im folgenden Bild ist die Framekette für das aktuelle Gesamtframe abgebildet. Die Framekette liegt zwischen dem Basiskoordinatensystem (BKS) und dem Werkstückkoordinatensystem (WKS).

Das Einstellbare Nullpunkt-System (ENS) entspricht dem durch das Programmierbare Frame transformierte WKS. Das Basis-Nullpunkt-System (BNS) beinhaltet noch das aktuelle Einstellbare Frame. Das Systemframe für die externe Nullpunktverschiebung ist nur vorhanden, wenn es projiziert wurde. Sonst wird die externe Nullpunktverschiebung als überlagerte Bewegung der Achse heraus gefahren.



Das aktuelle Gesamtframe `$P_ACTFRAME` ergibt sich aus der Verkettung aller aktiven Frames der Framekette:

```
$P_ACTFRAME =      $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME :
                   $P_ISO1FRAME : $P_ISO2FRAME : $P_ISO3FRAME :
                   $P_ACTBFRAME : $P_IFFRAME : $P_GFRAME :
                   $P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME :
                   $P_PFRAME : $P_ISO4FRAME : $P_CYCFRAME
```

Relative Koordinatensysteme dienen nur zur Anzeige von aktuellen Achs-Sollpositionen, die relativ zu einem vorgegebenen Referenzpunkt im aktiven angezeigten Koordinatensystem liegen. Bezüglich den relativen Koordinatensystemen kann nicht programmiert werden. Es können nur Achspositionen in diesen Systemen über Systemvariablen gelesen werden.

Die neuen Anzeige-Koordinatensysteme liegen relativ zum WKS- und ENS-Koordinatensystem und ergeben sich durch Transformation der WKS- oder ENS-Achspositionen mit dem aktiven Systemframe \$P_RELFRAME\$. Die relativen Koordinatensysteme können nicht nur linear verschoben sein, sondern auch noch gedreht, gespiegelt, gestaucht bzw. gedehnt werden.

Die Positionsanzeige für Achssollwerte erfolgt im WKS oder im ENS. Die Projektierung erfolgt über HMI-Maschinendaten. Es ist immer nur ein Anzeige-Koordinatensystem im Kanal aktiv. Deshalb wird auch nur ein relatives Frame zur Verfügung gestellt, welches die beiden relativen Koordinatensysteme im gleichen Verhältnis erzeugt. Das HMI zeigt die relativen Koordinaten entsprechend der Projektierung an.

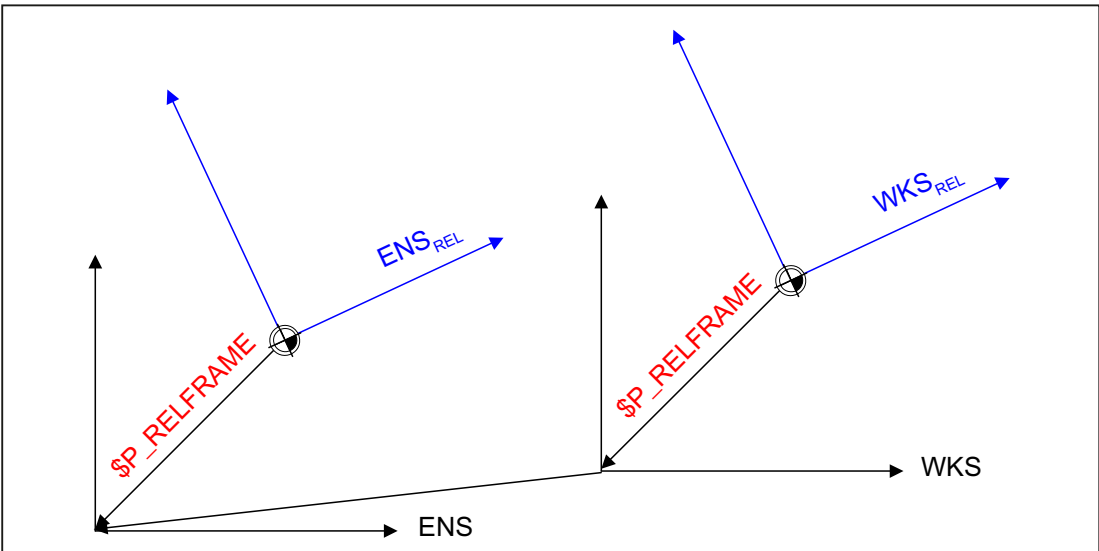


Bild 11-22 Relative Koordinatensysteme

Die Funktion "Relative Koordinatensysteme" wird mit MD51036 \$MNS_ENABLE_COORDINATE_REL=1 aktiviert.

Das Datenhaltungsframe \$P_RELFR kann im Teileprogramm und über BTSS beschrieben werden. Es können alle Framekomponenten verändert werden.

Das aktive Systemframe \$P_RELFRAME kann im Teileprogramm beschrieben und über BTSS gelesen werden.

Die Projektierung des Systemframes \$P_RELFR erfolgt über folgende Maschinendaten:

Maschinendatum	Bit	Bedeutung
MD28082 \$MC_MM_SYSTEM_FRAME_MASK	11	Anlegen von \$P_RELFR; damit werden relative Koordinatensysteme existent.
MD28083 \$MC_MM_SYSTEM_DATAFRAME_MASK	11	Datenhaltungsframe \$P_RELFR
MD24006 \$MC_CHSFRAME_RESET_MASK.	11	\$P_RELFR wird bei Reset aktiv
MD24007 \$MC_CHSFRAME_RESET_CLEAR_MASK	11	\$P_RELFR wird bei Reset gelöscht
MD24008 \$MC_CHSFRAME_POWERON_MASK	11	\$P_RELFR wird bei PowerOn gelöscht

Die Achspositionen im relativen Koordinatensystem WKS_{Rel} können über die Variable \$AA_PCS_REL[ax] gelesen werden. Die Variable ist im Teileprogramm, BTSS und über Synchronaktionen lesbar.

Die Achspositionen im relativen Koordinatensystem ENS_{Rel} können über die Variable \$AA_ACS_REL[ax] gelesen werden. Die Variable ist im Teileprogramm, BTSS und über Synchronaktionen lesbar.

Das Setzen eines relativen Bezugspunktes über die Bedienoberfläche erfolgt über die allgemeine Kommandoschnittstelle für die Werkstück- und Werkzeugvermessung. Das Systemframe \$P_RELFR für relative Koordinatensysteme wird folgendermaßen berechnet und aktiviert:

- \$AC_MEAS_TYPE = 14
- PI-Dienste _N_SETUDT(6, 7)

Ein Beispiel zum Setzen von relativen Achspositionen findet sich in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Messen (M5),
Kapitel "Messen von Geo- und Zusatzachsen (Mess Type 14, 15)"

11.5.4.3 Wählbares ENS

Innerhalb eines Zyklus erfolgt die Bearbeitung in einem Zyklus-spezifischen Werkstückkoordinatensystem (WKS). Das Zyklus-spezifische WKS entsteht dabei aus dem ENS transformiert durch für den Zyklus programmierten Frames Programmierbaren Frame \$P_PFRAME und/oder Zyklen-Frame \$P_CYCFRAME.

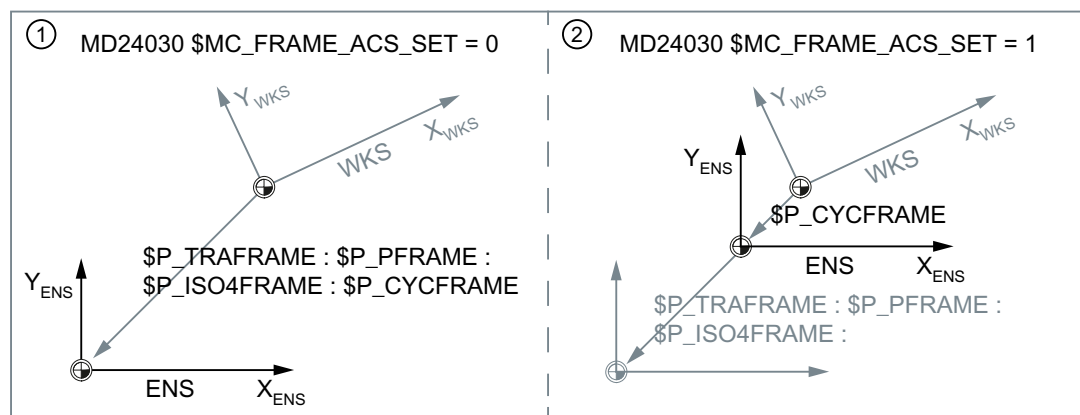
Wird ein Zyklus von einem Maschinenbediener, z.B. durch NC-Stop unterbrochen, soll anschließend im ursprünglichen, vor Aktivierung des Zyklus gültigen Koordinatensystem (ENS), verfahren werden.

Maschinendaten

Die Vorgabe, wie das ENS aus dem Zyklus-spezifischen WKS zu berechnen ist, kann über folgendes Maschinendatum eingestellt werden.

MD24030 \$MC_FRAME_ACS_SET = <Wert>

<Wert>	Bedeutung
0	ENS = WKS transformiert mit \$P_TRAFRAME, \$P_PFRAME, \$P_ISO4FRAME und \$P_CYCFRAME
1	ENS = WKS transformiert nur mit \$P_CYCFRAME



- ① ENS = WKS transformiert mit \$P_TRAFRAME, \$P_PFRAME, \$P_ISO4FRAME und \$P_CYCFRAME
- ② ENS = WKS transformiert nur mit \$P_CYCFRAME

Auswirkungen

Die Umprojektierung des ENS hat Rückwirkung auf:

- ENS-bezogenen Istwerte: Istwertanzeigen, Systemvariablen z.B. \$AA_IEN etc.
- Manuelles Verfahren (JOG) von Geometrieachsen im ENS

11.5.4.4 Manuelle Verfahren von Geometrieachsen wahlweise im WKS oder ENS (\$AC_JOG_COORD)

Die Geometrieachsen werden bisher beim manuellen Verfahren in der Betriebsart JOG im WKS verfahren. Zusätzlich dazu gibt es die Möglichkeit, das manuelle Verfahren im ENS-Koordinatensystem durchzuführen. Dazu gibt es die Variable \$AC_JOG_COORD, mit der zwischen manuellem Verfahren im WKS und ENS umgeschaltet werden kann. Der Anwender hat jetzt die Möglichkeit auszuwählen, ob er im ENS oder im WKS verfahren möchte.

Die Geometrieachsen können beim manuell Verfahren in der Betriebsart JOG wahlweise im Werkstückkoordinatensystem (WKS) oder im Einstellbaren Nullpunktsystem (ENS) verfahren werden.

Systemvariablen

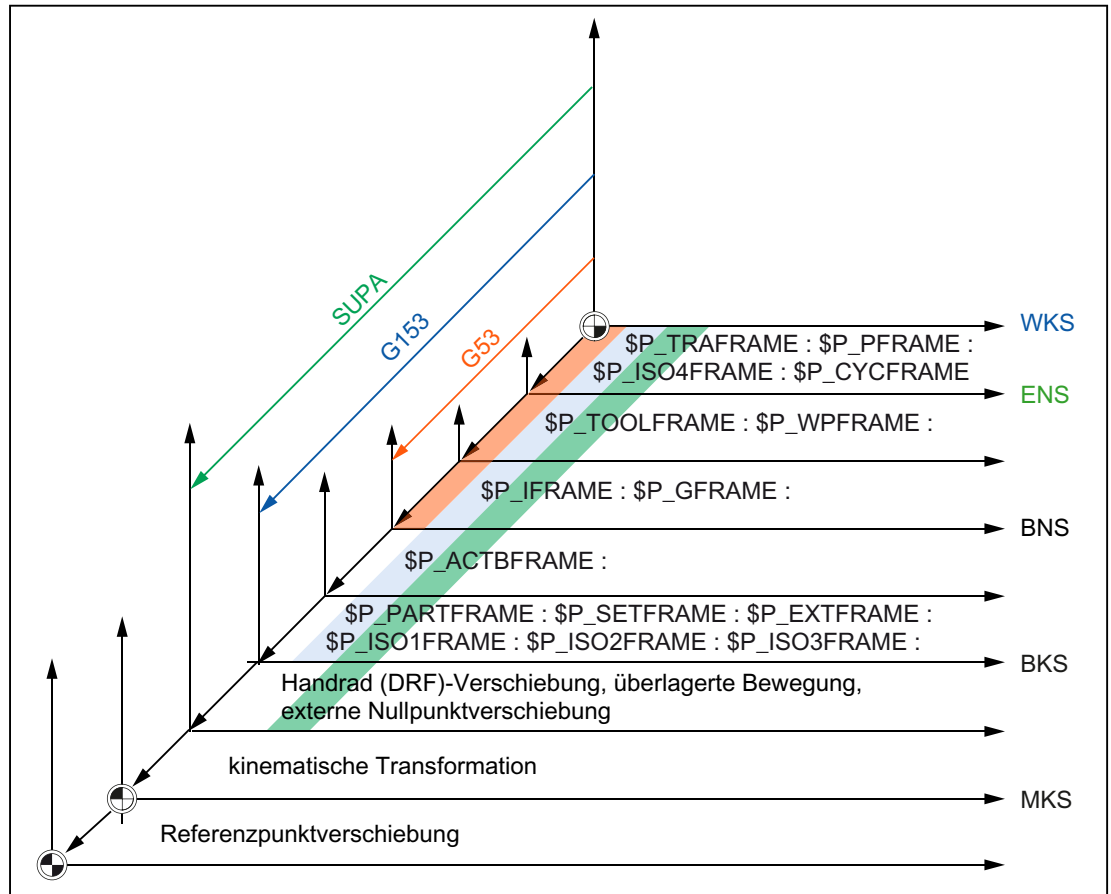
Die Geometrieachsen können beim manuellen Verfahren in der Betriebsart JOG wahlweise im Werkstückkoordinatensystem (WKS) oder in Einstellbaren Nullpunktsystem (ENS) verfahren werden. Die Auswahl erfolgt über die Systemvariable \$AC_JOG_COORD:

\$AC_JOG_COORD = <Wert>

<Wert>	Bedeutung
0	Werkstückkoordinatensystem (WKS)
1	Einstellbarens Nullpunktsystem (ENS)

11.5.4.5 Unterdrückung von Frames

Die Unterdrückung von Frames erfolgt kanalspezifisch über die nachfolgend beschriebenen Befehle `G53`, `G135` und `SUPA`. Eine Aktivierung der Frame-Unterdrückungen führt dazu, dass Positionsanzeigen (HMI) sowie Positionsangaben in Systemvariablen, die sich auf das WKS, ENS oder BNS beziehen, springen. Das Verhalten kann über Maschinendaten eingestellt werden.



Maschinendaten

Über das folgende Maschinendatum wird das Verhalten von Positionsanzeigen (HMI) und Positionsangaben in Systemvariablen festgelegt:

MD24020 \$MC_FRAME_SUPPRESS_MODE, Bit<n> = <Wert> (Positionen bei Frameunterdrückung)

Bit	Wert	Bedeutung
0	0	Positionsangaben (BTSS) mit Frame-Unterdrückung ¹⁾
	1	Positionsangaben (BTSS) ohne Frame-Unterdrückung ²⁾

Bit	Wert	Bedeutung
1	0	Positionsangaben in Systemvariablen mit Frame-Unterdrückung ¹⁾
	1	Positionsangaben in Systemvariablen ohne Frame-Unterdrückung ²⁾
1) Springen des Positionswertes		
2) kein Springen des Positionswertes		

Programmierung

Befehl	Bedeutung
G53:	Satzweises Unterdrücken folgender Frames: \$P_TRAFRAME : \$P_PFRAME : \$P_ISO4FRAME : \$P_CYCFRAME \$P_IFRAME : \$P_GFRAME : \$P_TOOLFRAME : \$P_WPFRAME :
G153:	Satzweises Unterdrücken der Frames wie bei G53 plus folgender Frames: \$P_PARTFRAME : \$P_SETFRAME : \$P_EXTFRAME : \$P_ACTBFRAME \$P_ISO1FRAME : \$P_ISO2FRAME : \$P_ISO3FRAME :
SUPA:	Impliziter Vorlaufstopp und satzweises Unterdrücken der Frames wie bei G53 und G135 plus folgender Frames: <ul style="list-style-type: none"> • Handradverschiebungen (DRF) • Überlagerte Bewegung • Externe Nullpunktverschiebung
G500:	Modales Aktivieren des G500-Frames. Der G500-Frame sollte normalerweise ein Nullframe sein.
DRFOF:	Ausschalten (Löschen) der Handradverschiebungen (DRF)

11.5.5 Frames der Framekette

11.5.5.1 Übersicht

Es stehen folgende Frames zur Verfügung:

- Einstellbare Frames (G500, G54 ... G57, G505 ... G599)
- Schleifframes (GFRAME0 ... GFRAME100)
- Basisframes
- Programmierbares Frame
- Systemframes

11.5.5.2 Einstellbare Frames \$P_UIFR[<n>]

Maschinendaten

Kanalspezifische Einstellbare Frames

Die Anzahl der kanalspezifischen Einstellbaren Frames wird mit folgendem Maschinendatum eingestellt:

MD28080 \$MC_MM_NUM_USER_FRAMES = <Anzahl>

Systemvariablen-Index n = 0, 1, 2, ... <Anzahl> - 1

NCU-globale Einstellbare Frames

Die Anzahl der NCU-globalen Einstellbaren Frames wird mit folgendem Maschinendatum eingestellt:

MD18601 \$MN_MM_NUM_GLOBAL_USER_FRAMES = <Anzahl>

Systemvariablen-Index n = 0, 1, 2, ... <Anzahl> - 1

Hat das Maschinendatum einen Wert > 0, existieren **keine kanalspezifischen** Einstellbaren Frames. Das Maschinendatum zum Einstellen der kanalspezifischen Einstellbaren Frames wird dann nicht ausgewertet.

Löschstellung der G-Gruppe der Einstellbare Frames

Die Löschstellung bzw. welche der G-Befehle der Einstellbare Frames-spezifischen **8. G-Gruppe** nach Kanal-Reset bzw. Power On wirksam wird, wird eingestellt in:

MD20150 \$MC_GCODE_RESET_VALUES[7] = <Wert>

Wert	G-Befehl
1	G500
2	G54
3	G55
4	G56
5	G57
6	G505
...	...
100	G599

Resetverhalten der schleifenspezifischen G-Gruppe

Das Resetverhalten der Einstellbaren Frames-spezifischen **8. G-Gruppe** wird eingestellt in:

MD20152 \$MC_GCODE_RESET_MODE[7] = <Wert>

Wert	Bedeutung Nach Kanal-Reset oder Teileprogrammende:
0	wird der Einstellbare Frames-spezifische G-Befehl entsprechend MD20150 aktiv.
1	bleibt der aktuell aktive Einstellbare Frames-spezifische G-Befehl weiter aktiv.

Hinweis**MD20110 \$MC_RESET_MODE_MASK**

Das Maschinendatum MD20152 \$MC_GCODE_RESET_MODE wird nur ausgewertet bei:

MD20110 \$MC_RESET_MODE_MASK, Bit0 == 1

Systemvariablen**\$P_UIFR[<n>] (Einstellbare Frames der Datenhaltung)**

Über die Systemvariable \$P_UIFR[<n>] können die Einstellbaren Frame der Datenhaltung gelesen und geschrieben werden. Beim Schreiben eines Einstellbaren Frames der Datenhaltung werden die neuen Werte nicht sofort im Kanal aktiv. Die Aktivierung im Kanal erfolgt erst mit Programmierung einer Nullpunktverschiebung G500, G54...G599.

Bei NCU-globalen Frames wird das geänderte Einstellbare Frame der Datenhaltung in jedem Kanal der NCU der einen G500, G54...G599-Befehl ausführt, aktiv.

Die Einstellbaren Frames in der Datenhaltung werden bei einer Datensicherung mit gesichert.

\$P_IFRAME (Aktives Einstellbares Frame)

Über die Systemvariable \$P_IFRAME kann der im Kanal **aktive** Einstellbare Frame gelesen und geschrieben werden. Beim Schreiben des Einstellbaren Frame werden die neuen Werte sofort im Kanal wirksam.

Bei NCU-globalen Einstellbaren Frames wirkt der geänderte aktive Einstellbare Frame nur in dem Kanal, in dem die neuen Werte programmiert wurde. Soll das geänderte NCU-globale Einstellbare Frame für alle Kanäle der NCU wirksam werden, muss das im Kanal aktive Einstellbare Frame **und** das entsprechende Einstellbare Frame der Datenhaltung gemeinsam geschrieben werden:

$\$P_UIFR[<n>] = \$P_IFRAME = <\text{neuer Wert}>$

- \$P_UIFR[<n>] (Einstellbares Frame in der Datenhaltung)
- \$P_IFRAME (im Kanal aktiver Einstellbarer Frame) .

Damit das geänderte Einstellbare Frame in einem anderen Kanal wirksam wird, muss es in diesem Kanal noch mit dem entsprechenden Befehl, z. B. G54, aktiviert werden.

\$P_UIFRNUM (Nummer des aktiven Einstellbaren Frames)

Über die Systemvariable \$P_UIFRNUM kann der Index <n> des im Kanal aktiven Einstellbaren Frames der Datenhaltung gelesen werden:

Im Kanal aktiver Einstellbarer Frame $\$P_IFRAME == \$P_UIFR[\$P_UIFRNUM]$

\$P_UIFRNUM	\$P_IFRAME == \$P_UIFR[<n>], mit n =
0	0
1	1
2	2
...	...
99	99

Programmierung

Befehle zur Aktivierung eines Einstellbaren Frames im Kanal

Durch Programmierung eines Befehles G500, G54...G599 wird der Einstellbare Frame der Datenhaltung \$P_UIFR[<n>] im Kanal aktiv bzw. der aktive Einstellbare Frame \$P_IFRAME wird gleich dem Einstellbaren Frame der Datenhaltung \$P_UIFR[<n>] gesetzt:

$G<x> \Rightarrow \$P_IFRAME = \$P_UIFR[<n>]$

Befehl	Aktiver Einstellbarer Frame \$P_IFRAME =
G500	\$P_UIFR[0]
G54	\$P_UIFR[1]
G55	\$P_UIFR[2]
G56	\$P_UIFR[3]
G57	\$P_UIFR[4]
G505	\$P_UIFR[5]
...	...
G599	\$P_UIFR[99]

Randbedingungen

Schreiben von Einstellbaren Frames durch HMI / PLC

Von HMI oder dem PLC-Anwenderprogramm können nur die Einstellbaren Frames der **Datenhaltung** geschrieben werden.

11.5.5.3 Schleifframes \$P_GFR[<n>]

Schleifframes sind speziell für die Technologie Schleifen zur Verfügung stehende zusätzliche Nullpunktverschiebungen oder auch „Sitzabhängige Korrekturen“. Sie wirken additiv zu den Nullpunktverschiebungen der Einstellbaren Frames (Seite 783).

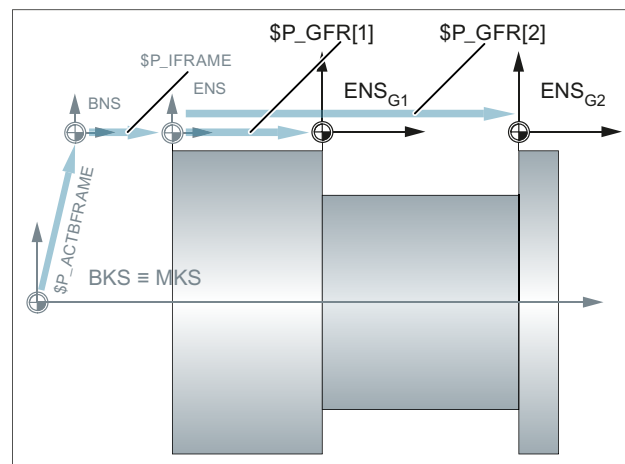


Bild 11-23 Schleifframes

Ausgehend vom Basis-Nullpunktsystem (BNS) ergibt sich somit das Einstellbare Nullpunktsystem (ENS) aus der Verkettung der im Kanal aktiven Frames:

\$P_IFRAME : **\$P_GFRAME** : \$P_TOOLFRAME : \$P_WPFRAME

Maschinendaten

Anzahl kanalspezifischer Schleifframes

Die Anzahl der kanalspezifischen Schleifframes wird eingestellt in:

MD28079 \$MN_MM_NUM_G_FRAMES = <Anzahl>

mit <Anzahl> = 0, 1, 2, ... maximale Anzahl

Anzahl NCU-globaler Schleifframes

Die Anzahl der NCU-globalen Schleifframes wird eingestellt in:

MD18603 \$MN_MM_NUM_GLOBAL_G_FRAMES = <Anzahl>

mit <Anzahl> = 0, 1, 2, ... maximale Anzahl

Hat das Maschinendatum einen Wert > 0, existieren **keine kanalspezifischen** Schleifframes. Das Maschinendatum zum Einstellen der kanalspezifischen Schleifframes wird dann nicht ausgewertet.

Löschstellung der schleifspezifischen G-Gruppe (64)

Die Löschstellung bzw. welche der G-Befehle der schleifspezifischen **64. G-Gruppe** nach Kanal-Reset bzw. Power On wirksam wird, wird eingestellt in:

MD20150 \$MC_GCODE_RESET_VALUES[63] = <Wert>

Wert	G-Befehl
1	GFRAME0 (Nullframe)
2	GFRAME1
3	GFRAME2
...	...
101	GFRAME100

Resetverhalten der schleifspezifischen G-Gruppe

Das Resetverhalten der schleifspezifischen **64. G-Gruppe** wird eingestellt in:

MD20152 \$MC_GCODE_RESET_MODE[63] = <Wert>

Wert	Bedeutung Nach Kanal-Reset oder Teileprogrammende:
0	wird der schleifspezifische G-Befehl entsprechend MD20150 aktiv.
1	bleibt die aktuell aktive schleifspezifische G-Befehl weiter aktiv.

Hinweis**MD20110 \$MC_RESET_MODE_MASK**

Das Maschinendatum MD20152 \$MC_GCODE_RESET_MODE wird nur ausgewertet bei:

MD20110 \$MC_RESET_MODE_MASK, Bit0 == 1

Systemvariablen**\$P_GFR[<n>] (Schleifframes der Datenhaltung)**

Über die Systemvariable \$P_GFR[<n>] können die Schleifframes der Datenhaltung gelesen und geschrieben werden. Beim Schreiben eines Schleifframes werden die neuen Werte nicht sofort im Kanal aktiv. Die Aktivierung im Kanal erfolgt erst mit Programmierung des entsprechenden Befehls GFRAME0 ... GFRAME100. Bei NCU-globalen Frames wird das geänderte Frame in jedem Kanal der NCU, der einen GFRAME0 ... GFRAME100-Befehl ausführt, aktiv.

Die Schleifframes der Datenhaltung werden bei einer Datensicherung mit gesichert.

Hinweis**Anzeige (SINUMERIK Operate)**

Die Schleifframes der Datenhaltung werden auf der Bedienoberfläche von SINUMERIK Operate in einem eigenen Fenster angezeigt.

- Literatur
Bedienhandbuch Schleifen; Kapitel "Maschine einrichten" > "Nullpunktverschiebungen" > "Sitzbezogene Feinverschiebung anzeigen und bearbeiten"

Löschen (SINUMERIK Operate)

Die Schleifframes der Datenhaltung können einzeln oder, z. B. nach einem Werkstückwechsel, alle gemeinsam über die Bedienoberfläche von SINUMERIK Operate gelöscht werden.

- Literatur
Bedienhandbuch Schleifen; Kapitel "Maschine einrichten" > "Nullpunktverschiebungen" > "Sitzbezogene Feinverschiebungen löschen"
-

\$P_GFRAME (Aktiver Schleifframe)

Über die Systemvariable \$P_GFRAME kann der im Kanal **aktive** Schleifframe gelesen und geschrieben werden. Beim Schreiben des aktiven Schleifframes werden die neuen Werte sofort im Kanal wirksam.

Bei NCU-globalen Schleifframes wirkt das geänderte Frame nur in dem Kanal, in dem die neuen Frame-Werte programmiert wurde.

Soll das geänderte NCU-globale Schleifframe von einem Kanal ausgehend für alle Kanäle der NCU wirksam werden, muss das im Kanal aktive Schleifframe \$P_GFRAME und das Schleifframe in der Datenhaltung \$P_GFR[<n>] gemeinsam geschrieben werden:

$$\$P_GFRAME = \$P_GFR[<n>] = <\text{neuer Wert}>$$

Damit das geänderte Schleiframe in der Datenhaltung \$P_GFR[<n>] in einem anderen Kanal wirksam wird, muss es in diesem Kanal noch mit dem entsprechenden Befehl GFRAME<n>, aktiviert werden.

\$P_GFRNUM (Nummer des aktiven Schleiframes)

Über die Systemvariable \$P_GFRNUM kann der Index <n> des im Kanal aktiven Schleiframes der Datenhaltung gelesen werden:

Im Kanal aktiver Schleiframe \$P_GFRAME == \$P_GFR[\$P_GFRNUM]

Durch G-Befehl GFRAME<n> aktivierter Schleiframe \$P_GFRAME:	\$P_GFRNUM
GFRAME0	0
GFRAME1	1
GFRAME2	2
...	...
GFRAME100	100

Programmierung

Befehl zur Aktivierung eines Schleiframes im Kanal

Durch Programmierung des Befehles GFRAME<n> wird der entsprechende Schleiframe der Datenhaltung \$P_GFR[<n>] im Kanal aktiv. Dazu wird der aktive Schleiframe \$P_GFRAME gleich dem Schleiframe der Datenhaltung \$P_GFR[<n>] gesetzt:

GFRAME<n> ⇒ \$P_GFRAME = \$P_GFR[<n>]

Befehl	Im Kanal aktivierter Schleiframe
GFRAME0	\$P_GFR[0] (Nullframe)
GFRAME1	\$P_GFR[1]
...	...
GFRAME100	\$P_GFR[100]

Syntax

GFRAME<n>

Bedeutung

GFRAME<n>:	Aktivierung des Schleiframes <n> der Datenhaltung	
	G-Gruppe:	64
	Grundstellung:	MD20150 \$MC_GCODE_RESET_VALUES[63]
	Wirksamkeit:	Modal
<n>:	Nummer des Schleiframes	
	Wertebereich:	0, 1, 2, ... 100

Randbedingungen

Schreiben von Schleifframes durch HMI / PLC

Von HMI oder dem PLC-Anwenderprogramm können nur die Schleifframes der **Datenhaltung** geschrieben werden.

11.5.5.4 Kanal-spezifische Basisframes[<n>]

Maschinendaten

Anzahl der kanalspezifischen Basisframes

Die Anzahl der kanalspezifischen Basisframes wird mit folgendem Maschinendatum eingestellt:

MD28081 \$MC_MM_NUM_BASE_FRAMES = <Anzahl>

Systemvariablen-Index n = 0, 1, 2, ... <Anzahl> - 1

Systemvariablen

\$P_CHBFR[<n>] (Kanalspezifische Basisframes der Datenhaltung)

Über die Systemvariablen \$P_CHBFR[<n>] können die kanalspezifischen Basisframes der Datenhaltung gelesen und geschrieben werden. Beim Schreiben eines kanalspezifischen Basisframes werden die neuen Werte nicht sofort im Kanal aktiv. Die Aktivierung im Kanal erfolgt erst mit Programmierung des entsprechenden Befehls G500, G54..G599.

Die kanalspezifischen Basisframes der Datenhaltung werden bei einer Datensicherung mit gesichert.

\$P_CHBFRAME[<n>] (Aktive kanalspezifische Basisframes)

Über die Systemvariablen \$P_CHBFRAME[<n>] können die im Kanal **aktiven** kanalspezifischen Basisframes gelesen und geschrieben werden. Beim Schreiben eines aktiven kanalspezifischen Basisframes werden die neuen Werte im Kanal sofort durch Neuberechnung des aktiven Gesamt-Basisframe \$P_ACTBFRAME wirksam.

Systemvariablen aus Kompatibilitätsgründen

\$P_UBFR (Erster kanalspezifischer Basisframe der Datenhaltung)

Die Systemvariable bleibt aus Kompatibilitätsgründen erhalten, obwohl sie redundant zu der Variablen \$P_CHBFR[0] ist.

Ein Schreiben auf die vordefinierte Variable \$P_UBFR aktiviert das Basisframe mit dem Feldindex 0 nicht gleichzeitig, sondern die Aktivierung erfolgt erst mit der Ausführung einer G500, G54, ..G599-Anweisung. Bei NCU-globalen Frames wird das geänderte Frame in jedem Kanal der NCU, der eine G500, G54..G599-Anweisung ausführt, aktiv. Die Variable dient vorwiegend als Speicher für Schreibvorgänge auf das Basisframe von HMI oder PLC. Die Variable kann auch im Programm geschrieben und gelesen werden.

\$P_UBFR ist identisch mit \$P_CHBFR[0]. Standardmäßig gibt es immer ein Basisframe im Kanal, so dass die Systemvariable kompatibel zu älteren Ständen ist. Gibt es kein kanalspezifisches Basisframe, so wird beim Schreiben oder Lesen der Alarm "Frame: Anweisung unzulässig" ausgegeben.

\$P_BFRAME (Erster aktiver kanalspezifischer Basisframe)

Die Systemvariable bleibt aus Kompatibilitätsgründen erhalten, obwohl sie redundant zu der Variablen \$P_CHBFRAME[0] ist.

Über die vordefinierte Framevariable \$P_BFRAME kann das aktuelle Basisframe mit dem Feldindex 0, welches im Kanal gültig ist, im Teileprogramm gelesen und geschrieben werden. Das geschriebene Basisframe wird sofort eingerechnet. Bei NCU-globalen einstellbaren Frames wirkt das geänderte Frame nur in dem Kanal, in dem das Frame programmiert wurde. Soll das Frame für alle Kanäle einer NCU geändert werden, so muss gleichzeitig \$P_UBFR und \$P_BFRAME beschrieben werden. Die anderen Kanäle müssen dann noch das entsprechende Frame mit z.B. G54 aktivieren.

\$P_BFRAME ist identisch mit \$P_CHBFRAME[0]. Die Systemvariable hat standardmäßig immer einen gültigen Wert. Gibt es kein kanalspezifisches Basisframe, so wird beim Schreiben oder Lesen der Alarm "Frame: Anweisung unzulässig" ausgegeben.

Randbedingungen

Schreiben von Basisframes durch HMI / PLC

Von HMI oder dem PLC-Anwenderprogramm können nur die Basisframes der **Datenhaltung** geschrieben werden.

11.5.5.5 NCU-globale Basisframes \$P_NCBFR[<n>]

Maschinendaten

Anzahl NCU-globale Basisframes

Die Anzahl der NCU-globalen Basisframes wird mit folgendem Maschinendatum eingestellt:

MD18602 \$MN_MM_NUM_GLOBAL_BASE_FRAMES = <Anzahl>

Systemvariablen-Index n = 0, 1, 2, ... <Anzahl> - 1

Systemvariablen

\$P_NCBFR[<n>] (NCU-globale Basisframes der Datenhaltung)

Über die Systemvariablen \$P_NCBFR[<n>] können die NCU-globalen Basisframes der Datenhaltung gelesen und geschrieben werden. Beim Schreiben eines NCU-globalen Basisframes werden die neuen Werte nicht sofort im Kanal aktiv. Die Aktivierung im Kanal erfolgt erst mit Programmierung des entsprechenden Befehls G500, G54..G599.

Die NCU-globalen Basisframes der Datenhaltung werden bei einer Datensicherung mit gesichert.

\$P_NCBFRAME[<n>] (Aktuelle NCU-globale Basisframes)

Über die Systemvariablen \$P_NCBFRAME[<n>] können die im Kanal **aktiven** NCU-globalen Basisframes gelesen und geschrieben werden. Beim Schreiben eines aktiven NCU-globalen Basisframes werden die neuen Werte im Kanal sofort durch Neuberechnung des aktiven Gesamt-Basisframe \$P_ACTBFRAME wirksam.

Soll das geänderte NCU-globale Basisframe von einem Kanal ausgehend für alle Kanäle der NCU wirksam werden, muss das im Kanal aktive NCU-globale Basisframe und das NCU-globale Basisframe der Datenhaltung gemeinsam geschrieben werden:

\$P_NCBFR[<n>] = \$P_NCBFRAME = <neuer Wert>

- \$P_NCBFR[<n>] (NCU-globaler Basisframe der Datenhaltung)
- \$P_NCBFRAME (im Kanal aktiver NCU-globaler Basisframe)

Damit das geänderte NCU-globale Basisframe in einem anderen Kanal wirksam wird, muss es in diesem Kanal noch mit dem entsprechenden Befehl G500, G54..G599, aktiviert werden.

Programmierung

Ein kanalspezifischer Einstellbarer Frame der Datenhaltung \$P_UIFR[<n>] wird durch den entsprechenden Befehl (G54 ... G57, G505 ... G599 und G500) zum im Kanal aktiven Einstellbarer Frame \$P_IFRAME.

Befehl	Aktivierung der NCU-globalen und kanalspezifischen Basisframes der Datenhaltung
G500	\$P_CHBFR[0] : \$P_NCBFR[0]
G54	\$P_CHBFR[1] : \$P_NCBFR[1]
G55	\$P_CHBFR[2] : \$P_NCBFR[2]
G56	\$P_CHBFR[3] : \$P_NCBFR[3]
G57	\$P_CHBFR[4] : \$P_NCBFR[4]
G505	\$P_CHBFR[5] : \$P_NCBFR[5]
...	...
G599	\$P_CHBFR[99] : \$P_NCBFR[99]

11.5.5.6 Aktiver Gesamt-Basisframe \$P_ACTBFRAME**Funktion**

Im Gesamt-Basisframe \$P_ACTBFRAME sind alle aktiven NCU-globalen und kanalspezifischen Basisframes zusammengefasst:

\$P_ACTBFRAME = \$P_NCBFRAME[0] : ... : \$P_NCBFRAME[<n>] :
\$P_CHBFRAME[0] : ... : \$P_CHBFRAME[<n>]

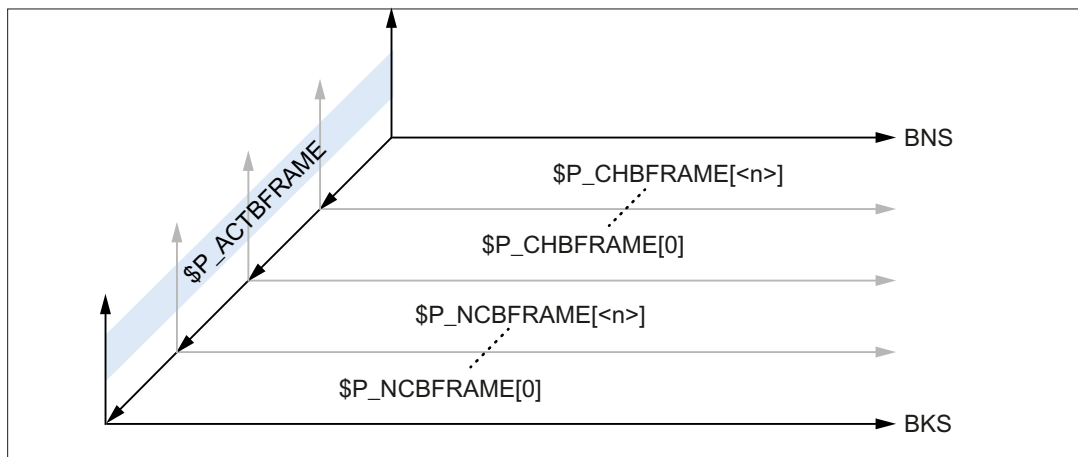


Bild 11-24 Gesamt-Basisframe

Maschinendaten

Reset-Verhalten

Welche Basisframes nach Reset (Kanal-Reset, Programmende-Reset bzw. Power On) aktiv sind, wird eingestellt über das Maschinendatum:

MD20110 \$MC_RESET_MODE_MASK, Bit0 = 1 und Bit14 = 1

- Bit0 = 1: Standardwert \Rightarrow Reset-Verhalten entsprechend der Einstellung der weiteren Bits
- Bit14 = 0: Mit Reset werden die Basisframes vollständig abgewählt.
- Bit14 = 1: Mit Reset werden die Maschineneinstellungen in die Systemvariablen übernommen und die darin angewählten Basisframes aktiv:
 - \$P_NCBFRMASK = MD10613 \$MN_NCBFRAME_RESET_MASK
 - \$P_CHBFRMASK = MD24002 \$MC_CHBFRAME_RESET_MASK

Beispiel

Maschineneinstellung	Aktive Basisframes
\$P_NCBFRMASK = MD10613 \$MN_NCBFRAME_RESET_MASK = 'H81'	\$P_NCBFRAME[0] : \$P_NCBFRAME[7]

Programmierung

Basisframemasken

Über die Basisframemasken \$P_NCBFRMASK und \$P_CHBFRMASK werden die Basisframes ausgewählt, die zum Gesamt-Basisframe verkettet werden.

Durch Setzen eines Bits in der Basisframemaske wird der entsprechende Basisframe ausgewählt:

- \$P_NCBFRMASK, Bit0, 1, 2, ... n \Rightarrow \$P_NCBFRAME[0, 1, 2, ... n]
- \$P_CHBFRMASK, Bit0, 1, 2, ... n \Rightarrow \$P_NCHFRAME[0, 1, 2, ... n]

Die Basisframemasken `$P_NCBFRMASK` und `$P_CHBFRMASK` können nur im NC-Programm gelesen / geschrieben werden. Über BTSS können die Basisframemasken gelesen werden.

Nach dem Schreiben einer Basisframemaske wird das aktive Gesamtbasisframe `$P_ACTBFRAME` und Gesamtframe `$P_ACTFRAME` neu berechnet.

Beispiel

Programmcode	Kommentar
<code>\$P_NCBFRMASK = 'H81'</code>	<code>; Aktive NCU-globale Basisframes: \$P_NCBFRA-</code> <code>ME[0] : \$P_NCBFRAME[7]</code>

11.5.5.7 Programmierbarer Frame `$P_PFRAME`

Den programmierbaren Frame gibt es nur als aktiven Frame. Dieser Frame ist für den Programmierer reserviert.

Das programmierbare Frame bleibt bei RESET erhalten, wenn:

`MD24010 $MC_PFRAME_RESET_MODE` (Reset-Modus für programmierbaren Frame) = 1

Diese Funktionalität ist vor allem wichtig, wenn man nach RESET noch aus einer schrägen Bohrung herausfahren möchte.

MIRROR

Spiegelungen einer Geometrieachse wurden bisher (bis SW-P4) anhand des Maschinendatums:

`MD10610 $MN_MIRROR_REF_AX` (Bezugsachse für das Spiegeln) auf eine festgelegte Bezugsachse bezogen.

Diese Festlegung ist aus Anwendersicht schwer nachvollziehbar. Bei Spiegelung der z-Achse wurde in der Anzeige übermittelt, dass die x-Achse gespiegelt ist und die y-Achse um 180° gedreht wurde. Bei Spiegelungen von zwei Achsen wird das noch komplexer und es ist nicht mehr leicht nachvollziehbar, welche Achsen gespiegelt wurden und welche nicht.

Ab SW-P5 gibt es die zusätzliche Möglichkeit das Spiegeln einer Achse eindeutig darzustellen. Ein Spiegeln wird dann nicht auf ein Spiegeln einer Bezugsachse und Verdrehungen von anderen Achsen abgebildet.

Projektiert werden kann diese Einstellung durch:

`MD10610 $MN_MIRROR_REF_AX = 0`

Die Programmierung des programmierbaren Frames mit `MIRROR` und `AMIRROR` wird erweitert. Bisher wurde der angegebene Wert der Koordinatenachse, wie z.B. bei `MIRROR X0` der Wert 0, nicht ausgewertet, sondern `AMIRROR` hatte eine Toggle-Funktion, d. h. `MIRROR X0` schaltet die Spiegelung ein und ein weiteres `AMIRROR X0` schaltet sie wieder aus. `MIRROR` wirkt immer absolut und `AMIRROR` additiv.

Mit der Maschinendatum-Einstellung:

`MD10612 $MN_MIRROR_TOGGLE = 0` ("Mirror Toggle")

kann festgelegt werden, dass die programmierten Werte ausgewertet werden.

Bei einem Wert von 0, wie bei `AMIRROR X0`, wird die Spiegelung der Achse ausgeschaltet, und bei Werte ungleich 0 wird die Achse gespiegelt, wenn sie noch nicht gespiegelt ist.

Das komponentenweise Lesen oder Schreiben von Spiegelungen ist unabhängig vom Maschinendatum:

MD10612 \$MN_MIRROR_TOGGLE

Ein Wert = 0 bedeutet, dass danach die Achse nicht gespiegelt ist und ein Wert = 1 heißt, dass die Achse danach immer gespiegelt wird, egal, ob die Achse schon gespiegelt war oder nicht.

```
$P_NCBFR[0,x,mi]=1      ; x-Achse wird immer gespiegelt.
$P_NCBFR[0,x,mi]=0      ; x-Achse spiegeln aus.
```

Achsspezifisches Ersetzen G58, G59 (nur 840D sl)

Der Translationsanteil des programmierbaren Frames wird in einen absoluten Anteil und in einen Anteil für die Summe aller additiv-programmierten Translationen aufgespalten. Der absolute Anteil kann über TRANS, CTRANS oder durch Schreiben der Translationskomponente geändert werden, wobei der additive Anteil zu Null gesetzt wird. G58 verändert nur den absoluten Translationsanteil für die angegebene Achse, die Summe der additiv-programmierten Translationen bleibt erhalten.

G58 X... Y... Z... A... ..

G59 dient zum achsspezifischen Überschreiben der additiv programmierten Translationen für die angegebenen Achsen, die mit ATRANS programmiert wurden.

G59 X... Y... Z... A... ..

Beispiel

```
TRANS X10 Y10 Z10
ATRANS X5 Y5          ; Gesamttranslationen X15 Y15 Z10
G58 X20               ; Gesamttranslationen X25 Y15 Z10
G59 X10 Y10           ; Gesamttranslationen X30 Y20 Z10
```

G58 und G59 ist nur einsetzbar, wenn:

MD24000 \$MC_FRAME_ADD_COMPONENTS (Framekomponenten für G58 / G59) == TRUE

Nachfolgende Tabelle beschreibt die Wirkung von diversen Programmbefehlen auf die absolute und die additive Translation.

	Grob- bzw. absolute Verschiebung	Fine- bzw. additive Verschiebung
TRANS X10	10	0
ATRANS X10	unverändert	alt_fine + 10
CTrans (X, 10)	10	0
CTrans ()	0	0
CFine (X, 10)	0	10
\$P_PFRAME[X,TR] = 10	10	unverändert
\$P_PFRAME[X,FI] = 10	unverändert	10
G58 X10	10	unverändert
G59 X10	unverändert	10

11.5.5.8 Kanalspezifische Systemframes

Kanalspezifische Systemframes werden nur von Systemfunktionen wie Istwertsetzen, Ankratzen, externe Nullpunktverschiebung, Schrägbearbeitung etc. geschrieben.

Maschinendaten

Parametrierung der kanalspezifischen Systemframes

Aus Speicherplatzgründen sollten nur die kanalspezifische Systemframes projiziert werden, deren Systemfunktionen tatsächlich verwendet werden.

Jedes Systemframe belegt pro Kanal ca. 1 kByte statischen und ca. 6 kByte dynamischen Speicher.

Die Parametrierung der kanalspezifischen Systemframes erfolgt in folgenden Maschinendatum:

MD28082 \$MC_MM_SYSTEM_FRAME_MASK, Bit<n>

Bit	Wert	Systemframe vorhanden:
0	1	\$P_SETFR: Istwertsetzen und Ankratzen
1	1	\$P_EXTFR: Externe Nullpunktverschiebung über Systemframe
2	1	\$P_PARTFR: TCARR und PAROT bei orientierbarem Werkzeugträger
3	1	\$P_TOOLFR: TOROT und TOFRAME
4	1	\$P_WPFR: Frame für Werkstückbezugspunkte
5	1	\$P_CYCFR: Frame für Zyklen
6	1	\$P_TRAFR: Frame für An- und Abwahl von Transformationen
7	1	\$P_ISO1FRAME: Frame für G51 . 1 Spiegeln (ISO)
8	1	\$P_ISO2FRAME: Frame für G68 2DROT (ISO)
9	1	\$P_ISO3FRAME: Frame für G68 3DROT (ISO)
10	1	\$P_ISO4FRAME: Frame für G51 Scale (ISO)
11	1	\$P_RELFR: Frame für relative Koordinatensysteme

Parametrierung des ENS (ACS)-Koordinatensystems

Über das folgende Maschinendatum wird festgelegt, welche Systemframes das ENS (ACS)-Koordinatensystem bilden

MD24030 \$MC_FRAME_ACS_SET = <Wert>

<Wert>	Bedeutung: Das ENS (ACS)-Koordinatensystem besteht aus
0	\$P_PARTFRAME : \$P_SETFRAME : \$P_EXTFRAME : \$P_ISO1FRAME : \$P_ISO2FRAME : \$P_ISO3FRAME : \$P_ACTBFRAME : \$P_IFRAME : \$P_GFRAME : \$P_TOOLFRAME : \$P_WPFRAME
1	\$P_PARTFRAME : \$P_SETFRAME : \$P_EXTFRAME : \$P_ISO1FRAME : \$P_ISO2FRAME : \$P_ISO3FRAME : \$P_ACTBFRAME : \$P_IFRAME : \$P_GFRAME : \$P_TOOLFRAME : \$P_WPFRAME : \$P_TRAFRAME : \$P_PFRAME : \$P_ISO4FRAME

Systemvariablen

Kanalspezifische Systemframes der Datenhaltung

Die kanalspezifischen Systemframes der Datenhaltung können über folgende Framevariablen geschrieben / gelesen werden:

Systemvariable	Bedeutung: Systemframe der Datenhaltung für
\$P_SETFR	Istwertsetzen und Ankratzen (Set-Frame)
\$P_EXTFR	Externe Nullpunktverschiebung (Ext-Frame)
\$P_PARTFR	TCARR und PAROT bei orientierbarem Werkzeugträger (Part-Frame)
\$P_TOOLFR	TOROT und TOFRAME (Tool-Frame)
\$P_WPFR	Werkstückbezugspunkte (Work-Piece-Frame)
\$P_CYCFR	Zyklen (Cycle-Frame)
\$P_TRAFR	Transformationen (Transformation-Frame)
\$P_ISO1FR	G51 . 1 Spiegeln (ISO)
\$P_ISO2FR	G68 2DROT (ISO)
\$P_ISO3FR	G68 3DROT (ISO)
\$P_ISO4FR	G51 Scale (ISO)
\$P_RELFR	relative Koordinatensysteme

Hinweis

Zyklenprogrammierung

Die Framevariablen der Systemframes sind ausschließlich für die Zyklen-Programmierung vorgesehen. In NC-Programmen sollten die Systemframes daher vom Anwender nicht direkt, sondern nur über Systemfunktionen wie TOROT, PAROT, etc. geschrieben werden.

Kanalspezifische aktive Systemframes

Systemvariablen der aktiven kanalspezifischen Systemframes:

Systemvariable	Bedeutung: Aktiver Systemframe für
\$P_SETFRAME	Istwertsetzen und Ankratzen (Set-Frame)
\$P_EXTFRAME	Externe Nullpunktverschiebung (Ext-Frame)
\$P_PARTFRAME	TCARR und PAROT bei orientierbarem Werkzeugträger (Part-Frame)
\$P_TOOLFRAME	TOROT und TOFRAME (Tool-Frame)
\$P_WPFRAME	Werkstückbezugspunkte (Work-Piece-Frame)
\$P_CYCFRAME	Zyklen (Cycle-Frame)
\$P_TRAFRAME	Transformationen (Transformation-Frame)
\$P_ISO1FRAME	G51 . 1 Spiegeln (ISO)
\$P_ISO2FRAME	G68 2DROT (ISO)
\$P_ISO3FRAME	G68 3DROT (ISO)
\$P_ISO4FRAME	G51 Scale (ISO)
\$P_RELFRAME	relative Koordinatensysteme

Ist ein kanalspezifisches Systemframe der Datenhaltung nicht parametrierbar, gilt für das entsprechende aktive Systemframe: \$P_<Systemframe> == Nullframe.

Kanalspezifische aktives ENS (ACS)-Gesamtframe

Über die Systemvariable \$P_ACSFRAME kann gelesen werden, welche Systemframes das ENS (ACS)-Koordinatensystem bilden. Die Festlegung erfolgt über das oben beschriebenen Maschinendatum MD24030 \$MC_FRAME_ACS_SET. Siehe Absatz "Maschinendaten" > "Parametrierung des ENS (ACS)-Koordinatensystems"

Systemvariable	Bedeutung: Aktiver Systemframe für
\$P_ACSFRAME	Systemframes, die das ENS (ACS)-Koordinatensystem entsprechend der Parametrierung in MD24030 \$MC_FRAME_ACS_SET bilden

11.5.6 Implizite Frame-Änderungen**11.5.6.1 Umschalten von Geometrieachsen**

Welche Kanalachsen die Geometrieachsen des Kanals sind, kann sich sowohl beim Ein-/ Ausschalten einer Transformation und beim Befehl `GEOAX()` ändern.

Über das folgende Maschinendatum können vier verschiedenen Einstellungen zur Behandlung des aktuellen Gesamtframe \$P_ACTFRAME vorgenommen werden:

MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = <Wert>

<Wert>	Bedeutung
0	<p>Löschen</p> <p>Bei Umschaltungen von Geometrieachsen, wie bei An- und Abwahl von Transformationen und <code>GEOAX()</code> wird das aktuelle Gesamtframe gelöscht. Erst nach Aktivieren eines neuen Frames wird dann die geänderte Geometrieachskonstellation berücksichtigt.</p>
1	<p>Neuberechnung mit Erhalt der Drehungen</p> <p>Das aktuelle Gesamtframe wird beim Umschalten von Geometrieachsen neu berechnet, wobei die Frame-Anteile der neuen Geometrieachsen wirksam werden. Die Drehungen der Geometrieachsen, die vor dem Umschalten programmiert wurden, bleiben auch für die neuen Geometrieachsen wirksam.</p> <p>Für <code>TRANSMIT</code>, <code>TRACYL</code> und <code>TRAANG</code> siehe Kapitel "An- und Abwahl von Transformationen: Allgemein (Seite 800)".</p>

<Wert>	Bedeutung
2	<p>Neuberechnung nur, wenn keine Drehungen aktiv waren</p> <p>Das aktuelle Gesamtframe wird beim Umschalten von Geometrieachsen neu berechnet, wobei die Frame-Anteile der neuen Geometrieachsen wirksam werden. Sind vor der Umschaltung in den aktuellen Basisframes, dem aktuellen Einstellbaren Frame oder im Programmierbaren Frame, Drehungen aktiv, wird die Umschaltung mit Alarm "Frame: Umschaltung der Geometrieachsen unzulässig" abgebrochen.</p> <p>Für TRANSMIT, TRACYL und TRAANG siehe Kapitel "An- und Abwahl von Transformationen: Allgemein (Seite 800)".</p>
3	<p>Transformation: Löschen / GEOAX(): Neuberechnung mit Erhalt der Drehungen</p> <ul style="list-style-type: none"> Transformation: Das aktuelle Gesamtframe wird bei An- und Abwahl von Transformationen gelöscht. GEOAX(): Bei GEOAX() wird das Gesamtframe neu berechnet, wobei die Translationen, Skalierungen und Spiegelungen der neuen achsspezifischesachsspezifisches wirksam werden. Die Drehungen der Geometrieachsen, die vor dem Umschalten programmiert wurden, bleiben auch für die neuen Geometrieachsen wirksam.

Die Werkstückgeometrie wird von einem Koordinatensystem beschrieben, das von den Geometrieachsen aufgespannt wird. Jeder Geometrieachse ist eine Kanalachse und jeder Kanalachse eine Maschinenachse zugeordnet. Für jedes Frame (Systemframe, Basisframes, einstellbare Frames, prog. Frame) gibt es für jede Maschinenachse ein achsspezifisches Frame. Wird einer Geometrieachse eine andere Maschinenachse zugeordnet, bringt die Maschinenachse ihren eigenen achsspezifischen Frameanteile mit. Die neue Geometrie im Kanal wird dann von den neuen Konturframes gebildet, die sich aus den bis zu drei neuen Geometrieachsen bilden.

Die aktuell gültigen Frames werden bei der Umschaltung von Geometrieachsen neu berechnet und ein resultierendes Gesamtframe daraus gewonnen. Die Datenhaltungsframes werden erst nach Aktivierung berücksichtigt.

Beispiel

Die Kanalachse A soll durch einen GeoAx-Tausch zur Geometrieachse X werden. Durch den Tausch soll das Programmierbare Frame einen Trans-Anteil von 10 in der X-Achse haben. Das aktuelle einstellbare Frame soll erhalten bleiben.

MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 1

Programmcode	Kommentar
\$P_UIFR[1] = CROT(X,10,Y,20,Z,30)	; Frame bleibt nach GeoAx-Tausch erhalten.
G54	; Einstellbares Frame wird aktiv.
TRANS A10	; Achsiale Verschiebung von A wird mit getauscht.
GEOAX(1, A)	; A wird zur X-Achse
	; \$P_ACTFRAME = CROT(X,10,Y,20,Z,30) : CTRANS(X10)

Bei Transformationswechsel können gleichzeitig mehrere Kanalachsen zu Geometrieachsen werden.

Beispiel

Durch eine 5-Achs-Orientierungstranformation werden die Kanalachsen 4, 5 und 6 zu Geometrieachsen der Transformation. Die Geometrieachsen vor der Transformation werden

also alle ersetzt. Beim Einschalten der Transformation ändern sich alle aktuellen Frames. Zur Berechnung des neuen WKS-Systems werden die achsialen Frameanteile der Kanalachsen, die zu Geometrieachsen werden, berücksichtigt. Programmierte Drehungen vor der Transformation werden beibehalten. Nach dem Ausschalten der Transformation wird das alte WKS wieder hergestellt. Der häufigste Anwendungsfall wird wohl der sein, dass sich die Geometrieachsen vor und nach der Transformation nicht ändern und die Frames so beibehalten werden sollen, wie sie vor der Transformation waren.

Maschinendaten

Programmcode

```
$MC_AXCONF_CHANAX_NAME_TAB[0] = "CAX"
$MC_AXCONF_CHANAX_NAME_TAB[1] = "CAY"
$MC_AXCONF_CHANAX_NAME_TAB[2] = "CAZ"
$MC_AXCONF_CHANAX_NAME_TAB[3] = "A"
$MC_AXCONF_CHANAX_NAME_TAB[4] = "B"
$MC_AXCONF_CHANAX_NAME_TAB[5] = "C"
```

```
$MC_AXCONF_GEOAX_ASSIGN_TAB[0] = 1
$MC_AXCONF_GEOAX_ASSIGN_TAB[1] = 2
$MC_AXCONF_GEOAX_ASSIGN_TAB[2] = 3
```

```
$MC_AXCONF_GEOAX_NAME_TAB[0] = "X"
$MC_AXCONF_GEOAX_NAME_TAB[1] = "Y"
$MC_AXCONF_GEOAX_NAME_TAB[2] = "Z"
```

```
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[0]=4
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[1]=5
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[2]=6
```

```
$MC_TRAFO_AXES_IN_1[0]=4
$MC_TRAFO_AXES_IN_1[1]=5
$MC_TRAFO_AXES_IN_1[2]=6
$MC_TRAFO_AXES_IN_1[3]=1
$MC_TRAFO_AXES_IN_1[4]=2
```

Programm:

Programmcode

```
$P_NCBFRAME[0] = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6)
$P_CHBFRAME[0] = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6)
$P_IFRAME = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6):CROT(Z,45)
$P_PFRAME = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6):CROT(X,10,Y,20,Z,30)
```

Programmcode	Kommentar
TRAORI	; Trafo setzt GEOAX(4,5,6)
	; \$P_NCBFRAME[0] = CTRANS(X,4,Y,5,Z,6,CAX,1,CAY,2,CAZ,3)

Programmcode	Kommentar
	; \$P_ACTBFRAME =CTTRANS(X,8,Y,10,Z,12,CAX,2,CAY,4,CAZ,6)
	; \$P_PFRAME = CTRANS(X,4,Y,5,Z,6,CAX,1,CAY,2,CAZ,3):CROT(X,10,Y,20,Z,30)
	; \$P_IFRAME = CTRANS(X,4,Y,5,Z,6,CAX,1,CAY,2,CAZ,3):CROT(Z,45)
TRAFOOF	; Ausschalten der Transformation setzt GEOAX(1,2,3)
	; \$P_NCBFRAME[0] = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6)
	; \$P_CHBFRAME[0] = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6)
	; \$P_IFRAME = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6):CROT(Z,45)
	; \$P_PFRAME = CTRANS(X,1,Y,2,Z,3,A,4,B,5,C,6):CROT(X,10,Y,20,Z,30)

11.5.6.2 An- und Abwahl von Transformationen: Allgemein

Bei An- und Abwahl von Transformationen ändert sich in der Regel die Zuordnung der Geometrieachsen zu den Kanalachsen. Bei Transformationen, bei denen Rundachsen zu Linearachsen und Linearachsen zu Rundachsen werden, ist keine eindeutige Zuordnung von achsspezifischen Frameanteilen zu geometrischen Konturframeanteilen möglich. Bei diesen nichtlinearen Transformationen muss eine Sonderbehandlung zur Aufbereitung des Konturframes erfolgen.

Es wird der Modus, der durch MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 1 und 2 eingestellt wird, so erweitert, dass auch die o. g. Transformationen berücksichtigt werden.

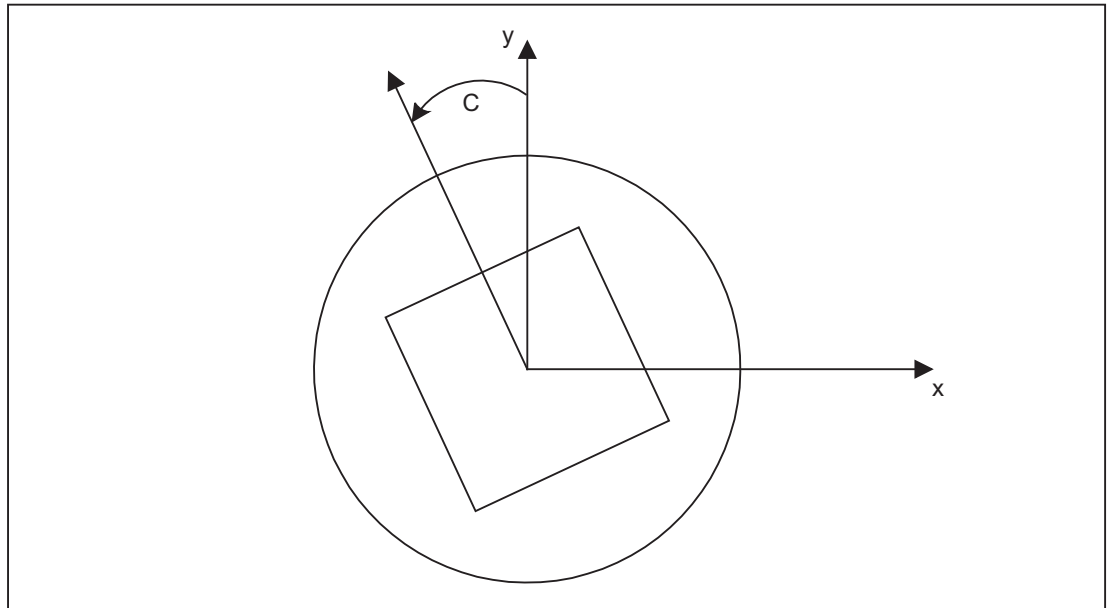
Bei Transformationsanwahl wird das Konturframe anhand der achsspezifischen Frames aufgebaut. Die virtuelle Geometrieachse der Transformationen TRANSMIT, TRACYL und TRAANG unterliegt dabei einer Sonderbehandlung.

Hinweis

Transformationen mit virtuellen Achsen

Bei der Anwahl von TRANSMIT oder TRACYL werden Verschiebungen, Skalierungen und Spiegelungen der realen Y-Achse nicht in die virtuelle Y-Achse übernommen. Verschiebungen, Skalierungen und Spiegelungen der virtuellen Y-Achse werden bei TRAFOOF gelöscht.

11.5.6.3 An- und Abwahl von Transformationen: TRANSMIT



Transmit-Erweiterungen

Der achsspezifische Gesamtframe der TRANSMIT-Rundachse, d. h. Translation, Spiegelung und Skalierung, kann über folgende Maschinendaten in der Transformation berücksichtigt werden:

- MD24905 \$MC_TRANSMIT_ROT_AX_FRAME_1 = 1
- MD24955 \$MC_TRANSMIT_ROT_AX_FRAME_2 = 1

Eine Verschiebung der Rundachse kann z. B. durch eine Kompensation der Schräglage eines Werkstückes in einem Frame innerhalb der Framekette eingetragen werden. Diese Verschiebung soll in der Regel auch in der Transformation als Offset der Rundachse berücksichtigt werden. Eine Verschiebung der C-Achse, wie im obigen Bild, führt dann zu entsprechenden X- und Y-Werten.

- MD24905 \$MC_TRANSMIT_ROT_AX_FRAME_1 = 2
- MD24955 \$MC_TRANSMIT_ROT_AX_FRAME_2 = 2

Mit dieser Einstellung wird die achsspezifische Verschiebung der Rundachse bis zum ENS in der Transformation berücksichtigt. Die in den ENS-Frames enthaltenen achsspezifischen Verschiebungen der Rundachse werden in das Transformationsframe als Drehung eingetragen. Diese Einstellung ist nur wirksam, wenn das Transformationsframe projiziert ist.

Frame-Erweiterungen

Nachfolgend beschriebene Erweiterungen gelten nur für folgende Maschinendaten-Einstellungen:

- MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 1
- MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 2

Mit Anwahl der Transformation TRANSMIT entsteht, gekoppelt über die Rundachse, eine virtuelle Geometrieachse, die keinen Bezug zu einem achsspezifischen Frame hat, sondern

nur im Konturframe berücksichtigt wird. Der geometrische Wert ergibt sich aus der Drehung einer Rundachse. Alle anderen Geometrieachsen übernehmen bei Trafo-Anwahl ihre achsspezifischen Anteile.

Komponenten:

- Translationen
Die Translationen der virtuellen Achse werden bei TRANSMIT-Anwahl gelöscht. Die Translationen der Rundachse können in der Transformation berücksichtigt werden.
- Drehungen
Die Drehungen vor der Transformation werden übernommen.
- Spiegelungen
Die Spiegelung der virtuellen Achse wird gelöscht. Die Spiegelung Rundachse kann in der Transformation berücksichtigt werden.
- Skalierungen
Die Skalierung der virtuellen Achse wird gelöscht. Die Skalierung der Rundachse kann in der Transformation berücksichtigt werden.

Beispiel: Maschinendaten

```

; FRAME-Projektierungen

$MC_MM_SYSTEM_FRAME_MASK='H41'           ; TRAFRAME, SETFRAME
$MC_CHSFRAME_RESET_MASK='H41'             ; Frames sind nach Reset aktiv.
$MC_CHSFRAME_POWERON_MASK='H41'           ; Frames werden bei Power On gelöscht.

$MN_FRAME_GEOAX_CHANGE_MODE=1              ; Frames werden nach GeoAx-Umschaltung
                                           umgerechnet.
$MC_RESET_MODE_MASK='H4041'               ; Basisframe wird nach RESET nicht abge-
                                           wählt.
; $MC_RESET_MODE_MASK='H41'                ; Basisframe wird nach RESET abgewählt.

; $MC_GCODE_RESET_VALUES[7]=2              ; G54 ist Voreinstellung.
$MC_GCODE_RESET_VALUES[7]=1               ; G500 ist Voreinstellung.

$MN_MM_NUM_GLOBAL_USER_FRAMES=0
$MN_MM_NUM_GLOBAL_BASE_FRAMES=3

$MC_MM_NUM_USER_FRAMES=10                  ; von 5 bis 100
$MC_MM_NUM_BASE_FRAMES=3                  ; von 0 bis 8

$MN_NCBFRAME_RESET_MASK='HFF'
$MC_CHBFRAME_RESET_MASK='HFF'

$MN_MIRROR_REF_AX=0                       ; Keine Normierung bei der Spiegelung.
$MN_MIRROR_TOGGLE=0

$MN_MM_FRAME_FINE_TRANS=1                 ; Feinverschiebung
$MC_FRAME_ADD_COMPONENTS=TRUE              ; G58, G59 ist möglich.

```

```
; TRANSMIT ist 1. Trafo
$MC_TRAFO_TYPE_1=256

$MC_TRAFO_AXES_IN_1[0]=1
$MC_TRAFO_AXES_IN_1[1]=6
$MC_TRAFO_AXES_IN_1[2]=3
$MC_TRAFO_AXES_IN_1[3]=0
$MC_TRAFO_AXES_IN_1[4]=0

$MA_ROT_IS_MODULO[AX6]=TRUE;

$MC_TRAFO_GEOAX_ASSIGN_TAB_1[0]=1
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[1]=6
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[2]=3

$MC_TRANSMIT_BASE_TOOL_1[0]=0.0
$MC_TRANSMIT_BASE_TOOL_1[1]=0.0
$MC_TRANSMIT_BASE_TOOL_1[2]=0.0

$MC_TRANSMIT_ROT_AX_OFFSET_1=0.0
$MC_TRANSMIT_ROT_SIGN_IS_PLUS_1=TRUE

$MC_TRANSMIT_ROT_AX_FRAME_1=1

; TRANSMIT ist 2. Trafo
$MC_TRAFO_TYPE_2=256

$MC_TRAFO_AXES_IN_2[0]=1
$MC_TRAFO_AXES_IN_2[1]=6
$MC_TRAFO_AXES_IN_2[2]=2
$MC_TRAFO_AXES_IN_2[3]=0
$MC_TRAFO_AXES_IN_2[4]=0

$MC_TRAFO_GEOAX_ASSIGN_TAB_2[0]=1
$MC_TRAFO_GEOAX_ASSIGN_TAB_2[1]=6
$MC_TRAFO_GEOAX_ASSIGN_TAB_2[2]=2

$MC_TRANSMIT_BASE_TOOL_2[0]=4.0
$MC_TRANSMIT_BASE_TOOL_2[1]=0.0
$MC_TRANSMIT_BASE_TOOL_2[2]=0.0

$MC_TRANSMIT_ROT_AX_OFFSET_2=19.0
$MC_TRANSMIT_ROT_SIGN_IS_PLUS_2=TRUE
```

```
$MC_TRANSMIT_ROT_AX_FRAME_2=1
```

Beispiel: Teileprogramm

```
; Frameeinstellungen
N820 $P_UIFR[1] = ctrans(x,1,y,2,z,3,c,4)
N830 $P_UIFR[1] = $P_UIFR[1] : crot(x,10,y,20,z,30)
N840 $P_UIFR[1] = $P_UIFR[1] : cmirror(x,c)
N850
N860 $P_CHBFR[0] = ctrans(x,10,y,20,z,30,c,15)
N870

; Werkzeuganwahl, Aufspannkomensation, Ebenenanwahl
N890 T2 D1 G54 G17 G90 F5000 G64 SOFT
N900

; Anfahren der Ausgangsstellung
N920 G0 X20 Z10
N930
N940 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,C,15)
N950 setal(61000)
N960 endif
N970 if $P_BFRAME <> $P_CHBFR[0]
N980 setal(61000)
N990 endif
N1000 if $P_IFRAME <>
CTRANS(X,1,Y,2,Z,3,C,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1010 setal(61000)
N1020 endif
N1030 if $P_IFRAME <> $P_UIFR[1]
N1040 setal(61000)
N1050 endif
N1060 if $P_ACTFRAME <>
CTRANS(X,11,Y,22,Z,33,C,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1070 setal(61000)
N1080 endif
N1090
N1100 TRANSMIT(2)
N1110
N1120 if $P_BFRAME <> CTRANS(X,10,Y,0,Z,20,CAZ,30,C,15)
N1130 setal(61000)
N1140 endif
N1180 if $P_IFRAME <>
CTRANS(X,1,Y,0,Z,2,CAZ,3,C,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1190 setal(61000)
```



```
N1200 endif
N1240 if $P_ACTFRAME <>
CTrans(X,11,Y,0,Z,22,CAZ,33,C,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1250 setal(61001)
N1260 endif
N1270
N1280
N1290 $P_UIFR[1,x,tr] = 11
N1300 $P_UIFR[1,y,tr] = 14
N1310
N1320 g54
N1330

; Frame einstellen
N1350 ROT RPL=-45
N1360 ATRANS X-2 Y10
N1370

; Vierkant schrappen
N1390 G1 X10 Y-10 G41 OFFN=1; Aufmass 1mm
N1400 X-10
N1410 Y10
N1420 X10
N1430 Y-10
N1440

; Werkzeugwechsel
N1460 G0 Z20 G40 OFFN=0
N1470 T3 D1 X15 Y-15
N1480 Z10 G41
N1490

; Vierkant schlichten
N1510 G1 X10 Y-10
N1520 X-10
N1530 Y10
N1540 X10
N1550 Y-10
N1560

; Frame abwählen
N2950 m30 N1580 Z20 G40
```

```

N1590 TRANS
N1600
N1610 if $P_BFRAME <> CTRANS(X,10,Y,0,Z,20,CAZ,30,C,15)
N1620 setal(61000)
N1630 endif
N1640 if $P_BFRAME <> $P_CHBFR[0]
N1650 setal(61000)
N1660 endif
N1670 if $P_IFRAME <>
TRANS(X,11,Y,0,Z,2,CAZ,3,C,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1680 setal(61000)
N1690 endif
N1730 if $P_ACTFRAME <>
TRANS(X,21,Y,0,Z,22,CAZ,33,C,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1740 setal(61001)
N1750 endif
N1760
N1770 TRAFOOF
N1780
N1790 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,C,15)
N1800 setal(61000)
N1810 endif
N1820 if $P_BFRAME <> $P_CHBFR[0]
N1830 setal(61000)
N1840 endif
N1850 if $P_IFRAME <>
TRANS(X,11,Y,2,Z,3,C,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1860 setal(61000)
N1870 endif
N1880 if $P_IFRAME <> $P_UIFR[1]
N1890 setal(61000)
N1900 endif
N1910 if $P_ACTFRAME <>
TRANS(X,21,Y,22,Z,33,C,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1920 setal(61002)
N1930 endif
N1940
N2010 $P_UIFR[1] = ctrans()
N2011 $P_CHBFR[0] = ctrans()
N2020 $P_UIFR[1] = ctrans(x,1,y,2,z,3,c,0)
N2021 G54
N2021 G0 X20 Y0 Z10 C0
N2030 TRANSMIT(1)
N2040 TRANS x10 y20 z30
N2041 ATRANS y200

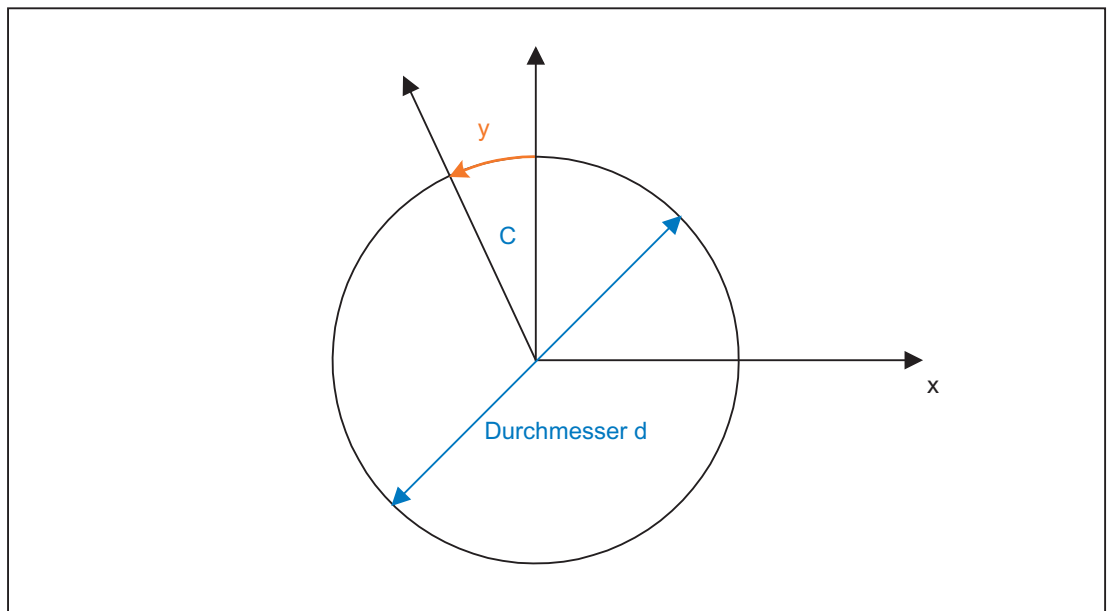
```

```

N2050 G0 X20 Y0 Z10
N2051 if $P_IFRAME <> CTRANS(X,1,Y,0,Z,3,CAY,2)
N2052 setal(61000)
N2053 endif
N2054 if $P_ACTFRAME <> CTRANS(X,11,Y,20,Z,33,CAY,2):CFINE(Y,200)
N2055 setal(61002)
N2056 endif
N2060 TRAFOOF
N2061 if $P_IFRAME <> $P_UIFR[1]
N2062 setal(61000)
N2063 endif
N2064 if $P_ACTFRAME <> CTRANS(X,11,Y,2,Z,33):CFINE(Y,0)
N2065 setal(61002)
N2066 endif

```

11.5.6.4 An- und Abwahl von Transformationen: TRACYL



TRACYL-Erweiterungen

Der achsspezifische Gesamtframe der TRACYL-Rundachse, d. h. die Translation, die Feinverschiebung, die Spiegelung und die Skalierung, kann über folgende Maschinendaten in der Transformation berücksichtigt werden:

- MD24805 \$MC_TRACYL_ROT_AX_FRAME_1 = 1
- MD24855 \$MC_TRACYL_ROT_AX_FRAME_2 = 1

Eine Verschiebung der Rundachse kann z. B. durch eine Kompensation der Schräglage eines Werkstückes in einem Frame innerhalb der Framekette eingetragen werden. Diese Verschiebung soll in der Regel auch in der Transformation als Offset der Rundachse oder als

y-Verschiebung berücksichtigt werden. Eine Verschiebung der C-Achse (wie im obigen Bild) führt dann zu entsprechenden X- und Y-Werten.

- MD24805 \$MC_TRACYL_ROT_AX_FRAME_1 = 2
- MD24855 \$MC_TRACYL_ROT_AX_FRAME_2 = 2

Mit dieser Einstellung wird die achsspezifische Verschiebung der Rundachse bis zum ENS in der Transformation berücksichtigt. Die in den ENS-Frames enthaltenen axialen Verschiebungen der Rundachse werden in das Transformationsframe als Verschiebung auf der Manteloberfläche eingetragen. Diese Einstellung ist nur wirksam, wenn das Transformationsframe projiziert ist.

Frame-Erweiterungen

Nachfolgend beschriebene Erweiterungen gelten nur für folgende Maschinendaten-Einstellungen:

- MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 1
- MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 2

Mit Anwahl der Transformation TRACYL entsteht gekoppelt über die Rundachse eine virtuelle Geometrieachse auf der Manteloberfläche, die keinen Bezug zu einem achsspezifischen Frame hat, sondern nur im Konturframe berücksichtigt wird. Alle Komponenten der virtuellen Geometrieachse werden abgelöscht. Alle anderen Geometrieachsen übernehmen bei Trafo-Anwahl ihre achsspezifischen Anteile.

Komponenten:

- Translationen
Die Translationen der virtuellen Achse werden bei TRACYL-Anwahl gelöscht. Die Translationen der Rundachse können in der Transformation berücksichtigt werden.
- Drehungen
Die Drehungen vor der Transformation werden übernommen.
- Spiegelungen
Die Spiegelung der virtuellen Achse wird gelöscht. Die Spiegelung Rundachse kann in der Transformation berücksichtigt werden.
- Skalierungen
Die Skalierung der virtuellen Achse wird gelöscht. Die Skalierung der Rundachse kann in der Transformation berücksichtigt werden.

Beispiel: Maschinendaten

```
; FRAME-Projektierungen
```

```
$MC_MM_SYSTEM_FRAME_MASK = 'H41'
```

```
; TRAFRAME, SETFRAME
```

```
$MC_CHSFRAME_RESET_MASK = 'H41'
```

```
; Frames sind nach Reset aktiv.
```

```
$MC_CHSFRAME_POWERON_MASK = 'H41'
```

```
; Frames werden bei Power On gelöscht.
```

```
$MN_FRAME_GEOAX_CHANGE_MODE = 1
```

```
; Frames werden nach GeoAx-Umschaltung umgerechnet.
```

```

$MC_RESET_MODE_MASK = 'H4041'           ; Basisframe wird nach Reset nicht abge-
                                         wählt.

; $MC_RESET_MODE_MASK = 'H41'           ; Basisframe wird nach Reset ausgewählt.

; $MC_GCODE_RESET_VALUES[7] = 2         ; G54 ist Voreinstellung.
$MC_GCODE_RESET_VALUES[7] = 1           ; G500 ist Voreinstellung.

$MN_MM_NUM_GLOBAL_USER_FRAMES = 0
$MN_MM_NUM_GLOBAL_BASE_FRAMES = 3

$MC_MM_NUM_USER_FRAMES = 10             ; von 5 bis 100
$MC_MM_NUM_BASE_FRAMES = 3             ; von 0 bis 8

$MN_NCBFRAME_RESET_MASK = 'HFF'
$MC_CHBFRAME_RESET_MASK = 'HFF'

$MN_MIRROR_REF_AX = 0                  ; Keine Normierung bei der Spiegelung.
$MN_MIRROR_TOGGLE = 0

$MN_MM_FRAME_FINE_TRANS = 1            ; Feinverschiebung
$MC_FRAME_ADD_COMPONENTS = TRUE         ; G58, G59 ist möglich

```

; TRACYL mit Nutwandkorrektur ist 3. Trafo

```

$MC_TRAFO_TYPE_3 = 513; TRACYL

$MC_TRAFO_AXES_IN_3[0] = 1
$MC_TRAFO_AXES_IN_3[1] = 5
$MC_TRAFO_AXES_IN_3[2] = 3
$MC_TRAFO_AXES_IN_3[3] = 2

$MC_TRAFO_GEOAX_ASSIGN_TAB_3[0] = 1
$MC_TRAFO_GEOAX_ASSIGN_TAB_3[1] = 5
$MC_TRAFO_GEOAX_ASSIGN_TAB_3[2] = 3

$MC_TRACYL_BASE_TOOL_1[0] = 0.0
$MC_TRACYL_BASE_TOOL_1[1] = 0.0
$MC_TRACYL_BASE_TOOL_1[2] = 0.0

$MC_TRACYL_ROT_AX_OFFSET_1 = 0.0
$MC_TRACYL_ROT_SIGN_IS_PLUS_1 = TRUE

$MC_TRACYL_ROT_AX_FRAME_1 = 1

```

Beispiel: Teileprogramm

```
;Einfacher Verfahrttest mit Nutwandkorrektur
N450 G603
N460

; Frameeinstellungen
N500 $P_UIFR[1] = CTRANS(x,1,y,2,z,3,b,4)
N510 $P_UIFR[1] = $P_UIFR[1] : CROT(x,10,y,20,z,30)
N520 $P_UIFR[1] = $P_UIFR[1] : CMIRROR(x,b)
N530
N540 $P_CHBFR[0] = CTRANS(x,10,y,20,z,30,b,15)
N550
N560 G54
N570

; Bahnsteuerbetrieb mit angewähltem Überschleifen
N590 G0 x0 y0 z-10 b0 G90 F50000 T1 D1 G19 G641 ADIS=1 ADISPOS=5
N600
N610 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,B,15)
N620 setal(61000)
N630 endif
N640 if $P_BFRAME <> $P_CHBFR[0]
N650 setal(61000)
N660 endif
N670 if $P_IFRAME <>
TRANS(X,1,Y,2,Z,3,B,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N680 setal(61000)
N690 endif
N700 if $P_IFRAME <> $P_UIFR[1]
N710 setal(61000)
N720 endif
N730 if $P_ACTFRAME <>
TRANS(X,11,Y,22,Z,33,B,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N740 setal(61000)
N750 endif
N760

; Transformation ein
N780 TRACYL(40.)
N790
N800 if $P_BFRAME <> CTRANS(X,10,Y,0,Z,30,CAY,20,B,15)
N810 setal(61000)
N820 endif
```

```

N830 if $P_CHBFR[0] <> CTRANS(X,10,Y,0,Z,30,CAY,20,B,15)
N840 setal(61000)
N850 endif
N860 if $P_IFRAME <>
TRANS(X,1,Y,0,Z,3,CAY,2,B,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N870 setal(61000)
N880 endif
N890 if $P_UIFR[1] <>
TRANS(X,1,Y,0,Z,3,CAY,2,B,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N900 setal(61000)
N910 endif
N920 if $P_ACTFRAME <>
TRANS(X,11,Y,0,Z,33,CAY,22,B,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N930 setal(61001)
N940 endif
N950
N960 $P_UIFR[1,x,tr] = 11
N970 $P_UIFR[1,y,tr] = 14
N980
N990 g54
N1000
N1010 if $P_BFRAME <> CTRANS(X,10,Y,0,Z,30,CAY,20,B,15)
N1020 setal(61000)
N1030 endif
N1040 if $P_BFRAME <> $P_CHBFR[0]
N1050 setal(61000)
N1060 endif
N1070 if $P_IFRAME <>
TRANS(X,11,Y,0,Z,3,CAY,2,B,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N1080 setal(61000)
N1090 endif
N1100 if $P_IFRAME <> $P_UIFR[1]
N1110 setal(61000)
N1120 endif
N1130 if $P_ACTFRAME <>
TRANS(X,21,Y,0,Z,33,CAY,22,B,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N1140 setal(61001)
N1150 endif
N1160

; Transformation aus
N1180 TRAFOOF
N1190
N1200 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,B,15)
N1210 setal(61000)

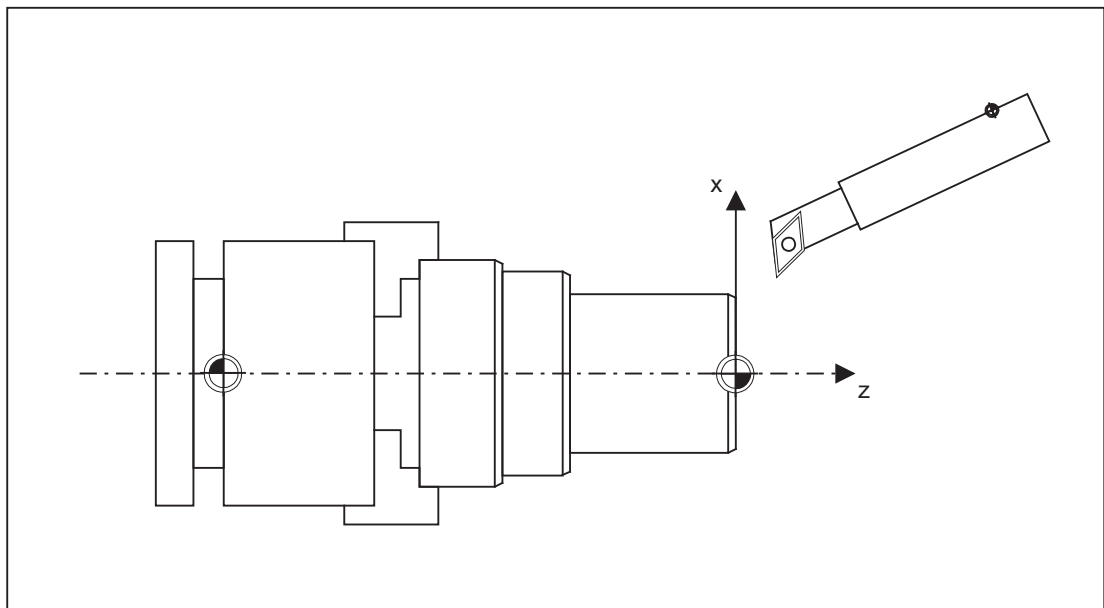
```

```

N1220 endif
N1230 if $P_BFRAME <> $P_CHBFR[0]
N1240 setal(61000)
N1250 endif
N1260 if $P_IFRAME <>
TRANS(X,11,Y,2,Z,3,B,4):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N1270 setal(61000)
N1280 endif
N1290 if $P_IFRAME <> $P_UIFR[1]
N1300 setal(61000)
N1310 endif
N1320 if $P_ACTFRAME <>
TRANS(X,21,Y,22,Z,33,B,19):CROT(X,10,Y,20,Z,30):CMIRROR(X,B)
N1330 setal(61002)
N1340 endif
N1350
N1360 G00 x0 y0 z0 G90
N1370
N1380 m30

```

11.5.6.5 An- und Abwahl von Transformationen: TRAANG



Frame-Erweiterungen:

Nachfolgend beschriebene Erweiterungen gelten nur für folgende Maschinendaten-Einstellungen:

- MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 1
- MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE = 2

Komponenten:

- Translationen
Die Translationen der virtuellen Achse werden bei TRAANG-Anwahl beibehalten.
- Drehungen
Die Drehungen vor der Transformation werden übernommen.
- Spiegelungen
Die Spiegelung der virtuellen Achse wird übernommen.
- Skalierungen
Die Skalierung der virtuellen Achse wird übernommen.

Beispiel: Maschinendaten

```

; FRAME - Projektierungen

$MC_MM_SYSTEM_FRAME_MASK = 'H1'           ; SETFRAME
$MC_CHSFRAME_RESET_MASK = 'H41'           ; Frames sind nach RESET aktiv.
$MC_CHSFRAME_POWERON_MASK = 'H41'         ; Frames werden bei "Power On" gelöscht.

$MN_FRAME_GEOAX_CHANGE_MODE = 1           ; Frames werden nach GeoAx-Umschaltung um-
                                           ; gerechnet.

$MC_RESET_MODE_MASK = 'H4041'             ; Basisframe wird nach RESET nicht abge-
                                           ; wählt.
; $MC_RESET_MODE_MASK = 'H41'             ; Basisframe wird nach RESET ausgewählt.

; $MC_GCODE_RESET_VALUES[7] = 2           ; G54 ist Voreinstellung.
$MC_GCODE_RESET_VALUES[7] = 1             ; G500 ist Voreinstellung.

$MN_MM_NUM_GLOBAL_USER_FRAMES = 0
$MN_MM_NUM_GLOBAL_BASE_FRAMES = 3

$MC_MM_NUM_USER_FRAMES = 10               ; von 5 bis 100
$MC_MM_NUM_BASE_FRAMES = 3               ; von 0 bis 8

$MN_NCBFRAME_RESET_MASK = 'HFF'
$MC_CHBFRAME_RESET_MASK = 'HFF'

$MN_MIRROR_REF_AX = 0                     ; Keine Normierung bei der Spiegelung.
$MN_MIRROR_TOGGLE = 0
$MN_MM_FRAME_FINE_TRANS = 1              ; Feinverschiebung
$MC_FRAME_ADD_COMPONENTS = TRUE           ; G58, G59 ist möglich.

; TRAANG ist 1. Trafo

$MC_TRAFO_TYPE_1 = 1024

```

```

$MC_TRAFO_AXES_IN_1[0] = 4           ; schräge Achse
$MC_TRAFO_AXES_IN_1[1] = 3           ; Achse parallel zu z
$MC_TRAFO_AXES_IN_1[2] = 2
$MC_TRAFO_AXES_IN_1[3] = 0
$MC_TRAFO_AXES_IN_1[4] = 0

$MC_TRAFO_GEOAX_ASSIGN_TAB_1[0] = 4
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[1] = 2
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[2] = 3

$MC_TRAANG_ANGLE_1 = 85.
$MC_TRAANG_PARALLEL_VELO_RES_1 = 0.
$MC_TRAANG_PARALLEL_ACCEL_RES_1 = 0.

$MC_TRAANG_BASE_TOOL_1[0] = 0.0
$MC_TRAANG_BASE_TOOL_1[1] = 0.0
$MC_TRAANG_BASE_TOOL_1[2] = 0.0

```

; TRAANG ist 2. Trafo

```

$MC_TRAFO_TYPE_2 = 1024

$MC_TRAFO_AXES_IN_2[0] = 4
$MC_TRAFO_AXES_IN_2[1] = 3
$MC_TRAFO_AXES_IN_2[2] = 0
$MC_TRAFO_AXES_IN_2[3] = 0
$MC_TRAFO_AXES_IN_2[4] = 0

$MC_TRAFO_GEOAX_ASSIGN_TAB_2[0] = 4
$MC_TRAFO_GEOAX_ASSIGN_TAB_2[1] = 0
$MC_TRAFO_GEOAX_ASSIGN_TAB_2[2] = 3

$MC_TRAANG_ANGLE_2 = -85.
$MC_TRAANG_PARALLEL_VELO_RES_2 = 0.2
$MC_TRAANG_PARALLEL_ACCEL_RES_2 = 0.2

$MC_TRAANG_BASE_TOOL_2[0] = 0.0
$MC_TRAANG_BASE_TOOL_2[1] = 0.0
$MC_TRAANG_BASE_TOOL_2[2] = 0.0

```

Beispiel: Teileprogramm

```

; Frameeinstellungen
N820 $P_UIFR[1] = ctrans(x,1,y,2,z,3,b,4,c,5)

```

```

N830 $P_UIFR[1] = $P_UIFR[1] : crot(x,10,y,20,z,30)
N840 $P_UIFR[1] = $P_UIFR[1] : cmirror(x,c)
N850
N860 $P_CHBFR[0] = ctrans(x,10,y,20,z,30,b,40,c,15)
N870

; Werkzeuganwahl, Aufspannkompensation, Ebenenanwahl
N890 T2 D1 G54 G17 G90 F5000 G64 SOFT
N900

; Anfahren der Ausgangsstellung
N920 G0 X20 Z10
N930
N940 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,B,40,C,15)
N950 setal(61000)
N960 endif
N970 if $P_BFRAME <> $P_CHBFR[0]
N980 setal(61000)
N990 endif
N1000 if $P_IFRAME <>
TRANS(X,1,Y,2,Z,3,B,4,C,5):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1010 setal(61000)
N1020 endif
N1030 if $P_IFRAME <> $P_UIFR[1]
N1040 setal(61000)
N1050 endif
N1060 if $P_ACTFRAME <>
TRANS(X,11,Y,22,Z,33,B,44,C,20):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1070 setal(61000)
N1080 endif
N1090
N1100 TRAANG(,1)
N1110
N1120 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,CAX,10,B,40,C,15)
N1130 setal(61000)
N1140 endif
N1150 if $P_BFRAME <> $P_CHBFR[0]
N1160 setal(61000)
N1170 endif
N1180 if $P_IFRAME <>
CTTRANS(X,1,Y,2,Z,3,CAX,1,B,4,C,5):CROT(X,10,Y,20,Z,30):CMIRROR(X,CAX,C)
N1190 setal(61000)
N1200 endif
N1210 if $P_IFRAME <> $P_UIFR[1]

```

11.5 Frames

```
N1220 setal(61000)
N1230 endif
N1240 if $P_ACTFRAME <>
  TRANS(X,11,Y,22,Z,33,CAX,11,B,44,C,20):CROT(X,10,Y,20,Z,
  30):CMIRROR(X,CAX,C)
N1250 setal(61001)
N1260 endif
N1270
N1280
N1290 $P_UIFR[1,x,tr] = 11
N1300 $P_UIFR[1,y,tr] = 14
N1310
N1320 g54
N1330

; Frame einstellen
N1350 ROT RPL=-45
N1360 ATRANS X-2 Y10
N1370

; Vierkant schrappen
N1390 G1 X10 Y-10 G41 OFFN=1; Aufmass 1mm
N1400 X-10
N1410 Y10
N1420 X10
N1430 Y-10
N1440

; Werkzeugwechsel
N1460 G0 Z20 G40 OFFN=0
N1470 T3 D1 X15 Y-15
N1480 Z10 G41
N1490

; Vierkant schlichten
N1510 G1 X10 Y-10
N1520 X-10
N1530 Y10
N1540 X10
N1550 Y-10
N1560
```

```

; Frame abwaehlen
N1580 Z20 G40
N1590 TRANS
N1600
N1610 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,CAX,10,B,40,C,15)
N1620 setal(61000)
N1630 endif
N1640 if $P_BFRAME <> $P_CHBFR[0]
N1650 setal(61000)
N1660 endif
N1670 if $P_IFRAME <>
TRANS(X,11,Y,14,Z,3,CAX,1,B,4,C,5):CROT(X,10,Y,20,Z,30):CMIRROR(X,CAX,C)
N1680 setal(61000)
N1690 endif
N1700 if $P_IFRAME <> $P_UIFR[1]
N1710 setal(61000)
N1720 endif
N1730 if $P_ACTFRAME <>
TRANS(X,21,Y,34,Z,33,CAX,11,B,44,C,20):CROT(X,10,Y,20,Z,
30):CMIRROR(X,CAX,C)
N1740 setal(61001)
N1750 endif
N1760
N1770 TRAFOOF
N1780
N1790 if $P_BFRAME <> CTRANS(X,10,Y,20,Z,30,B,40,C,15)
N1800 setal(61000)
N1810 endif
N1820 if $P_BFRAME <> $P_CHBFR[0]
N1830 setal(61000)
N1840 endif
N1850 if $P_IFRAME <>
TRANS(X,1,Y,14,Z,3,B,4,C,5):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1860 setal(61000)
N1870 endif
N1880 if $P_IFRAME <> $P_UIFR[1]
N1890 setal(61000)
N1900 endif
N1910 if $P_ACTFRAME <>
TRANS(X,11,Y,34,Z,33,B,44,C,20):CROT(X,10,Y,20,Z,30):CMIRROR(X,C)
N1920 setal(61002)
N1930 endif
N1940
N1950 m30

```

11.5.6.6 Adaptionen von aktiven Frames

Die Geometrie-Achskonstellation kann sich während der Programmbearbeitung oder bei **RESET** ändern. Die Anzahl der vorhandenen Geometrieachsen können dabei von null bis drei variieren. Bei nicht-vorhandenen Geometrieachsen können Komponenten in den aktiven Frames (z.B. Drehungen) dazu führen, dass die aktiven Frames für diese Achskonstellation ungültig werden. Dies wird durch Alarm "Kanal %1 Satz %2 Rotation fuer nicht vorhandene Geometrieachse programmiert" angezeigt. Der Alarm bleibt solange anstehen, bis die Frames entsprechend geändert wurden.

Mit folgendem Maschinendatum kann die automatische Anpassung von aktiven Frames eingeschaltet werden:

MD24040 \$MC_FRAME_ADAPT_MODE, Bit<n> = <Wert>

Bit	<Wert>	Bedeutung
0	1	Drehungen in aktiven Frames, die Koordinatenachsen verdrehen, fuer die es keine Geometrieachsen gibt, werden aus den aktiven Frames geloescht.
1	1	Scherungswinkel in aktiven Frames werden orthogonalisiert.
2	1	Skalierungen aller Geometrieachsen in den aktiven Frames werden auf den Wert 1 gesetzt.

Über folgendes Maschinendatum werden alle Drehungen in den aktiven Frames gelöscht, die zu Koordinaten-Achsbewegungen für nicht vorhandene Geometrieachsen führen könnten:

MD24040 \$MC_FRAME_ADAPT_MODE = 1

Die Datenhaltungsframes werden dabei nicht verändert. Bei der Aktivierung von Datenhaltungsframes, werden dann auch nur die möglichen Drehungen übernommen.

Beispiel

Es existiert keine Y-Achse:

- MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB[0] = 1
- MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB[1] = 0
- MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB[2] = 3
- \$P_UIFR[1] = CROT(X,45,Y,45,Z,45)

Programmcode	; Kommentar
N390 G54 G0 X10 Z10 F10000	
IF \$P_IFRAME <> CROT(Y,45)	; es wird nur die Drehung um Y übernommen
SETAL(61000)	
ENDIF	

11.5.6.7 Mapped Frames

Übersicht

Die Funktion "Mapped Frames" unterstützt die kanalübergreifende konsistente Änderung achsspezifischer Frames innerhalb kanalspezifischer oder globaler Datenhaltungsframes. In achsspezifischen Maschinendaten wird dazu festgelegt zwischen welchen Achsen das Mapping erfolgen soll.

Ist das Frame-Mapping z.B. für die Maschinenachsen AX1 und AX4 aktiv und es wird in einem kanalspezifischen Datenhaltungsframe (z.B. Basisframe \$P_CHBFR[x]) der achsspezifische Frame der Achse AX1 verändert (Translation, Fine-Translation, Skalierung, Spiegelung), werden diese Frame-Daten für AX1 und AX4 auf alle kanalspezifischen Datenhaltungsframes (z.B. Basisframe \$P_CHBFR[x]) in allen Kanälen in denen sie als Kanalachsen parametrisiert sind, übertragen.

Kein Frame-Mapping erfolgt bei Änderung der achsspezifischen Frame-Daten für die Drehung.

Voraussetzungen

Für das Frame-Mapping müssen folgende Voraussetzungen erfüllt sein:

- Die Datenhaltungsframes die für das Mapping verwendet werden, müssen projiziert sein: MD28083 \$MC_MM_SYSTEM_DATAFRAME_MASK (Systemframes)
- **Kanalspezifische** Datenhaltungsframes müssen für das Mapping explizit freigegeben werden: MD10616 \$MN_MAPPED_FRAME_MASK (Freigabe Frame-Mapping)

Hinweis

Bei **globalen** Datenhaltungsframes wird das Mapping immer durchgeführt. Eine Freigabe ist nicht erforderlich.

Parametrierung

Die Parametrierung der Mapping-Beziehungen erfolgt in den achsspezifischen Maschinendaten:

MD32075 \$MA_MAPPED_FRAME[<AXn>] = "AXm"

AXn, AXm: Maschinenachsname mit n, m = 1, 2, ... max. Anzahl Maschinenachsen

Mapping-Regeln

Für das Frame-Mapping gelten folgende Regeln:

- Das Mapping ist bidirektional.
Ein achsspezifischer Frame kann für die Achse AXn oder AXm geschrieben werden. Die Frame-Daten werden immer für die jeweils andere Achse übernommen.
- Es werden immer alle parametrisierten Mapping-Beziehungen ausgewertet.
Beim Schreiben eines achsspezifischen Frames einer Achsen AXn werden alle Mapping-Beziehungen ausgewertet und die Frame-Daten für alle direkt und indirekt betroffenen Achsen übernommen.

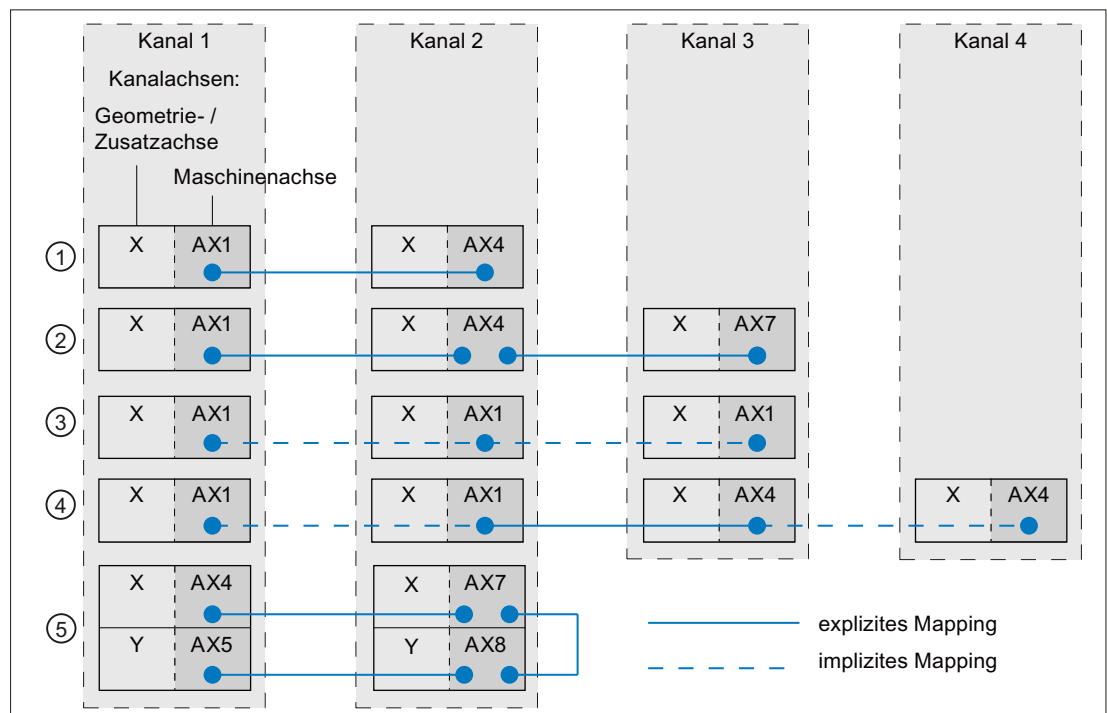
11.5 Frames

- Das Mapping ist kanalglobal.
Beim Schreiben eines achsspezifischen Frames der Achse AXn oder AXm für einen kanalspezifischen Frame, werden die Frame-Daten für alle Kanäle übernommen in denen AXn oder AXm als Kanalachsen parametrier sind.
- Beim Schreiben eines achsspezifischen Frames mittels Geometrie- oder Zusatzachsname, erfolgt die Auswertung der Mapping-Beziehungen über die der Geometrie- oder Zusatzachse aktuell zugeordneten Maschinenachse.
- Das Mapping ist Frame-spezifisch.
Beim Schreiben eines achsspezifischen Frames erfolgt das Mapping der Frame-Daten nur innerhalb des gleichen kanalspezifischen oder globalen Datenhaltungsframes.

Hinweis

Datenkonsistenz

Es liegt ausschließlich in der Verantwortung des Anwenders/Maschinenhersteller, z.B. durch Kanalsynchronisation, dass nach dem Schreiben eines Frames in allen Kanälen konsistente Frame-Daten vorliegen.

**Beschreibung**

- ① Einfache Mapping-Beziehung:
AX1(K1) ↔ AX4(K2)
- ② Verkettete Mapping-Beziehungen:
AX1(K1) ↔ AX4(K2) ↔ AX7(K3)
- ③ Mapping-Beziehung auf sich selbst, mit AX1 als
Kanalachse von Kanal 1, 2 und 3:
AX1(K1+K2+K3)
- ④ Mapping-Beziehung zwischen zwei Achsen, die
Kanalachsen in jeweils zwei Kanälen sind:
AX1(K1+K2) ↔ AX4(K3+K4)
- ⑤ Verkettete Mapping-Beziehungen bei der meh-
rere Kanalachsen im gleichen Kanal geschrie-
ben werden:
AX4(K1) ↔ AX7(K2) ↔ AX8(K2) ↔ AX5(K1)

Parametrierung: \$MA_

MAPPED_FRAME[<AX1>] = "AX4"

MAPPED_FRAME[<AX1>] = "AX4"
MAPPED_FRAME[<AX4>] = "AX7"

MAPPED_FRAME[<AX1>] = "AX1"

MAPPED_FRAME[<AX1>] = "AX4"

MAPPED_FRAME[<AX4>] = "AX7"
MAPPED_FRAME[<AX7>] = "AX8"
MAPPED_FRAME[<AX8>] = "AX5"

Bild 11-25 Mapping Beispiele

Aktivieren der Datenhaltungsframes

Die Datenhaltungsframes können im Teileprogramm und über die Bedienoberfläche von SINUMERIK Operate geschrieben werden. Bei der Aktivierung der direkt und über Frame-Mapping geschriebenen Datenhaltungsframes in den Kanälen ist folgendes zu beachten:

- Schreiben im Teileprogramm
Die Datenhaltungsframes müssen in jedem Kanal explizit (G500, G54 ... G599) aktiviert werden
- Schreiben über Bedienoberfläche
Datenhaltungsframes werden über die Bedienoberfläche z.B. durch die Eingabe neuer Nullpunktverschiebungen geschrieben. Ein veränderter Datenhaltungsframe wird in allen betroffenen Kanälen sofort aktiv, wenn sich keiner dieser Kanäle im Zustand "Kanal aktiv" befindet. Ist einer der betroffenen Kanäle im Zustand "Kanal aktiv", wird der Datenhaltungsframe in keinem Kanal aktiv. Die Aktivierung muss dann in jedem Kanal explizit im Teileprogramm (G500, G54 ... G599) programmiert werden. Oder er wird mit dem nächsten Wechsel des Kanalzustandes nach "Kanal-Reset" aktiv.

Beispiel

An einer Steuerung sind folgende Kanäle und Kanalachsen parametrisiert:

- Kanal 1
 - Z: Geometrieachse
 - AX1: Maschinenachse der Geometrieachse Z
- Kanal 2
 - Z: Geometrieachse
 - AX4: Maschinenachse der Geometrieachse Z

Der Nullpunkt der Z-Achse soll in beiden Kanäle immer gleich sein:

- Mapping-Beziehung: \$MA_MAPPED_FRAME[AX1] = "AX4"

Teileprogramme im Kanal 1 und 2

Kanal 1	Kanal 2
...	...
N100 WAIT (10,1,2)	N200 WAIT (10,1,2)
N110 \$P_UIFR[1] = CTRANS(Z, 10)	
N120 WAIT (20,1,2)	N220 WAIT (20,1,2)
N130 G54	N230 G54
N140 IF (\$P_IFRAME[0, Z, TR] <> 10)	N230 IF (\$P_IFRAME[0, Z, TR] <> 10)
N150 SETAL(61000)	N250 SETAL(61000)
N160 ENDIF	N260 ENDIF
...	...

Beschreibung:

N100 / N200	Kanalsynchronisation für konsistentes Schreiben und Mapping der Frame-Daten
N110	Schreiben des einstellbaren Datenhaltungs-Frames \$P_UIFR[1]: Verschieben des Nullpunktes der Z-Achse auf 10 mm Mapping der achsspezifischen Frame-Daten: Kanal1: $Z \triangleq AX1 \Leftrightarrow$ Kanal2: $Z \triangleq AX4$
N120 / N220	Kanalsynchronisation für konsistentes Aktivieren der neuen Frame-Daten
N130 / N230	Aktivieren der neuen Frame-Daten
N140 / N240	Überprüfung des Nullpunktes der Z-Achse auf: 10 mm

11.5.7 Vordefinierte Frame-Funktionen

11.5.7.1 Inverses Frame

Die Funktion INVFRAME() berechnet aus einem Frame den entsprechenden inversen Frame.

Funktionsbeschreibung

Die Frame-Verkettung eines Frames mit seinem inversen Frame ergibt immer einen Nullframe.

FRAME : INVFRAME(FRAME) \square Null-Frame

Die Frame-Invertierung ist ein Hilfsmittel für die Koordinatentransformationen. Die Berechnung von Messframes erfolgt meist im WKS. Möchte man dieses berechnete Frame in ein anderes Koordinatensystem transformieren, d.h. das berechnete Frame soll in einen beliebigen Frame innerhalb der Frame-Kette eingetragen werden, so ist dies mit folgenden Berechnungen möglich:

Das neue Gesamtframe ergibt sich als Verkettung des alten Gesamtframes mit dem berechneten Frame:

$\$P_ACTFRAME = \$P_ACTFRAME : \$AC_MEAS_FRAME$

Das neue Frame in der Framekette ergibt sich danach:

- Zielframe ist `$P_SETFRAME`:
`$P_SETFRAME = $P_ACTFRAME : $AC_MEAS_FRAME :`
`INVFRAME($P_ACTFRAME) : $P_SETFRAME`
- Ziel-Frame ist n-tes Kanalbasisframe `$P_CHBFRAME[<n>]`:
`k = $MN_MM_NUM_GLOBAL_BASE_FRAMES`
 - Bei `n = 0` ergibt sich TMP zu:
`TMP = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME :`
`$P_NCBFRAME[<0...k>]`
 - Bei `n ≠ 0` ergibt sich TMP zu:
`TMP = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME :`
`$P_NCBFRAME[<0...k>] : $P_CHBFRAME[<0...n> - 1]``$P_CHBFRAME[<n>] = INVFRAME(TMP) : $P_ACTFRAME : $AC_MEAS_FRAME :`
`INVFRAME($P_ACTFRAME) : TMP : $P_CHBFRAME[<n>]`
- Ziel-Frame ist `$P_IFRAME`:
`TMP = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_BFRAME`
`$P_IFRAME = INVFRAME(TMP) : $P_ACTFRAME : $AC_MEAS_FRAME :`
`INVFRAME($P_ACTFRAME) : TMP : $P_IFRAME`

Beispiel

Ein Frame, der z.B. über eine Messfunktion ermittelt wurde, soll im aktuellen `SETFRAME` so eingetragen werden, dass das neue Gesamtframe sich als Verkettung des alten Gesamtframes mit dem Mess-Frame ergibt. Das `SETFRAME` wird mit Hilfe von Frame-Invertierungen entsprechend umgerechnet.

Programmcode	Kommentar
DEF INT RETVAL	
DEF FRAME TMP	
\$TC_DP1[1,1]=120	; Typ
\$TC_DP2[1,1]=20.	; 0
\$TC_DP3[1,1]= 10.	; (z) Laengenkorrekturvektor
\$TC_DP4[1,1]= 0.	; (y)
\$TC_DP5[1,1]= 0.	; (x)
\$TC_DP6[1,1]= 2.	; Radius
T1 D1	
g0 x0 y0 z0 f10000	
G54	
\$P_CHBFRAME[0] = CROT(Z,45)	
\$P_IFRAME[X,TR] = -SIN(45)	
\$P_IFRAME[Y,TR] = -SIN(45)	
\$P_PFRAME[Z,TR] = -45	
\$AC_MEAS_VALID = 0	; Ecke mit 4 Messpunkten vermessen
G1 X-1 Y-3	; 1. Messpunkt anfahren
\$AC_MEAS_LATCH[0] = 1	; 1. Messpunkt abspeichern
G1 X5 Y-3	; 2. Messpunkt anfahren

Programmcode	Kommentar
\$AC_MEAS_LATCH[1] = 1	; 2. Messpunkt abspeichern
G1 X-4 Y4	; 3. Messpunkt anfahren
\$AC_MEAS_LATCH[2] = 1	; 3. Messpunkt abspeichern
G1 X-4 Y1	; 4. Messpunkt anfahren
\$AC_MEAS_LATCH[3] = 1	; 4. Messpunkt abspeichern
\$AA_MEAS_SETPOINT[X] = 0	; Sollposition der Ecke setzen
\$AA_MEAS_SETPOINT[Y] = 0	
\$AA_MEAS_SETPOINT[Z] = 0	
\$AC_MEAS_CORNER_SETANGLE = 90	; Sollschnittwinkel vorgeben
\$AC_MEAS_WP_SETANGLE = 30	
\$AC_MEAS_ACT_PLANE = 0	; Ebene fuer die Messung ist G17
\$AC_MEAS_T_NUMBER = 1	; Werkzeug auswaehlen
\$AC_MEAS_D_NUMBER = 1	
\$AC_MEAS_TYPE = 4	; Messtype auf Ecke 1 setzen
RETVAL = MEASURE()	; Messvorgang starten
IF RETVAL <> 0	
SETAL(61000 + RETVAL)	
ENDIF	
IF \$AC_MEAS_WP_ANGLE <> 30	
SETAL(61000 + \$AC_MEAS_WP_ANGLE)	
ENDIF	
IF \$AC_MEAS_CORNER_ANGLE <> 90	
SETAL(61000 + \$AC_MEAS_CORNER_ANGLE)	
ENDIF	
; Gemessenes Frame so transformieren und nach \$P_SETFRAME schreiben, dass ein Gesamtframe entsteht,	
; welches aus dem alten Gesamtframe verkettet mit dem Messframe ergibt.	
\$P_SETFRAME = \$P_ACTFRAME : \$AC_MEAS_FRAME : INVFRAME(\$P_ACTFRAME) : \$P_SETFRAME	
\$P_SETFR = \$P_SETFRAME	; Systemframe in der Datenhaltung beschreiben
G1 X0 Y0	; Fahre die Ecke an
G1 X10	; Um 30 Grad gedrehtes Rechteck abfahren
Y10	
X0	
Y0	
M30	

11.5.7.2 Additives Frame in der Framekette

Durch Messungen am Werkstück oder durch Berechnungen im Teileprogramm oder Zyklus ergibt sich oftmals ein Frame, der additiv zum aktiven Gesamtframe wirken sollen. Dadurch sollen z.B. das WKS und damit der Nullpunkt der Programmierung verschoben und/oder gedreht werden. Der gemessene Frame liegt dabei in einer Frame-Variablen vor und ist noch nicht in die Framekette eingerechnet worden.

Funktionsbeschreibung

Die Funktion `ADDFRAME()` berechnet aus dem temporären Frame den angegebenen Zielframe, der durch den Parameter `<STRING>` spezifiziert ist, so, dass sich das neue aktive Gesamtframe `$P_ACTFRAME` aus der Verkettung des alten aktiven Gesamtframes mit dem temporären Frame ergibt:

$$ERG = ADDFRAME(TMPFRAME, "$P_SETFRAME") \Rightarrow \$P_SETFRAME_{neu} = \$P_SETFRAME_{alt} \text{ ADD } TMPFRAME \text{ und } \$P_ACTFRAME_{neu} = \$P_ACTFRAME_{alt} : TMPFRAME$$

Wurde als Zielframe ein aktiver Frame angegeben, wird das neue Gesamtframe **im Vorlauf** aktiv.

Ist das Zielframe ein Datenhaltungs-Frame, wird der Frame erst aktiv, wenn er im Kanal explizit, z.B. Teileprogramm, Zyklus, aktiviert wird.

Die Funktion gibt einen Rückgabewert zurück, auf den anwenderspezifisch, z.B. in einem Anwender-Zyklus, reagiert werden kann.

Programmierung

Syntax

`INT ADDFRAME(<FRAME>, <STRING>)`

Bedeutung

<FRAME>:	Frame-Variable mit den additiv einzurechnenden Werten	
	Typ	FRAME
<STRING>:	Name eines aktiven oder Datenhaltungs-Frames:	
	<ul style="list-style-type: none"> Aktive Frames "\$P_CYCFRAME", "\$P_ISO4FRAME", "\$P_PFRAME", "\$P_WPFRAME", "\$P_TOOLFRAME", "\$P_IFRAME", "\$P_GFRAME", "\$P_CHBFRAME[<n>]", "\$P_NCBFRAME[<n>]", "\$P_ISO1FRAME", "\$P_ISO2FRAME", "\$P_ISO3FRAME", "\$P_EXTFRAME", "\$P_SETFRAME", "\$P_PARTFRAME" Datenhaltungs-Frames "\$P_CYCFR", "\$P_ISO4FR", "\$P_TRAFR", "\$P_WPFR", "\$P_TOOLFR", "\$P_UIFR[<n>]", "\$P_GFR", "\$P_CHBFR[<n>]", "\$P_NCBFR[<n>]", "\$P_ISO1FR", "\$P_ISO2FR", "\$P_ISO3FR", "\$P_EXTFR", "\$P_SETFR", "\$P_PARTFR" 	
Rückgabewert:	Typ	STRING
	Mögliche Rückgabewerte:	
	<ul style="list-style-type: none"> 0: OK 1: Zielangabe (String) ist falsch 2: Zielframe ist nicht projiziert 3: Drehung im Frame ist nicht erlaubt 	
	Typ	INT

11.5.8 Funktionen

11.5.8.1 Setzen von Nullpunkten, Werkstück- und Werkzeugvermessung

Das Iswertsetzen erfolgt über die HMI-Bedienung oder über Messzyklen. Das berechnete Frame wird in das Systemframe `SETFRAME` geschrieben werden. Beim Istwertsetzen kann die Sollposition einer Achse im WKS geändert werden.

Unter dem Begriff "Ankratzen" verstehen wir Werkstück- und Werkzeugvermessung. Bei der Werkstückvermessung kann die Lage des Werkstückes bzgl. einer Kante, einer Ecke oder einer Bohrung vermessen werden. Zur Festlegung der Nulllage des Werkstückes oder der Bohrung können die gemessenen Positionen dann mit Sollpositionen im WKS beaufschlagt werden. Die resultierenden Verschiebungen können dabei in ein ausgewähltes Frame eingetragen werden. Bei der Werkzeugvermessung kann die Länge oder der Radius eines Werkzeuges anhand eines vermessenen Referenzteiles gemessen werden.

Die Messungen können über die Bedienung oder über Messzyklen erfolgen. Als Kommunikation mit dem NC dienen vordefinierte Systemvariablen. Die Berechnung erfolgt im NC durch Aktivierung eines PI-Dienstes über die HMI-Bedienung oder über einen Teileprogrammbefehl aus den Messzyklen. Als Grundlage für die Berechnung kann ein Werkzeug und eine Ebene ausgewählt werden. Der berechnete Frame wird in das Ergebnis-Frame eingetragen.

11.5.8.2 Achsspezifische Externe Nullpunktverschiebung

Maschinendaten

Die Aktivierung der Externe Nullpunktverschiebung bzw. Systemframe `$P_EXTFRAME` erfolgt über das folgende Maschinendatum:

MD28082 `$MC_MM_SYSTEM_FRAME_MASK`, Bit1 = TRUE

Der Betrag für die Externe Nullpunktverschiebung kann manuell über die HMI-Bedienoberfläche und das PLC-Anwenderprogramm über BTSS vorgegeben oder im Teileprogramm über die achsiale Systemvariable `$AA_ETRANS[<Achse>]` programmiert werden.

Aktivierung

Die Aktivierung der Externen Nullpunktverschiebung erfolgt über das Nahtstellensignal: DB31, ... DBX3.0 (Externe Nullpunktverschiebung übernehmen)

Verhalten

Bei Aktivierung der Externen Nullpunktverschiebung werden die Verfahrbewegungen aller Achsen, außer Kommando- und PLC-Achsen, sofort gestoppt und der Vorlauf reorganisiert. Die Grobverschiebung des aktuellen Systemframe und des Systemframe in der Datenhaltung wird auf den Wert der achsialen Systemvariablen `$AA_ETRANS[<Achse>]` gesetzt. Anschließend wird zuerst die Verschiebung herausgefahren und dann die unterbrochene Bewegung fortgesetzt.

Verhalten bei Kettenmaßangabe

Bei aktiver Kettenmaßangabe `G91` und Maschinendatum:

MD42440 `$MC_FRAME_OFFSET_INCR_PROG` (Nullpunktverschiebungen in Frames) = 0 wird im Rahmen der Externe Nullpunktverschiebung über Systemframe die Verschiebung,

trotz gegenteiliger Projektierung des Maschinendatums, mit dem Anfahrsatz herausgefahren, obwohl sie durch einen Frame vorgegeben wird.

Hinweis

Die Externe Nullpunktverschiebung wirkt immer absolut.

11.5.8.3 Werkzeugträger

Translationen

Bei Kinematiken vom Typ "P" und "M" wird bei der Anwahl eines Werkzeugträgers ein additiver Frame aktiviert (Tischoffset des orientierbaren Werkzeugträgers), der die Verschiebung des Nullpunktes als Folge der Drehung des Tisches berücksichtigt. Die Nullpunktverschiebung wird in das Systemframe `$P_PARTFR` eingetragen. Dabei wird der translatorische Anteil dieses Frames überschrieben. Die anderen Framekomponenten bleiben erhalten.

Das Systemframe `$P_PARTFR` muss über folgendes Maschinendatum freigegeben werden:

MD28082 `$MC_MM_SYSTEM_FRAME_MASK`, Bit 2 = 1 (Systemframe für TCARR und PAROT)

Hinweis

Alternativ kann die Verschiebung auch über das Maschinendatum zur Aufnahme des Tischoffsets parametrisiert werden:

MD20184 `$MC_TOCARR_BASE_FRAME_NUMBER` = <Nummer des Basisframes>

Diese Möglichkeit besteht nur aus Kompatibilitätsgründen zu älteren Softwareständen. Es wird dringend empfohlen diese Vorgehensweise **nicht** mehr zu verwenden.

Eine Frameverschiebung als Folge eines Werkzeugträgerwechsels wird sofort bei der Anwahl mit `TCARR=...` wirksam. Dagegen wird eine Änderung der Werkzeuglänge nur dann sofort wirksam, wenn ein Werkzeug aktiv ist.

Eine Framedrehung wird mit der Aktivierung nicht ausgeführt, bzw. eine bereits wirksame Drehung wird nicht verändert. Die Stellung der Rundachsen, die für die Berechnung verwendet werden, wird wie im Fall T (nur das Werkzeug ist drehbar) abhängig vom Befehl `TCOFR / TCOABS` aus dem Drehanteil eines aktiven Frames bzw. aus den Einträgen `$TC_CARR` ermittelt. Durch die Aktivierung eines Frames verändert sich die Position im Werkstückkoordinatensystem entsprechend, ohne dass es zu einer Ausgleichsbewegung der Maschine kommt.

Die Verhältnisse sind im folgenden Bild dargestellt:

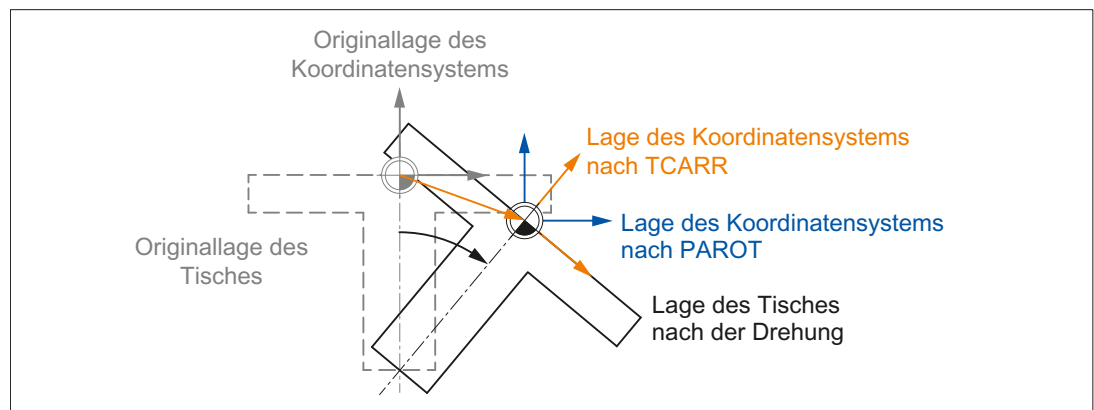


Bild 11-26 Frame bei Aktivierung eines drehbaren Werkzeugtisches mit TCARR

Bei Kinematiken des Typs M (Werkzeug und Tisch sind jeweils um eine Achse drehbar), bewirkt die Aktivierung eines Werkzeugträgers mit `TCARR` gleichzeitig eine entsprechende Änderung der effektiven Werkzeuglänge (falls ein Werkzeug aktiv ist) und der Nullpunktverschiebung.

Drehungen

Je nach Bearbeitungsaufgabe muss beim Einsatz drehbarer Werkzeugträger bzw. Tische nicht nur eine Nullpunktverschiebung (sei es als Frame oder als Werkzeuglänge), sondern auch eine Drehung berücksichtigt werden. Die Aktivierung eines orientierbaren Werkzeugträgers führt jedoch in keinem Fall unmittelbar zu einer Drehung des Koordinatensystems.

Ist nur das Werkzeug drehbar, kann dafür ein Frame mittels `TOFRAME` bzw. `TOROT` usw. definiert werden.

Bei drehbaren Tischen (Kinematiktypen P und M) hat die Aktivierung mit `TCARR` zunächst ebenfalls keine Drehung des Koordinatensystems zur Folge, d.h. der Nullpunkt des Koordinatensystems verschiebt sich zwar in Bezug auf die Maschine und bleibt fest in Bezug auf den Nullpunkt des Werkstücks, die Orientierung bleibt jedoch unverändert raumfest.

Wird ein Koordinatensystem benötigt, das in Bezug auf das Werkstück fest ist, d.h. gegenüber der Originallage nicht nur verschoben sondern auch entsprechend der Tischdrehung gedreht ist, so kann analog zur Situation bei drehbarem Werkzeug mit `PAROT` eine entsprechende Drehung aktiviert werden.

Mit `PAROT` bleiben Translationen, Skalierungen und Spiegelungen im aktiven Frame erhalten, der Rotationsanteil wird jedoch durch den Rotationsanteil eines orientierbaren Werkzeugträgers, der dem Tisch entspricht gedreht. Dabei bleibt der gesamte programmierbare Frame, einschließlich seines Drehanteils, unverändert.

Der Rotationsanteil, der die Drehung des Werkzeugtisches beschreibt, wird dann entweder in den Systemframe `$PARTFR` oder in den durch MD20184

`$MC_TOCARR_BASE_FRAME_NUMBER` parametrisierten Basisframe eingetragen:

`$MC_MM_SYSTEM_FRAME_MASK`, Bit 2 = <Wert>

Wert	Bedeutung
1	Rotationsanteil → <code>\$PARTFR</code>
0	Rotationsanteil → MD20184 <code>\$MC_TOCARR_BASE_FRAME_NUMBER</code>

Entsprechend dem Hinweis bei der Beschreibung der Tischverschiebung gilt auch hier, dass empfohlen wird, die zweite Alternative für Neuanlagen nicht mehr zu verwenden.

Der Rotationsanteil des Partframes kann mit `PAROTOF` gelöscht werden, unabhängig davon, ob dieser Frame in einem Basis- oder in einem Systemframe steht.

Der Translationsanteil wird gelöscht, indem ein Werkzeugträger aktiviert wird, der keine Verschiebung bewirkt, bzw. durch Abwahl eines eventuell aktiven orientierbaren Werkzeugträgers mit `TCARR=0`.

`PAROT` bzw. `TOROT` berücksichtigen in den Fällen, in denen der Tisch bzw. das Werkzeug mit zwei Drehachsen orientiert werden, die gesamte Orientierungsänderung. Bei gemischten Kinematiken wird jeweils nur der entsprechende, durch eine Drehachse verursachte Anteil berücksichtigt. Damit ist es z. B. bei der Verwendung von `TOROT` möglich, ein Werkstück so zu drehen, dass eine schräge Ebene parallel zur raumfesten X-Y-Ebene liegt, bei der Bearbeitung aber eine Drehung des Werkzeugs zu berücksichtigen, wenn beispielsweise Bohrungen angebracht werden müssen, die nicht senkrecht zu dieser Ebene sind.

Beispiel

Bei einer Maschine zeige die Drehachse des Tisches in die positive Y-Richtung. Der Tisch ist um +45 Grad gedreht. Mit `PAROT` wird dann ein Frame definiert, der ebenfalls eine Drehung von 45 Grad um die Y-Achse beschreibt. Das gegenüber der Außenwelt nicht gedrehte Koordinatensystem (im Bild mit "Lage des Koordinatensystems nach `TCARR`" gekennzeichnet) ist gegenüber dem mitgeführten Koordinatensystem (Lage nach `PAROT`) aber um -45 Grad gedreht. Wird dieses Koordinatensystem z. B. mit `ROT Y-45` definiert und anschließend der Werkzeugträger bei aktivem `TCOFR` angewählt, wird für die Drehachse des Werkzeugträgers ein Winkel von +45 Grad ermittelt.

Der Sprachbefehl `PAROT` wird nicht abgelehnt, wenn kein orientierbarer Werkzeugträger aktiv ist. Allerdings bewirkt ein solcher Aufruf dann keine Frameänderung.

Bearbeitung in Richtung der Werkzeugorientierung

Vor allem an Maschinen mit orientierbarem Werkzeug soll mitunter in Werkzeugrichtung verfahren werden (typischerweise beim Bohren), ohne dass ein Frame aktiviert wird (z. B. mittels `TOFRAME` oder `TOROT`), bei dem eine der Achsen in Richtung des Werkzeugs zeigt. Das Problem stellt sich auch dann, wenn bei einer Schrägbearbeitung ein Frame aktiv ist, der die schräge Ebene definiert, das Werkzeug aber nicht exakt senkrecht eingestellt werden kann, weil wegen eines indexierten Werkzeugträgers (Hirth-Verzahnung) die Werkzeugorientierung nicht beliebig einstellbar ist. In diesen Fällen muss dann - abweichend von der eigentlich verlangten Bewegung senkrecht zur Ebene - in Werkzeugrichtung gebohrt werden, da andernfalls der Bohrer nicht in Richtung seiner Längsachse geführt würde (Werkzeugbruch).

Inkrementelles Verfahren

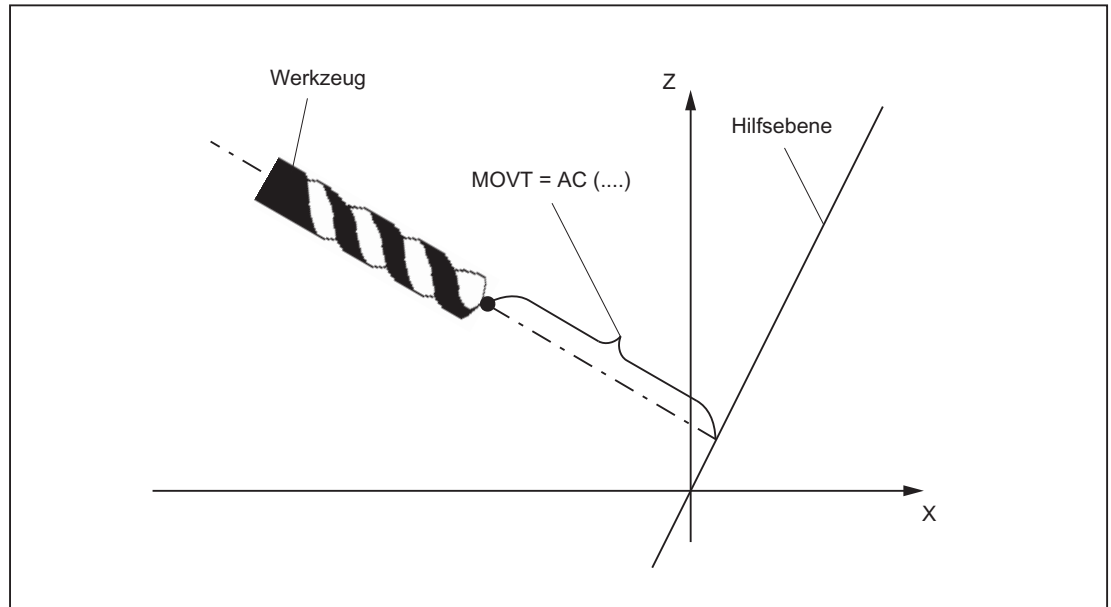
Der Endpunkt für eine inkrementelle Verfahrbewegung in Werkzeugrichtung wird mit `MOV T = <Wert>` oder `MOV T=IC (<Wert>)` programmiert.

Die positive Verfahrrichtung ist dabei von der Werkzeugspitze zur Werkzeugaufnahme definiert. Entsprechend der achsparallelen Bearbeitung z. B. mit `G91 Z...`

Absolutes Verfahren

Der Endpunkt für eine absolute Verfahrbewegung in Werkzeugrichtung wird mit `MOV T=AC (<Wert>)` programmiert.

Dazu wird eine Ebene definiert, die durch den aktuellen Nullpunkt verläuft, und deren Flächennormalenvektor parallel zur Werkzeugorientierung ist. `MOV T` gibt dann die Lage bezüglich dieser Ebene an (siehe Bild). Die Bezugsebene dient nur zur Berechnung der Endposition. Aktive Frames werden durch diese interne Berechnung nicht beeinflusst.



Die Programmierung mit `MOV T` ist unabhängig vom Vorhandensein eines orientierbaren Werkzeugträgers. Die Richtung der Bewegung ist von der aktiven Ebene abhängig. Sie verläuft in Richtung der Applikate, d.h. bei `G17` in Z-Richtung, bei `G18` in Y-Richtung und bei `G19` in X-Richtung. Dies gilt sowohl für den Fall, dass kein orientierbarer Werkzeugträger aktiv ist, als auch für den Fall eines orientierbaren Werkzeugträgers ohne drehbares Werkzeug oder mit drehbarem Werkzeug in Grundstellung.

`MOV T` wirkt bei aktiver Orientierungstransformation (3-4-5-Achs-Transformation) in gleicher Weise.

Wird in einem Satz mit `MOV T` gleichzeitig die Werkzeugorientierung verändert (z. B. bei aktiver 5-Achstransformation durch gleichzeitige Interpolation der Rundachsen), ist die Orientierung am Satzanfang für die Bewegungsrichtung von `MOV T` maßgebend.

Bei aktiver 5-Achstransformation wird die Bahn des "tool center points" (TCP) durch die Orientierungsänderung nicht beeinflusst, d. h. die Bahn bleibt eine Gerade, deren Richtung durch die Werkzeugorientierung im Satzangang bestimmt ist.

Wird `MOV T` programmiert, muss Linear- oder Spline-Interpolation (`G0`, `G1`, `ASPLINE`, `BSPLINE`, `CSPLINE`) aktiv sein. Andernfalls wird ein Alarm ausgegeben.

Ist eine Spline-Interpolationsart aktiv, ist die resultierende Bahn im allgemeinen keine Gerade, da der von `MOV T` ermittelte Endpunkt so behandelt wird, als sei er explizit mit X, Y, Z programmiert worden.

In einem Satz mit `MOV T` dürfen keine Geometrieachsen programmiert werden.

Definition von Framedrehungen mit Raumwinkeln

Soll ein Frame, der eine Drehung um mehr als eine Achse beschreibt, definiert werden, so geschieht das durch die Verkettung von Einzeldrehungen. Dabei erfolgt die nachfolgende Drehung im neuen gedrehten Koordinatensystem.

Das gilt sowohl bei Programmierung in einem Satz als auch beim Aufbau eines Frames in mehreren aufeinander folgenden Sätzen:

- Ein Satz: `N10 ROT X... Y... Z...`
- Mehrere aufeinander folgenden Sätze:
`N10 ROT Y...`
`N20 AROT X...`
`N30 AROT Z...`

Raumwinkel

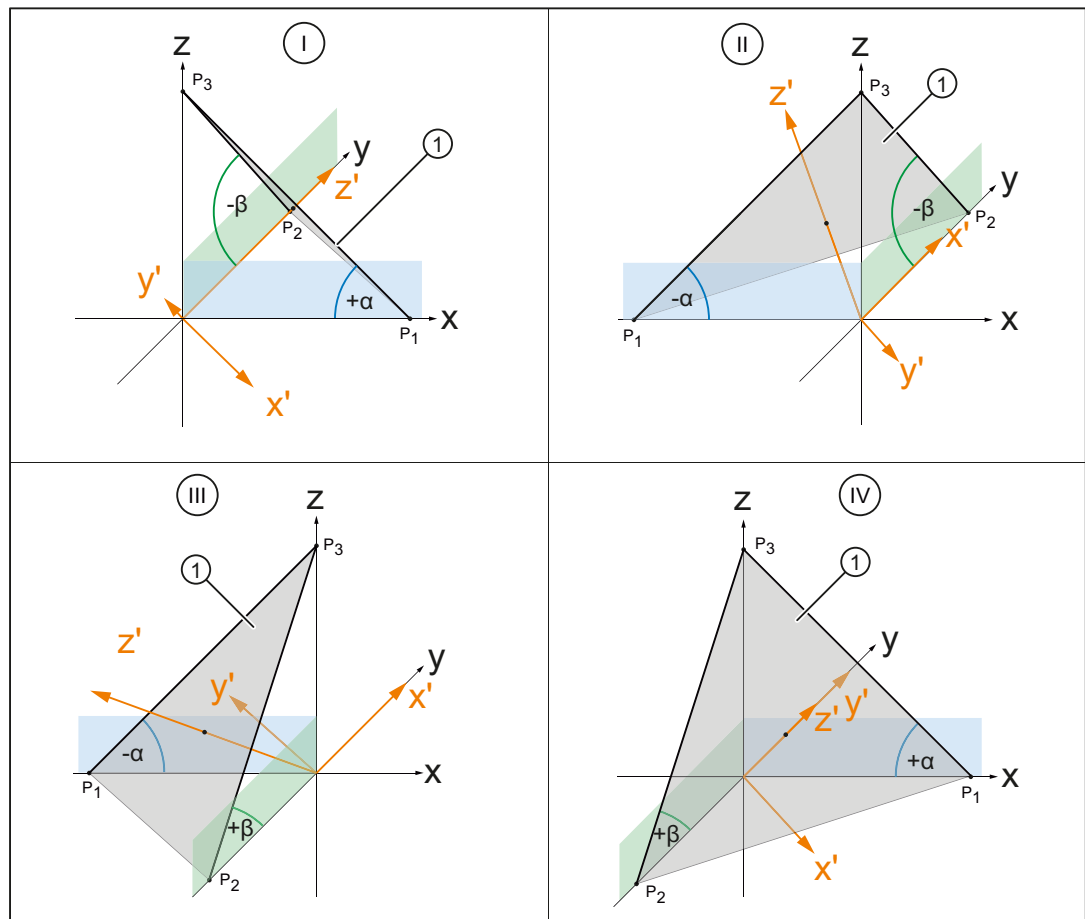
In Werkstückzeichnungen werden zur Beschreibung schräger Flächen oftmals Raumwinkel angegeben. Raumwinkel sind dabei die Winkel welche die Schnittgeraden der schrägen Ebene mit den Hauptebenen (X-Y, Y-Z, Z-X-Ebene) des Werkstückkoordinatensystems bilden (siehe nachfolgendes Bild). Die Orientierung einer Ebene im Raum ist durch die Angabe zweier Raumwinkel eindeutig bestimmt. Der dritte Raumwinkel ergibt sich aus den beiden ersten.

Mit den Befehlen `ROTS`, `AROTS` und `CROTS` können die Drehungen unmittelbar als Raumwinkel beschrieben werden.

Die Angabe eines einzelnen Raumwinkels ist zulässig. Die Drehung, die in diesem Fall bei `ROTS` bzw. `AROTS` ausgeführt wird, ist identisch zu denen bei `ROT` bzw. `AROT`.

Die beiden im Befehl programmierten Achsen definieren eine Ebene. Die nicht programmierte Achse definiert die zugehörige dritte Achse des kartesischen Koordinatensystems. Damit ist für die beiden programmierten Achsen auch eindeutig definiert, welches die erste und welches die zweite Achse ist. Die Definition entspricht dabei den Festlegungen der Ebenendefinition bei `G17/G18/G19`.

Der mit dem Achsbuchstaben einer Achse der Ebene programmierte Winkel gibt dann die Achse an, um die man die andere Achse der Ebene drehen muss, um diese in die Schnittgerade zu überführen, die die gedrehte Ebene mit der aus der anderen und der dritten Achse aufgespannten Ebene bildet. Mit dieser Definition ist gewährleistet, dass für den Fall, dass einer der beiden programmierten Winkel gegen Null geht, die derart definierte Ebene in die Ebene übergeht, die sich ergibt, wenn nur eine Achse (z. B. auch mit `ROT` oder `AROT`) programmiert ist.



I, ..., Quadrant 1 bis 4
IV

① Schräge Ebene als Vorgabe für die neue G17-Ebene

α, β Raumwinkel der schrägen Ebene

Bild 11-27 Drehung um Raumwinkel

Im Bild sind die Raumwinkel für eine beispielhafte Ebene in den Quadranten I bis IV aufgezeigt. Die schräge Ebene definiert die Ausrichtung der G17-Ebene nach der Drehung des Werkstück-

Koordinatensystems WKS. Die Vorzeichen der Raumwinkel geben die Richtung an, um die das Koordinatensystem um die jeweilige Achse gedreht wird:

1. Drehung um y:
Drehung des Werkstück-Koordinatensystems WKS um die y-Achse um den vorzeichenbehafteten Winkel $\alpha \Rightarrow$
x'-Achse ist parallel (kollinear) zur Schnittgeraden der xz-Ebene mit der schrägen Ebene ausgerichtet
2. Drehung um x':
Drehung des neuen Werkstück-Koordinatensystems WKS' um x'-Achse um den vorzeichenbehafteten Winkel $\beta \Rightarrow$
 - y'-Achse ist parallel (kollinear) zur Schnittgeraden der zy-Ebene mit der schrägen Ebene ausgerichtet
 - z'-Achse steht senkrecht auf der schrägen Ebene
 - G17' liegt parallel zur schrägen Ebene

Die entsprechende Programmierung zur Ausrichtung der G17-Ebene des Werkstück-Koordinatensystems WKS auf die schräge Ebene lautet pro Quadrant:

- Quadrant I: ROTs X<+ α > Y<- β >
- Quadrant II: ROTs X<- α > Y<- β >
- Quadrant III: ROTs X<- α > Y<+ β >
- Quadrant IV: ROTs X<+ α > Y<+ β >

Orientierung

Mit der Angabe der Raumwinkel ist die Orientierung des zweidimensionalen Koordinatensystems innerhalb der Ebene (d.h. der Drehwinkel um den Flächennormalenvektor) nicht definiert. Die Lage des Koordinatensystems wird deshalb so festgelegt, dass die gedrehte erste Achse in der Ebene liegt, die von der ersten und dritten Achse des nicht gedrehten Koordinatensystems aufgespannt wird.

Das bedeutet:

- Bei Programmierung von X und Y liegt die neue X-Achse in der ursprünglichen Z-X-Ebene.
- Bei Programmierung von Z und X liegt die neue Z-Achse in der ursprünglichen Y-Z-Ebene.
- Bei Programmierung von Y und Z liegt die neue Y-Achse in der ursprünglichen X-Y-Ebene.

Wird eine von dieser Voreinstellung abweichende Lage des Koordinatensystems benötigt, muss eine zusätzliche Drehung mit `AROT` . . . ausgeführt werden.

Parametrierung: zy'x"- (RPY-Winkel) oder zx'z"-Konvention

Die programmierten Raumwinkel werden bei der Eingabe abhängig von folgendem Maschinendatum in die äquivalenten Euler-Winkel nach zy'x"-Konvention (RPY-Winkel) oder zx'z"-Konvention umgerechnet:

MD10600 \$MN_FRAME_ANGLE_INPUT_MODE

Framedrehung in Werkzeugrichtung

Mit dem bereits in älteren Softwareständen vorhandenen Sprachbefehl `TOFRAME` besteht die Möglichkeit, einen Frame zu definieren, dessen Z-Achse in Werkzeugrichtung zeigt. Ein vorhandener programmierter Frame wird dabei durch einen Frame überschrieben, der eine reine Drehung beschreibt. Im vorher aktiven Frame eventuell vorhandene Nullpunktverschiebungen, Spiegelungen oder Skalierungen werden gelöscht. Mitunter ist dieses Verhalten störend. Insbesondere ist es oft sinnvoll, eine Nullpunktverschiebung, mit der der Bezugspunkt im Werkstück definiert wird, beizubehalten.

Es wird deshalb zusätzlich der Sprachbefehl `TOROT` eingeführt, der im programmierten Frame nur den Rotationsanteil überschreibt, die übrigen Komponenten aber unverändert lässt. Die durch `TOROT` definierte Drehung ist die gleiche wie bei `TOFRAME`.

`TOROT` ist ebenso wie `TOFRAME` unabhängig vom Vorhandensein eines orientierbaren Werkzeugträgers. Insbesondere ist dieser Sprachbefehl auch bei 5-Achs-Transformationen verwendbar.

Mit dem neuen Sprachbefehl `TOROT` wird außerdem eine konsistente Programmierung bei aktiven orientierbaren Werkzeugträgern für jeden Kinematiktyp erreicht.

Mit `TOFRAME` bzw. `TOROT` werden Frames definiert, deren Z-Richtung in Werkzeugrichtung zeigt. Diese Definition passt zu Fräsbearbeitungen, bei denen typischerweise `G17` aktiv ist. Insbesondere bei Drehbearbeitungen oder allgemein bei aktivem `G18` oder `G19` ist es jedoch wünschenswert, dass Frames definiert werden können, bei denen die Ausrichtung an der X- bzw. Y-Achse erfolgt. Dazu sind in der G-Gruppe 53 folgende Befehle vorhanden:

- `TOFRAMEX`, `TOFRAMEY`, `TOFRAMEZ`
- `TOROTX`, `TOROTY`, `TOROTZ`

Mit diesen Befehlen ist es möglich, entsprechende Frames zu definieren. Dabei ist die Funktionalität von `TOFRAME` und `TOFRAMEZ` bzw. von `TOROT` und `TOROTZ` jeweils identisch.

Die durch `TOROT` bzw. `TOFRAME` entstehenden Frames können in einen eigenen Systemframe (`$P_TOOLFR`) geschrieben werden. Der programmierbare Frame bleibt dann unverändert erhalten.

- Voraussetzung: MD28082 `$MC_MM_SYSTEM_FRAME_MASK`, Bit 3 = 1

Bei der Programmierung von `TOROT` bzw. `TOFRAME` usw. ist das Verhalten mit bzw. ohne Systemframe identisch. Unterschiede ergeben sich dann, wenn der programmierbare Frame weiter bearbeitet wird.

Hinweis

Es wird empfohlen, für die von den Befehlen der G-Gruppe 53 erzeugten Frames bei Neuanlagen nur noch den dafür vorgesehenen Systemframe zu verwenden.

Beispiel

Nach `TOROT` wird `TRANS` programmiert. `TRANS` ohne Angabe von Parametern löscht den programmierbaren Frame. In der Variante ohne Systemframe wird damit auch der durch `TOROT` verursachte Frameanteil des programmierbaren Frames gelöscht, steht der `TOROT`-Anteil im Systemframe, bleibt er erhalten.

`TOROT` bzw. `TOFRAME` usw. werden mit dem Sprachbefehl `TOROTOF` ausgeschaltet. `TOROTOF` löscht den gesamten Systemframe `$P_TOOLFR`. Beschreiben die Befehle

TOFRAME usw. nicht das Systemframe, sondern das programmierbare Frame (alte Variante), löscht TOROT nur den Rotationsanteil und lässt die übrigen Frameanteile unverändert.

Ist vor der Aktivierung der Sprachbefehle TOFRAME oder TOROT bereits ein drehender Frame aktiv, besteht oft die Forderung, dass der neu definierte Frame vom alten Frame möglichst wenig abweicht. Das ist z.B. dann der Fall, wenn eine Framedefinition leicht modifiziert werden muss weil die Werkzeugorientierung wegen Hirth-verzählter Rundachsen nicht beliebig eingestellt werden kann. Durch die genannten Sprachbefehle wird die Z-Richtung des neuen Frames eindeutig definiert.

Parametrierung: Frame-Definition bei TOFRAME, TOROT und PAROT (SD42980)

Mit dem folgenden Settingdatum wird die Richtung der Geometrieachsen der aktuellen Bearbeitungsebene (G17: X- und Y-Achse) bei der Frame-Definition mittels TOFRAME, TOROT (TOROTY, TOROTX) oder PAROT festgelegt.

Bei einer Frameberechnung wird die Werkzeugrichtung so festgelegt, dass Werkzeugrichtung und Applikate (G17: Z-Achse) des Frames parallel sind und senkrecht auf der Bearbeitungsebene stehen.

Die Drehung um die Werkzeugachse ist zunächst beliebig. Mit dem Settingdatum kann diese freie Drehung so bestimmt werden, dass der neu definierte Frame von einem vorher aktiven Frame möglichst wenig abweicht.

In allen Fällen, in denen das Settingdatum ungleich Null ist, bleibt ein aktiver Frame unverändert, wenn die Werkzeugrichtung des alten und des neuen Frame übereinstimmen.

SD42980 \$SC_TOFRAME_MODE

TCARR (Werkzeugträger anfordern) und PAROT (Werkstückkoordinatensystem am Werkstück ausrichten)

TCARR nutzt das durch das folgende Maschinendatum bezeichnete Basisframe:
MD20184 \$MC_TOCARR_BASE_FRAME_NUMBER.

Um Konflikte mit Systemen zu vermeiden, die bereits alle Basisframes nutzen, kann für TCARR und PAROT ein eigenes Systemframe angelegt werden.

PAROT, TOROT und TOFRAME verEändern bisher den Rotationsanteil des programmierbaren Frames. Ein separates Abschalten von PAROT oder TOROT ist in diesem Falle nicht möglich. Bei RESET wird das programmierbare Frame gelöscht, was bedeutet, dass nach Betriebsartenwechsel nach JOG der Rotationsanteil von PAROT und TOROT nicht mehr vorhanden ist. Auch muss das programmierbare Frame dem Anwender uneingeschränkt zur Verfügung stehen. Die durch PAROT und TOROT erzeugten Frames müssen über die Datensicherung archivierbar und wieder ladbar sein.

Das Systemframe für TCARR und PAROT wird projiziert mit:
MD28082 \$MC_MM_SYSTEM_FRAME_MASK, Bit 2 = 1

Folgendes Maschinendatum wird dann nicht mehr ausgewertet:
MD20184 \$MC_TOCARR_BASE_FRAME_NUMBER

Ist das Systemframe für TCARR projiziert, so beschreibt TCARR und PAROT das entsprechende Systemframe, ansonsten wird das durch Maschinendatum MD20184 bezeichnete Basisframe beschrieben.

TCARR trägt bei Kinematiken des Typs P und des Typs M den Tischoffset des orientierbaren Werkzeugträgers (Verschiebung des Nullpunktes als Folge der Drehung des Tisches), als Translation in das Systemframe ein. PAROT rechnet das Systemframe so um, dass sich ein werkstückbezogenes Werkstückkoordinatensystem ergibt.

Die Systemframes werden remanent gespeichert und bleiben deshalb nach Reset erhalten. Auch bei Betriebsartenwechsel bleiben die Systemframes aktiv.

Für die Anzeige werden die Befehle PAROT und TOROT, TOFRAME jeweils einer eigenen G-Gruppe zugeordnet.

PAROTOF

PAROTOF ist der Ausschaltbefehl zu PAROT. Dieser Befehl löscht die Drehungen im Systemframe für PAROT ab. Es werden dabei die Drehungen im aktuellen \$P_PARTFRAME und im Datenhaltungsframe \$P_PARTFR gelöscht. Damit wird die Lage des Koordinatensystems nach TCARR wieder hergestellt. PAROTOF ist in der gleichen G-Gruppe, wie PAROT und erscheint deshalb in der G-Befehlsanzeige.

TOROT (Z-Achse des WKS durch Frame-Drehung parallel zur Werkzeugorientierung ausrichten) und TOFRAME (dto.)

Das Systemframe für TOROT und TOFRAME wird über folgendes Maschinendatum aktiviert: MD28082 \$MC_MM_SYSTEM_FRAME_MASK, Bit 3 = 1

Dieses Systemframe liegt in der Framekette vor dem programmierbaren Frame. Das ENS-Koordinatensystem liegt entsprechend vor dem programmierbaren Frame.

TOROTOF

TOROTOF ist der Ausschaltbefehl zu TOROT und TOFRAME. Dieser Befehl löscht den entsprechenden Systemframe. Dabei wird der aktuelle \$P_TOOLFRAME und der Datenhaltungsframe \$P_TOOLFR gelöscht. TOROTOF ist in der gleichen G-Gruppe wie TOROT und TOFRAME und erscheint deshalb in der G-Befehlsanzeige.

Beispiel

Verwendung eines orientierbaren Werkzeugträgers mit aufgelöster Kinematik.

```
N10 $TC_DP1[1,1]= 120
N20 $TC_DP3[1,1]= 13           ; Werkzeugläng 13mm
; Definition des Werkzeugträgers 1:
N30 $TC_CARR1[1] = 0           ; X-Komponente des 1. Offsetvektors
N40 $TC_CARR2[1] = 0           ; Y-Komponente des 1. Offsetvektors
N50 $TC_CARR3[1] = 0           ; Z-Komponente des 1. Offsetvektors
N60 $TC_CARR4[1] = 0           ; X-Komponente des 2. Offsetvektors
N70 $TC_CARR5[1] = 0           ; Y-Komponente des 2. Offsetvektors
N80 $TC_CARR6[1] = -15.        ; Z-Komponente des 2. Offsetvektors
N90 $TC_CARR7[1] = 1           ; X-Komponente der 1. Achse
N100 $TC_CARR8[1] = 0          ; Y-Komponente der 1. Achse
```

```

N110 $TC_CARR9[1] = 0           ; Z-Komponente der 1. Achse
N120 $TC_CARR10[1] = 0          ; X-Komponente der 2. Achse
N130 $TC_CARR11[1] = 1          ; Y-Komponente der 2. Achse
N140 $TC_CARR12[1] = 0          ; Z-Komponente der 2. Achse
N150 $TC_CARR13[1] = 30.        ; Drehwinkel der 1. Achse
N160 $TC_CARR14[1] = -30.       ; Drehwinkel der 2. Achse
N170 $TC_CARR15[1] = 0          ; X-Komponente des 3. Offsetvektors
N180 $TC_CARR16[1] = 0          ; Y-Komponente des 3. Offsetvektors
N190 $TC_CARR17[1] = 0          ; Z-Komponente des 3. Offsetvektors
N200 $TC_CARR18[1] = 0          ; X-Komponente des 4. Offsetvektors
N210 $TC_CARR19[1] = 0          ; Y-Komponente des 4. Offsetvektors
N220 $TC_CARR20[1] = 15.        ; Z-Komponente des 4. Offsetvektors
N230 $TC_CARR21[1] = A          ; Bezug fuer 1. Achse
N240 $TC_CARR22[1] = B          ; Bezug fuer 2. Achse
N250 $TC_CARR23[1] = "M"        ; Typ des Werkzeugtraegers
N260 X0 Y0 Z0 A0 B45 F2000
N270 TCARR=1 X0 Y10 Z0 T1 TCOABS ; Anwahl des orient. Werkzeugträgers
N280 PAROT                      ; Drehung des Tisches
N290 TOROT                      ; Drehung der z-Achse in WZ-Orient.
N290 X0 Y0 Z0
N300 G18 MOVT=AC(20)            ; Bearbeitung in G18 Ebene
N310 G17 X10 Y0 Z0             ; Bearbeitung in G17 Ebene
N320 MOVT=-10
N330 PAROTOF                   ; Drehung des Tisches abschalten
N340 TOROTOF                   ; WKS nicht mehr am WZ ausrichten.
N400 M30

```

11.5.9 Unterprogramme mit Attribut SAVE

Bei verschiedenen Frames ist das Verhalten bezüglich Unterprogrammen mit dem Attribut SAVE einstellbar.

Einstellbare Frames G54 - G599

Mit MD10617 \$MN_FRAME_SAVE_MASK.BIT0 kann das Verhalten der einstellbaren Frames eingestellt werden:

- BIT0 = 0
Werden durch das Unterprogramm nur die Werte des aktiven einstellbaren Frames über die Systemvariable \$P_IFRAME verändert, der G-Befehl aber beibehalten, bleibt die Veränderung auch nach Unterprogrammende erhalten.
- BIT0 = 1
Mit dem Unterprogrammende wird der vor dem Unterprogrammaufruf aktive einstellbare Frame, G-Befehl und Werte, reaktiviert.

Basisframes \$P_CHBFR[] und \$P_NCBFR[]

Mit MD10617 \$MN_FRAME_SAVE_MASK.BIT1 kann das Verhalten der Basisframes eingestellt werden:

- BIT1 = 0
Wird durch das Unterprogramm der aktive Basisframe verändert, bleibt die Veränderung auch nach Unterprogrammende erhalten.
- BIT1 = 1
Mit dem Unterprogrammende wird der vor dem Unterprogrammaufruf aktive Basisframe reaktiviert.

Programmierbarer Frame

Mit dem Unterprogrammende wird der vor dem Unterprogrammaufruf aktive programmierbare Frame reaktiviert.

Systemframes

Werden durch das Unterprogramm Systemframes verändert, bleibt die Veränderung nach Unterprogrammende erhalten.

11.5.10 Datensicherung**Datenbausteine**

Die Datensicherung der Frames erfolgt in folgende Datenbausteine:

- Kanalspezifische Frames ⇒ Datenbausteine _N_CHAN<x>_UFR
- Globale Frames ⇒ Datenbaustein _N_NC_UFR

Hinweis

- Es wird dringen empfohlen, zwischen dem Sichern und Wiedereinspielen der gesicherten Systemframes das folgende Maschinendatum nicht zu ändern. Ansonsten können die gesicherte Systemframes eventuell nicht mehr geladen werden.
MD28082 \$MC_MM_SYSTEM_FRAME_MASK (Projektierung von kanalspezifischen Systemframes, die im Kanal eingerechnet werden)
- Die Datensicherung erfolgt immer nach den aktuell im Kanal gültigen Geometrieachs-Zuordnungen und nicht nach den in den Maschinendaten ursprünglich eingestellten Achskonstellationen.

Datensicherung von Systemframes

Bei der Datensicherung von Systemframes, werden nur die in der Datenhaltung angelegten Systemframes berücksichtigt. Diese Systemframes wurden bei der Inbetriebnahme der Steuerung über das folgende Maschinendatum ausgewählt:

MD28083 \$MC_MM_SYSTEM_DATAFRAME_MASK

Nicht in der Datenhaltung angelegte Frames werden nicht gesichert.

Datensicherung von NC-globalen Frames

Eine Datensicherung von NC-globalen Frames erfolgt nur, wenn in einem der folgenden Maschinendaten mindestens ein NC-globaler Frame parametrier ist:

- MD18601 \$MN_MM_NUM_GLOBAL_USER_FRAMES
- MD18602 \$MN_MM_NUM_GLOBAL_BASE_FRAMES
- MD18603 \$MN_MM_NUM_GLOBAL_G_FRAMES

11.5.11 Positionen in den Koordinatensystemen

Die aktuellen Sollwert-Positionen in den Koordinatensystemen können über folgende Systemvariablen gelesen werden. Über PLC können wahlweise die Istwerte im WKS, ENS, BNS oder MKS angezeigt werden. Es gibt dazu einen Softkey Istwertanzeige im MKS/WKS. Der Maschinenhersteller kann von PLC-Seite vorgeben, welches Koordinatensystem bei seiner Maschine dem Werkstückkoordinatensystem entspricht. Der HMI fordert die entsprechenden Istwerte von NC an.

\$AA_IM[Achse]

Die Sollwerte im Maschinenkoordinatensystem können für jede Achse mit der Variablen \$AA_IM[Achse] gelesen werden.

\$AA_IEN[Achse]

Die Sollwerte im einstellbaren Nullpunkt-Koordinatensystem ENS können für jede Achse mit der Variablen \$AA_IEN[Achse] gelesen werden.

\$AA_IBN[Achse]

Mit \$AA_IBN[Achse] können die Sollwerte im Basisnullpunkt-Koordinatensystem BNS gelesen werden.

\$AA_IW[Achse]

Mit \$AA_IW[Achse] können die Sollwerte im Werkstück-Koordinatensystem WKS gelesen werden.

11.5.12 Steuerungsverhalten

11.5.12.1 POWER ON

Frame	Zustand nach POWER ON
Programmierbarer Frame \$P_PFRAME	Gelöscht.
Einstellbare Frame \$P_IFRAME	Bleibt erhalten, abhängig von: <ul style="list-style-type: none"> • MD24080 \$MC_USER_FRAME_POWERON_MASK, Bit 0 • MD20152 \$MC_GCODE_RESET_MODE[7]

Frame	Zustand nach POWER ON
Schleifframe \$P_GFRAME	Bleibt erhalten, abhängig von: <ul style="list-style-type: none"> MD24080 \$MC_USER_FRAME_POWERON_MASK, Bit 0 MD20152 \$MC_GCODE_RESET_MODE[63]
Gesamt-Basisframe \$P_ACTBF-RAME	Bleibt erhalten, abhängig von: <ul style="list-style-type: none"> MD20110 \$MC_RESET_MODE_MASK Bit 0 und Bit 14 Über Maschinendaten können Basisframes gelöscht werden: <ul style="list-style-type: none"> MD10615 \$MN_NCBFRAME_POWERON_MASK MD24004 \$MC_CHBFRAME_POWERON_MASK
Systemframes: \$P_PARTFRAME, \$P_SETFRAME, \$P_ISO1FRAME, \$P_ISO2FRAME, \$P_ISO3FRAME, \$P_TOOLFRAME, \$P_WPFRAME, \$P_TRAFRAME, \$P_ISO4FRAME, \$P_RELFRAME, \$P_CACFRAME	Bleiben erhalten. Über das Maschinendatum können einzelne Systemframes gelöscht werden: <ul style="list-style-type: none"> MD24008 \$MC_CHSFRAME_POWERON_MASK Das Löschen von Systemframes wird erstrangig in der Datenhaltung durchgeführt.
Externe Nullpunktverschiebung \$P_EXTFRAME	Bleibt erhalten, muss aber erneut aktiviert werden.
DRF-Verschiebung	Gelöscht.

11.5.12.2 Betriebsartenwechsel

Systemframes

Die Systemframes bleiben erhalten und aktiv.

Betriebsart JOG

In JOG werden nur die Framekomponenten des aktuellen Frames für die Geometrieachsen berücksichtigt, wenn eine Rotation aktiv ist. Alle anderen achsspezifischen Frames werden nicht berücksichtigt.

PLC- und Kommando-Achsen

Für PLC- und Kommando-Achsen ist das Verhalten einstellbar über das Maschinendatum:

MD32074 \$MA_FRAME_OR_CORRPOS_NOTALLOWED (Frame oder HL-Korrektur sind unzulässig)

11.5.12.3 Kanal-Reset / Teileprogrammende

Reset-Verhalten der Basisframes

Das Reset-Verhalten der Basisframes wird eingestellt über das Maschinendatum:

MD20110 \$MC_RESET_MODE_MASK (Festlegung der Steuerungs-Grundstellung nach Kanal-Reset / Teileprogrammende)

Reset-Verhalten der Systemframes

Die Systemframes bleiben auch nach Kanal-Reset / Teileprogrammende in der Datenhaltung erhalten.

Die Aktivierung der einzelnen Systemframes kann über die folgenden Maschinendaten projiziert werden:

MD24006 \$MC_CHSFRAME_RESET_MASK, Bit<n> = <Wert> (Aktive Systemframes nach Kanal-Reset / Teileprogrammende)

Bit	Wert	Bedeutung
0	1	Systemframe für Istwertsetzen und Ankratzen ist nach Kanal-Reset / Teileprogrammende aktiv.
1	1	Systemframe für externe Nullpunktverschiebung ist nach Kanal-Reset / Teileprogrammende aktiv.
2	---	Wird nicht ausgewertet.
3	---	Wird nicht ausgewertet.
4	1	Systemframe für Werkstückbezugspunkte ist nach Kanal-Reset / Teileprogrammende aktiv.
5	1	Systemframe für Zyklen ist nach Kanal-Reset / Teileprogrammende aktiv.
6	---	Reserviert, RESET-Verhalten abhängig von MD20110 \$MC_RESET_MODE_MASK.
7	1	Systemframe \$P_ISO1FR ist nach Kanal-Reset / Teileprogrammende aktiv.
8	1	Systemframe \$P_ISO2FR ist nach Kanal-Reset / Teileprogrammende aktiv.
9	1	Systemframe \$P_ISO3FR ist nach Kanal-Reset / Teileprogrammende aktiv.
10	1	Systemframe \$P_ISO4FR ist nach Kanal-Reset / Teileprogrammende aktiv.
11	1	Systemframe \$P_RELFR ist nach Kanal-Reset / Teileprogrammende aktiv.
Bei Bit<n> = 0 wird der entsprechende Systemframe nicht aktiv.		

Reset-Verhalten der Systemframes für TCARR, PAROT, TOROT und TOFRAME

Das Reset-Verhalten der Systemframes für TCARR, PAROT, TOROT und TOFRAME nach Kanal-Reset / Teileprogrammende ist von der Reset-Einstellung der G-Befehle abhängig.

Die Einstellung erfolgt mit den Maschinendaten:

- MD20110 \$MC_RESET_MODE_MASK, Bit<n> = <Wert>

Bit	Wert	Bedeutung	
0	0	Der aktuelle Systemframe für TCARR und PAROT bleibt erhalten.	
0	1	Weitere relevante Maschineneinstellungen	Auswirkung
		MD20152 \$MC_GCODE_RESET_MODE[51] = 0 UND MD20150 \$MC_GCODE_RESET_VALUES[51] = 1	PAROTOF
		MD20152 \$MC_GCODE_RESET_MODE[51] = 0 UND MD20150 \$MC_GCODE_RESET_VALUES[51] = 2	PAROT
		MD20152 \$MC_GCODE_RESET_MODE[51] = 1	TCARR und PAROT bleiben er- halten
		MD20152 \$MC_GCODE_RESET_MODE[52] = 0 UND MD20150 \$MC_GCODE_RESET_VALUES[52] = 1	TOROTOF
		MD20152 \$MC_GCODE_RESET_MODE[52] = 0 UND MD20150 \$MC_GCODE_RESET_VALUES[52] = 2	TOROT
		MD20152 \$MC_GCODE_RESET_MODE[52] = 0 UND MD20150 \$MC_GCODE_RESET_VALUES[52] = 3	TOFRAME
		MD20152 \$MC_GCODE_RESET_MODE[52] = 1	TOROT und TOFRAME bleiben erhalten
Beschreibung der weiteren Bits bei Bit 0 == 1			
14	0	Verkettetes Gesamtbasisframe wird gelöscht.	
	1	Das Gesamtbasisframe ergibt sich anhand folgender weiterer Maschinenda- ten::	
		MD24002 \$MC_CHBFRAME_RESET_MASK, Bit<n> = 1 n: der n-te kanalspezifischer Basisframe wird in das verkettete Gesamtbasisf- rame eingerechnet.	
		MD10613 \$MN_NCBFRAME_RESET_MASK, Bit<n> = 1 n: der n-te NCU-globaler Basisframe wird in das verkettete Gesamtbasisframe eingerechnet.	

TCARR und PAROT sind zwei eigenständige Funktionen, die das gleiche Frame beschreiben. Bei Kanal-Reset / Teileprogrammende wird mit PAROTOF auch der Anteil von TCARR nicht aktiviert.

- MD20152 \$MC_GCODE_RESET_MODE[] (RESET-Verhalten der G-Gruppen)
- MD20150 \$MC_GCODE_RESET_VALUES[] (Löschstellung der G-Gruppen)

Framezustände nach Kanal-Reset / Teileprogrammende

Frame	Zustand nach Kanal-Reset / Teileprogrammende
Programmierbarer Frame: \$P_PFRAME	Gelöscht.
Einstellbare Frame \$P_IFRAME	Bleibt erhalten, abhängig von: <ul style="list-style-type: none"> MD20110 \$MC_RESET_MODE_MASK MD20152 \$MC_GCODE_RESET_MODE[7]
Schleifframe \$P_GFRAME	Bleibt erhalten, abhängig von: <ul style="list-style-type: none"> MD20110 \$MC_RESET_MODE_MASK MD20152 \$MC_GCODE_RESET_MODE[63]
Gesamt-Basisframe \$P_ACTBFRAME	Bleibt erhalten, abhängig von: <ul style="list-style-type: none"> MD20110 \$MC_RESET_MODE_MASK Bit 0 und Bit 14 MD10613 \$MN_NCBFRAME_RESET_MASK MD24002 \$MC_CHBFRAME_RESET_MASK
Systemframes: \$P_PARTFRAME, \$P_SETFRAME, \$P_ISO1FRAME, \$P_ISO2FRAME, \$P_ISO3FRAME, \$P_TOOLFRAME, \$P_WPFRAME, \$P_TRAFRAME, \$P_ISO4FRAME, \$P_RELFRAME, \$P_CACFRAME	Bleiben erhalten, abhängig von: <ul style="list-style-type: none"> MD24006 \$MC_CHSFRAME_RESET_MASK MD20150 \$MC_GCODE_RESET_VALUES[<n>]
Externe Nullpunktverschiebung \$P_EXTFRAME	Bleibt erhalten.
DRF-Verschiebung	Bleibt erhalten.

Löschen von Systemframes

Die Systemframes in der Datenhaltung können bei Kanal-Reset / Teileprogrammende über folgendes Maschinendatum gelöscht werden:

MD24007 \$MC_CHSFRAME_RESET_CLEAR_MASK, Bit<n> = <Wert>

Bit	Wert	Bedeutung
0	1	Systemframe für Istwertsetzen und Ankratzen wird bei Kanal-Reset / Teileprogrammende gelöscht.
1	1	Systemframe für externe Nullpunktverschiebung wird bei Kanal-Reset / Teileprogrammende gelöscht.
2	---	Reserviert, TCARR und PAROT siehe MD20150 \$MC_GCODE_RESET_VALUES[].
3	---	Reserviert, TOROT und TOFRAME siehe MD20150 \$MC_GCODE_RESET_VALUES[].
4	1	Systemframe für Werkstückbezugspunkte wird bei Kanal-Reset / Teileprogrammende gelöscht.
5	1	Systemframe für Zyklen wird bei Kanal-Reset / Teileprogrammende gelöscht.
6	---	Reserviert, Reset-Verhalten abhängig von MD20110 \$MC_RESET_MODE_MASK.
7	1	Systemframe \$P_ISO1FR wird bei Kanal-Reset / Teileprogrammende gelöscht.

Bit	Wert	Bedeutung
8	1	Systemframe \$P_ISO2FR wird bei Kanal-Reset / Teileprogrammende gelöscht.
9	1	Systemframe \$P_ISO3FR wird bei Kanal-Reset / Teileprogrammende gelöscht.
10	1	Systemframe \$P_ISO4FR wird bei Kanal-Reset / Teileprogrammende gelöscht.
11	1	Systemframe \$P_RELFR wird bei Kanal-Reset / Teileprogrammende gelöscht.
Bei Bit<n> = 0 wird der entsprechende Systemframe nicht gelöscht.		

11.5.12.4 Teileprogrammstart

Framezustände nach Teileprogrammstart

Frame	Zustand nach Teileprogrammstart
Programmierbarer Frame \$P_PFRAME	Gelöscht.
Einstellbare Frame \$P_IFRAME	Bleibt erhalten, abhängig von: MD20112 \$MC_START_MODE_MASK
Schleifframe \$P_GFRAME	Bleibt erhalten, abhängig von: MD20112 \$MC_START_MODE_MASK
Gesamt-Basisframe \$P_ACTBFRAME	Bleibt erhalten.
Systemframes: \$P_PARTFRAME, \$P_SETFRAME, \$P_ISO1FRAME, \$P_ISO2FRAME, \$P_ISO3FRAME, \$P_TOOLFRAME, \$P_WPFRAME, \$P_TRAFRAME, \$P_ISO4FRAME, \$P_RELFRAME, \$P_CACFRAME	Bleiben erhalten.
Externe Nullpunktverschiebung \$P_EXTFRAME	Bleibt erhalten.
DRF-Verschiebung	Bleibt erhalten.

11.5.12.5 Satzsuchlauf

Satzsuchlauf mit Berechnung

Beim Satzsuchlauf mit Berechnung werden auch Datenhaltungsframes modifiziert.

Abbruch des Satzsuchlaufs

Wird ein Satzsuchlauf mit Kanal-Reset abgebrochen, so kann mit dem folgenden Maschinendatum eingestellt werden, dass alle Datenhaltungsframes auf den Wert vor dem Satzsuchlauf zurückgesetzt werden:

MD28560 \$MC_MM_SEARCH_RUN_RESTORE_MODE, Bit<n>

Bit	Wert	Bedeutung
0	1	Alle Frames in der Datenhaltung werden restauriert.

Bei kaskadiertem Satzsuchlauf, werden die Frames auf den Zustand des Vorgänger-Satzsuchlaufes gesetzt.

SERUPRO

Die Funktion "SERUPRO" wird nicht unterstützt.

11.5.12.6 REPOS

Es gibt keine Sonderbehandlung für Frames. Wird ein Frame in einem ASUP geändert, so bleibt es im Programm erhalten. Beim Wiederaufstart mit REPOS wird ein geändertes Frame berücksichtigt, sofern die Änderung im ASUP aktiviert wurde.

11.6 Werkstücknahes Istwertsystem

11.6.1 Übersicht

Definition

Unter dem Begriff "Werkstücknahes Istwertsystem" werden eine Reihe von Funktionen zusammengefasst, die dem Anwender folgendes Vorgehen ermöglichen:

- Nach Hochlauf auf ein über Maschinendaten definiertes Werkstückkoordinatensystem aufsetzen.
Merkmale:
 - keine zusätzlichen Bedienhandlungen nötig
 - in der Betriebsart JOG und AUTOMATIK wirksam
- Gültige Einstellungen bei Teileprogrammende für das folgende Teileprogramm beibehalten bezüglich:
 - aktiver Ebene
 - einstellbarer Frame (G54-G57)
 - kinematischer Transformation
 - aktiver Werkzeugkorrektur
- Über die Bedienoberfläche zwischen Werkstückkoordinatensystem und Maschinenkoordinatensystem wechseln.
- Das Werkstückkoordinatensystem per Bedienung ändern (z.B. Änderung des einstellbaren Frame oder der Werkzeugkorrektur).

11.6.2 Benutzung des werkstücknahen Istwertsystems

Voraussetzungen, Grundsätzliches

Für das System wurden die im vorigen Abschnitt beschriebenen Einstellungen getroffen. Die Voreinstellung nach Hochlauf der HMI-Software ist MKS.

Umschalten auf WKS

Die Umschaltung auf das WKS über die Bedienoberfläche bewirkt, dass die Achspositionen bezüglich des Ursprungs des WKS angezeigt werden.

Umschalten auf MKS

Die Umschaltung auf das MKS über die Bedienoberfläche bewirkt, dass die Achspositionen bezüglich des Ursprungs des MKS angezeigt werden.

Zusammenhänge zwischen Koordinatensystemen

Das folgende Bild stellt die Zusammenhänge vom Maschinenkoordinatensystem MKS bis zum Werkstückkoordinatensystem WKS dar.

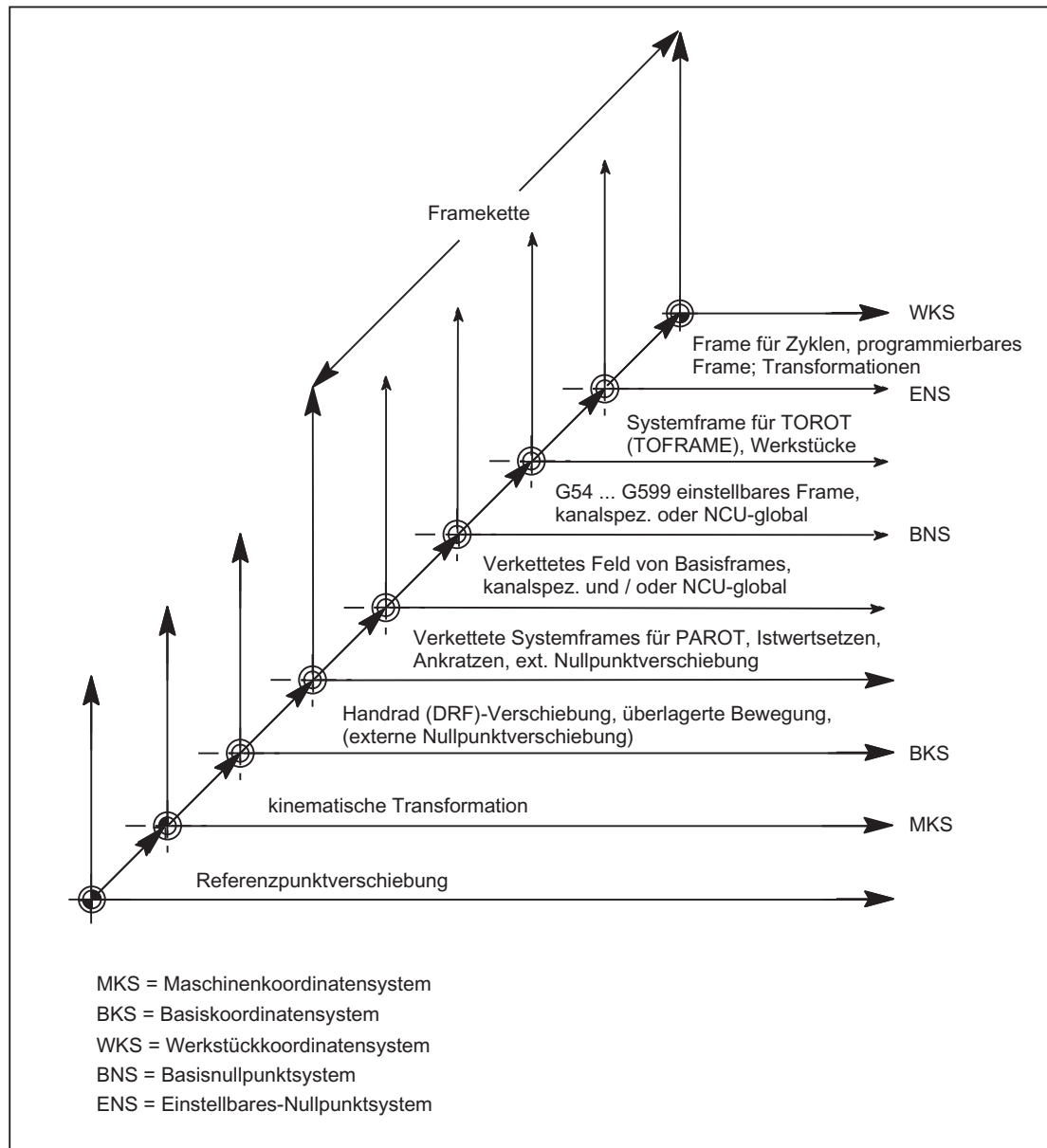


Bild 11-28 Zusammenhang Koordinatensysteme

Für weitere Informationen siehe "H2: Hilfsfunktionsausgaben an PLC (Seite 399)" und "W1: Werkzeugkorrektur (Seite 1475)".

Literatur:

- Programmieranleitung Grundlagen
- Funktionshandbuch Erweiterungsfunktionen; Kinematische Transformation (M1)

- Funktionshandbuch Sonderfunktionen; Achskopplungen und ESR (M3); Kapitel: Mitschleppen, Kapitel: Leitwertkopplung
- Funktionshandbuch Sonderfunktionen; Tangentialsteuerung (T3)

11.6.3 Besondere Reaktionen

Überspeichern

Überspeichern im `RESET`-Zustand von:

- Frames (Nullpunktverschiebungen)
- Aktiver Ebene
- Aktivierter Transformation
- Werkzeugkorrektur

wirkt sofort auf die Istwert-Anzeige aller Achsen im Kanal.

Eingabe über Bedientafelfront

Werden die Werte für:

"Aktiver Frame" (Nullpunktverschiebungen, Bedienbereich Parameter)

und

"Aktive Werkzeuglängenkorrektur" (Bedienbereich Parameter)

durch Bedienung an der Bedientafelfront geändert, so werden sie in der Anzeige durch eine der folgenden Maßnahmen wirksam:

- Betätigen der `RESET`-Taste
- Neuanwahl von:
 - Nullpunktverschiebung durch das Teileprogramm
 - Werkzeugkorrektur durch das Teileprogramm
- Neueinstellung von:
 - Nullpunktverschiebung durch Überspeichern
 - Werkzeugkorrektur durch Überspeichern
- Teileprogramm-Start

MD9440

Ist das HMI-Maschinendatum für die Bedientafelfront:

MD9440 `ACTIVATE_SEL_USER_DATA`

gesetzt, so werden die eingegebenen Werte im Reset-Zustand sofort aktiv.

Bei erfolgter Eingabe in gestoppter Teileprogrammbearbeitung werden die Werte mit dem Fortsetz-Start aktiv.

Istwertlesen

Wird aus \$AA_IW nach dem Aktivieren eines Frames (Nullpunktverschiebung) oder einer Werkzeugkorrektur der Istwert im WKS gelesen, so sind die aktivierten Änderungen im gelesenen Ergebnis bereits enthalten, auch wenn die Achsen noch nicht mit den aktivierten Änderungen bewegt wurden.

Die Istwerte im Einstellbaren-Nullpunktsystem ENS können für jede Achse mit der Variablen \$AA_IEN[Achse] aus dem Teileprogramm gelesen werden.

Mit \$AA_IBN[Achse] können die Istwerte im Basisnullpunkt-Koordinatensystem BNS aus dem Teileprogramm gelesen werden.

Istwertanzeige

Im WKS wird immer die programmierte Kontur angezeigt.

Auf das MKS werden folgende Verschiebungen aufgerechnet:

- Kinematische Transformation
- DRF-Verschiebung / Externe Nullpunktverschiebung
- Aktiver Frame
- Aktive Werkzeugkorrektur des aktuellen Werkzeugs

Umschalten durch PLC

Über die PLC können wahlweise die Istwerte im WKS, ENS, BNS oder MKS angezeigt werden. Von PLC-Seite kann vorgegeben werden, welches Koordinatensystem bei einer Maschine dem Werkstückkoordinatensystem entspricht.

Nach Hochlauf ist standardmäßig das MKS angewählt.

Durch das Signal DB19 DBB0.7 "MKS/WKS-Umschaltung" kann auch von der PLC aus auf das WKS umgeschaltet werden.

Übergaben an PLC

In Abhängigkeit vom Maschinendatum:

MD20110 / MD20112, Bit 1

werden bei der Anwahl der Werkzeuglängenkorrektur die Hilfsfunktionen (D, T, M) an die PLC ausgegeben (oder nicht).

Hinweis

Ist von der PLC das WKS angewählt, kann per Bedienung für die jeweilige Betriebsart dennoch zwischen WKS und MKS umgeschaltet werden.

Bei Betriebsarten- oder/und Bereichswechsel wird jedoch das von der PLC angewählte WKS ausgewertet und aktiviert (siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)").

11.7 Randbedingungen

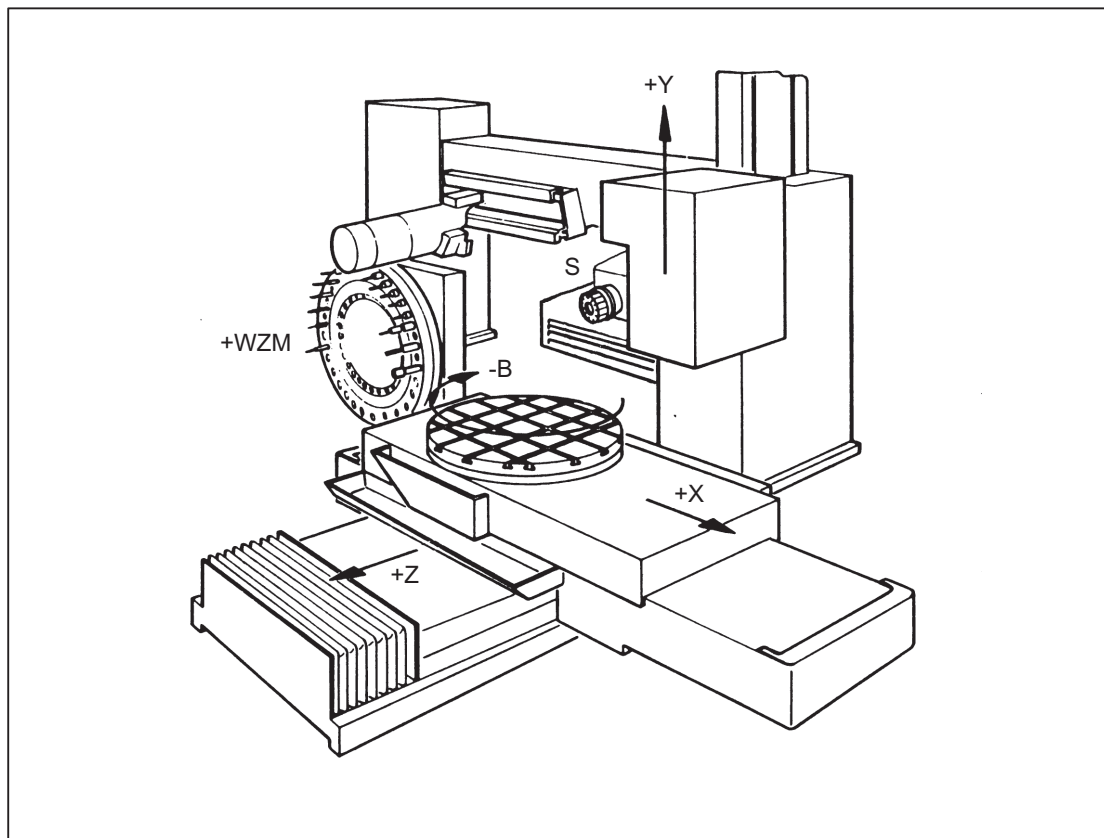
Es sind keine Randbedingungen zu beachten.

11.8 Beispiele

11.8.1 Achsen

Achskonfiguration für eine 3-Achs-Fräsmaschine mit Rundtisch

1. Maschinenachse: X1	Linearachse
2. Maschinenachse: Y1	Linearachse
3. Maschinenachse: Z1	Linearachse
4. Maschinenachse: B1	Rundtisch (zum Drehen für Mehrseitenbearbeitung)
5. Maschinenachse: W1	Rundachse für Werkzeugmagazin (WZ-Teller)
6. Maschinenachse: C1	(Spindel)
1. Geometrieachse: X	(1. Kanal)
2. Geometrieachse: Y	(1. Kanal)
3. Geometrieachse: Z	(1. Kanal)
1. Zusatzachse: B	(1. Kanal)
2. Zusatzachse: WZM	(1. Kanal)
1. Spindel: S1/C	(1. Kanal)



Parametrierung der Maschinendaten

Maschinendatum	Wert
MD10000 AXCONF_MACHAX_NAME_TAB[0]	= X1
MD10000 AXCONF_MACHAX_NAME_TAB[1]	= Y1
MD10000 AXCONF_MACHAX_NAME_TAB[2]	= Z1
MD10000 AXCONF_MACHAX_NAME_TAB[3]	= B1
MD10000 AXCONF_MACHAX_NAME_TAB[4]	= W1
MD10000 AXCONF_MACHAX_NAME_TAB[5]	= C1
MD20050 AXCONF_GEOAX_ASSIGN_TAB[0]	= 1
MD20050 AXCONF_GEOAX_ASSIGN_TAB[1]	= 2
MD20050 AXCONF_GEOAX_ASSIGN_TAB[2]	= 3
MD20060 AXCONF_GEOAX_NAME_TAB[0]	= X
MD20060 AXCONF_GEOAX_NAME_TAB[1]	= Y
MD20060 AXCONF_GEOAX_NAME_TAB[2]	= Z
MD20070 AXCONF_MACHAX_USED[0]	= 1
MD20070 AXCONF_MACHAX_USED[1]	= 2

Maschinendatum	Wert
MD20070 AXCONF_MACHAX_USED[2]	= 3
MD20070 AXCONF_MACHAX_USED[3]	= 4
MD20070 AXCONF_MACHAX_USED[4]	= 5
MD20070 AXCONF_MACHAX_USED[5]	= 6
MD20080 AXCONF_CHANAX_NAME_TAB[0]	= X
MD20080 AXCONF_CHANAX_NAME_TAB[1]	= Y
MD20080 AXCONF_CHANAX_NAME_TAB[2]	= Z
MD20080 AXCONF_CHANAX_NAME_TAB[3]	= B
MD20080 AXCONF_CHANAX_NAME_TAB[4]	= WZM
MD20080 AXCONF_CHANAX_NAME_TAB[5]	= S1
MD30300 IS_ROT_AX[3]	= 1
MD30300 IS_ROT_AX[4]	= 1
MD30300 IS_ROT_AX[5]	= 1
MD30310 ROT_IS_MODULO[3]	= 1
MD30310 ROT_IS_MODULO[4]	= 1
MD30310 ROT_IS_MODULO[5]	= 1
MD30320 DISPLAY_IS_MODULO[3]	= 1
MD30320 DISPLAY_IS_MODULO[4]	= 1
MD20090 SPIND_DEF_MASTER_SPIND	= 1
MD35000 SPIND_ASSIGN_TO_MACHAX[AX1]	= 0
MD35000 SPIND_ASSIGN_TO_MACHAX[AX2]	= 0
MD35000 SPIND_ASSIGN_TO_MACHAX[AX3]	= 0
MD35000 SPIND_ASSIGN_TO_MACHAX[AX4]	= 0
MD35000 SPIND_ASSIGN_TO_MACHAX[AX5]	= 0
MD35000 SPIND_ASSIGN_TO_MACHAX[AX6]	= 1

11.8.2 Koordinatensysteme

Projektierung eines globalen Basisframes

Vorausgesetzt ist eine NC mit 2 Kanälen. Dabei gilt:

- Beide Kanäle können den globalen Basisframe schreiben.
- Der jeweils andere Kanal erkennt die Änderung nach erneuter Aktivierung des globalen Basisframes.

11.8 Beispiele

- Beide Kanäle können den globalen Basisframe lesen.
- Beide Kanäle können für sich den globalen Basisframe aktivieren

Maschinendaten

Maschinendatum	Wert
MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[0]	= X1
MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[1]	= X2
MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[2]	= X3
MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[3]	= X4
MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[4]	= X5
MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[5]	= X6
MD18602 \$MN_MM_NUM_GLOBAL_BASE_FRAMES	= 1
MD28081 \$MC_MM_NUM_BASE_FRAMES	= 1

Maschinendaten für Kanal 1	Wert	Maschinendaten für Kanal 1	Wert
\$MC_AXCONF_CHANAX_NAME_TAB[0]	= X	\$MC_AXCONF_CHANAX_NAME_TAB[0]	= X
\$MC_AXCONF_CHANAX_NAME_TAB[1]	= Y	\$MC_AXCONF_CHANAX_NAME_TAB[1]	= Y
\$MC_AXCONF_CHANAX_NAME_TAB[2]	= Z	\$MC_AXCONF_CHANAX_NAME_TAB[2]	= Z
\$MC_AXCONF_MACHAX_USED[0]	= 1	\$MC_AXCONF_MACHAX_USED[0]	= 4
\$MC_AXCONF_MACHAX_USED[1]	= 2	\$MC_AXCONF_MACHAX_USED[1]	= 5
\$MC_AXCONF_MACHAX_USED[2]	= 3	\$MC_AXCONF_MACHAX_USED[2]	= 6
\$MC_AXCONF_GEOAX_NAME_TAB[0]	= X	\$MC_AXCONF_GEOAX_NAME_TAB[0]	= X
\$MC_AXCONF_GEOAX_NAME_TAB[1]	= Y	\$MC_AXCONF_GEOAX_NAME_TAB[1]	= Y
\$MC_AXCONF_GEOAX_NAME_TAB[2]	= Z	\$MC_AXCONF_GEOAX_NAME_TAB[2]	= Z
\$MC_AXCONF_GEOAX_ASSIGN_TAB[0]	= 1	\$MC_AXCONF_GEOAX_ASSIGN_TAB[0]	= 4
\$MC_AXCONF_GEOAX_ASSIGN_TAB[1]	= 2	\$MC_AXCONF_GEOAX_ASSIGN_TAB[1]	= 5
\$MC_AXCONF_GEOAX_ASSIGN_TAB[2]	= 3	\$MC_AXCONF_GEOAX_ASSIGN_TAB[2]	= 6

Teileprogramm im 1. Kanal

Code (Ausschnitt)	; Kommentar
...	
N100 \$P_NCBFR[0] = CTRANS(x, 10)	; Aktivierung des NC-globalen Basisframes
...	
N130 \$P_NCBFRAME[0] = CROT(X, 45)	; Aktivierung des NC-globalen Basisframes mit Drehung =>
	; Alarm 18310, da Drehungen bei NC-globalen Frames
	; nicht erlaubt sind
...	

Teileprogramm im 2. Kanal

Code (Ausschnitt)	; Kommentar
...	
N100 \$P_NCBFR[0] = CTRANS(x, 10)	; Der NC-globale Basisframe wirkt auch im 2.Kanal
...	
N510 G500 X10	; Basisframe aktivieren
N520 \$P_CHBFRAME[0] = CTRANS(x, 10)	; Aktueller Frame des 2. Kanals wird mit einer Verschiebung aktiviert
...	

11.8.3 Frames

Beispiel 1

Die Kanalachse soll durch einen Geometrieachsen-Tausch zur Geometrieachse werden.

Durch den Tausch soll der programmierbare Frame einen Translationsanteil von 10 in der x-Achse haben.

Der aktuelle einstellbare Frame soll erhalten bleiben:

FRAME_GEOX_CHANGE_MODE = 1

\$P_UIFR[1] = CROT(x,10,y,20,z,30)	; Frame bleibt nach Geometrieachsen- Tausch erhalten
G54	; einstellbarer Frame wird aktiv
TRANS a10	; achsiale Verschiebung von a wird getauscht
GEOAX(1,a)	; a wird zur x-Achse
	; \$P_ACTFRAME= CROT(x,10,y,20,z,30):CTTRANS(x10)

Bei Transformationswechsel können gleichzeitig mehrere Kanalachsen zu Geometrieachsen werden.

Beispiel 2

Durch eine 5-Achs-Orientierungs-Transformation werden die Kanalachsen 4, 5 und 6 zu Geometrieachsen der Transformation. Die Geometrieachsen vor der Transformation werden also alle ersetzt.

Beim Einschalten der Transformation ändern sich alle aktuellen Frames.

Zur Berechnung des neuen WKS werden die achsialen Frame-Anteile der Kanalachsen, die zu Geometrieachsen werden, berücksichtigt. Programmierte Drehungen vor der Transformation werden beibehalten. Nach dem Ausschalten der Transformation wird das alte WKS wieder hergestellt.

Der häufigste Anwendungsfall wird wohl der sein, dass sich die Geometrieachsen vor und nach der Transformation nicht ändern und die Frames so beibehalten werden sollen, wie sie vor der Transformation waren.

Maschinendaten:

```
$MN_FRAME_GEOAX_CHANGE_MODE = 1
```

```
$MC_AXCONF_CHANAX_NAME_TAB[0] = "CAX"
$MC_AXCONF_CHANAX_NAME_TAB[1] = "CAY"
$MC_AXCONF_CHANAX_NAME_TAB[2] = "CAZ"
$MC_AXCONF_CHANAX_NAME_TAB[3] = "A"
$MC_AXCONF_CHANAX_NAME_TAB[4] = "B"
$MC_AXCONF_CHANAX_NAME_TAB[5] = "C"
```

```
$MC_AXCONF_GEOAX_ASSIGN_TAB[0] = 1
$MC_AXCONF_GEOAX_ASSIGN_TAB[1] = 2
$MC_AXCONF_GEOAX_ASSIGN_TAB[2] = 3
```

```
$MC_AXCONF_GEOAX_NAME_TAB[0] = "X"
$MC_AXCONF_GEOAX_NAME_TAB[1] = "Y"
$MC_AXCONF_GEOAX_NAME_TAB[2] = "Z"
```

```
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[0] = 4
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[1] = 5
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[2] = 6
```

```
$MC_TRAFO_AXES_IN_1[0] = 4
$MC_TRAFO_AXES_IN_1[1] = 5
$MC_TRAFO_AXES_IN_1[2] = 6
$MC_TRAFO_AXES_IN_1[3] = 1
$MC_TRAFO_AXES_IN_1[4] = 2
```

Programm:

```
$P_NCBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
$P_CHBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
$P_IFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(z,45)
$P_PFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(x,10,y,20,z,30)

TRAORI                                ; Trafo setzt GeoAx(4,5,6)
                                      ; $P_NCBFRAME[0] = ctrans(x,4,y,5,z,6,cax,1,cay,2,caz,
                                      ; 3)
                                      ; $P_ACTBFRAME = ctrans(x,8,y,10,z,12,cax,2,cay,4,caz,
                                      ; 6)
                                      ; $P_PFRAME = ctrans(x,4,y,5,z,6,cax,1,cay,2,caz,3):
                                      ; crot(x,10,y,20,z,30)
                                      ; $P_IFRAME = ctrans(x,4,y,5,z,6,cax,1,cay,2,caz,
                                      ; 3):crot(z,45)
```

```

TRAFOOF;                                ; Ausschalten der Transformation setzt GeoAx(1,2,3)
; $P_NCBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
; $P_CHBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
; $P_IFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(z,
45)
; $P_PFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(x,
10,y,20,z,30)

```

11.9 Datenlisten

11.9.1 Maschinendaten

11.9.1.1 Anzeige-Maschinendaten

Nummer	Bezeichner: \$MM_	Beschreibung
SINUMERIK Operate		
9242	MA_STAT_DISPLAY_BASE	Zahlenbasis für die Anzeige der Gelenkstellung STAT
9243	MA_TU_DISPLAY_BASE	Zahlenbasis für die Anzeige der Rundachsstellung TU
9244	MA_ORIAXES_EULER_ANGLE_NAME	Anzeige der Orientierungsachsen als Euler-Winkel
9245	MA_PRESET_FRAMEIDX	Wertablage Ankratzen und Istwertsetzen
9247	USER_CLASS_BASE_ZERO_OFF_PA	Verfügbarkeit der Basisverschiebung in Bedienbereich "Parameter"
9248	USER_CLASS_BASE_ZERO_OFF_MA	Verfügbarkeit der Basisverschiebung in Bedienbereich Maschine
9424	MA_COORDINATE_SYSTEM	Koordinatensystem für Istwertanzeige
9440	ACTIVE_SEL_USER_DATA	Aktive Daten (Frames) werden nach Änderung sofort wirksam
9449	WRITE_TOA_LIMIT_MASK	Wirksamkeit von MD9203 bezüglich Schneidendaten und ortsabhängige Korrekturen
9450	MM_WRITE_TOA_FINE_LIMIT	Grenzwert für Verschleiß fein
9451	MM_WRITE_ZOA_FINE_LIMIT	Grenzwert für Verschiebung fein

11.9.1.2 NC-spezifischen Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10000	AXCONF_MACHAX_NAME_TAB	Maschinenachsname
10600	FRAME_ANGLE_INPUT_MODE	Drehreihenfolge in Frame
10602	FRAME_GEOAX_CHANGE_MODE	Frames beim Umschalten von Geometrieachsen
10610	MIRROR_REF_AX	Bezugsachse für das Spiegeln
10612	MIRROR_TOGGLE	Mirror umschalten
10613	NCBFRAME_RESET_MASK	Aktive NCU-globalen Basisframes nach Reset
10615	NCBFRAME_POWERON_MASK	Globalen Basisframes nach Power On zurücksetzen
10617	FRAME_SAVE_MASK	Verhalten von Frames bei SAVE- Unterprogrammen
10650	IPO_PARAM_NAME_TAB	Name der Interpolationsparameter
10660	INTERMEDIATE_POINT_NAME_TAB	Name der Zwischenpunktkoordinaten bei G2/G3
11640	ENABLE_CHAN_AX_GAP	Kanalachslücken werden erlaubt
18600	MM_FRAME_FINE_TRANS	Feinverschiebung bei FRAME (SRAM)
18601	MM_NUM_GLOBAL_USER_FRAMES	Anzahl der globalen vordefinierten Anwender-Frames (SRAM)
18602	MM_NUM_GLOBAL_BASE_FRAMES	Anzahl der globalen Basisframes (SRAM)

11.9.1.3 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20050	AXCONF_GEOAX_ASSIGN_TAB	Zuordnung Geometrieachse zu Kanalachse
20060	AXCONF_GEOAX_NAME_TAB	Geometrieachsname im Kanal
20070	AXCONF_MACHAX_USED	Maschinenachsnummer gültig im Kanal
20080	AXCONF_CHANAX_NAME_TAB	Kanalachsname im Kanal
20110	RESET_MODE_MASK	Festlegung der Steuerungs-Grundstellung nach RESET / TP-Ende
20118	GEOAX_CHANGE_RESET	Automatischen Geometrieachswechsel erlauben
20126	TOOL_CARRIER_RESET_VALUE	Wirksamer Werkzeugträger bei RESET
20140	TRAFO_RESET_VALUE	Transformationsdatensatz im Hochlauf (RESET / TP-Ende)
20150	GCODE_RESET_VALUES	Löschstellung der G-Gruppen
20152	GCODE_RESET_MODE	RESET-Verhalten der G-Gruppen
20184	TOCARR_BASE_FRAME_NUMBER	Nummer des Basisframes für Aufnahme des Tischoffsets
21015	INVOLUTE_RADIUS_DELTA	Endpunktüberwachung bei Evolvente
22532	GEOAX_CHANGE_M_CODE	M-Code bei Umschaltung der Geometrieachsen
22534	TRAFO_CHANGE_M_CODE	M-Code bei Transformationswechsel
24000	FRAME_ADD_COMPONENTS	Framekomponenten für G58 und G59
24002	CHBFRAME_RESET_MASK	RESET-Verhalten der kanalspezifischen Basisframes
24004	CHBFRAME_POWERON_MASK	Kanalspezifische Basisframes nach Power On zurücksetzen

Nummer	Bezeichner: \$MC_	Beschreibung
24006	CHSFRAME_RESET_MASK	Aktive Systemframes nach Reset
24007	CHSFRAME_RESET_CLEAR_MASK	Löschen von Systemframes bei RESET
24008	CHSFRAME_POWERON_MASK	Systemframes nach POWER ON zurücksetzen
24010	PFRAME_RESET_MODE	RESET-Mode für programmierbaren Frame
24020	FRAME_SUPPRESS_MODE	Positionen bei Frameunterdrückung
24030	FRAME_ACT_SET	Einstellung des ENS-Koordinatens
24040	FRAME_ADAPT_MODE	Anpassungen von aktiven Frames
24050	FRAME_SAA_MODE	Speichern und Aktivieren von Datenhaltungsframes
24805	TRACYL_ROT_AX_FRAME_1	Rundachs-Verschiebung TRACYL 1
24855	TRACYL_ROT_AX_FRAME_2	Rundachs-Verschiebung TRACYL 2
24905	TRANSMIT_ROT_AX_FRAME_1	Rundachsverschiebung TRANSMIT 1
24955	TRANSMIT_ROT_AX_FRAME_2	Rundachsverschiebung TRANSMIT 2
28080	MM_NUM_USER_FRAMES	Anzahl der einstellbaren Frames (SRAM)
28081	MM_NUM_BASE_FRAMES	Anzahl Basisframes
28082	MM_SYSTEM_FRAME_FRAMES	Systemframes (SRAM)
28560	MM_SEARCH_RUN_RESTORE_MODE	Restore von Daten nach einer Simulation

11.9.1.4 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
32074	FRAME_OR_CORRPOS_NOTALLOWED	FRAME oder HL-Korrektur ist unzulässig
35000	SPIND_ASSIGN_TO_MACHAX	Zuordnung Spindel zu Maschinenachse

11.9.2 Settingdaten

11.9.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42440	FRAME_OFFSET_INCR_PROG	Nullpunktverschiebungen in Frames
42980	TOFRAME_MODE	Framedefinition bei TOFRAME, TOROT und PAROT

11.9.3 Systemvariablen

Bezeichner	Beschreibung
\$AA_ETRANS[Achse]	Externe Nullpunktverschiebung
\$AA_IBN [Achse]	Istwert im Basisnullpunkt-Koordinatensystem (BNS)
\$AA_IEN [Achse]	Aktueller Istwert im einstellbaren Nullpunkt-Koordinatensystem (ENS)
\$AA_OFF[Achse]	Überlagerte Bewegung für die programmierte Achse
\$AC_DRF[Achse]	Handradüberlagerung einer Achse
\$AC_JOG_COORD	Koordinatensystem für das manuelle Verfahren
\$P_ACSFRAME	Aktives Frame zwischen BKS und ENS
\$P_ACTBFRAME	Aktives Gesamt-Basisframe
\$P_ACTFRAME	Aktives Gesamtframe
\$P_BFRAME	1. aktives Basisframe. Entspricht \$P_CHBFRAME[0]
\$P_CHBFR[<n>]	Datenhaltungsframe: Basisframe über G500, G54...G599 aktivierbar
\$P_CHBFRAME[<n>]	Aktives Basisframe
\$P_CHBFRMASK	Basisframemaske im Kanal
\$P_CHSFRMASK	Systemframe-Maske
\$P_CYCFR	Datenhaltungsframe: Systemframe für Zyklen
\$P_CYCFRAME	Aktives Systemframe für Zyklen
\$P_EXTFR	Datenhaltungsframe: Systemframe für Externe Nullpunktverschiebung
\$P_EXTFRAME	Aktives Systemframe für Externe Nullpunktverschiebung
\$P_IFRAME	Aktives einstellbares Frame
\$P_ISO1FR	Datenhaltungsframe: Systemframe für ISO G51 . 1 Spiegeln
\$P_ISO2FR	Datenhaltungsframe: Systemframe für ISO G68 2DROT
\$P_ISO3FR	Datenhaltungsframe: Systemframe für ISO G68 3DROT
\$P_ISO4FR	Datenhaltungsframe: Systemframe für ISO G51 Scale
\$P_ISO1FRAME	Aktives Systemframe für ISO G51 . 1 Spiegeln
\$P_ISO2FRAME	Aktives Systemframe für ISO G68 2DROT
\$P_ISO3FRAME	Aktives Systemframe für ISO G68 3DROT
\$P_ISO4FRAME	Aktives Systemframe für ISO G51 Scale
\$P_NCBFR[n]	Datenhaltungsframe: NCU-globales Basisframe in der Datenhaltung, über G500, G54...G599 aktivierbar
\$P_NCBFRAME[n]	Aktives NCU-globales Basisframe
\$P_NCBFRMASK	Globale Basisframe-Maske
\$P_PARTFR	Datenhaltungsframe: Systemframe für TCARR und PAROT
\$P_PARTFRAME	Aktives Systemframe für TCARR und PAROT bei orientierbarem Werkzeugträger
\$P_PFRAME	Aktiver Programmierbarer Frame
\$P_SETFR	Datenhaltungsframe: Systemframe für Istwertsetzen
\$P_SETFRAME	Aktives Systemframe für Istwertsetzen
\$P_TOOLFR	Datenhaltungsframe: Systemframe für TOROT und TOFRAME
\$P_TOOLFRAME	Aktives Systemframe für TOROT und TOFRAME

Bezeichner	Beschreibung
\$P_TRAFR	Datenhaltungsframe: Systemframe für Transformationen
\$P_TRAFRAME	Aktives Systemframe für Transformationen
\$P_UBFR	1. Basisframe im Kanal in der Datenhaltung, das nach G500, G54...G599 aktiviert wird. Entspricht \$P_CHBFR[0].
\$P_UIFR[n]	Datenhaltungsframe: Einstellbarer Frame, über G500, G54...G599 aktivierbar
\$P_UIFRNUM	Nummer des aktiven Einstellbaren Frames
\$P_WPFR	Datenhaltungsframe: Systemframe für das Werkstück
\$P_WPFRAME	Aktives Systemframe für das Werkstück

11.9.4 Signale

11.9.4.1 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
T-Funktionsänderung	DB21,DBX61.0-.2	-
D-Funktionsänderung	DB21,DBX62.0-.2	-
T-Funktion 1	DB21,DBB118-119	DB250x.DBD2000
D-Funktion 2	DB21,DBB129	DB250x.DBD5000
Nummer aktive Fkt. G-Gruppe 1 8 (Bit -Int)	DB21,DBB208	DB350x.DBB0
Nummer aktive Fkt. G-Gruppe 2 8 (Bit -Int)	DB21,DBB209	DB350x.DBB1
...
Nummer aktive Fkt. G-Gruppe 29 8 (Bit -Int)	DB21,DBB236	DB350x.DBB28

11.9.4.2 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Externe Nullpunktverschiebung übernehmen	DB31,DBX3.0	-

11.9.4.3 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Spindel / Rundachse	DB31,DBX60.0	DB390x.DBX0.0

N2: Not-Halt

12.1 Kurzbeschreibung

Funktion

Die Steuerung unterstützt den Maschinenhersteller bei der Realisierung der Not-Halt-Funktion durch folgende Funktionen:

- An allen SINUMERIK-Maschinensteuertafeln ist ein Not-Halt-Taster für den Maschinenbediener leicht erreichbar angebracht. Der Funktionalität des Not-Halt-Tasters umfasst die Zwangsöffnung der elektrischen Schaltkontakte und eine mechanisch selbsttätige Verrastung/Verriegelung.
- Die Not-Halt-Anforderung an die NC erfolgt über die PLC über die NC/PLC-Nahtstelle.
- Bei Not-Halt muss ein Stillsetzen nach Stopp-Kategorie 0 oder 1 (EN 60204) erfolgen.
- Alle durch die PLC gesteuerten Maschinenfunktionen können bei Not-Halt einen, durch den Maschinenhersteller einstellbaren, sicheren Zustand einnehmen.
- Ein Entriegeln des Not-Halt-Tasters bewirkt kein Aufheben des Not-Halt-Zustandes bzw. löst keinen Wiederanlauf aus.
- Nach dem Aufheben des Not-Halt-Zustandes ist kein Referenzieren der Maschinenachsen bzw. Synchronisieren der Spindeln nötig. Die Istpositionen der Maschinenachsen werden während des Not-Halt-Ablaufs kontinuierlich nachgeführt.

12.2 Normen

Relevante Normen

Im Rahmen der Not-Halt-Funktion sind folgende Normen unbedingt zu beachten:

- EN ISO 12000-1
- EN ISO 12000-2
- EN 418
- EN 60204

Not-Halt

Entsprechend EN 418 ist Not-Halt eine Funktion, die:

- aufkommende oder bestehende Gefahren für Personen, Schäden an der Maschine oder dem Arbeitsgut abwenden oder vermindern soll.
- durch eine einzige Handlung einer Person ausgelöst wird, wenn die normale Haltefunktion dafür nicht angemessen ist.

Gefahren

Gefahren im Sinne der EN 418 sind solche, die herrühren können von:

- funktionalen Unregelmäßigkeiten (Fehlfunktionen der Maschine, nicht hinnehmbare Eigenschaften des bearbeiteten Materials, menschliche Fehler, ...).
- normalem Betrieb.

Norm EN ISO 12000-2

Gemäß einer grundlegenden Sicherheitsanforderung der EG-Richtlinie Maschinen hinsichtlich Not-Halt müssen Maschinen mit einer Not-Halt-Einrichtung versehen sein.

Ausnahmen

Keine Not-Halt-Einrichtung wird bei Maschinen benötigt:

- an denen eine Not-Halt-Einrichtung das Risiko nicht verringern würde, weil dadurch entweder die Stillsetzzeit nicht verringert würde oder weil die dafür zu ergreifenden Maßnahmen nicht geeignet wären, das Risiko zu beeinflussen.
- die von Hand getragen und geführt werden.

Hinweis

Der Maschinenhersteller wird auf die Einhaltung der nationalen und internationalen Normen ausdrücklich hingewiesen. Die SINUMERIK-Steuerungen unterstützen den Maschinenhersteller bei der Realisierung der Not-Halt-Funktion entsprechend den Festlegungen der folgenden Funktionsbeschreibung. Die Verantwortung für die Not-Halt-Funktion (Auslösung, Ablauf, Quittierung) liegt aber ausschließlich beim Maschinenhersteller selbst.

12.3 Not-Halt-Stellteile

Not-Halt-Stellteile

Nach EN 418 müssen Not-Halt-Stellteile so konstruiert sein, dass sie mechanisch selbsttätig verrasten und für die Bedienperson und andere im Notfall leicht zu betätigen sind.

Folgende Typen von Stellteilen können unter anderem eingesetzt werden:

- Pilztaster (drucktastenbetätigter Schalter)
- Drähte/Drahtseile, Leinen, Stangen
- Griffe
- In besonderen Fällen: Fußschalter ohne Schutzhaube

Not-Halt-Taster und Steuerung

Die Betätigung des Not-Halt-Tasters oder ein direkt daraus abgeleitetes Signal muss als PLC-Eingang zur Steuerung (PLC) geführt werden. Im PLC-Anwenderprogramm muss dieser PLC-Eingang weitergeleitet werden an die NC auf das Nahtstellensignal:

DB10 DBX56.1 (Not-Halt)

Das Rückstellen des Not-Halt-Tasters oder ein direkt daraus abgeleitetes Signal muss als PLC-Eingang zur Steuerung (PLC) geführt werden. Im PLC-Anwenderprogramm muss dieser PLC-Eingang weitergeleitet werden an die NC auf das Nahtstellensignal:

DB10 DBX56.2 (Not-Halt quittieren)

Anschlussbedingungen

Zum Anschluss des Not-Halt-Tasters siehe:

Literatur:

Gerätehandbuch Bedienkomponenten

12.4 Not-Halt-Ablauf

Nach Betätigung des Not-Halt-Stellteils muss die Not-Halt-Einrichtung in einer Weise arbeiten, dass die Gefahr automatisch auf die bestmögliche Weise abgewendet oder verringert wird.

"Auf bestmögliche Weise" bedeutet, dass die günstigste Verzögerungsrate gewählt und die richtige Stopp-Kategorie (definiert in EN 60204) entsprechend einer Risikoabschätzung festgelegt werden kann.

Not-Halt-Ablauf in der NC

Der nach EN 418 vorbestimmte Ablauf interner Funktionen zum Not-Halt-Zustand sieht in der Steuerung wie folgt aus:

1. Die Teileprogrammbearbeitung wird unterbrochen.
Alle Maschinenachsen werden abgebremst in der jeweiligen achsspezifisch parametrisierten Zeit:
MD36610 \$MA_AX_EMERGENCY_STOP_TIME (Zeit der Bremsrampe bei Fehlern)
Die maximale Bremsrampe, die dabei erzielt werden kann, wird definiert durch den maximalen Bremsstrom des jeweiligen Antriebs. Der maximale Bremsstrom wird erreicht durch Vorgabe von Sollwert = 0 (Schnellbremsung).
2. Zurücksetzung des Nahtstellensignals:
DB11 DBX6.3 (BAG-betriebsbereit)
3. Setzen des Nahtstellensignals:
DB10 DBX106.1 (Not-Halt aktiv)
4. Alarm 3000 "Not-Halt" wird angezeigt.
5. Nach Ablauf einer parametrierbaren Verzögerungszeit werden die Reglerfreigaben der Maschinenachsen zurückgesetzt.
Die Einstellung der Verzögerungszeit erfolgt im Maschinendatum:
MD36620 \$MA_SERVO_DISABLE_DELAY_TIME (Abschaltverzögerung
Reglerfreigabe)
Folgende Einstellregel ist zu beachten: MD36620 \geq MD36610
6. Alle Maschinenachsen werden steuerungsintern in den Nachführbetrieb geschaltet.
Die Maschinenachsen sind dabei nicht mehr in Lageregelung.

Not-Halt-Ablauf an der Maschine

Der Not-Halt-Ablauf an der Maschine wird ausschließlich vom Maschinenhersteller bestimmt.

Dabei ist in Verbindung mit dem Ablauf in der NC Folgendes zu beachten:

- Der Ablauf in der NC wird gestartet mit dem Nahtstellensignal:
DB10 DBX56.1 (Not-Halt)
Nachdem die Maschinenachsen im Stillstand sind, muss nach EN 418 die Energiezufuhr unterbrochen werden.

Hinweis

Das Unterbrechen der Energiezufuhr liegt in der Verantwortung des Maschinenherstellers.

- Die digitalen und analogen Ausgänge der PLC-Peripherie werden vom Not-Halt-Ablauf in der NC nicht beeinflusst.
Sollen einzelne Ausgänge bei Not-Halt einen bestimmten Zustand oder Spannungspegel einnehmen, ist dies vom Maschinenhersteller im PLC-Anwenderprogramm vorzunehmen.
- Die schnellen digitalen Ausgänge der NCK-Peripherie werden vom Not-Halt-Ablauf in der NC nicht beeinflusst.
Sollen einzelne Ausgänge bei Not-Halt einen bestimmten Zustand einnehmen, muss der Maschinenhersteller im PLC-Anwenderprogramm den gewünschten Zustand an die NC übergeben über die Nahtstellensignale:
DB10 DBB4-7

Hinweis

Soll bei Not-Halt der Ablauf in der NC nicht wie oben beschrieben ablaufen, darf bis zum Erreichen eines durch den Maschinenhersteller im PLC-Anwenderprogramm festgelegten Not-Halt-Zustandes das Nahtstellensignal DB10 DBX56.1 (Not-Halt) nicht gesetzt werden.

Solange das Nahtstellensignal nicht gesetzt ist und kein anderer Alarm ansteht, sind in der NC alle Nahtstellensignale wirksam. Dadurch kann jeder herstellerspezifische Not-Halt-Zustand (auch Achs-, Spindel- und Kanal-spezifisch) eingenommen werden.

12.5 Not-Halt-Quittierung

Nach EN 418 darf das Rückstellen des Not-Halt-Stellteils nur als Ergebnis einer von Hand ausgeführten Handlung am Not-Halt-Stellteil möglich sein.

Das Rückstellen des Not-Halt-Stellteils allein darf keinen Wiederanlauf-Befehl auslösen.

Der Wiederanlauf der Maschine darf nicht möglich sein, bis alle betätigten Not-Halt-Stellteile von Hand einzeln und bewusst rückgestellt worden sind.

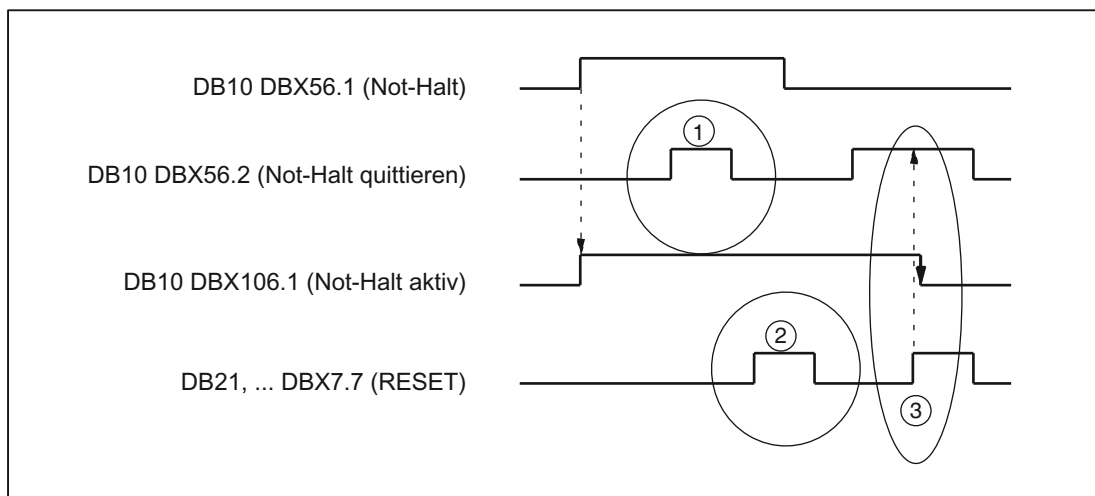
Not-Halt-Quittierung

Der Not-Halt-Zustand wird nur dann wieder zurückgesetzt, wenn zuerst das Nahtstellensignal DB10 DBX56.2 (Not-Halt quittieren) und anschließend das Nahtstellensignal DB11 DBX0.7 (BAG-Reset) gesetzt wird.

Dabei ist zu beachten, dass das Nahtstellensignal DB10 DBX56.2 (Not-Halt quittieren) und das Nahtstellensignal DB21, ... DBX7.7 (Reset) gemeinsam mindestens so lange gesetzt sind, bis das Nahtstellensignal DB10 DBX106.1 (Not-Halt aktiv) zurückgesetzt wurde.

Hinweis

Allein mit dem Nahtstellensignal DB21, ... DBX7.7 (Reset) kann der Not-Halt-Zustand nicht zurückgesetzt werden.



- (1) DB10 DBX56.2 (Not-Halt quittieren) ist wirkungslos
- (2) DB21, ... DBX7.7 (Reset) ist wirkungslos
- (3) DB10 DBX56.2 und DB21, ... DBX7.7 setzen DB10 DBX106.1 (Not-Halt aktiv) zurück

Bild 12-1 Not-Halt-Zustand rücksetzen

Auswirkungen

Durch Rücksetzen des Not-Halt-Zustandes werden:

- steuerungsintern für alle Maschinenachsen:
 - die Reglerfreigaben gesetzt.
 - der Nachführbetrieb aufgehoben.
 - die Lageregelung aktiviert.
- die folgenden Nahtstellensignale gesetzt:
 - DB31, ... DBX60.5 (Lageregelung aktiv)
 - DB11 DBX6.3 (BAG betriebsbereit)
- das folgende Nahtstellensignal zurückgesetzt:
 - DB10 DBX106.1 (Not-Halt aktiv)
- der Alarm 3000 "Not-Halt" gelöscht.
- die Teileprogrammbearbeitung in allen Kanälen der NC abgebrochen.

PLC- und NC-Peripherie

Die PLC- und NC-Peripherie müssen durch das PLC-Anwenderprogramm wieder in den Zustand zum Betrieb der Maschine versetzt werden.

POWER OFF/ON (Netz aus/ein)

Der Not-Halt-Zustand kann auch durch Aus- und Einschalten der Steuerung (POWER OFF/ON) zurückgesetzt werden.

Voraussetzung:

Beim Hochlauf der Steuerung darf das Nahtstellensignal DB10 DBX56.1 (Not-Halt) nicht gesetzt sein.

12.6 Datenlisten

12.6.1 Maschinendaten

12.6.1.1 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
36610	AX_EMERGENCY_STOP_TIME	Zeitdauer der Bremsrampe bei Fehlerzuständen
36620	SERVO_DISABLE_DELAY_TIME	Abschaltverzögerung Reglerfreigabe

12.6.2 Signale

12.6.2.1 Signale an NC

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Not-Halt	DB10.DBX56.1	DB2600.DBX0.1
Not-Halt quittieren	DB10.DBX56.2	DB2600.DBX0.2

12.6.2.2 Signale von NC

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Not-Halt aktiv	DB10.DBX106.1	DB2700.DBX0.1

12.6.2.3 Signale an BAG

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
BAG-Reset	DB11.DBX0.7	DB3000.DBX0.7

P1: Planachsen

13.1 Funktion

Planachse

Mit Planachse wird im Rahmen der Technologie "Drehen" die Maschinenachse bezeichnet, die senkrecht zur Symmetrieachse der Spindel bzw. zur Längsachse Z verfährt.

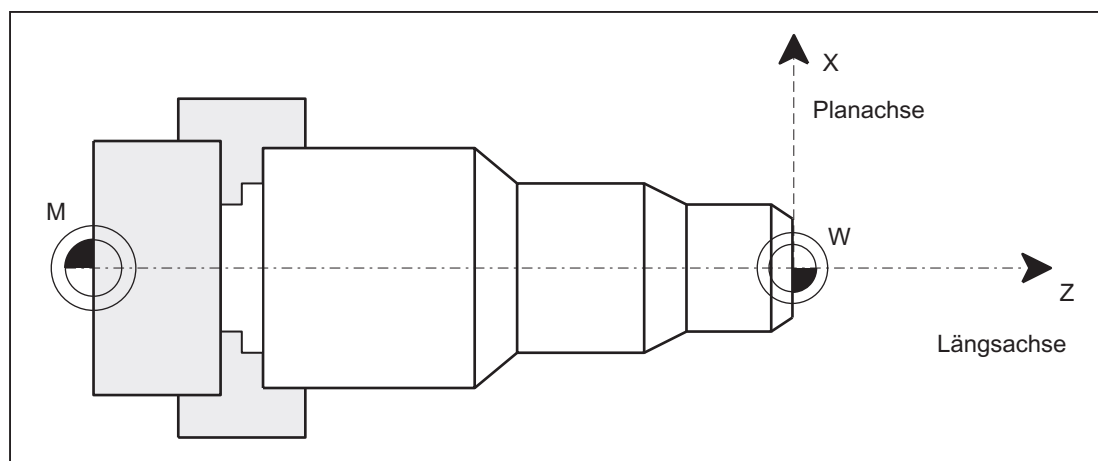


Bild 13-1 Lage der Planachse im Maschinenkoordinatensystem

Geometrieachse als Planachse

Jede Geometrieachse eines Kanals kann als Planachse definiert werden.

Für die Geometrieachse als Planachse sind folgende Funktionen gleichzeitig oder getrennt zugelassen und aktivierbar:

- Programmierung und Anzeige im Durchmesser
- Bezugsachse für konstante Schnittgeschwindigkeit G96/G961/G962

Mehrere Planachsen im Kanal

Die Einführung mehrerer Planachsen im Kanal bringt eine funktionelle Entkopplung von Durchmesserprogrammierung und Bezugsachse für G96/G961/G962 mit sich. Durchmesserprogrammierung und Bezugsachse für G96/G961/G962 können für unterschiedliche Planachsen aktiv sein (siehe folgende Tabelle).

	Programmierung und Anzeige im Durchmesser			Bezugsachse für G96 / G961 / G962	
Zulässiger Achsentyp:	Geometrieachse		Lineare Kanal- achsen	Geometrieachse	
Auswahl im Kanal:	eine	m aus 3	m aus n	eine	eine aus 3
Spezifische Wirkung:	Kanal	Achse		Kanal	
Maschinendaten:	MD20100	MD30460		MD20100	
Programmierung:	DIAM* kanalspezifisch modal G-Gruppe 29			SCC [<Achse>] kanalspezifisch modal Bezugsachse für G96/G961/G962	
Übernahme bei Achstausch:	DIAM*A [<Achse>] achsspezifisch modal				
Achsspezifische nicht moda- le Durchmesser-/ Radiuspro- grammierung	DAC, DIC; RAC, RIC achsspezifisch satzweise nur Programmierung				

DIAM*: DIAMOF, DIAMON, DIAM90, DIAMCYCOF

DIAM*A[<Achse>]: DIAMOFA[<Achse>], DIAMONA[<Achse>], DIAM90A[<Achse>], DIACYCOFA[<Achse>], DIAMCHANA[<Achse>]

<Achse>: Achsname für Geo-/Kanal- oder Maschinenachsname

Hinweis

Rundachsen sind nicht als Planachsen zugelassen.

Durchmesser-bezogene Daten

DIAMON/DIAMONA [<Achse>]

Nach dem Einschalten der Durchmesserprogrammierung mit DIAMON/DIAMONA [<Achse>] (siehe "Programmierung (Seite 877)") beziehen sich folgende Daten auf Durchmesserangaben:

- Anzeigedaten der Planachse im Werkstückkoordinatensystem:
 - Soll- und Istposition
 - Restweg
 - REPOS-Verschiebung
- Betriebsart "JOG":
 - Schrittmass (INC) durch inkrementelles manuelles Verfahren per Handrad (abhängig von wirksamen MD)

- Teileprogrammprogrammierung:
 - Endpositionen, unabhängig vom Bezugsmodus (G90 / G91)
 - Interpolationsparameter der Kreisprogrammierung (G2 / G3), falls diese mit Teileprogrammanweisung AC absolut programmiert sind.
- Mit Bezug auf das Werkstückkoordinatensystem (WKS) gelesene Istwerte:
 - \$AA_MW[<Planachse>
Systemvariable der Messfunktionen MEAS (Messen mit Restweglöschen) und MEAW (Messen ohne Restweglöschen)
 - \$P_EP[<Planachse>]
 - \$AA_IW[<Planachse>]

DIAM90/DIAM90A[<Achse>]

Nach dem Einschalten der Bezugsmodus-abhängigen Durchmesserprogrammierung mit DIAM90/DIAM90A[<Achse>] werden folgende Daten unabhängig vom Bezugsmodus (G90 / G91) immer Durchmesser-bezogen angezeigt:

- Istwert
- Mit Bezug auf das Werkstückkoordinatensystem (WKS) gelesene Istwerte:
 - \$AA_MW[<Planachse>
Systemvariable der Messfunktionen MEAS (Messen mit Restweglöschen) und MEAW (Messen ohne Restweglöschen)
 - \$P_EP[<Planachse>]
 - \$AA_IW[<Planachse>]

DIAMCYCOF/DIACYCOFA[<Achse>]

Bei DIAMCYCOF bzw. DIACYCOFA[<Achse>] erfolgt steuerungsintern eine Umschaltung auf Radiusprogrammierung. Es wird der vor DIAMCYCOF bzw. DIACYCOFA[AX] aktive Zustand bezüglich Durchmesserprogrammierung weiterhin zum HMI angezeigt.

Fest Radius-bezogene Daten

Für Planachsen werden folgende Daten **immer** Radius-bezogen eingegeben, programmiert und angezeigt:

- Teileprogrammprogrammierung
 - Interpolationsparameter der Kreisprogrammierung mit CIP
- Verschiebungen:
 - Werkzeugkorrekturen
 - programmierbare und einstellbare Frames
 - externe Nullpunktverschiebung
 - DRF- und Preset-Verschiebung
 - etc.

- Arbeitsfeldbegrenzung
- Software-Endschalter
- Vorschub
- Anzeigedaten bezogen auf das Maschinenkoordinatensystem
- Anzeigedaten der Servicebilder für Achse, VSA und HSA

Funktionserweiterungen für immer Radius-bezogene Daten:

Für PLC-Achsen, über FC18 oder ausschließlich vom PLC kontrollierte Achsen gilt:

- Die Maßangabe für PLC-Achsen im Radius gilt auch für mehrere Planachsen mit Durchmesserfunktion und ist unabhängig von kanalspezifischer oder achsspezifischer Durchmesserprogrammierung.
- In der Betriebsart JOG (Inc) unterliegt eine PLC-Achse dem Kanalzustand. Ist Durchmesserprogrammierung aktiv und MD20624 \$MC_HANDWH_CHAN_STOP_COND Bit 15 = 0, wird nur der halbe Weg des vorgegebenen Inkrements verfahren.

13.2 Parametrierung

Geometrieachse als Planachse definieren

Die Definition **einer** Geometrieachse als Planachse im Kanal erfolgt mit dem Maschinendatum:

MD20100 \$MC_DIAMETER_AX_DEF (Geometrieachse mit Planachsfunktion)

Beispiel:

MD20100 \$MC_DIAMETER_AX_DEF="X" ; Die Geometrieachse X ist Planachse im Kanal.

Für diese Achse ist sowohl Durchmesserprogrammierung als auch die Zuordnung einer konstanten Schnittgeschwindigkeit mit G96/G961/G962 zugelassen.

Mehrere Planachsen im Kanal

Die Definition weiterer Planachsen im Kanal, für die die Funktionalität der achsspezifischen Durchmesserprogrammierung zugelassen ist, erfolgt mit dem Maschinendatum:

MD30460 BASE_FUNCTION_MASK (Achsfunktionen)

Bit	Wert	Bedeutung
2	0	Achsspezifische Durchmesserprogrammierung ist nicht zugelassen.
	1	Achsspezifische Durchmesserprogrammierung ist zugelassen.

Hinweis

Die Einstellung MD30460 Bit 2 = 1 ist nur für Linearachsen möglich.

Eine Achse darf gleichzeitig im MD20100 und im MD30460 (Bit 2) definiert sein. Dabei hat das kanalspezifische MD20100 gegenüber dem achsspezifischen MD30460 eine höhere Priorität.

Mit MD20100 wird der Planachse im Hochlauf:

- die Funktion G96/G961/G962 zugeordnet.
- die kanalspezifische Durchmesserprogrammierung DIAMON, DIAMOF, DIAM90, DIAMCYCOF zugeordnet.

Diese Achse besitzt nach Hochlauf die achsspezifische Grundstellung DIAMCHANA[<Achse>].

Mit MD30460 Bit2 erfolgt die zusätzliche Freigabe der achsspezifischen Anweisungen DIAMONA[<Achse>], DIAMOFA[<Achse>], DIAM90A[<Achse>], DIACYCOFA[<Achse>], DIMCHANA[<Achse>].

Maßangabe für Werkzeugparameter

Per MD20360 \$MC_TOOL_PARAMETER_DEF_MASK sind für alle im Kanal definierten Planachsen die folgenden Werkzeugparameter als Durchmesserwerte aktivierbar:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK		
Bit	Wert	Bedeutung
1	1	Werkzeuglänge Planachse als Durchmesser
2	1	Alarm bei Verschleiß oder Werkzeuglänge als Durchmesser und Ebenenwechsel
3	1	Nullpunktverschiebung (NPV) in Frames in der Planachse als Durchmesser Der Frame legt die Nullpunktverschiebungen intern als Radiuswert ab. Es erfolgt keine Umrechnung bei Wechsel von Durchmesser– auf Radiusprogrammierung bzw. umgekehrt.
4	1	Preset-Wert als Durchmesser
5	1	Externe NPV in der Planachse als Durchmesser Es erfolgt keine Umrechnung bei Wechsel von Durchmesser– auf Radiusprogrammierung bzw. umgekehrt.
6	1	Istwerte der Planachse als Durchmesser
7	1	Anzeige Istwerte der Planachse als Durchmesserwert
10	1	Werkzeuganteil eines aktiven orientierbaren Werkzeugträgers wenn kein Werkzeug aktiv ist
11	1	Bewertung von \$TC_DP6 als Durchmesser
12	1	Bewertung von \$TC_DP15 als Verschleiß des Werkzeugdurchmessers
15	1	Inkrementalwerte der Planachse bei Zyklenmasken als Durchmesser

Radiusprogrammierung aus MD20100 \$MC_DIAMETER_AX_DEF und MD30460 \$MA_BASE_FUNCTION_MASK Bit 2 wird abhängig von MD20360 \$MC_TOOL_PARAMETER_DEF_MASK wie folgt berücksichtigt:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK		
Bit	Wert	Bedeutung
3	0	Nullpunktverschiebung \$P_EXTFRAME und Frames Für Planachsen werden Nullpunktverschiebungen in Frames immer als Radiuswerte eingerechnet.
5	0	Externe Nullpunktverschiebung (Achsenüberlagerung) Für Planachsen wird die externe Nullpunktverschiebung immer als Radiuswerte eingerechnet.
8	1	Anzeige Restweg im WKS immer als Radius
9		Für alle Planachsen wird mit MD11346 \$MN_HANDWH_TRUE_DISTANCE==1
	0	<ul style="list-style-type: none"> der halbe Weg des vorgegebenen Handradinkrements verfahren, falls für diese Achse kanalspezifische oder achsspezifische Durchmesserprogrammierung aktiv ist.
	1	<ul style="list-style-type: none"> immer der halbe Weg des vorgegebenen Handradinkrements verfahren.
13	1	Beim Joggen von Kreisen ist die Kreismittelpunktskoordinate immer ein Radiuswert, siehe SD42690 \$SC_JOG_CIRCLE_CENTRE
14	1	Bei Zyklenmasken sind Absolutwerte der Planachse im Radius.

Verhalten beim manuellen Verfahren per Handrad

Das Verhalten beim manuellen Verfahren per Handrad (MD11346 \$MN_HANDWH_TRUE_DISTANCE == 1 oder 3) einer Planachse bei aktiver Durchmesserprogrammierung kann mit dem folgenden Maschinendatum angepasst werden:

MD20624 \$MC_HANDWH_CHAN_STOP_COND

Bit	Wert	Bedeutung
15	0	Es wird nur der halbe Weg des vorgegebenen Inkrements verfahren.
	1	Es wird das vorgegebene Inkrement vollständig verfahren.

Positionswerte im Durchmesser anzeigen

Positionswerte von Planachsen werden immer als Durchmesserwert angezeigt, wenn: MD27100 \$MC_ABSBLOCK_FUNCTION_MASK Bit 0 = 1

Kanalspezifische Grundstellung nach Hochlauf, Reset

Die kanalspezifische Grundstellung nach Hochlauf bzw. Reset oder Teileprogrammende der G-Gruppe 29 (DIAMON, DIAM90, DIAMOF, DIAMCYCOF) wird festgelegt durch das Maschinendatum:

- MD20150 \$MC_GCODE_RESET_VALUE

bzw. (bei MD20110 \$MC_RESET_MODE_MASK Bit 0 = 1):

- MD20152 \$MC_GCODE_RESET_MODE

Der Anwender kann über einen ereignisgesteuerten Programmaufruf (Prog-Event) den jeweils gewünschten Zustand einstellen.

Ist G96/G961/G962 Grundstellung nach Hochlauf, muss mit MD20100 \$MC_DIAMETER_AX_DEF eine Planachse definiert sein, anderenfalls erfolgt die Alarmmeldung 10870.

Bezugsachse für G96/G961/G962 beibehalten:

MD20110 \$MC_RESET_MODE_MASK, Bit 18=1 bei Reset oder Teileprogrammende

MD20112 \$MC_START_MODE_MASK, Bit 18=1 bei Teileprogrammstart

Eine Bezugsachse für G96/G961/G962 kann auch ohne Applikation einer Planachse in MD20100 über `SCC[AX]` zugeordnet werden. Für diesen Fall darf die Konstante Schnittgeschwindigkeit nicht mit G96 eingeschaltet sein. Weitere Informationen siehe:

Literatur:

Programmierhandbuch Grundlagen, Vorschubregelung und Spindelbewegung

"Konstante Schnittgeschwindigkeit (G96/G961/G962, G97/G971/G972, G973, LIMS, SCC)

13.3 Programmierung

Planachsen können sowohl Durchmesser- als auch Radius-bezogen programmiert werden. In der Regel werden sie Durchmesser-bezogen, d. h. mit doppeltem Wegmaß programmiert, um aus technischen Zeichnungen die entsprechenden Maßangaben direkt in das Teileprogramm übernehmen zu können.

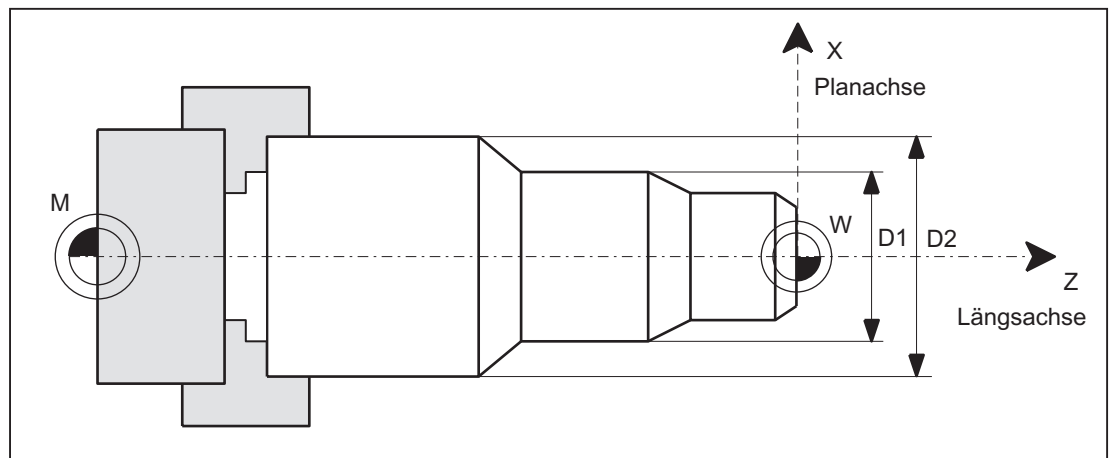


Bild 13-2 Planachse mit Durchmesserangabe (D1, D2)

Ein/Ausschalten der Durchmesserprogrammierung

Kanalspezifische Durchmesserprogrammierung

Das Ein bzw. Ausschalten der Durchmesserprogrammierung erfolgt über die modal wirksamen Teileprogrammanweisungen der G-Gruppe 29:

- `DIAMON`: Durchmesserprogrammierung EIN
- `DIAMOF`: Durchmesserprogrammierung AUS bzw. Radiusprogrammierung EIN

- DIAM90: Durchmesser- bzw. Radiusprogrammierung abhängig vom Bezugsmodus:
 - Durchmesserprogrammierung EIN im Zusammenhang mit Absolutmaßangabe G90
 - Radiusprogrammierung EIN im Zusammenhang mit Kettenmaßangabe G91
- DIAMCYCOF: Radiusprogrammierung für G90 und G91 EIN, zum HMI bleibt der zuletzt aktive G-Befehl dieser Gruppe weiterhin aktiv

Es ist ausschließlich auf die Planachse des Kanals bezogen.

Achsspezifische Durchmesserprogrammierung für mehrere Planachsen im Kanal

Hinweis

Die zusätzlich angegebene Achse muss über das MD30460 \$MA_BASE_FUNCTION_MASK mit Bit 2 = 1 aktiviert werden.

Die angegebene Achse muss eine im Kanal bekannte Achse sein. Zugelassen sind Geometrie-, Kanal- oder Maschinenachsen.

Eine Programmierung in Synchronaktionen ist nicht zugelassen.

Folgende achsspezifische modale Anweisungen können mehrmals in einem Teileprogrammsatz programmiert werden:

- DIAMONA [<Achse>]: Durchmesserprogrammierung für G90, G91, AC und IC EIN
- DIAMOFA [<Achse>]: Durchmesserprogrammierung AUS bzw. Radiusprogrammierung EIN
- DIAM90A [<Achse>]: Durchmesser- bzw. Radiusprogrammierung abhängig vom Bezugsmodus:
 - Durchmesserprogrammierung EIN im Zusammenhang mit Absolutmaßangabe G90 und AC
 - Radiusprogrammierung EIN im Zusammenhang mit Kettenmaßangabe G91 und IC
- DIACYCOFA [Achse]: Radiusprogrammierung für G90 und G91 EIN, zum HMI bleibt der zuletzt aktive G-Befehl dieser Gruppe weiterhin aktiv
- DIAMCHANA [Achse]: Übernahme Kanalzustand Durchmesserprogrammierung
- DIAMCHAN: alle Achsen mit MD30460 Bit 2 = 1 übernehmen den Kanalzustand Durchmesserprogrammierung

Achsspezifische modale Anweisungen sind dominant gegenüber der Kanaleinstellung.

13.4 Randbedingungen

Achstausch

Aufgrund einer GET-Anforderung aus dem Teileprogramm wird beim Achstausch mit `RELEASE [<Achse>]` der Zustand Durchmesserprogrammierung für eine zusätzliche Planachse im neuen Kanal übernommen.

Achstausch in Synchronaktionen

Bei Achstausch in Synchronaktionen nimmt eine Planachse den Zustand der achsspezifischen Durchmesserprogrammierung in den neuen Kanal mit, falls:

- für die Planachse mit MD30460 Bit 2 = 1 achsspezifische Durchmesserprogrammierung zugelassen ist.
- die Planachse nicht der kanalspezifischen Durchmesserprogrammierung im abgebenden Kanal unterstellt ist.

Über die Systemvariable \$AA_DIAM_STAT[<Achse>] kann die aktive Maßangabe abgefragt werden.

Achstausch über Achscontainer-Drehung

Durch Achscontainer-Drehung kann sich die Zuordnung einer Kanalachse zu einer Maschinenachse ändern. Der aktuelle Zustand der Durchmesserprogrammierung bleibt aber für die Kanalachse nach der Drehung erhalten. Dies betrifft auch den aktuellen Kanal- und Achszustand, da zum Zeitpunkt von "Maschinendaten wirksam setzen" der Zustand aus dem MD30460 \$MA_BASE_FUNCTION_MASK für alle Achsen des Achscontainers gleich ist.

13.5 Beispiele

Beispiel 1

X ist eine über MD20100 \$MC_DIAMETER_AX_DEF definierte Planachse.

Y ist eine Geometrieachse und U eine Zusatzachse. Diese beiden Achsen sind weitere im MD30460 \$MA_BASE_FUNCTION_MASK mit Bit 2 = 1 definierte Planachsen mit Durchmesserangabe.

DIAMON ist nach Hochlauf nicht aktiv.

Programmcode	Kommentar
N10 G0 G90 X100 Y50	;keine Durchmesserprogrammierung aktiv
N20 DIAMON	;Durchmesserprogrammierung kanalspezifisch, wirksam für X
N30 Y200 X200	;Maßangabe: X im Durchmesser, Y im Radius
N40 DIAMONA[Y]	;achsspezifische modale Durchmesserprogrammierung, wirksam für Y
N50 Y250 X300	;Maßangabe: X und Y im Durchmesser
N60 DIAM90	;Maßangabe: X G90/AC im Durchmesser, G91/IC im Radius
N70 Y200	;Y: weiterhin achsspezif. modale Durchmesserprogrammierung
N75 G91 Y20 U=DIC(40)	;Maßangabe: Y im Durchmesser, U satzweise IC im Durchmesser
N80 X50 Y100	;Maßangabe: X im Radius (G91), Y im Durchmesser
N85 G90 X100 U200	;Maßangabe: X im Durchmesser, U im Radius
N90 DIAMCHANA[Y]	;Y übernimmt den Kanalzustand DIAM90
N95 G91 X100 Y100	;Maßangabe: X und Y im Radius(G91)
N100 G90 X200 Y200	;Maßangabe: X und Y im Durchmesser

Beispiel 2

Planachsen mit Durchmesserangabe wie im vorherigen Beispiel appliziert.

X und Y befinden sich im Kanal 1 und sind zusätzlich im Kanal 2 bekannt, d. h. für Achstausch zugelassen.

Programmcode	Kommentar
Kanal 1	
N10 G0 G90 X100 Y50	;keine Durchmesserprogrammierung aktiv
N20 DIAMON	;Durchmesserprogrammierung kanalspezifisch für X
N30 Y200 X200	;Maßangabe: X im Durchmesser, Y im Radius
N40 DIAMONA[Y]	;Y achsspezifische modale Durchmesserprogrammierung
N50 Y250 X300	;Maßangabe: X und Y im Durchmesser
N60 SETM(1)	;Synchronmarke 1
N70 WAIT(1,2)	;warten auf Synchronmarke 1 im Kanal 2
Kanal 2	
...	
N50 DIAMOF	;Kanal 2 keine Durchmesserprogrammierung aktiv
...	
N100 WAIT(1,1)	;warten auf Synchronmarke 1 im Kanal 1
N110 GETD(Y)	;Achstausch direkt Y
N120 Y100	;Y wird der kanalspezifischen Durchmesserprogrammierung ;im Kanal 2 unterstellt, d.h. Maßangabe im Radius

13.6 Datenlisten

13.6.1 Maschinendaten

13.6.1.1 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20050	AXCONF_GEOAX_ASSIGN_TAB[n]	Zuordnung Geometrieachse zu Kanalachse
20060	AXCONF_GEOAX_NAME_TAB[n]	Geometrieachsname im Kanal
20100	DIAMETER_AX_DEF	Geometrieachse mit Planachsfunktion
20110	RESET_MODE_MASK	Festlegung der Steuerungs-Grundstellung nach Hochlauf und RESET/Teileprogrammende
20112	START_MODE_MASK	Festlegung der Steuerungs-Grundstellung bei NC-Start
20150	GCODE_RESET_VALUES[n]	Löschstellung der G-Gruppen
20152	GCODE_RESET_MODE[n]	G-Befehl Grundstellung bei RESET/Teileprogrammende
20360	TOOL_PARAMETER_DEF_MASK	Definition der WZ-Parameter

Nummer	Bezeichner: \$MC_	Beschreibung
20624	HANDWH_CHAN_STOP_COND	Festlegung des Verhaltens des manuellen Verfahrens per Handrad
27100	ABSBLOCK_FUNCTION_MASK	Satzanzeige mit Absolutwerten parametrieren

13.6.1.2 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30460	BASE_FUNCTION_MASK	Achsfunktionen

14.1 Kurzbeschreibung

Allgemeines

Das PLC-Grundprogramm organisiert den Austausch von Signalen und Daten zwischen dem PLC-Anwenderprogramm und dem NC-, HMI- und MCP-Bereich. Bei den Signalen und Daten wird zwischen folgenden Gruppen unterschieden:

- Zyklischer Signalaustausch
- Ereignisgesteuerter Signalaustausch
- Meldungen

Zyklischer Signalaustausch

Die Signale, die zyklisch ausgetauscht werden, bestehen im Wesentlichen aus Bitfeldern.

- Sie enthalten **Befehle**, die von der PLC an den NC übergeben werden (z. B. Start, Stopp etc.) und **Status-Informationen** des NC (z. B. Programm läuft, Programm unterbrochen etc.).
- Die Bitfelder sind gegliedert in Signale für:
 - BAG
 - Kanäle
 - Achsen/Spindeln
 - allgemeine NC-Signale

Der zyklische Signalaustausch wird vom Grundprogramm am Zyklusanfang der PLC (OB1) durchgeführt. Dadurch ist sichergestellt, dass z. B. die Signale vom NC über einen Zyklus konstant bleiben.

Ereignisgesteuerter Signalaustausch NC → PLC

PLC-Funktionen, die abhängig vom Werkstückprogramm ausgeführt werden müssen, werden über Hilfsfunktionen im Werkstückprogramm angestoßen. Gelangt ein Satz mit Hilfsfunktionen zur Ausführung, so hängt es von der Art der Hilfsfunktion ab, ob der NC die Ausführung dieser Funktion abwarten muss (z. B. Werkzeugwechsel) oder ob diese Funktion begleitend zur Werkstückbearbeitung zur Ausführung kommt (z. B. Werkzeugbereitstellung bei Fräsmaschinen mit Kettenmagazinen).

Damit die NC-Bearbeitung möglichst wenig beeinflusst wird, muss die Datenübergabe möglichst schnell und trotzdem sicher erfolgen. Sie erfolgt deshalb alarm- und quittungsgesteuert. Das Grundprogramm wertet die Signale und Daten aus, quittiert dies zum NC und überträgt die Daten am Zyklusanfang zur Anwendernahtstelle. Erfordern die Daten keine Anwenderquittung, so wird die NC-Bearbeitung dadurch nicht beeinflusst.

Ereignisgesteuerter Signalaustausch PLC → NC

Immer dann, wenn die PLC an den NC einen Auftrag übergibt (z. B. Verfahren einer Hilfsachse), findet ein "ereignisgesteuerter Signalaustausch PLC → NC" statt. Auch hier erfolgt die Datenübergabe quittungsgesteuert. Vom Anwenderprogramm aus wird ein derartiger Signalaustausch über einen FB bzw. FC angestoßen.

Die zugehörigen FBs (Funktionsbausteine) und FCs (Function Calls) werden zusammen mit dem Grundprogramm ausgeliefert.

Meldungen

Die Erfassung und Aufbereitung der Anwendermeldungen erfolgt durch das Grundprogramm. Über ein vereinbartes Bitfeld werden die Meldesignale an das Grundprogramm übergeben. Dort werden diese Signale ausgewertet und bei Auftreten der Meldeereignisse über die ALARM S/SQ-Funktionen in den Alarm-Puffer der PLC eingetragen. Ist ein HMI (z. B. SINUMERIK Operate) vorhanden, so werden die Meldungen an den HMI übertragen und angezeigt.

Datenaustausch PLC / HMI

Bei diesem Datenaustausch übernimmt der HMI die Initiative als sogenannter Client am Bussystem. Der HMI fordert Daten an bzw. beschreibt Daten. Die PLC wickelt über das Betriebssystem diese Aufträge zum Zykluskontrollpunkt ab. Das PLC-Grundprogramm ist an diesen Übertragungen nicht beteiligt.

Hinweis

Die Funktion der Maschine wird maßgebend vom PLC-Programm bestimmt. Jedes im Arbeitsspeicher vorhandene PLC-Anwenderprogramm kann mit dem Programmiergerät geändert werden.

Know How-Schutz für Anwenderbausteine

Zum Schutz des in Anwenderbausteinen (OB, FB und FC) enthaltenen Know How, können diese mit dem in SIMATIC STEP 7 enthaltenen SBP-Tool (SIMATIC Block Protection) verschlüsselt werden. Ohne Angabe des bei der Verschlüsselung verwendeten Passworts, ist dann für diese Bausteine Öffnen, Debuggen und Modifizieren nicht mehr möglich.

Bei der Verschlüsselung ist das Automatisierungssystem anzugeben, auf deren PLC-CPU die Bausteine ausführbar sein sollen: SIMATIC und/oder SINUMERIK PLC-CPU.

Die Handhabung der Bausteine, z. B. Laden in die CPU, wird durch die Verschlüsselung nicht beeinflusst.

Voraussetzung

SIMATIC STEP 7 ab Version 5.5 SP3

14.2 Eckdaten der PLC-CPU

Eckdaten der PLC-CPU

Literatur:

Die Übersicht der Eckdaten der in die SINUMERIK NCU integrierten PLC-CPU findet sich in: Gerätehandbuch NCU 7x0.3 PN, Kapitel "Technische Daten"

Hinweis
E/A-Adressen für integrierte Antriebe

Die E/A-Adressen oberhalb 4096 sind für die integrierten Antriebe der NCU reserviert und dürfen nicht anderweitig belegt werden.

Funktionen des PLC-Grundprogramms

Umfang		
	Achsen / Spindeln	31
	Kanäle	10
	BAGs	10
Funktionen		
	Status- / Steuersignale	+
	M-Decoder (M00-99)	+
	G-Gruppen-Decoder	+
	Hilfsfunktionsverteiler	+
	Hilfsfunktionsübergabe alarmgesteuert	+
	M-Dekodierung nach Liste	+
	Achsen / Spindeln bewegen von PLC	+
	ASUP-Schnittstelle	+
	Fehler- / Betriebsmeldungen	+
	MCP- und BHG-Signale übertragen	+
	Display-Steuerung Bedienhandgerät	+
	Lesen / Schreiben von NC-Variablen und GUDs	+
	PI-Dienste	+
	Werkzeugverwaltung	+
	Stern / Dreieck Umschaltung	+
	M zu N	+
	Safety Integrated	+
	Programm-Diagnose	+

14.3 PLC-Betriebssystemversion

Die Version des PLC-Betriebssystems wird angezeigt unter:

- Bedienoberfläche von SINUMERIK Operate: "Bedienbereichsumschaltung" > "Diagnose" > "Version" ⇒ Versionsdaten /Systemsoftware NCU: Auswahl "PLC" > "Details" ⇒ Versionsdaten /Systemsoftware NCU/PLC: Die PLC-Betriebssystemversion wird in der erste Zeile unter "PLC 3xx..." angezeigt.

Hinweis

Die angezeigte Version ist SINUMERIK-spezifisch. Sie ist nicht kompatibel zur zugrunde liegenden SIMATIC CPU.

- SIMATIC STEP 7, HW Konfig: In den Eigenschaften der PLC-CPU im SINUMERIK Baugruppenträger: "Eigenschaften – CPU 3xx..." > "Bestell-Nr. / Firmware": xxxx / **Vx.y.z**

Hinweis

Es wird die Version der zugrunde liegenden SIMATIC CPU angezeigt.

14.4 PLC-Betriebsartenschalter

Der PLC-Betriebsartenschalter befindet sich auf Frontseite der NCU-Baugruppe. Über den PLC-Betriebsartenschalter können folgende PLC-Betriebsarten eingestellt werden:

S ¹⁾	Bedeutung	Bemerkung
0	RUN-P	Änderungen am PLC-Programm können ohne Aktivierung des Kennwortes erfolgen
1	RUN	Über ein Programmiergerät (PG) sind nur lesende Zugriffe möglich. Erst nach dem Setzen des Kennwortes, können Änderungen am PLC-Programm durchgeführt werden.
2	STOP	Die Bearbeitung des PLC-Anwenderprogramms ist angehalten und alle PLC-Ausgänge auf Ersatzwerte gesetzt.
3	MRES	Die PLC wird in den STOP-Zustand versetzt und anschließend PLC-Urlöschen (Default Data) durchgeführt.
1) Schalterstellung des PLC-Betriebsartenschalters		

Literatur

Eine ausführliche Beschreibung der Lage des PLC-Betriebsartenschalters auf Frontseite der NCU-Baugruppe, sowie seine Verwendung im Zusammenhang mit NC- und PLC-Urlöschen findet sich in:

Inbetriebnahmehandbuch CNC: NC, PLC, Antrieb:

- Kapitel "Einschalten Hochlauf" > "Bedien- und Anzeigeelemente für den Hochlauf"
- Kapitel "Einschalten Hochlauf" > "NC- und PLC-Urlöschen"
- Kapitel "Allgemeine Tipps" > "Getrenntes NC- und PLC-Urlöschen"

14.5 Ressourcen (Timer, Zähler, FC, FB, DB, Peripherie) reservieren

Reservierung von Ressourcen (Timer, Zähler, FC, FB, DB, Peripherie)

Folgende Komponenten sind für das Grundprogramm reserviert:

- **Timer**
Keine Reservierung.
- **Zähler**
Keine Reservierung.
- **FC, FB, DB**
Für das Grundprogramm sind FC0 bis FC29 und FB0 bis FB29 reserviert. Für FC, FB ist zusätzlich das Nummernband von 1000 bis 1023 reserviert. Bei Datenbausteinen sind DB1 bis DB61, DB71 bis DB80 reserviert. Für DB ist zusätzlich das Nummernband 1000 bis 1099 reserviert. Datenbausteine von nicht aktivierten Kanälen, Achsen / Spindeln, Werkzeugverwaltung sind frei für den Anwender.
- **Peripheriebereich**
Die PLC haben ein Peripherie-Adressvolumen von je 16384 Byte für Ein- / Ausgänge. Hiervon sind für integrierte Antriebe die Adressbereiche ab 4096 reserviert bzw. belegt. Diagnoseadressen für Baugruppen können aber in dem obersten Adressbereich platziert werden, so wie es STEP 7 vorschlägt. Weiterhin ist der Adressbereich 4080 bis 4097 belegt für NC, CP und HMI im Rack 0 der SIMATIC 300 Station (bei NCU 7x0.3).

14.6 Inbetriebnahme Hardware-Konfiguration der PLC-CPU

Die Inbetriebnahme der PLC-CPU ist ausführlich beschrieben in:

Literatur

Inbetriebnahmehandbuch CNC: NC, PLC, Antrieb:

- Kapitel: "PG/PC mit PLC verbinden"
- Kapitel: "Inbetriebnahme PLC"
- Kapitel: "Grundlagen" > "PLC-Programm"
- Kapitel: "Allgemeine Tipps" > "Getrenntes NC- und PLC-Urlöschen"
- Kapitel: "Allgemeine Tipps" > "Einbringen PG/PC ins Netzwerk (NetPro)"

14.7 Inbetriebnahme PLC-Programm

14.7.1 Installation des Grundprogramms

Die Installation des Grundprogramms ist ausführlich beschrieben in:

Literatur

Inbetriebnahmehandbuch CNC: NC, PLC, Antrieb; Kapitel: "Inbetriebnahme PLC" > "PLC-Programm erstellen"

Hinweis

Installation / Update

Vor Installation der Toolbox für SINUMERIK 840D sl muss SIMATIC STEP 7 installiert sein.

Es wird nach einem Update von STEP 7 empfohlen, die Hardware-Ergänzungen für STEP 7 aus der Toolbox neu zu installieren.

Inhalt

Die Quellprogramme der OBs inklusive Standard-Parametrierung, die Nahtstellensymbolik und DB-Vorlagen für Bedienhandgerät und M-Dekodierung sind im Grundprogramm enthalten.

14.7.2 Anwendung des Grundprogramms

Für jede Anlage (Maschine) ist ein neues CPU-Programm in einem Projekt über die STEP 7-Software anzulegen (z. B. "Drehma1").

Anmerkung

Die Katalogstrukturen eines Projekts und die Vorgehensweise zur Erstellung von Projekten und Anwenderprogrammen sind in der zugehörigen SIMATIC-Dokumentation enthalten.

Vorgehensweise

Das Kopieren der Grundprogrammbausteine erfolgt über den SIMATIC Manager mit "Datei" > "Öffnen" > "Bibliothek".

Aus der Bibliothek müssen folgende Teile kopiert werden:

- Aus dem Baustein-Container: FCs, FBs, DBs, OBs, SFC, SFB, UDT
- Die Source_files (aus dem Source-Container): GPOB840D
- Gegebenenfalls MDECLIST, BHG_DB und weitere
- Die Symboltabelle (aus dem Symbol-Container)

Kompatibilität zu STEP 7

Es sind keine Abhängigkeiten des Grundprogramms zu den derzeitigen STEP 7-Versionen vorhanden.

14.7.3 Versionskennzeichnungen

Grundprogramm

Die Version des Grundprogramms einschließlich des Steuerungstyps wird im Versionsbild der Bedienoberfläche angezeigt.

Der Steuerungstyp ist folgendermaßen verschlüsselt:

Linksbündige Dekade von DB17.DBD0 (Byte 0)	Steuerungstyp
03	SINUMERIK 840D sl (NCU 7x0)

Versionskennzeichnungen Anwenderprogramm

Der Anwender kann seine eigene PLC-Versionskennzeichnung auch auf dem HMI im Versionsbild darstellen. Hierzu ist in einem beliebigen Datenbaustein ein Datum vom Datentyp STRING mit max. **54** Zeichen zu definieren. Es wird allerdings keine Interpretation der Version durchgeführt, sondern lediglich der eingetragene String übernommen. Die Parametrierung auf diesen String erfolgt über einen Zeiger am FB1. Hierzu muss der Datenbaustein symbolisch definiert sein. Siehe hierzu Bausteinbeschreibung FB1 (Seite 980).

Die Versionskennzeichnung kann wie folgt im String formatiert werden:

- xx.yy
- xx.yy.zz
- ww.xx.yy.zz
- vv.ww.xx.yy.zzz
- x.y
- x.y.z
- w.x.y.z
- v.w.x.y.z

Zusätzlich zur Versionskennung kann noch ein Datum im String eingetragen werden, welches bei entsprechender Formatierung im HMI im Versionsbild angezeigt wird. Dabei wird allerdings keine Interpretation des Datums durchgeführt sondern lediglich der eingetragene String übernommen. Eine Kombination mit Versionskennungen ist möglich. Aktuell werden die folgenden Formate unterstützt:

- 00/00/0000
- 0000/00/00
- 00/00/00

Beispiele:

- "Testprojekt Version 01.02.03 01/01/2015"
- "1.2 2015/01/01 Testprojekt"
- "01/01/15 Version 01.02 Testprojekt"

14.7.4 Maschinenprogramm

Das Maschinenprogramm wird durch den Maschinenhersteller unter Zuhilfenahme der Bibliotheksroutinen des Grundprogramms erstellt. Im Maschinenprogramm sind die logischen Verknüpfungen und Abläufe der Maschine enthalten. Weiterhin werden die Nahtstellensignale zur NC bedient. Komplexere Kommunikationsfunktionen zur NC wie z. B. NC-Daten lesen, schreiben, Werkzeugverwaltungs-Quittungen, usw. werden über die Bausteine FCs, FBs des Grundprogramms angestoßen und ausgeführt.

Das Maschinenprogramm kann in unterschiedlichen STEP 7-Erstellungssprachen wie z. B. AWL, KOP, FUP, S7-HIGRAPH, S7GRAPH, SCL erstellt werden. Das komplette Maschinenprogramm muss in der richtigen Reihenfolge erstellt und kompiliert sein.

Das bedeutet, dass Bausteine die von anderen Bausteinen aufgerufen werden, generell **vor diesen** Bausteinen kompiliert werden müssen.

Wenn diese aufgerufenen Bausteine nachträglich im Zuge der Programmentwicklung in der Schnittstelle (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) geändert werden, dann sind auch der rufende Baustein und alle damit verbundenen Bausteine anschließend zu kompilieren. Diese Vorgehensweise gilt sinngemäß auch für Instanz-Datenbausteine für FBs. Wenn diese Reihenfolge nicht eingehalten wird, treten Zeitstempel-Konflikte bei der Rückübersetzung in STEP 7 auf. Somit ist die Rückübersetzbarkeit der Bausteine nicht sichergestellt und bei der Funktion "Status Baustein" treten unter anderen unnötige Konflikte auf. Es ist außerdem empfehlenswert die im Kontaktplan oder in Einzelanweisung (inkrementeller Modus) erstellten Bausteine in ASCII-AWL durch den STEP 7-Editor zu generieren.

14.7.5 Datensicherung

Die PLC-CPU speichert keine symbolischen Namen ab, sondern nur die Datentypbeschreibungen der Bausteinparameter VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR und die Datentypen der globalen Datenbausteine.

Hinweis

Ohne das zugehörige Projekt für diese Maschine ist keine vernünftige Rückübersetzung möglich. Dies betrifft insbesondere z. B. die Funktion "Status Baustein" oder nachträglich notwendige Änderungen der PLC-CPU-Programme. Es ist deshalb erforderlich, das sich in der PLC-CPU befindende STEP 7-Projekt an der Maschine unbedingt aufzubewahren. Für einen Servicefall ist dies eine große Hilfe und erspart einen unnötigen Zeitaufwand das ursprüngliche zugehörige Projekt wieder herzustellen.

Wenn das STEP 7-Projekt vorhanden ist und nach den genannten Regeln erstellt wurde, kann an dieser Maschine in der PLC-CPU symbolisch gearbeitet werden. Gegebenfalls sind auch die Quell-Programme der Maschine als ".awl"-Dateien mit zu hinterlegen für eine eventuell notwendige Hochrüstung.

Als Minimum sollten die Quellprogramme aller Organisationsbausteine und auch aller Instanz-Datenbausteine vorhanden sein.

14.7.6 PLC-Serieninbetriebnahme, PLC Archive

Nach dem Laden der Bausteine in die PLC-CPU kann über die Bedienoberfläche HMI ein Serien-Archiv erzeugt werden zur Datensicherung an der Maschine. Die Datensicherung soll direkt nach dem Laden der Bausteine im PLC-Stop-Zustand erfolgen um Konsistenz der Daten zu erreichen. Diese Datensicherung ersetzt nicht die Sicherung des SIMATIC-Projekts, da im Serienarchiv nur Binärdaten abgespeichert werden. Hier stehen z. B. keine symbolischen Informationen zur Verfügung. Weiterhin werden keine CPU-DBs (SFC22-DBs) und in der CPU erzeugte SDBs gesichert.

Auswahl des SINUMERIK Archivierungsprogramms

Alternativ kann das PLC-Serienarchiv direkt aus dem SIMATIC-Projekt erzeugt werden:

- Öffnen sie im SIMATIC Manager den Dialog "Einstellungen": Menüleiste "Extras" > "Einstellungen"
- Öffnen sie die Registerkarte "Archivieren"
- Wählen sie im Dropdown-Listefeld "Bevorzugtes Archivier-Programm" das SINUMERK Archivierungsprogramm "SINUMERIK (*.arc)" aus.

Start des SINUMERIK Archivierungsprogramms

Das SINUMERIK Archivierungsprogramm wird im SIMATIC Manager über den Menüpunkt "Datei" > "Archivieren" gestartet.

Nach der Vergabe des Archivnamens wird das PLC-Archiv erzeugt. Enthält das Projekt mehrere Programmpfade, kann im Dialog das S7-Programm ausgewählt werden, für welches das PLC-Archiv angelegt wird. Es werden alle Bausteine archiviert, die im ausgewählten Programmpfad enthalten sind (außer Datenbausteine, die mit SFC22 (online) in der CPU erstellt wurden).

Für das Archivier-Programm kann die Funktion "SDB Archiv" aktiviert bzw. deaktiviert werden. Ist "SDB Archiv" aktiviert, wird ein PLC-Archiv erstellt, in dem sich nur die Systemdatenbausteine (SDB) des ausgewählten Programmpfades befinden.

Automatisierung

Die Erzeugung des Serienarchivs ist auch automatisierbar (vergleichbar mit der Kommandoschnittstelle von STEP 7). Diese Erzeugung stellt eine Erweiterung der Kommandoschnittstelle dar.

Folgende Funktionen stehen in dieser Erweiterung zur Verfügung:

Die Funktionen (hier mit VB-Script dargestellt) sind erst nach Aufruf von Server-Instanzierungen und Magic-Aufruf verfügbar:

```
Const S7BlockContainer = 1138689, S7PlanContainer = 17829889
Const S7SourceContainer = 1122308
set S7 = CreateObject("Simatic.Simatic.1")
rem Kommandoschnittstelle von STEP 7 instanzieren
Set S7Ext = CreateObject("SimaticExt.S7ContainerExt")
Call S7Ext.Magic("")
```

Funktionen:

- Function Magic(bstrVal As String) As Long
- Function MakeSerienIB (FileName As String, Option As Long, Container As S7Container) As Long

Beschreibung

Function **Magic**(bstrVal As String) As Long

Über Aufruf wird Zugang zu bestimmten Funktionen erreicht. Die Funktion muss nach Serverinstanzierung einmalig aufgerufen werden. Der Wert von bstrVal kann leer sein. Hiermit wird die korrekte STEP 7-Version und Path-Angabe in Autoexec geprüft. Bei Rückgabe von 0 sind die Funktionen freigeschaltet.

Rückgabe (-1) = falsche STEP 7-Version

Rückgabe (-2) = kein Eintrag in Autoexec.bat

Function **MakeSerienIB**(FileName As String, Option As Long, Container As S7Container) As Long

Parameter "Option":

0:	Normale Serien-IBN-Datei mit Urlöschen.
Bit 0 = 1:	Serien-IBN-Datei ohne Urlöschen. Wenn SDBs im Projekt sind, ist diese Option nicht wirksam. Es wird dann immer ein Urlöschen durchgeführt.
Bit 1 = 1:	Serien-IBN-Datei mit PLC-Neustart

Rückgabe-Wert:

0	= OK
-1	= Funktion nicht verfügbar, Funktion "Magic" vorher aufrufen
-2	= Dateiname nicht erzeugbar
-4	= Parameter Container ungültig oder Baustein Container leer
-5	= interner Fehler (Speicheranforderung durch Windows abgelehnt)
-6	= interner Fehler (Problem im STEP 7-Projekt)
-7	= Schreibfehler beim Erzeugen der Serien-IBN-Datei (z. B. Diskette voll)

Nutzung im Script

Programmcode

```
If S7Ext.Magic("") < 0 Then
    Wscript.Quit(1)
End If

Set Proj1 = s7.Projects("neu")
set S7Prog = Nothing
Set s7prog = Proj1.Programs.Item(1) 'wenn es nur ein Programm gibt
For Each cont In s7prog.Next
    If (Cont.ConcreteType = S7BlockContainer) Then
```


Programmcode

```

'Bausteincontainer prüfen
Exit For
End if
Cont = Nothing
Next
Fehler = S7Ext.MakeSerienIB("f:\dh\arc.dir\PLC.arc", 0, Cont)
'Jetzt Fehlerauswertung

```

Der oben programmierte For Each ... Next -Block kann in der Programmiersprache Delphi wie folgt programmiert werden (ähnliche Programmierung gilt auch für die Programmiersprache C, C++):

Programmcode

```

Var
    EnumVar: IEnumVariant;
    rgvar: OleVariant;
    fetched: Cardinal;

//For Each Next
EnumVar := (S7Prog.Next._NewEnum) as IEnumVariant;
While (EnumVar.Next(1, rgvar, fetched) = S_OK) Do Begin
    Cont := IS7Container(IDispatch(rgvar)); // Bausteincontainer,
    Quellen prüfen
    If (Cont.ConcreteType = S7BlockContainer) Then Break;
    Cont := NIL;
End;

```

14.7.7 Software-Hochrüstung

Für einen definierten Ausgangszustand ist vor einer PLC-Software-Hochrüstung ein PLC-Urlöschen durchzuführen. Dabei werden unter anderem alle Anwenderdaten (Programm- und Datenbeisteine) gelöscht. Das PLC-Urlöschen ist beschrieben in:

Literatur:

Inbetriebnahmehandbuch IBN CNC: NC, PLC, Antrieb; Allgemeine Tipps, Kapitel: PLC-Urlöschen

Neues SIMATIC S7-Projekt erzeugen

Im Regelfall ist bei einem neuen NCU-Softwarestand das neue PLC-Grundprogramm mit einzubinden. Hierzu ist die Übertragung der Grundprogrammteile in das Anwenderprojekt erforderlich. Befinden sich folgende Programm- und Datenbausteine schon im Anwenderprojekt sollten sie nicht mit den Bausteinen des PLC-Grundprogramms übertragen werden: OB1, OB40, OB82, OB86, OB100, FC12 und DB4. Diese sind evtl. vom Anwender verändert worden und sollten nicht überschrieben werden. Das neue Grundprogramm ist mit dem Anwenderprogramm zusammenzubinden. Folgende Vorgehensweise ist dabei zu beachten:

1. Vor dem Kopieren des PLC-Grundprogramms von allen Anwender-Bausteinen die Text- bzw. Quell-Datei generieren.
2. Die neuen Grundprogramm-Bausteine in das SIMATIC S7-Projekt kopieren (Beschreibung siehe Kapitel "Anwendung des Grundprogramms (Seite 888)")
3. Alle Anwenderprogramme "*.awl" in korrekter Reihenfolge neu zu übersetzen! (siehe auch " Maschinenprogramm (Seite 890)")
4. Dieses neu kompilierte SIMATIC S7-Projekt ist dann mit STEP 7 in die PLC zu laden.

Im Normalfall reicht jedoch ein neues Übersetzen der Organisationsbausteine (OBs) und der Instanz-Datenbausteine des S7-Projekts aus. D. h. es brauchen vor der Hochrüstung nur Quellen für die Organisationsbausteine und die Instanz-Datenbausteine erzeugt werden.

NC-Variablen

Für jeden NC-Softwarestand (auch ältere Versionen) kann der neueste NC-VAR-Selector verwendet werden. Für ältere NC-Softwarestände können die Variablen auch aus der neuesten Gesamtliste selektiert werden. Der Informationsinhalt im DB120 (Standard-DB für Variablen) ist nicht abhängig vom Softwarestand. D. h. selektierte Variable in einem älteren Softwarestand müssen bei einer Software-Hochrüstung nicht neu selektiert werden.

14.7.8 Peripheriebaugruppen (FM-, CP-Baugruppen)

Für komplexere Peripheriebaugruppen (FM-, CP-Baugruppen) werden in der Regel Zusatzpakete zu STEP 7 benötigt. In diesen Zusatzpaketen werden Unterstützungsbausteine (FC / FB) mitgeliefert. Diese Bausteine beinhalten spezifische Funktionen zum Betrieb der jeweiligen Baugruppe. Diese Funktionen können im Anwenderprogramm parametrisiert und aufgerufen werden.

Nummerngleichheit

Bei Nummerngleichheit von Hantierungs- und Grundprogrammteilen müssen die Baustein-Nummern des Grundprogramms unverändert bleiben. Die Baustein-Nummern der Hantierungsbausteine müssen über STEP 7 auf freie Nummern umbenannt werden.

14.7.9 Fehler-Beseitigung

Dieser Abschnitt soll Hinweise zu Problemfällen und deren Beseitigung bzw. auch Ursachen geben, bevor ein Hardware-Tausch erfolgt.

Fehler, Ursache/Beschreibung und Abhilfe			
lfd. Nr. Fehler- hinweis	Fehler	Ursache / Beschreibung	Abhilfe
1	Keine Verbindung über MPI zur PLC.	Das MPI-Kabel ist nicht gesteckt oder defekt. Möglicherweise ist auch die STEP 7-Software für die MPI-Karte nicht richtig konfiguriert.	Versuch: Mit dem PG im STEP 7-Editor über die Verbindung "Direct_PLC" an-koppeln. Hierbei müssen einige Teil-nehmeradressen eingeblendet wer-den. Fehlen die, dann MPI-Kabel de-fekt / nicht gesteckt.
2	Trotz Urlö-schen der PLC ist kein Zugriff zur PLC möglich.	Ein Systemdatenbaustein SDB 0 wurde mit einer geänderten MPI-Adresse geladen. Hierdurch tritt ein MPI-Buskonflikt wegen dop-pelt vergebenen Adressen auf.	Alle MPI-Kabel zu anderen Kompon-ten trennen. Mit dem PG die Verbin-dung "Direct_PLC" aufbauen. Durch Taufe der MPI-Adresse die MPI-Adres-se korrigieren.
3	Alle 4 LEDs der PLC blin-ken (DI-Kata-strophe)	Ein Systemfehler in der PLC ist aufgetreten. Maßnahmen: Zur genaueren Analyse des Sys-temfehlers ist der Diagnosepuffer der PLC auszulesen. Hierzu ist die PLC zuerst in den Stopp-Zu-stand zu bringen (z. B. Schalter "PLC" auf Stellung 2. Anschlie-ßend ist ein Hardware-RESET er-forderlich. Danach kann mit STEP 7 der Diagnosepuffer aus-gelesen werden. Die Informatio-nen aus dem Diagnosepuffer sind der Hotline / Entwicklung zu mel-den. Wenn nach dem Hardware-RESET zusätzlich ein Urlöschen angefordert wird, ist erst das Ur-löschen durchzuführen. Anschlie-ßend kann dann im Stopp-Zu-stand der Diagnosepuffer gele-sen werden.	Nach RESET bzw. nach Neuladen des PLC-Programms läuft das System un-ter Umständen normal weiter. In jedem Fall ist der Diagnosepuffer der Entwick-lung zur Verfügung zu stellen.

14.8 Ankopplung der PLC-CPU

14.8.1 Allgemeine Informationen

Als PLC wird bei SINUMERIK 840D sl eine CPU des Automatisierungssystems S7-300 verwendet. Die PLC-CPU ist als Submodul in die NCU-Komponente integriert. Ein Verweis auf die Leistungsdaten der PLC-CPU findet sich im Kapitel "Eckdaten der PLC-CPU (Seite 885)".

14.8.2 Eigenschaften der PLC-CPU

Die in die SINUMERIK 840D sl integrierte PLC hat im Allgemeinen die gleiche Funktionalität wie die entsprechende SIMATIC S7-300 PLC.

Zu Abweichungen, siehe Verweis in Kapitel "Eckdaten der PLC-CPU (Seite 885)".

Durch das teilweise unterschiedliche Speicherkonzept gegenüber einer SIMATIC S7-300 PLC sind bestimmte Funktionen nicht vorhanden (z. B. Bausteine auf Memory Card, Projekt auf Memory Card speichern).

Hinweis

Bei einer SIMATIC S7-300 PLC wird, wie auch bei der in SINUMERIK integrierten PLC, bei Auslösen eines "PLC-Stop" durch Bedienung am Programmiergerät kein automatisches Starten der PLC nach Spannungsausfall und -wiederkehr durchgeführt. Die PLC verbleibt aus Sicherheitsgründen mit entsprechendem Diagnoseeintrag im Stopp-Zustand. Ein Starten der PLC ist nur durch Bedienung am Programmiergerät "Neustart ausführen" oder über den Betriebsartenschalter "Stop" > "Run" (Warmstart) möglich.

14.8.3 Nahtstelle mit integrierter PLC

Physikalische Nahtstellen

Die in die NCU integrierte PLC bietet bei SINUMERIK 840D sl die Möglichkeit, den Austausch der Signale zwischen NC und PLC direkt über ein Dual-Port-RAM vorzunehmen.

Datenaustausch mit der Bedientafel

Der Datenaustausch mit der Bedientafel (z. B. TCU / OP) kann über Ethernet oder PROFIBUS erfolgen. Bei Anschluss über Ethernet erfolgt die Kommunikation über den integrierten Kommunikationsprozessor (CP 840D sl).

Der Datenaustausch mit Maschinensteuertafel (MCP) und Bedienhandgerät (BHG) kann über MPI, PROFIBUS oder Ethernet erfolgen.

Programmiergeräte werden vorzugsweise über Ethernet oder über MPI (Multi-Point-Interface) direkt an die PLC angeschlossen.

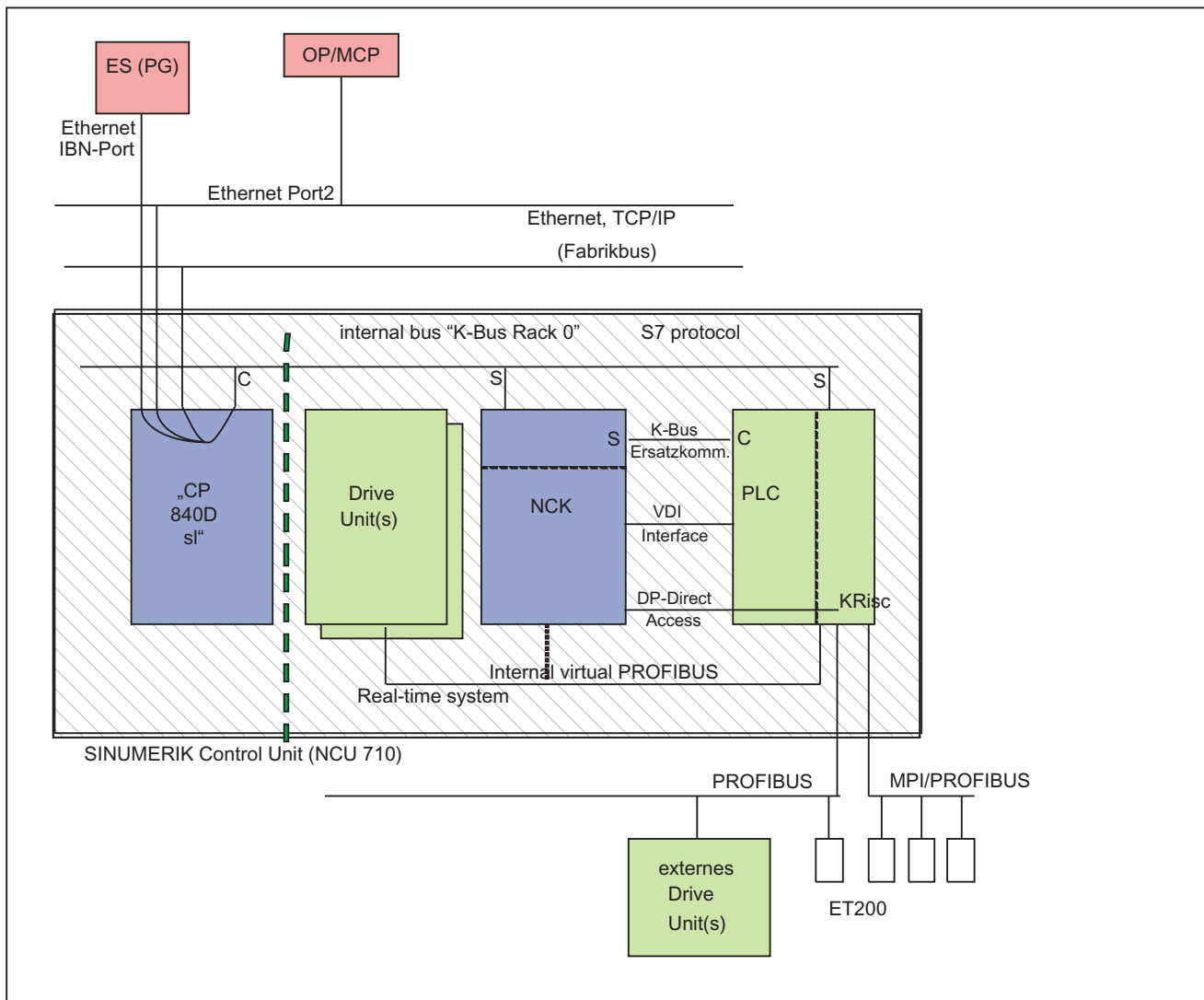


Bild 14-1 NC-PLC-Kopplung bei SINUMERIK 840D sl (integrierte PLC)

Nahtstelle: NC / PLC

Der Datenaustausch zwischen NC und PLC wird auf PLC-Seite vom Grundprogramm organisiert. Die von der NC in der NC/PLC-Nahtstelle abgelegten **Statusinformationen**, wie z. B. "Programm läuft", werden vom Grundprogramm am Zyklusanfang (OB1) in Datenbausteine kopiert, auf die im Anwenderprogramm dann zugegriffen werden kann (Anwendernahtstelle). Die vom Anwender in die Nahtstellen-DB eingetragenen **Steuersignale** an die NC, wie z. B. "NC-Start", werden ebenso am Zyklusanfang ins interne DPR zur NC übertragen.

Die abhängig vom Werkstückprogramm an die PLC übergebenen **Hilfsfunktionen** werden zunächst alarmgesteuert vom Grundprogramm ausgewertet und dann am Anfang von OB1 an die Anwendernahtstelle übergeben. Sind im betreffenden NC-Satz Hilfsfunktionen enthalten, die eine Unterbrechung der NC-Bearbeitung erfordern (wie z. B. M06 für Werkzeugwechsel), wird die NC-Satz-Decodierung vom Grundprogramm zunächst für eine PLC-Zykluszeit angehalten. Der Anwender kann dann über das Nahtstellensignal

"Einleesesperre" die Decodierung so lange anhalten, bis z. B. der Werkzeugwechsel abgeschlossen ist. Sind dagegen im betreffenden NC-Satz nur Hilfsfunktionen enthalten, die keine Unterbrechung der Decodierung erfordern (wie z. B. M08 für Kühlmittel Ein), so wird die Übergabe dieser "schnellen" Hilfsfunktionen direkt im OB40 quittiert, sodass die Decodierung von der Übergabe an die PLC nur unwesentlich beeinflusst wird.

Die vom NC übergebenen **G-Befehle** werden ebenfalls alarmgesteuert ausgewertet und quittiert, dann jedoch direkt an die Anwendernahtstelle übergeben. Wird ein G-Befehl an mehreren Stellen im PLC-Programm ausgewertet, können sich Unterschiede in der Information des G-Befehls innerhalb eines PLC-Zyklus ergeben.

Bei **NC-Aktionen**, die vom PLC aus angestoßen und parametrierbar werden (z. B. konkurrierende Achsen verfahren), erfolgt der Anstoß und die Parametrierung nicht über Nahtstellen-DBs, sondern über FCs und FBs. Die zu den NC-Aktionen gehörenden FCs werden zusammen mit dem Grundprogramm ausgeliefert. Die jeweils benötigten FCs und FBs müssen vom Anwender geladen und in dem PLC-Programm des Maschinenherstellers (Maschinenprogramm) aufgerufen werden. Eine Übersicht der Bausteine FC, FB und DB getrennt nach Basisfunktion und erweiterter Funktion finden Sie im Kapitel "Inbetriebnahme PLC-Programm".

Nahtstelle: HMI / PLC

Der Datenaustausch HMI / PLC erfolgt über den integrierten CP, wobei der HMI immer der aktive Partner (Client) und die PLC immer der passive Partner (Server) ist. Die vom HMI angeforderten bzw. übergebenen Daten werden vom PLC-Betriebssystem aus dem Nahtstellenbereich HMI / PLC ausgelesen bzw. in ihn eingetragen (Zeitpunkt: Zykluskontrollpunkt). Aus PLC-Anwendersicht verhalten sich die Daten wie E/A-Signale.

Nahtstelle: MCP / PLC bzw. BHG / PLC (Anschluss: Ethernet)

Der Datenaustausch MCP / PLC, BHG (HT 2) / PLC erfolgt über den integrierten CP. Vom CP werden die MCP / BHG-Signale in das interne DPR (Dual Port RAM) der PLC abgelegt bzw. von ihm abgeholt. Auf PLC-Seite übernimmt das Grundprogramm den Austausch mit der Anwendernahtstelle. Über die Parameter des Grundprogramms (FB1, DB7) werden die Operandenbereiche (z. B. E/A-Bereiche) und die Anfangsadressen festgelegt.

Nahtstelle: MCP / PLC (Anschluss: PROFIBUS)

Der Datenaustausch MCP / PLC erfolgt über den PROFIBUS der PLC. Die E/A-Adressen der MCP sind in den Prozess-Abbildbereich der PLC über die HW-Projektierung in STEP 7 zu legen. Die Pointer Variablen MCP*In, MCP*Out sind auf die gleichen Adressen zu setzen. In MCP*BusAdr ist die eingestellte DP-Slave-Nummer einzutragen.

Nahtstelle: BHG / PLC (Anschluss: MPI)

Der Datenaustausch BHG / PLC erfolgt über die MPI-Nahtstelle der PLC. Es wird hierfür der Dienst "Kommunikation mit Globaldaten (GD)" verwendet (siehe hierzu STEP 7 Benutzerhandbuch). Das PLC-Betriebssystem übernimmt den Transfer der Signale von und zur Anwendernahtstelle. Über das STEP 7-Projektierwerkzeug "Communication-Configuration" werden neben den GD-Parametern auch die Operandenbereiche (z. B. E/A-Bereiche) und deren Anfangsadressen festgelegt.

14.8.4 Diagnosepuffer der PLC

Im Diagnosepuffer der PLC (auslesbar mit STEP 7) werden Diagnoseinformationen des PLC-Betriebssystems eingetragen.

14.9 Struktur der Nahtstelle

Nahtstellen-DBs

Aufgrund der Vielzahl der Signale zwischen NC und PLC ist die Abbildung in Nahtstellen-DBs notwendig. Aus PLC-Programmsicht sind dies globale DBs. Das Grundprogramm erzeugt diese DBs beim Systemanlauf anhand von aktuellen NC-Maschinendaten (Anzahl der Kanäle, Achsen etc.). Dies hat den Vorteil, dass nur soviel PLC-RAM belegt wird, wie für die aktuelle Maschinenkonfiguration benötigt wird.

14.9.1 Nahtstelle PLC/NCK

Allgemeines

Die Nahtstelle PLC / NC wird einerseits durch eine Datenschnittstelle und andererseits durch eine Funktionsschnittstelle gebildet. In der Datenschnittstelle sind Status- und Steuersignale, Hilfs- und G-Befehle enthalten, während über die Funktionsschnittstelle Aufträge von der PLC an den NC übergeben werden.

Datenschnittstelle

Die Datenschnittstelle wird in folgende Gruppen unterteilt:

- NC-spezifische Signale
- BAG-spezifische Signale
- Kanal-spezifische Signale
- Achs-/Spindel-/Antriebsspezifische Signale

Funktionsschnittstelle

Die Funktionsschnittstelle wird durch FBs und FCs gebildet. Das folgende Bild zeigt die generelle Struktur der Nahtstelle zwischen PLC und NC.

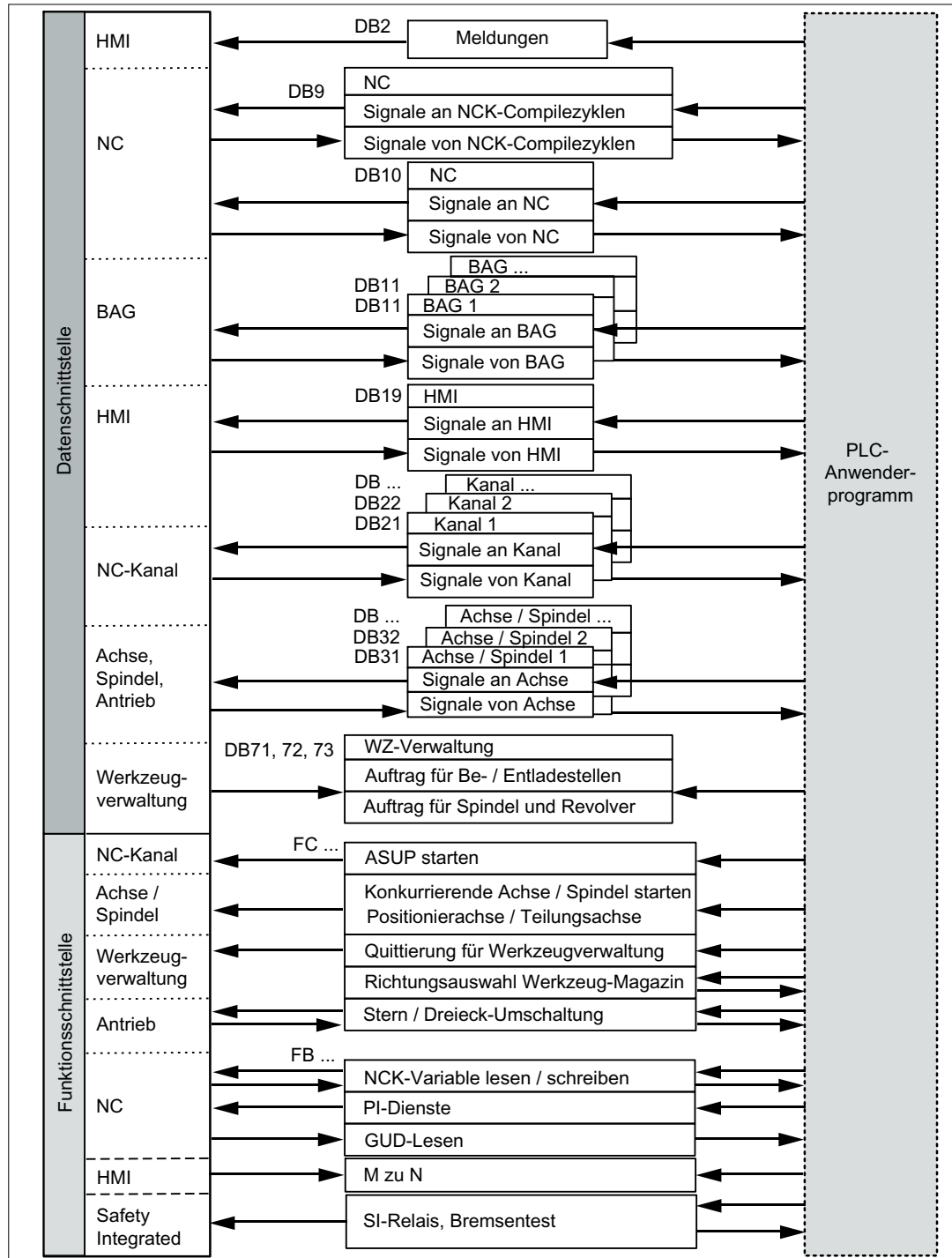


Bild 14-2 Anwendernahtstelle PLC / NC

Compile-Zyklen-Signale

Neben den standardmäßig vorhandenen Signalen zwischen PLC und NC wird bei Bedarf ein Nahtstellen-DB für Compile-Zyklen erzeugt (DB9). Die zugehörigen Signale, die abhängig von den jeweiligen Compile-Zyklen sind, werden zyklisch zu Beginn des OB1 übertragen. Die Übertragung erfolgt von niedriger nach höherer Adresse durch das Grundprogramm. Zuerst werden die Signale von PLC an NC übertragen, danach die Signale von NC an PLC. Notwendige Synchronisationen zwischen NC und PLC sind durch den Anwender (z. B. durch Semaphorentechnik) vorzusehen. Die Signalübertragungen erfolgen asynchron zwischen NC und PLC. Hierbei kann z. B. eine begonnene Datenübertragung des NC vom PLC unterbrochen werden. Damit sind die Daten unter Umständen nicht konsistent.

Signale PLC / NC

In die Gruppe der Signale von PLC an NC fallen:

- Signale für die Beeinflussung der digitalen und analogen E-/A-Signale der NC
- Schlüsselschalter- und Not-Halt-Signale

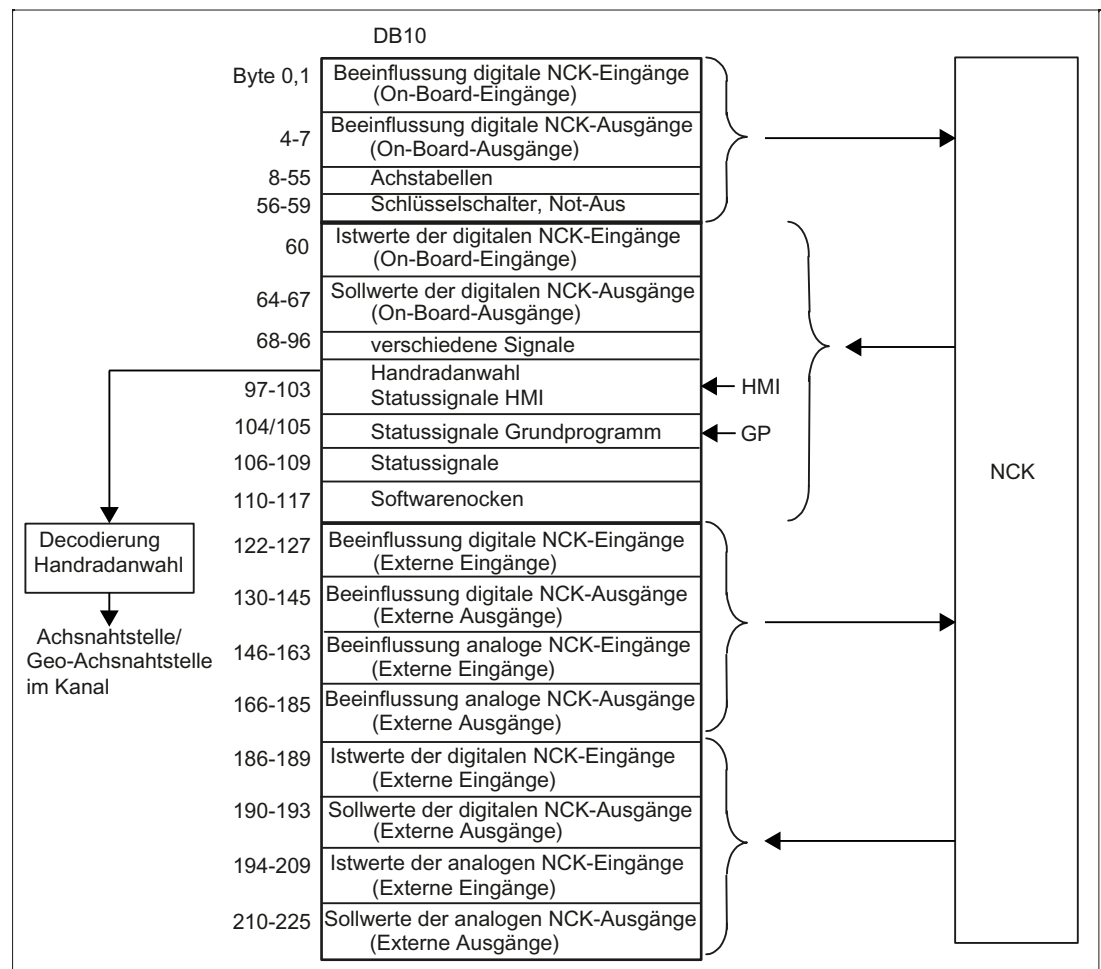


Bild 14-3 Nahtstelle PLC / NC

Signale NC / PLC

In die Gruppe der Signale von NC an PLC fallen:

- Istwerte der digitalen und analogen E/A-Signale der NC
- Bereitschafts- und Statussignale der NC

Weiterhin sind hier auch die Handradanwahlsignale und die Kanal-Statussignale vom HMI abgelegt.

Die Signale zur Handradanwahl werden vom Grundprogramm decodiert und in die maschinen- bzw. kanalachsspezifische Nahtstelle eingetragen.

Digitale/Analoge Ein-/Ausgänge des NC

Bei den digitalen und analogen Ein- und Ausgängen des NC ist Folgendes zu beachten:

Eingänge:

- Alle Eingangssignale bzw. Eingangswerte des NC werden auch an die PLC übergeben.
- Die Weitergabe der Signale an das NC-Teilprogramm kann von der PLC unterdrückt werden. Stattdessen kann ein Signal bzw. Wert von der PLC vorgegeben werden.
- Es kann auch dann ein Signal bzw. Wert von der PLC an den NC übergeben werden, wenn auf NC-Seite für diesen Kanal keine Hardware vorhanden ist.

Ausgänge:

- Alle auszugebenden Signale bzw. Werte werden auch an die PLC übergeben.
- Es können auch dann Signale bzw. Werte vom NC an die PLC übergeben werden, wenn auf NC-Seite für diesen Kanal keine Hardware vorhanden ist.
- Die vom NC vorgegebenen Werte können von der PLC überschrieben werden.
- Von der PLC aus können auch direkt Signale bzw. Werte über die NC-Peripherie ausgegeben werden.

Hinweis

Bei der Realisierung von digitaler und analoger NC-Peripherie sind die Informationen in folgender Dokumentation zu berücksichtigen:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Digitale und analoge NC-Peripherie (A4)

Signale PLC / BAG

Zur Betriebsartengruppe (BAG) des NC werden die von der Maschinensteuertafel oder vom HMI vorgegebenen Betriebsartensignale übertragen. Diese gelten für alle NC-Kanäle. Im NC können optional mehrere BAGs definiert sein.

Von der BAG wird deren aktueller Zustand an die PLC gemeldet.

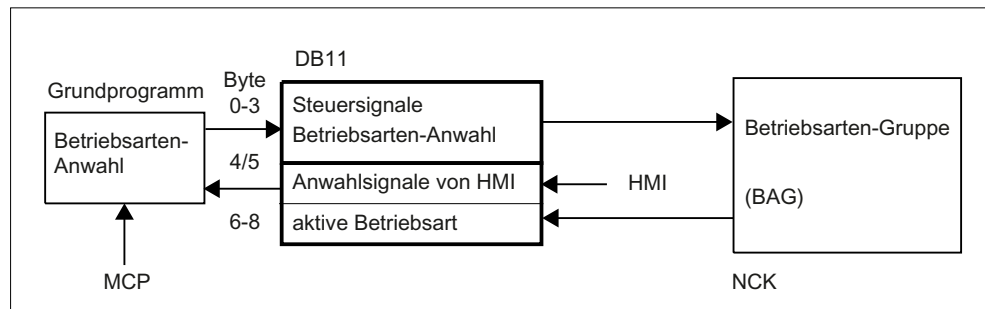


Bild 14-4 Nahtstelle PLC / BAG

Signale PLC / NC-Kanäle

Bei der Nahtstelle sind folgende Signalgruppen zu betrachten:

- Steuer- / Status-Signale
- Hilfs- / G-Befehle
- Signale der Werkzeugverwaltung
- NC-Funktionen

Die **Steuer- / Status-Signale** werden zyklisch am Anfang von OB1 übertragen. Auch die vom HMI in die kanalspezifische Nahtstelle eingetragenen Signale (der Eintrag der HMI-Signale erfolgt durch das Betriebssystem der PLC) werden zu diesem Zeitpunkt übertragen, wenn diese Signale nicht über die MCP, sondern über die HMI-Bedientafel vorgegeben werden.

Die **Hilfs- und G-Befehle** werden auf zweierlei Weise in die Nahtstellen-DBs eingetragen. Zunächst werden sie zusammen mit den Änderungssignalen eingetragen.

- Die **M-Signale** M00 - M99 (sie werden vom NC mit der erweiterten Adresse 0 übergeben) werden zusätzlich decodiert und die zugehörigen Nahtstellenbits für eine Zyklusdauer gesetzt.
- Bei den **G-Befehlen** werden nur die durch Maschinendaten ausgewählten Gruppen in den Nahtstellen-DB eingetragen.
- Die **S-Werte** werden zusätzlich zusammen mit den zugehörigen M-Signalen (M03, M04, M05) in die spindelspezifische Nahtstelle eingetragen. Ebenso werden achsspezifische Vorschübe in die entsprechende achsspezifische Nahtstelle eingetragen.

Bei aktivierter **Werkzeugverwaltung (Magazinverwaltung)** im NC wird die Belegung von Spindel bzw. Revolver und der Be-/Entladestellen in eigene Nahtstellen-DBs eingetragen (DB71 - 73).

Die **NC-Funktionen** werden über PLC-Funktionsaufrufe angestoßen und parametrisiert. Es stehen z. B. folgende Funktionsaufrufe zur Verfügung:

- Positionieren einer Linear- bzw. Rundachse
- Positionieren einer Teilungsachse
- Starten eines vorbereiteten asynchronen Unterprogramms (ASUP)
- Lesen/Schreiben von NC-Variablen
- Magazin- und Werkzeugbewegung aktualisieren

Die vorgenannten Funktionen sind zum Teil in eigenen Funktionsdokumentationen beschrieben.

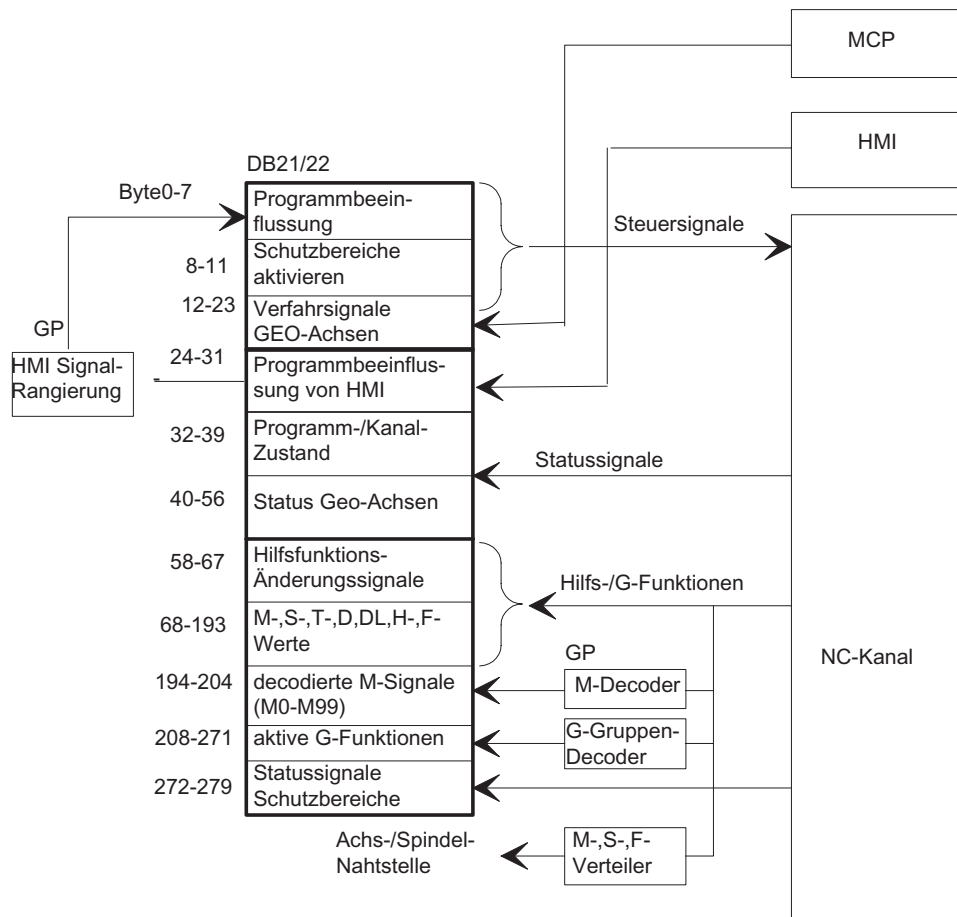


Bild 14-5 Nahtstelle PLC/NC-Kanal

Signale PLC / Achsen, Spindel, Antrieb

Die achs- und spindelspezifischen Signale sind in folgende Gruppen aufgeteilt:

- Gemeinsame Achs- / Spindelsignale
- Achssignale
- Spindelsignale
- Antriebssignale

Die Signale werden bis auf die im Folgenden beschriebenen Ausnahmen zyklisch am Anfang des OB1 übertragen.

Zu den Ausnahmen gehören:

- achsspezifischer F-Wert
- M-Wert
- S-Wert

Ein **achsspezifischer F-Wert** wird über den M-, S-, F-Verteiler des Grundprogramms dann eingetragen, wenn er im Zuge der NC-Programmbearbeitung an die PLC übertragen wird.

M- und S-Wert wird ebenfalls dann über den M-, S-, F-Verteiler des Grundprogramms eingetragen, wenn einer oder beide Werte zur Abarbeitung kommen.

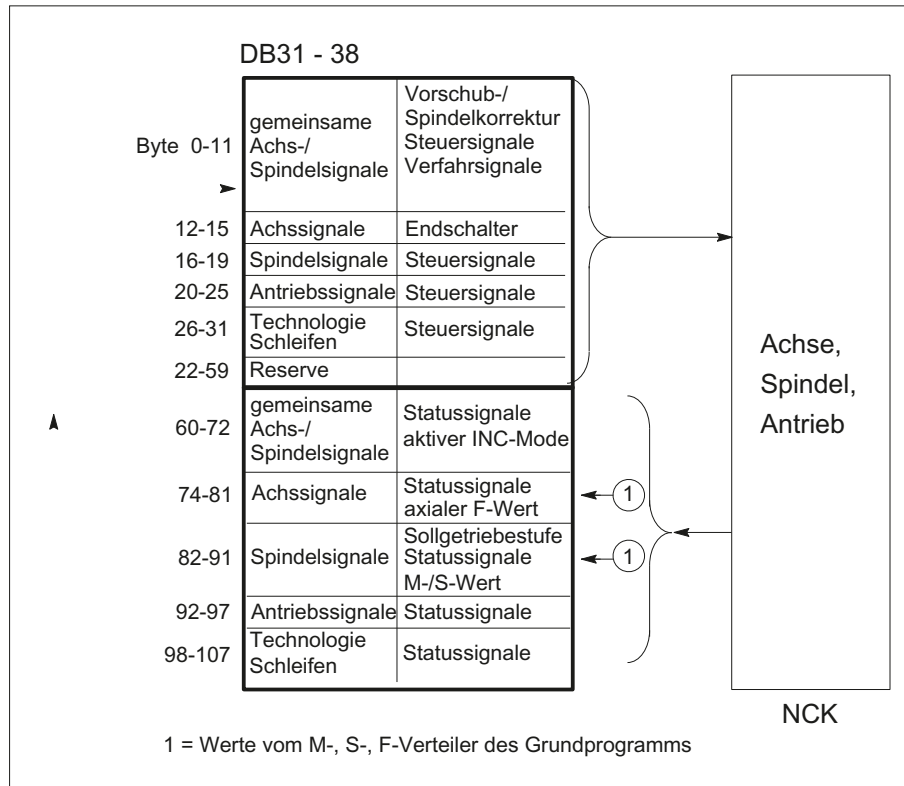


Bild 14-6 Nahtstelle PLC / Achsen, Spindeln, Antriebe

14.9.2 Nahtstelle PLC/HMI

Allgemeines

Bei der Nahtstelle PLC/HMI müssen folgende Funktionskomplexe betrachtet werden:

- Steuer-signale
- Maschinenbedienung
- PLC-Meldungen
- PLC-Status-Anzeige

Steuersignale

Bei den Steuersignalen handelt es sich um Signale, die u. a. von der Maschinensteuertafel vorgegeben werden und vom HMI berücksichtigt werden müssen. Zu diesen Signalen gehören z. B. Istwerte im MKS oder WKS anzeigen, Tastensperre usw. Diese Signale werden über einen eigenen Nahtstellen-DB (DB19) mit dem HMI ausgetauscht.

Maschinenbedienung

Alle Bedienhandlungen, die zu Aktionen an der Maschine führen, werden von der PLC kontrolliert. Im Normalfall werden diese über die Maschinensteuertafel (MCP) vorgenommen. Es ist jedoch auch möglich, ein Teil der Bedienhandlungen wahlweise vom HMI aus vorzunehmen wie z. B. Betriebsartenwahl.

Die vom HMI kommenden Bediensignale werden vom PLC-Betriebssystem direkt in die Nahtstellen-DBs eingetragen. Im Standardfall werden diese Bediensignale so vom Grundprogramm rangiert, dass - sofern eine gleichartige Bedienmöglichkeit von der MCP besteht - die Bedienung wahlweise vom HMI **oder** von der MCP vorgenommen werden kann. Bei Bedarf kann die Bedienung über HMI durch einen Parameter "MMCToIF" des FB1 vom Anwender ausgeschaltet werden.

PLC-Meldungen

Basis der Meldefunktionen sind die in das Betriebssystem der AS 300 integrierten Systemdiagnose-Funktionen, die folgende Eigenschaften besitzen:

- In einer **Diagnose-Zustandsliste** werden vom PLC-Betriebssystem alle wichtigen Systemzustände sowie Zustandsübergänge eingetragen. Weiterhin werden Kommunikationsereignisse und (Peripherie-) Baugruppen-Diagnosedaten (bei Baugruppen mit Diagnosefunktionen) eingetragen.
- Zusätzlich werden die Diagnose-Ereignisse, die zum Systemstopp führen, in einen **Diagnosepuffer** (Ringpuffer) in der zeitlichen Reihenfolge des Auftretens mit Uhrzeitstempel eingetragen.
- Die in den Diagnosepuffer eingetragenen Ereignisse werden automatisch an Bedien-/ Beobachtungssysteme (OP bzw. HMI) über die Bussysteme versandt, sofern diese sich als empfangsbereit angemeldet haben (Meldedienst). Die Übertragung an den angemeldeten Teilnehmer ist eine Funktion des Betriebssystems der PLC. Der Empfang und die Interpretation der Meldungen werden durch die HMI-Software sichergestellt.
- Vom PLC-Anwenderprogramm können über SFCs (System-Function-Calls) ebenfalls Meldungen in den Diagnosepuffer bzw. ALARM_S/ ALARM_SQ -Puffer eingetragen werden.
- Die Ereignisse werden in den Alarm-Puffer eingetragen.
Die zugehörigen Texte müssen im OP bzw. HMI vorgehalten werden.

Zusammen mit dem Grundprogramm wird ein FC zur Meldeerfassung (FC10) bereitgestellt, der die zu meldenden Ereignisse - in Signalgruppen eingeteilt - erfasst und über den Alarm-Puffer zum HMI meldet.

Die Struktur der Meldeerfassung ist in Bild "Erfassung und Meldung von PLC-Ereignissen" dargestellt. Sie weist folgende Merkmale auf:

- Die Bitfelder für Ereignisse, die die NC/PLC-Nahtstelle betreffen, sind zusammen mit den Bitfeldern für die Anwendermeldungen in dem DB2 zusammengefasst.
- Die Bitfelder werden mehrfach durch den FC10 ausgewertet.
 - **Auswertung 1; Erfassung von SammelSignalen**
Je Signalgruppe wird ein SammelSignal erzeugt, wenn mindestens ein Bit Signal "1" führt. Dieses Signal wird im Normalfall auf das zugehörige Sperrsignal der NC/PLC-Nahtstelle (bei Baugruppen mit Diagnosefunktionen) geführt. Die SammelSignale werden zyklisch komplett erfasst.
 - **Auswertung 2; Erfassung von Alarmmeldungen**
Es gibt eine feste Vorgabe, welche Signale einer Gruppe bei ihrem Wechsel von "0" nach "1" eine Alarmmeldung erzeugen.
 - **Auswertung 3; Erfassung von Betriebsmeldungen**
Es gibt eine feste Vorgabe, welche Signale einer Gruppe eine Betriebsmeldung erzeugen.
- Der Umfang der Anwender-Bitfelder (Bereich User) ist standardmäßig auf 10 Bereiche zu je 8 Byte festgelegt, kann jedoch über Grundprogramm-Parameter am FB1 bezüglich der Anzahl der Bereiche auf die Bedürfnisse des Maschinenherstellers angepasst werden.

Quittungskonzept

Bei den Fehler- und Betriebsmeldungen werden folgende Quittungskonzepte verfolgt:

Die **Betriebsmeldungen** zeichnen sich dadurch aus, dass mit ihnen normale Betriebszustände der Maschine als Information für den Bediener angezeigt werden sollen. Deshalb entfällt bei ihnen die Notwendigkeit von Quittungssignalen. Bei ihnen wird sowohl das Kommen als auch das Gehen des Ereignisses erfasst und ein Eintrag in die Diagnosezustandsliste vorgenommen. Der HMI erzeugt anhand der Kennungen "BM gekommen" und "BM gegangen" ein aktuelles Abbild der anstehenden Betriebsmeldungen.

Mit **Alarmmeldungen** werden Fehlerzustände an der Maschine angezeigt, die im Normalfall zum Maschinenstillstand führen. Treten mehrere Fehler "gleichzeitig" auf, ist es für die Fehlersuche wichtig, die Reihenfolge des Auftretens zu kennen. Dies wird einerseits durch die Reihenfolge des Eintrags in den Diagnosepuffer angezeigt und andererseits durch den Uhrzeitstempel, den jeder Eintrag erhält.

Verschwindet die Fehlerursache, so wird die zugehörige Alarmmeldung erst dann gelöscht, wenn eine Anwenderquittung vorliegt (z. B. Kundentaste an der MCP). Als Reaktion auf dieses Signal untersucht der FC "Meldungserfassung", welche der bereits gemeldeten Fehler verschwunden sind und trägt diese mit der Kennung "Alarm gegangen" in den Diagnosepuffer ein. Somit kann der HMI auch bei den Alarmmeldungen ein aktuelles Abbild der anstehenden Meldungen erzeugen. Bei den noch anstehenden Meldungen bleibt die Uhrzeit ihres Entstehens erhalten (im Gegensatz zu einer Neuabfrage).

STEP 7

Im SIMATIC Manager kann mit dem Menüpunkt "Zielsystem" > "CPU-Meldungen" ein Tool gestartet werden. Mit dem Tool können die Alarmer und Meldungen nummernmäßig angezeigt werden. Hierzu ist der Tabulator "Alarm" zu aktivieren und ein Haken im oberen Bildteil unter "A" zu setzen.

Anwenderprogramm

Das Anwender PLC-Programm muss im zyklischen Programmteil lediglich den Grundprogrammbaustein FC10 mit entsprechender Parametrierung aufrufen und die Bitfelder im DB2 setzen bzw. rücksetzen. Alles Weitere wird durch das Grundprogramm und den HMI erledigt.

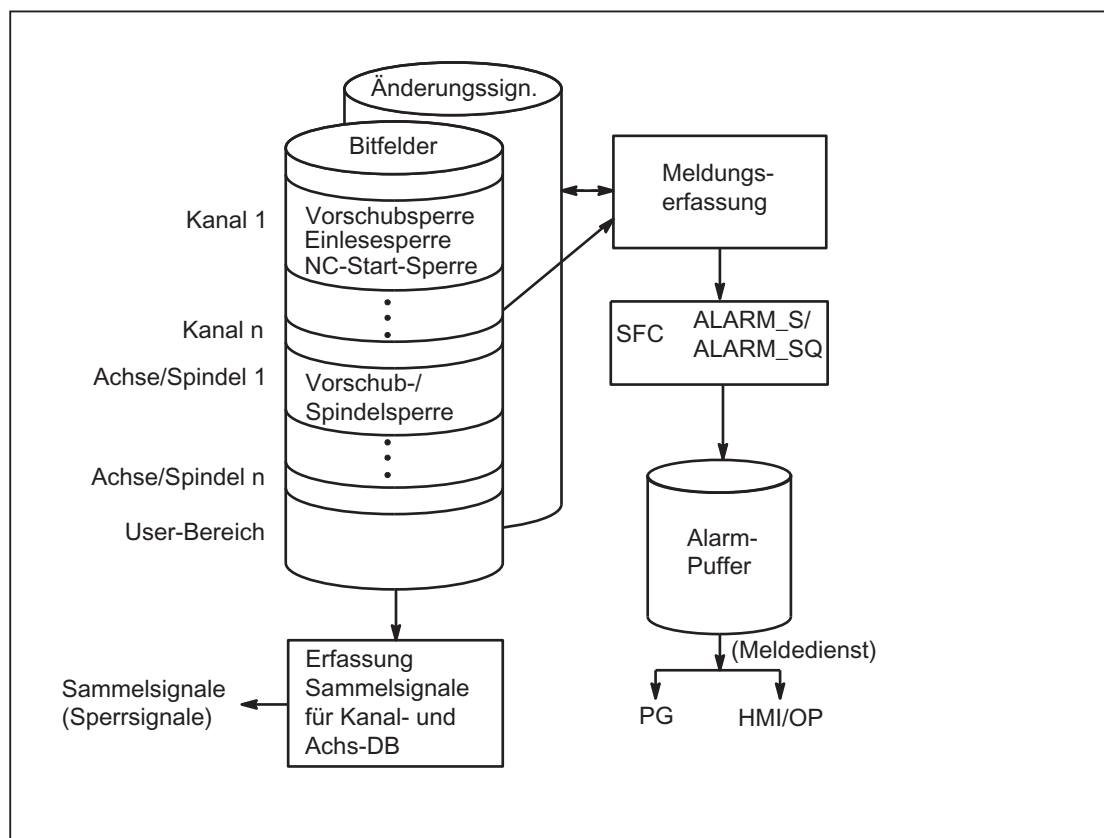


Bild 14-7 Erfassung und Meldung von PLC-Ereignissen

Erweiterungen der PLC-Alarmer über den Baustein FC10

Mit dem FB1-Parameter "ExtendAIMsg" wird eine Auswahl über den PLC-Alarmmechanismus getroffen.

Bei "ExtendAIMsg:= FALSE" ist das bisherige Verfahren des FC10 mit dem DB2 als Bitfeld-Datenbaustein aktiv. Es gelten die bekannten Einschränkungen bezüglich Kanal- und Achs-Anzahl.

Dagegen wird bei "ExtendAlMsg:= TRUE" die Erweiterung des FC10 aktiv. Der DB2 und der DB3 werden wie bisher angelegt. Der Anwender muss im DB2 die Bits setzen bzw. rücksetzen. Die Parametrierung über Meldung und Alarm und einer Parametrierung des Zahlenwerts der 2. Dekade der Useralarme sind im DB5 enthalten.

Die Erweiterungen sind:

- Unterstützung für 10 Kanäle, 31 Achsen, 64 Anwenderbereiche (die Anzahl der Anwenderbereiche ist am FB1-Parameter "MsgUser" einzutragen).
- Es stehen Bereiche für Vorschubhalt, Einlesesperre usw. ohne Meldungen zur Verfügung. Die Informationen aus diesem Bereich werden in Abhängigkeit vom FC10-Parameter "ToUserIF" zusammen mit den zugehörigen Meldungsbits als Sammelsignale auf die Nahtstelle im DB21, DB31 gelegt. Somit entfällt das bisher recht umständliche Handling dieser Signale.
- Die Alarme / Meldungen bekommen für den Userbereich 0 zusätzlich zur Alarmnummer auch 16 Bit Integer Zusatzwerte (%Z Parameter im Alarmtext). Die 16 Bit Integerwerte sind vom Anwender in den DB2 in die Array-Variable ZInfo0 parallel zum Setzen eines Alarmbits zu schreiben. Für jedes Bit im User Bereich 0 ist ein Integerwert vorhanden, siehe UDT1002 im Grundprogramm.
- Die Usermeldungen sind in der 2. Dekade der Meldenummer im Zahlenbereich 0 bis 9 parametrierbar. Der Anzeigewert der 2. Dekade muss vom Anwender in den DB5 in die Array-Variable UserDek2No geschrieben werden. Für jeden Userbereich kann eine Zahl festgelegt werden, siehe DB5 im Grundprogramm.
Standardmäßig ist der Wert 0 für die 2. Dekade vorbelegt.

Die Strukturierung des DB2 ist im UDT1002 (Grundprogramm) erkennbar. Der UDT1002 sollte bei den neuen Alarmfunktionen in der Symboltabelle dem DB2 symbolisch zugeordnet werden.

Am Beginn des DB2 liegen Bitfelder für Signale ohne Meldungserzeugung. Danach folgt ein 64 Integer großes Array für die Zusatzinfo des Userbereichs 0.

Anschließend folgen die Bereiche, die auch Alarme / Meldungen absetzen (siehe Listenhandbuch). Diese Bereiche sind ergänzt auf 10 Kanäle, 31 Achsen.

Einfache Umsetzung eines Anwenderprogramms auf die neuen Alarme

Im Source Container des Grundprogramms gibt es das File "udt2_for_Convert.awl", das folgende Strukturelemente aus UDT1002 enthält:

- ChanA als Array von 1 ... 8
- AxisA als Array von 1 ... 18
- UserA als Array von 0 ... 63

Dieser UDT2 ist über den KOP/FUP/AWL-Editor zu übersetzen. Der UDT2 muss in der Symboltabelle dem DB2 zugewiesen sein.

Von den Bausteinen, die Zuweisungen auf den DB2 haben, sollen Quellen erzeugt werden. Alternativ können natürlich auch von allen Bausteinen Quellen erstellt werden. Der UDT1002 muss nun noch in der Symboltabelle dem DB2 zugeordnet werden. Anschließend sind die Quellen neu zu übersetzen.

Jetzt sind alle Alarmzuweisungen auf die neuen Datenbereiche im DB2 zugeordnet und es muss nur noch der FB1-Parameter "ExtendAlMsg" auf TRUE gesetzt werden.

Nach einem Power-On-Reset ist das Alarmverhalten identisch wie bisher.

14.9.3 Nahtstelle PLC/MCP/BHG

Allgemeines

Es gibt verschiedene Anschlussmöglichkeiten für die Maschinensteuertafel (MCP) und das Bedienhandgerät (BHG). Dies resultiert teilweise aus der Historie der MCP und des BHG. In dieser Beschreibung ist vorrangig der Anschluss der Ethernet-Komponenten beschrieben.

Die Maschinensteuertafel (MCP) und das Bedienhandgerät (BHG) sind bei SINUMERIK 840D sl über den Ethernet-Bus angekoppelt, der auch die TCU mit der NCU verbindet. Dies hat den Vorteil, dass nur eine Busleitung zur Bedieneinheit verlegt werden muss.

Topologie SINUMERIK 840D sl

Bei 840D sl wird die Maschinensteuertafel und auch das Bedienhandgerät an den Ethernet-Bus des CP 840D sl angeschlossen (siehe folgendes Bild). Sollen weitere Tasten und Anzeigen für Kundenbedientafeln angeschlossen werden, so kann dies über ein weiteres Tastatur-Interface (MCP ohne Bedienteil) erfolgen. Je Tastatur-Interface können über Flachbandkabel 64 Taster, Schalter etc. und 64 Anzeigeelemente angeschlossen werden.

Die von der MCP kommenden Signale werden vom integrierten Ethernet des CP-840D sl ins DPR (Dual-Port-RAM) zur PLC kopiert. Das Grundprogramm der PLC trägt die ankommenden Signale in das am FB1 projektierte Eingangsabbild ein. Im Normalfall werden die NC-relevanten Signale vom Grundprogramm auf die NC/PLC-Nahtstelle verteilt. Bei Bedarf können die Signale vom Anwender modifiziert werden.

Die Signale von der PLC zur MCP (Anzeigen) nehmen den umgekehrten Weg.

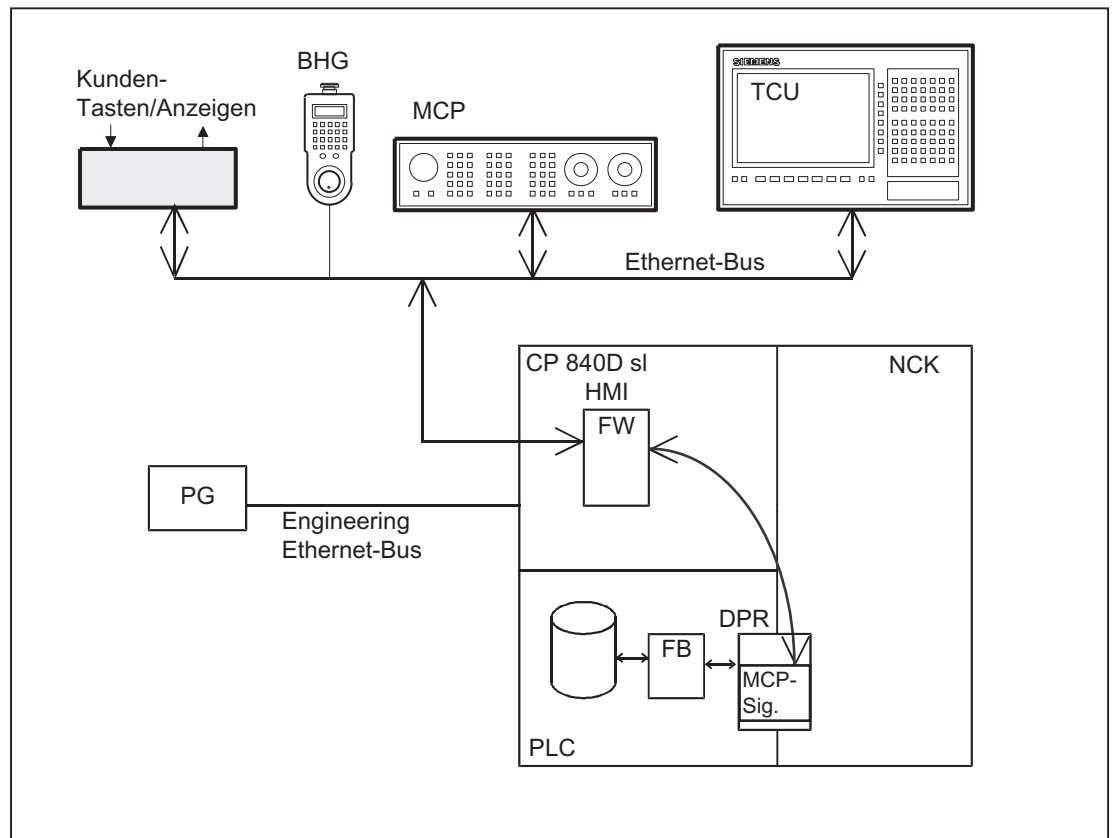


Bild 14-8 Ankopplung der Maschinensteuertafel bei 840D sl

Busadressen

Bei den Ethernet-Komponenten sind MAC- und IP-Adressen bzw. logische Namen für die Kommunikation bestimmend. Die Umsetzung von logischen Namen in die MAC bzw. IP Adressen erfolgt durch die Systemprogramme der Steuerung. In der PLC wird der numerische Teil des logischen Namens für die Kommunikation verwendet. Dieser numerische Teil wird vom Anwender über den Parameter "MCPxBusAdr" an den FB1 übergeben.

Der logische Name einer MCP oder BHG beginnt immer mit "DIP". Danach folgt eine Nummer, die der Schalterstellung der MCP-Komponente entspricht (z. B. DIP192, DIP17).

MCP-Nahtstelle in der PLC

Die Signale der Maschinensteuertafel werden standardmäßig über die E-/A-Nahtstelle in den PLC-Bereich geführt. Es ist dabei zwischen den NC- und den Maschinenspezifischen Signalen zu unterscheiden. Die NC-spezifischen Tastensignale werden standardmäßig vom FC19 (oder FC24, FC25, FC26 je nach MCP-Variante) auf die jeweilige BAG-, NC-, Achs- und Spindel-spezifische Nahtstelle verteilt. Umgekehrt werden die zugehörigen Statussignale auf die MCP-Nahtstelle rangiert. Der FC19 oder die weiteren oben erwähnten Bausteine müssen dafür im **Anwenderprogramm** aufgerufen werden.

Die Kundentasten, mit denen diverse Maschinenfunktionen ausgelöst werden können, müssen direkt vom Anwenderprogramm ausgewertet werden. Ebenso sind von diesem auch die zugehörigen Statussignale auf den Ausgabebereich für die LEDs zu rangieren.

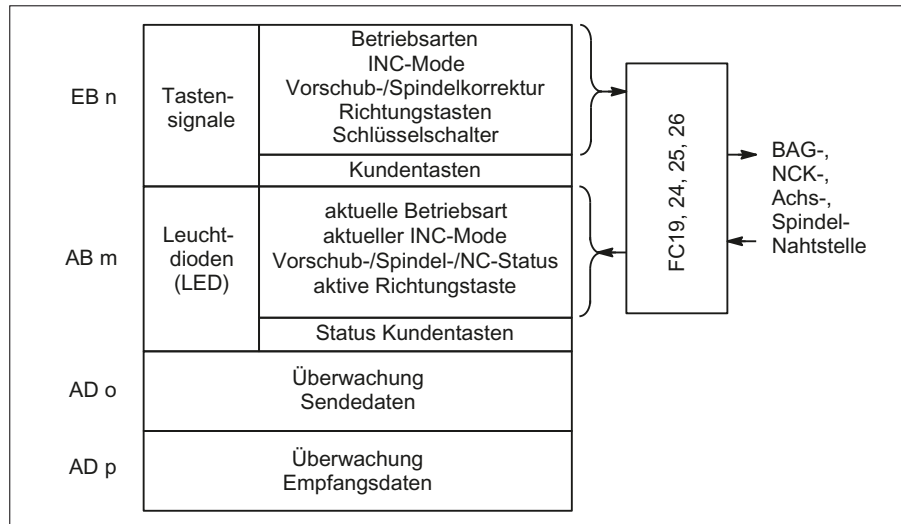


Bild 14-9 Nahtstelle von und zur Maschinensteuertafel

14.10 Struktur und Funktionen des Grundprogramms

Allgemeines

Das PLC-Programm ist modular aufgebaut. Die Organisationsbausteine (OB) bilden die Schnittstelle zwischen dem Betriebssystem und dem Grund- und Anwenderprogramm.

- Neustart (Warmstart) mit Anlauf und Synchronisation (OB100)
- Zyklischer Betrieb (OB1)
- Prozessalarme (OB40)
- Asynchrone Fehler: Diagnose-Alarm (OB82), Baugruppenausfall (OB86)

In den Organisationsbausteinen (OB) müssen vom Anwender die Aufrufe der Funktionsbausteine des Grund- und Anwenderprogramms programmiert werden.

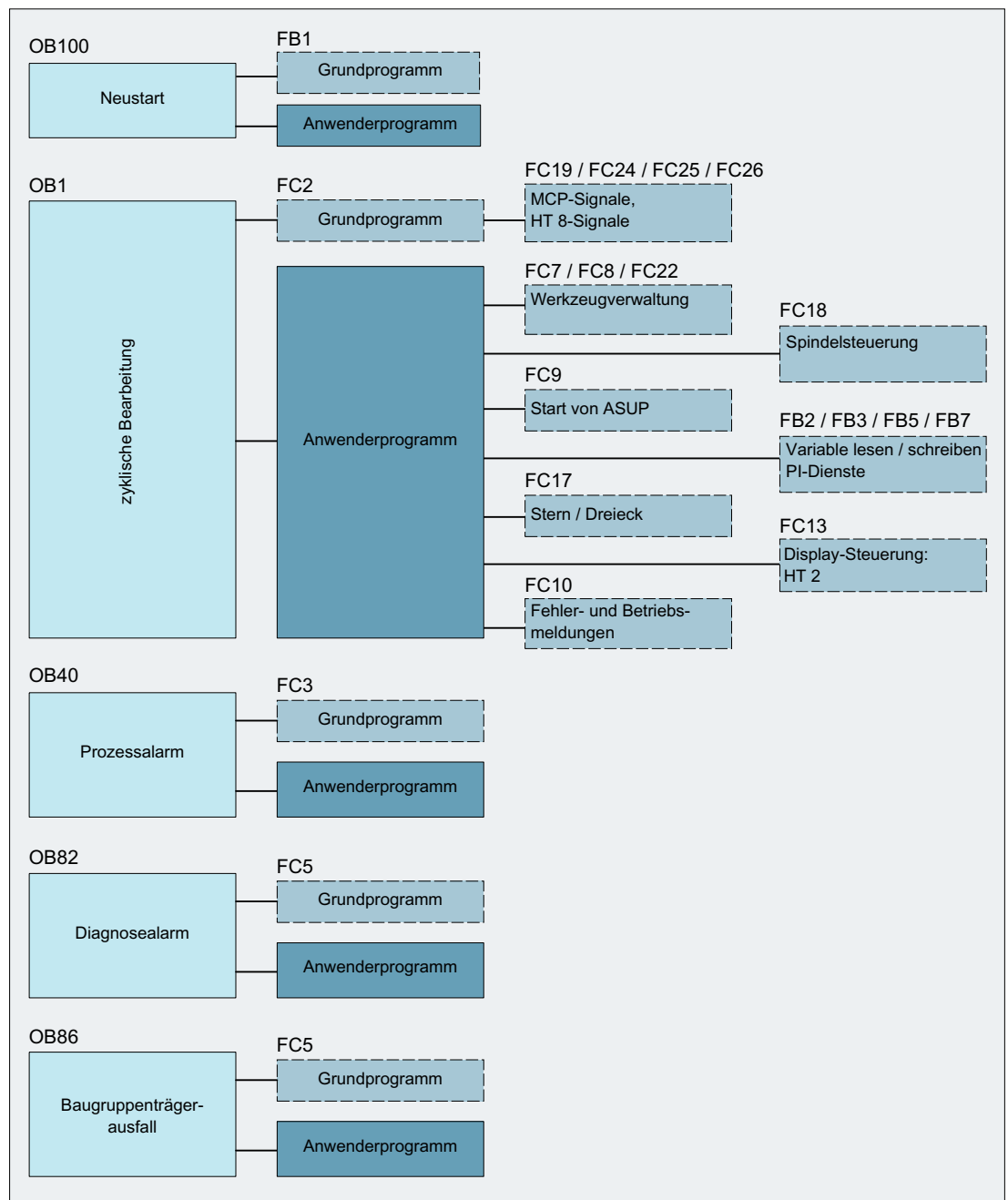


Bild 14-10 Struktur des Grundprogramms (Prinzip)

14.10.1 Anlauf und Synchronisation NCK-PLC

Laden des Grundprogramms

Das Laden des Grundprogramms mit dem S7-Tool muss im Stopp-Zustand der PLC erfolgen. Es wird so sichergestellt, dass alle Bausteine des Grundprogramms beim nächsten Anlauf richtig initialisiert werden. Es kann sonst zu undefinierten Zuständen der PLC kommen (z. B. Blinken aller PLC-LEDs).

Anlauf

Im Anlauf erfolgt die Synchronisation zwischen NC und PLC. Es werden die System- und Anwenderdatenbausteine auf Vollständigkeit und die wichtigsten GP-Parameter auf Plausibilität geprüft. Im Fehlerfall meldet das Grundprogramm einen Alarm (am HMI sichtbar) und überführt die PLC in den Stopp-Zustand.

Das Betriebssystem durchläuft nach der Systeminitialisierung den Organisationsbaustein OB100 und beginnt den zyklischen Betrieb immer am Anfang des OB1.

Synchronisation

Beim Hochlauf synchronisiert sich die PLC mit HMI, NC und CP.

Lebenszeichen

Nach ordnungsgemäßigem Anlauf und erstem vollständigen OB1-Zyklus (Grundstellungszyklus) tauschen PLC und NC fortlaufend Lebenszeichen aus. Bleibt das Lebenszeichen des NC aus, wird die PLC/NC-Nahtstelle initialisiert und das Signal "NC-CPU-ready" im DB10 wird auf FALSE gesetzt.

14.10.2 Zyklischer Betrieb (OB1)

Allgemeines

Die komplette Bearbeitung der NC/PLC-Schnittstelle erfolgt ausschließlich im zyklischen Betrieb. Das Grundprogramm läuft - zeitlich gesehen - vor der Bearbeitung des Anwenderprogramms. Um die Grundprogrammlaufzeit gering zu halten, werden nur die Steuer-/Statussignale zyklisch übertragen, die Hilfs- und G-Befehlssübergabe wird nur auf Anforderung bearbeitet.

Folgende Funktionen werden im zyklischen Teil des Grundprogramms ausgeführt:

- Übertragung der Steuer-/Statussignale
- Verteilung der Hilfsfunktionen
- M-Decodierung (M00 - M99)
- M-, S-, F-Verteilung

- Übertragung der MCP-Signale über NC
- Erfassung und Aufbereitung der Anwender-Fehler- und Betriebsmeldungen

Steuer-/ Statussignale

Gemeinsames Merkmal der Steuer- und Statussignale ist, dass es sich um Bitfelder handelt. Das Grundprogramm aktualisiert sie am Anfang des OB1.

Die Signale sind in folgende Gruppen eingeteilt:

- Allgemeine Signale
- BAG-spezifische Signale (wie z. B. Betriebsarten)
- Kanalspezifische Signale (wie z. B. Programm- und Vorschub-Beeinflussung)
- Achs- und Spindel-spezifische Signale (wie z. B. Vorschub-Sperre)

Hilfs- und G-Befehle

Die Hilfs- und G-Befehle weisen folgende Merkmale auf:

- Sie werden jeweils satzsynchron zur PLC übertragen (bezogen auf einen Teileprogrammsatz).
- Die Übergabe erfolgt quittungsgesteuert.
- Die Quittungszeiten beeinflussen unmittelbar die Satzausführungszeit von NC-Sätzen mit quittierungspflichtigen Hilfsfunktionen.

Der Wertebereich ist in folgender Tabelle dargestellt:

Funktion	Struktur		Wertebereich		Datentyp	
	1. Wert	2. Wert	1. Wert	2. Wert	1. Wert	2. Wert
G-Befehl		G-Befehl		255 ¹⁾		Byte
M-Wort	M-Gruppe	M-Wort	99	99.999.999	Word	DWord
S-Wort	Spindel-Nr.	S-Wort	6	Gleitpunkt ²⁾	Word	DWord
T-Wort	Magazin-Nr.	T-Wort	99	65535	Word	Word
D-Wort	-	D-Wort	99	255	Byte	Byte
H-Wort	H-Gruppe	H-Wort	99	Gleitpunkt	Word	DWord
F-Wort	Achs-Nr.	F-Wort	18	Gleitpunkt	Word	DWord

¹⁾ relative Nummer, die je G-Gruppe übergeben wird

²⁾ entsprechend STEP 7-Format (24 Bit Mantisse, 8 Bit Exponent)

Über den Hilfs-/G-Befehlsverteiler werden die vom NC übergebenen M-, S-, T-, H-, D-, F-Werte zusammen mit den zugehörigen Änderungssignalen auf die Nahtstelle **KANAL-DB** (siehe Listenhandbuch) ausgegeben. Dabei werden jeweils die beiden Werte der Hilfsfunktion in die entsprechenden Datenworte übertragen. Das dazugehörige Änderungssignal wird für einen PLC-Zyklus auf 1 aktiv geschaltet. Mit dem Rücksetzen des Änderungssignals erfolgt die Quittierung an den NC. Die Quittierung von schnellen Hilfsfunktionen erfolgt gleich nachdem das Grundprogramm die Hilfsfunktion erkannt hat durch das Grundprogramm.

Zusätzlich zur Verteilung der Hilfs- und G-Befehle erfolgt für ausgewählte Signale eine weitere Aufbereitung wie im Folgenden ausgeführt.

M-Decoder

Mit M-Funktionen können sowohl Schaltbefehle als auch Festpunkt-Werte übergeben werden. Für die Standard M-Funktionen (Bereich M00 - M99) werden ausdecodierte dynamische Signale auf die Nahtstelle **KANAL-DB** ausgegeben (Signaldauer = 1 Zykluszeit).

G-Gruppen-Decoder

Bei den vom NC übergebenen G-Befehle werden die zugehörigen Gruppen ausdecodiert und die jeweils aktuelle G-Nummer in das entsprechende Nahtstellenbyte des KANAL-DB eingetragen, d. h. in den Kanal-DBs sind alle aktiven G-Befehle eingetragen. Die eingetragenen G-Befehle bleiben auch nach NC-Programm-Ende oder Programm-Abbruch erhalten.

Hinweis

Bei Systemanlauf wird in alle G-Gruppenbytes der Wert "0" eingetragen.

M-, S-, F-Verteiler

Mit dem M-, S-, F-Verteiler werden spindelspezifische M-Worte M(1...6)=[3,4,5], S-Worte und F-Worte für axiale Vorschübe in die zugehörigen **Spindel- und Achs-DBs** eingetragen. Kriterium für die Verteilung ist die erweiterte Adresse, die bei M-Worten, S-Worten und axialen F-Worten an die PLC übergeben wird.

MCP-Signal-Übertragung

Die MCP-Signale werden abhängig vom Bus-Anschluss entweder direkt zur PLC oder indirekt über interne Verfahrensweisen mit Hilfe des Grundprogramms in die parametrisierten E/A Bereiche übertragen.

Anwendermeldungen

Die Erfassung und Aufbereitung der Anwender-Fehler- und Betriebsmeldungen erfolgt durch einen FC des Grundprogramms.

14.10.3 Zeitalarm Bearbeitung (OB35)

Für die Zeitalarmbearbeitung ist der **OB35** vom Anwender zu programmieren. Das Zeitraster des OB 35 ist im Standard auf 100 ms eingestellt. Durch das STEP 7-Tool "HW-Konfig" kann jedoch ein anderes Zeitraster gewählt werden. Allerdings darf eine Zeiteinstellung des OB35 nicht kleiner als etwa 3 ms verwendet werden, da es sonst zum Stopp der PLC-CPU kommt. Der Stopp wird verursacht durch das Lesen der Systemzustandsliste (SZL) des HMI bei dessen Hochlauf. Dieses Lesen bedeutet eine Blockierung der Ablaufebenen-Steuerung (AES) für ca. 2 ms. Dadurch wird der OB35 bei kleineren Zeiteinstellungen nicht mehr korrekt bearbeitet.

14.10.4 Prozessalarm Bearbeitung (OB40)

Ein Prozessalarm **OB40** (Interrupt) kann z. B. durch entsprechend projektierte Peripherie oder durch bestimmte NC-Funktionen ausgelöst werden. Wegen der unterschiedlichen Herkunft des Interrupts muss das PLC-Anwenderprogramm im OB40 zuerst die Interrupt-Ursache interpretieren. Die Interrupt-Ursache ist in den Lokaldaten des OB40 enthalten.

Literatur:

SIMATIC STEP 7-Beschreibung oder Online-Hilfe von STEP 7

14.10.5 Diagnosealarm, Baugruppenausfall Bearbeitung (OB82, OB86)

Allgemeines

Ein Diagnosealarm bzw. ein Baugruppenausfall von einer Peripherie-Baugruppe führt zum Aufruf des Grundprogramm-Bausteins OB82 bzw. OB86. Von diesem wird der Grundprogramm-Baustein FC5 "Diagnosealarm und Baugruppenausfall" (Seite 1064) aufgerufen.

Bus-Diagnose

Der Status der DP-Slave-Baugruppen an den PROFIBUS-Anschlüssen MPI/DP, DP1 bzw. dem PROFINET-Anschluss PN werden durch das Grundprogramm als Sammelsignal "Slaves OK" des jeweiligen Bussystems dem Anwenderprogramm gemeldet:

- DB10.DBX92.0 == 1 (MPI/DP Bus: Slaves OK)
- DB10.DBX92.1 == 1 (DP1-Bus: Slaves OK)
- DB10.DBX92.2 == 1 (PN-Bus: Slaves OK)

Das Sammelsignal wird aus dem LED-Status des jeweiligen Bussystems abgeleitet (Systemzustandsliste SZL 0x174) .

Alarmausgabe

Wird eine Störung oder Ausfall eines Slaves am jeweiligen Bussystem erkannt, wird der Alarm angezeigt:

- Alarm 400551 (MPI/DP Bus)
- Alarm 400552 (DP1-Bus)
- Alarm 400553 (PN-Bus)

Bei Wegfall der Störung wird der Alarm automatisch wieder gelöscht.

Unterdrückung der Alarmausgabe 400551, 400552, 400553

Durch Setzen eines der nachfolgenden Signale wird für das jeweilige Bussystem die Anzeige des Alarms unterdrückt:

- DB10.DBX92.4 = 1 (Unterdrückung Alarm 400551)
- DB10.DBX92.5 = 1 (Unterdrückung Alarm 400552)
- DB10.DBX92.6 = 1 (Unterdrückung Alarm 400553)

Die Unterdrückung der Alarmausgabe wirkt sobald das Signal gesetzt wird. Vor Auftreten des Fehlers oder während ein Fehler bereits ansteht.

Mit dem Rücksetzen eines Signals wird im Fehlerfall der entsprechende Alarm angezeigt.

Im Hochlauf der Steuerung werden die Signale zurückgesetzt.

Hinweis

Durch Setzen des Signals wird die Störungsüberwachung für den kompletten Busstrang deaktiviert!

14.10.6 Verhalten bei NCK-Ausfall

Allgemeines

Während des zyklischen Betriebs erfolgt eine ständige Überwachung der NC-Bereitschaft durch das PLC-Grundprogramm mittels Abfrage eines Lebenszeichens. Reagiert der NC nicht mehr, so wird die Nahtstelle NC/PLC initialisiert und das Signal **NC-CPU-Ready** im Bereich der **Signale von NC (DB10.DBX104.7)** wird zurückgesetzt. Des Weiteren werden die Signale, die vom NC an die PLC und von PLC an NC übergeben werden, in einen Grundzustand gesetzt.

Die PLC selbst bleibt aktiv, so dass weiterhin Maschinenfunktionen von ihr gesteuert werden können. Es bleibt aber dem Anwenderprogramm überlassen, einen sicheren Zustand der Maschine einzuleiten.

Signale NC → PLC

Bei den Signalen, die von NC an PLC übergeben werden, müssen folgende Gruppen unterschieden werden:

- Statussignale von NC, Kanälen, Achsen und Spindeln
- Änderungssignale der Hilfsfunktionen
- Werte der Hilfsfunktionen
- Werte der G-Befehle

Statussignale:

Die Statussignale von NC, den Kanälen, Achsen und Spindeln werden zurückgesetzt.

Änderungssignale der Hilfsfunktionen:

Die Änderungssignale der Hilfsfunktionen werden ebenfalls zurückgesetzt.

Werte der Hilfsfunktionen:

Die Werte der Hilfsfunktionen bleiben erhalten, so dass rekonstruierbar ist, welche Funktionen als letzte vom NC angestoßen wurden.

Werte der G-Befehle:

Die Werte der G-Befehle werden rückgesetzt (d. h. es wird jeweils der Wert 0 eingetragen).

Signale PLC → NC

Bei den Signalen, die von der PLC an den NC übergeben werden, wird zwischen Steuersignalen und Aufträgen, die per FCs an den NC übergeben werden, unterschieden.

Steuersignale:

Die Steuersignale von PLC an NC werden eingefroren; die zyklische Aktualisierung durch das PLC-Grundprogramm wird unterbrochen.

Aufträge von PLC an NC:

Die FCs und FBs, mit denen Aufträge an den NC übergeben werden, dürfen durch das PLC-Anwenderprogramm nicht mehr bearbeitet werden, da unter Umständen fehlerhafte Rückmeldungen auftreten können. Beim Hochlauf der Steuerung muss im Anwenderprogramm auch solange gewartet werden mit einer Auftragsaktivierung (z. B. NC-Daten lesen), bis das Signal **NC-CPU ready** gesetzt ist.

14.10.7 Funktionen des Grundprogramms mit Aufruf vom Anwenderprogramm

Allgemeines

Zusätzlich zu den Modulen des Grundprogramms, die am Anfang vom OB1, OB40 und OB100 aufzurufen sind, werden Funktionen bereitgestellt, die an geeigneter Stelle im Anwenderprogramm aufgerufen und parametrisiert werden müssen.

Mit diesen Funktionen können z. B. folgende Aufträge von PLC an den NC übergeben werden:

- Verfahren von konkurrierenden Achsen (FC18)
- Starten asynchroner Unterprogramme (ASUPs) (FC9)
- Anwahl von NC-Programmen (FB4)

- Steuern der Spindel (FC18)
- Lesen/Schreiben von Variablen (FB2, FB3)

Hinweis

Kontrolle und Diagnose eines Funktionsaufrufs des PLC-Grundprogramms

Zur Vereinfachung der Kontrolle und Diagnose eines Funktionsaufrufs (FB oder FC) des PLC-Grundprogramms, die über einen Anstoß (z. B. über Parameter Req, Start) gesteuert werden und die als Ausgangsparameter eine Ausführungsquittierung (z. B. über Parameter Done, NDR, Error) liefern, ist wie folgt vorzugehen.

Es sollte eine von anderen Signalen zusammengefasste Variable gesetzt werden, die den Anstoß für den Funktionsaufruf herbeiführen. Das Rücksetzen der Startbedingungen darf nur von den Parametern Done, NDR, Error abgeleitet werden.

Das entsprechende Steuerwerk kann entweder vor oder hinter dem Funktionsaufruf platziert werden. Wenn das Steuerwerk hinter dem Aufruf platziert ist, können die Ausgangsvariablen als lokale Variablen definiert sein (Vorteil: Reduzierung der globalen Variablen Merker, Datenvariable und zeitliche Vorteile gegenüber Datenvariable).

Der Anstoßparameter muss eine globale Variable sein (z. B. Merker, Datenvariable).

Im OB100 sind vom Anwenderprogramm noch aktive Aufträge zurückzusetzen (Parameter Req, Start, etc. von TRUE \Rightarrow FALSE). Durch ein Aus- und Wiedereinschalten kann ein Zustand mit noch aktivierten Aufträgen entstehen.

Konkurrierende Achsen

Konkurrierende Achsen zeichnen sich durch folgende Eigenschaften aus:

- Sie müssen als solche per NC-Maschinendatum definiert sein.
- Sie können entweder von der PLC oder von NC mittels JOG-Tasten verfahren werden.
- Der Start von der PLC ist in den NC-Betriebsarten MDA und AUTOMATIK per FC möglich.
- Der Start erfolgt unabhängig von NC-Satzgrenzen.

Es stehen FCs für Positionierachsen, Teilungsachsen und Spindeln (FC18) zur Verfügung.

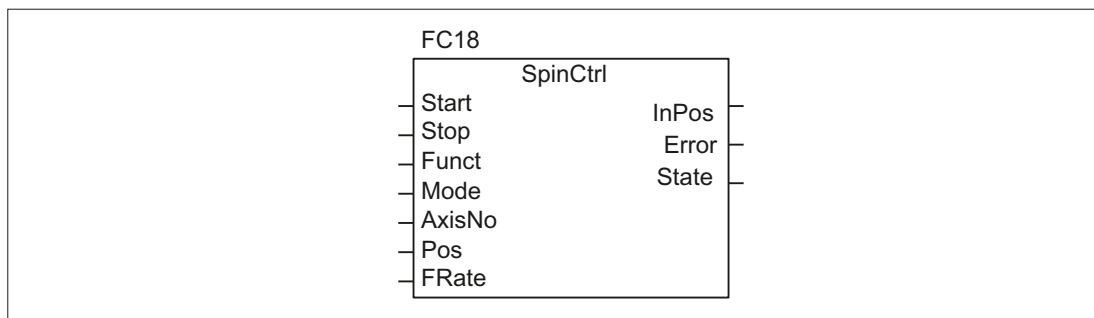
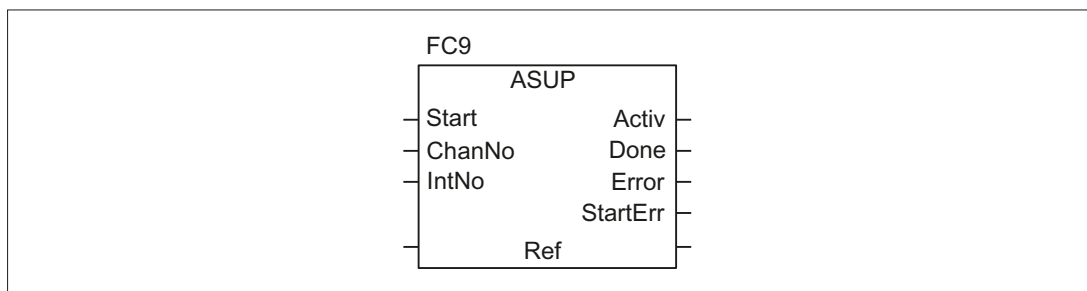


Bild 14-11 FC18 Ein-/Ausgabe-Parameter

Asynchrone Unterprogramme (ASUP)

Mit ASUP können beliebige Funktionen in der NC ausgelöst werden. Voraussetzung dafür, dass ein ASUP von der PLC aus gestartet werden kann, ist dessen Existenz und Vorbereitung vom NC-Programm bzw. durch FB4 PI-Dienste (ASUP).

Ein derart vorbereitetes ASUP kann von der PLC zu einem beliebigen Zeitpunkt gestartet werden. Das in einem parametrisierten Kanal des FC9 laufende NC-Programm wird durch das ASUP unterbrochen. Der Start eines ASUP erfolgt durch Aufruf des FC9 vom Anwenderprogramm durch den Parameter Start = 1.



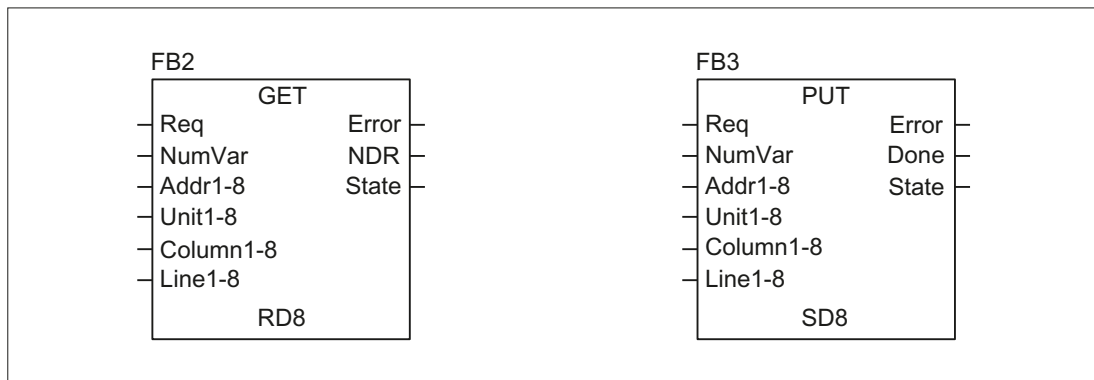
Hinweis

Ist ein ASUP noch nicht von einem NC-Programm oder vom FB4 (ASUP) vorbereitet worden (z. B. keine Interrupt-Nr. vergeben), wird Start-Fehler (StartErr = TRUE) gemeldet.

NC-Variable lesen/schreiben

Mit dem FB GET können Variable des NC gelesen und mit dem FB PUT können Werte in NC-Variable eingetragen werden. Die NC-Variablen werden über Bezeichner an den Eingängen Addr1 bis Addr8 adressiert. Die Bezeichner (Symbole) zeigen auf Adressangaben, die in einem globalen DB hinterlegt sein müssen. Zur Erzeugung dieses DB wird zusammen mit dem Grundprogramm eine PC-Software (NC-Var-Selector) ausgeliefert, mit der aus einer ebenfalls mitgelieferten Tabelle die gewünschten Variablen selektiert werden können. Die selektierten Variablen werden zunächst in einer zweiten, projektabhängigen Liste gesammelt. Mit dem Kommando **DB erzeugen** wird eine "*.AWL"-Datei erzeugt, die in die Programmdatei für die betreffende Maschine einzubinden und mit dem Maschinenprogramm zusammen zu kompilieren ist.

Es können 1 bis 8 Werte mit einem Lese- oder Schreibauftrag gelesen bzw. beschrieben werden. Die Werte werden - wenn notwendig - konvertiert [z. B. werden die NC-Gleitpunktwerte (64 Bit) in das PLC-Format (32 Bit mit 24 Bit Mantisse und 8 Bit Exponent) gewandelt und umgekehrt]. Durch die Wandlung von 64 Bit auf 32 Bit REAL geht Genauigkeit verloren. Die max. Genauigkeit der 32 Bit REAL-Zahlen liegt etwa bei 10 hoch 7.



Funktionen AG_SEND / AG_RECV

Die Funktionen AG_SEND / AG_RECV entsprechen den Funktionen der Bibliothek "SIMATIC_NET_CP" der S7-300 CPU in STEP 7. Generell gilt für diese Funktionen die Online-Hilfe zu diesen Funktionen.

Die Funktionen AG_SEND / AG_RECV können über die integrierte "CP 840D sl" zum Datenaustausch mit einer weiteren Station genutzt werden. Eine Beschreibung der Funktionen ist im Kapitel "Bausteinbeschreibungen (Seite 980)" zu finden.

Hinweis

Weitere Kommunikationsbausteine (z. B. BSEND, USEND), die über eine CP343-1 verfügbar sind, werden bei der SINUMERIK 840D sl nicht unterstützt.

14.10.8 Symbolische Programmierung des Anwenderprogramms mit Nahtstellen-DB

Allgemeines

Hinweis

Auf der Liefer-CD des der Toolbox 840D sl in der Bibliothek des Grundprogramms werden die Dateien NST_UDTB.AWL und TM_UDTB.AWL mitgeliefert.

In dem CPU-Programm des Grundprogramms sind die kompilierten UDT-Bausteine aus diesen beiden Dateien abgelegt.

Ein UDT ist ein vom Anwender definierter Datentyp, der z. B. einem Datenbaustein zugeordnet werden kann, der in der CPU erzeugt wurde.

In diesen UDT-Bausteinen sind symbolische Namen nahezu aller Nahtstellensignale definiert.

Verwendet werden die UDT-Nummern 2, 10, 11, 19, 21, 31, 71, 72, 73, 77, 1002, 1071, 1072, 1073.

Folgende Zuordnung ist hierbei getroffen worden:

UDT-Zuordnungen		
UDT-Nummer	Zuordnung Nahtstellen-DB	Bedeutung
UDT2	DB2	Alarmer / Meldungen
UDT10	DB10	NCK-Signale
UDT11	DB11	BAG-Signale
UDT19	DB19	HMI-Signale
UDT21	DB21 bis DB30	Kanal-Signal
UDT31	DB31 bis 61	Achs/Spindel-Signale
UDT71	DB71	Werkzeugverwaltung: Be-/Endladestellen
UDT72	DB72	Werkzeugverwaltung: Wechsel in Spindel
UDT73	DB73	Werkzeugverwaltung: Wechsel in Revolver
UDT77	DB77	MCP- und BHG-Signale mit Standard SDB210
UDT1002	DB2	erweiterte Alarmer / Meldungen (FB1-Parameter "ExtendAIMsg:=TRUE")
UDT 1071	DB 1071	Werkzeugverwaltung: Be-/Entladestellen (Multitool)
UDT 1072	DB 1072	Werkzeugverwaltung: Wechsel in Spindel (Multi-tool)
UDT 1073	DB 1073	Werkzeugverwaltung: Wechsel in Revolver (Multi-tool)

Für eine symbolische Programmierung der Nahtstellensignale müssen zuerst die Datenbausteine der Nahtstelle mit dem Symboleditor symbolisch zugeordnet werden.

Hierzu wird z. B. das Symbol "AchseX" dem Operand DB31 mit dem Datentyp UDT31 in der Symboldatei zugeordnet.

Nach dieser Eingabe kann die Programmierung des STEP 7-Programms für diese Nahtstelle symbolisch erfolgen.

Hinweis

Bisher erstellte Programme, die mit diesen Nahtstellen-DBs arbeiten, können auch in diese symbolische Form umgesetzt werden. Hierzu ist dann im bisher erstellten Programm eine voll qualifizierte Anweisung beim Datenzugriff z. B. "U DB31.DBX60.0" (Spindel / Rundachse) notwendig. Dieser Befehl wird bei Einschaltung der Symbolik im Editor in "AchseX.E_SpKA" umgesetzt.

Beschreibung

In den beiden AWL-Dateien NST_UDTB.AWL und TM_UDTB.AWL sind abgekürzte symbolische Namen der Nahtstellensignale definiert.

Um den Bezug auf die Namen der Nahtstellensignale herzustellen, ist hinter jedem Signal der Name im Kommentar mit aufgeführt.

Die Namen sind englischsprachig abgeleitet. Die Kommentare sind englischsprachig.

Über den STEP 7-Editor können beim Aufschlagen des UDT-Bausteins die symbolischen Namen, Kommentare und Absolutadressen sichtbar gemacht werden.

Hinweis

Die nicht verwendeten Bits und Bytes werden z. B. mit der Bezeichnung "f56_3" aufgeführt:

- "56": Byteadresse des jeweiligen Datenbausteins
 - "3": Bit-Nummer in diesem Byte
-

14.10.9 M-Dekodierung nach Liste

Funktionsbeschreibung

Über die Funktion "M-Dekodierung nach Liste" können bis zu 256 M-Funktionen mit erweiterter Adresse vom Grundprogramm dekodiert werden. Die Aktivierung der Funktion erfolgt über den FB1-Parameter "ListMDecGrp" (Anzahl M-Gruppen zur Dekodierung). Die Zuordnung einer M-Funktion mit erweiterter Adresse zu einem Signal in der Signalliste erfolgt über die Dekodierliste. Dabei erfolgt auch die Einteilung in Gruppen.

Dekodierliste (DB75)

Die Quelldatei für die Dekodierliste (MDECLIST.AWL) wird mit dem Grundprogramm ausgeliefert. Nach der Übersetzung der AWL-Quelle entsteht der Datenbaustein DB75. Vor Aktivierung der Funktion muss die Dekodierliste (DB75) in das AG übertragen und ein Neustart durchgeführt werden.

Eine M-Funktion wird dekodiert, wenn sie in der Dekodierliste enthalten ist. Mit Dekodierung der M-Funktion wird gruppenspezifisch das entsprechende Signal in der Signalliste gesetzt. Mit dem Setzen eines Signals in der Signalliste wird durch das Grundprogramm im zugehörigen Kanal der NC das Nahtstellensignal "Einlesesperre" gesetzt. Das Nahtstellensignal wird für den Kanal wieder zurückgesetzt, sobald vom Anwender alle von diesem Kanal ausgegebenen Signale in der Signalliste zurückgesetzt d.h. quittiert worden sind.

Signalliste (DB76)

Die Signalliste wird vom Grundprogramm bei Aktivierung der Funktion im Datenbaustein DB76 eingerichtet. Ab dann wird für jedes nach Liste dekodierte M-Signal in der Signalliste (DB76) in der entsprechenden Gruppe ein Signal gesetzt. Gleichzeitig wird in dem Kanal, in dem die M-Funktion ausgegeben wurde, das Nahtstellensignal "Einlesesperre" gesetzt. Das Nahtstellensignal wird für den Kanal wieder zurückgesetzt, sobald vom Anwender alle von diesem Kanal ausgegebenen Signale in der Signalliste zurückgesetzt d.h. quittiert worden sind.

Schnelle Hilfsfunktionen

Bei Ausgabe einer in der Dekodierliste enthaltenen M-Funktion als "schnelle Hilfsfunktion" wird keine Einlesesperre für den entsprechenden Kanal der NC gesetzt.

Folgendes Bild stellt die Struktur der **M-Dekodierung nach Liste** dar:

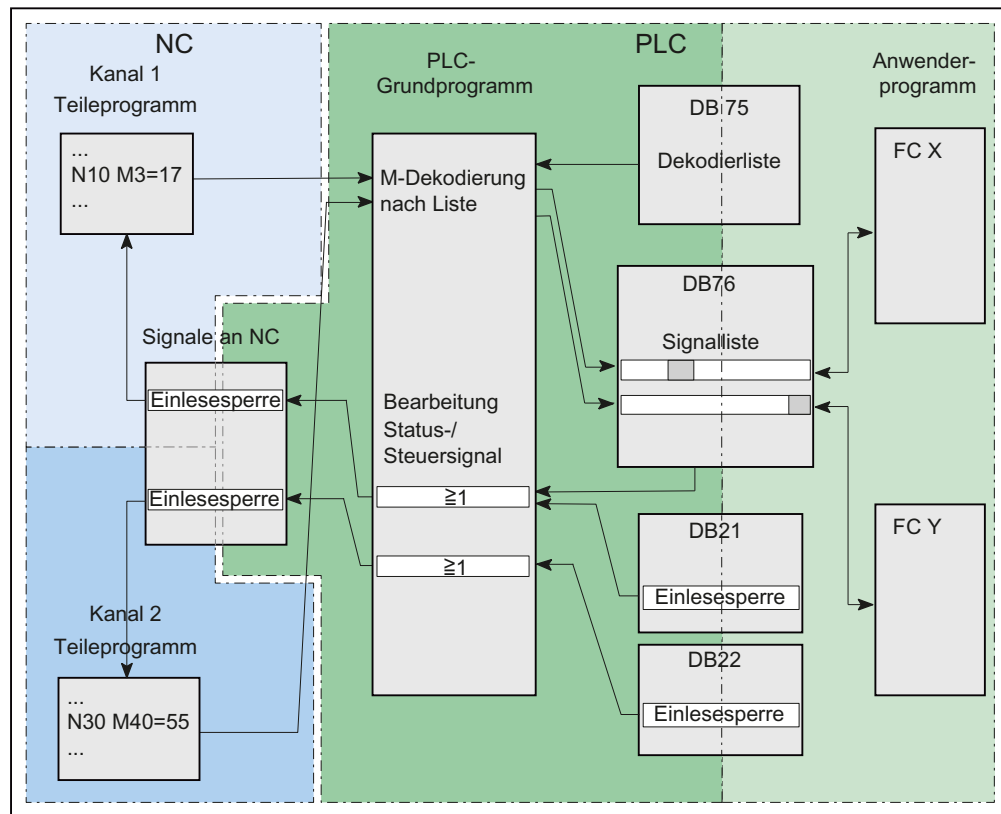


Bild 14-12 M-Dekodierung nach Liste

Aktivierung

Die Aktivierung der M-Dekodierung erfolgt über den FB1-Parameter "**ListMDecGrp**"

Über den Parameter wird die Anzahl der auszuwertenden bzw. zu dekodierenden M-Gruppen angegeben. Bei einem Parameterwert = 1 ... 16 wird die Funktion aktiv.

- Grundprogramm, OB100, FB1-Parameter **ListMDecGrp** = <Anzahl M-Gruppen> (siehe auch " FB1: RUN_UP - Grundprogramm, Anlaufteil (Seite 980) ").

Eigenschaften und Aufbau der Dekodierliste (DB75)

Eigenschaften der Dekodierliste (DB75):

- Die Dekodierliste existieren kanalunabhängig nur ein Mal.
- Die Dekodierliste kann maximal 16 Gruppen enthalten.
- Eine Gruppe umfasst maximal 16 Signale

- Für jede zu dekodierende Gruppe von M-Funktionen muss ein Eintrag in der Dekodierliste enthalten sein
- Die Zuordnung einer M-Funktion mit erweiterter Adresse zum zu setzenden Signal in der Signalliste wird in der Dekodierliste über die erste und letzte M-Funktion der zugehörigen Gruppe angegeben.
 - erste M-Funktion: Parameter: "MFirstAdr" \triangleq Signal bzw. Bit 0
 - letzte M-Funktion: Parameter: "MLastAdr" \triangleq abhängig von der Differenz zur ersten M-Funktion maximal Signal bzw. Bit 15

Aufbau der Dekodierliste (DB75):

Ein Eintrag in der Dekodierliste besteht aus 3 Parametern, die jeweils einer Gruppe zugeordnet sind:

Gruppe	Erweiterte M-Adresse	erste M-Adresse der Gruppe	letzte M-Adresse der Gruppe
1	MSigGrp[1].MExtAdr	MSigGrp[1].MFirstAdr	MSigGrp[1].MLastAdr
2	MSigGrp[2].MExtAdr	MSigGrp[2].MFirstAdr	MSigGrp[2].MLastAdr
...
16	MSigGrp[16].MExtAdr	MSigGrp[16].MFirstAdr	MSigGrp[16].MLastAdr

Typ und Wertebereich der Signale:

Signal	Typ	Wertebereich	Bedeutung
MExtAdr	INT	0 bis 99	Erweiterte M-Adresse
MFirstAdr	DINT	0 bis 99.999.999	erste M-Adresse in Gruppe
MLastAdr	DINT	0 bis 99.999.999	letzte M-Adresse in Gruppe

Eigenschaften der Signalliste (DB76)

Die Signalliste (DB76) hat folgende Eigenschaften:

- Die Signalliste existieren kanalunabhängig nur ein Mal.
- Die Signalliste kann maximal 16 Signalen pro M-Gruppe enthalten.

Beispiel

Es sollen 3 Gruppen von M-Funktionen dekodiert werden:

- Gruppe 1: M2 = 1 bis M2 = 5
- Gruppe 2: M3 = 12 bis M3 = 23
- Gruppe 3: M40 = 55

Aufbau der Dekodier- und Signalliste

Gruppe	Dekodierliste (DB75)			Signalliste (DB76)
	Erweiterte M-Adresse	Erste M-Adresse der Gruppe	Letzte M-Adresse der Gruppe	
1	2	1	5	DB76.DBX0.0 ... DBX0.4
2	3	12	23	DB76.DBX2.0 ... DBX3.3
3	40	55	55	DB76.DBX4.0

Programmcode

```
DATA_BLOCK DB 75
TITLE =
VERSION : 0.0
STRUCT
    MSigGrp : ARRAY [1 .. 16 ] OF STRUCT
        MExtAdr : INT;
        MFirstAdr : DINT;
        MLastAdr : DINT;
    END_STRUCT ;
BEGIN
    MSigGrp[1].MExtAdr      :=      2; Erweiterte M-Adresse der 1. Gruppe
    MSigGrp[1].MFirstAdr   :=   L#1; Erste M-Adresse der Gruppe
    MSigGrp[1].MLastAdr    :=   L#5; Letzte M-Adresse der Gruppe
    MSigGrp[2].MExtAdr      :=      3; Erweiterte M-Adresse der 2. Gruppe
    MSigGrp[2].MFirstAdr   :=   L#12; Erste M-Adresse der Gruppe
    MSigGrp[2].MLastAdr    :=   L#23; Letzte M-Adresse der Gruppe
    MSigGrp[3].MExtAdr      :=     40; Erweiterte M-Adresse der 3. Gruppe
    MSigGrp[3].MFirstAdr   :=   L#55; Erste M-Adresse der Gruppe
    MSigGrp[3].MLastAdr    :=   L#55; Letzte M-Adresse der Gruppe
END_DATA_BLOCK
```

Aufbau des FB1 im OB100

Zur Aktivierung der Funktion den Parameter für die Anzahl der zu dekodierenden M-Gruppen "ListMDecGrp" einfügen.

```
Call FB 1, DB 7(
    ...
    ListMDecGrp := 3;      // M-Dekodierung von 3 Gruppen
    ...
);
```

Beschreibung

Nachdem der Eintrag im OB100 angefügt ist und die Dekodierliste (DB75) in das AG übertragen wurde, muss ein Neustart durchgeführt werden. Das Grundprogramm richtet im anschließenden Neuanlauf die Signalliste (DB76) ein.

Anschließend wird ein NC-Programm z.B. im 1. Kanal gestartet. In diesem ist eine erweiterte M-Funktion (M3=17) enthalten. Mit Dekodierung der M-Funktion (M3 \triangleq Gruppe 2) wird in der Signalliste (DB76) das zugehörige Signal (DBW1.5) und im 1. Kanal das Nahtstellensignal "Einlesesperre" gesetzt. Die Abarbeitung des NC-Programms wird gestoppt. Im Kanal-DB des 1. Kanals werden desweiteren die "erweiterten Adresse M-Funktion" und die "M-Funktions-Nr.") angezeigt.

Das Signal "Einlesesperre" wird im 1. Kanal zurückgesetzt, sobald der Anwender alle von diesem Kanal ausgegebenen Signale in der Signalliste (DB76) zurückgesetzt und damit quittiert hat.

14.10.10 PLC-Maschinendaten

Allgemeines

Es besteht für den Anwender die Möglichkeit, PLC-spezifische Maschinendaten in der NC zu speichern. Diese Maschinendaten können nach Hochlauf der PLC (OB100) vom Anwender verarbeitet werden. Dadurch können z. B. Anwender-Optionen, Maschinenausbaustufen, Maschinenkonfiguration, etc. realisiert werden.

Die Nahtstelle zum Lesen dieser Daten liegt im DB20. Der DB20 wird aber nur bei Verwendung der Anwender-Maschinendaten, d. h. Summe aus den GP-Parametern "UDInt", "UDHex" und "UDReal" ist größer NULL, vom Grundprogramm im Hochlauf angelegt.

Die Größen der einzelnen Bereiche, und damit auch die Gesamtlänge des DB20, werden durch folgende PLC-Maschinendaten eingestellt:

MD14504 \$MN_MAXNUM_USER_DATA_INT

MD14506 \$MN_MAXNUM_USER_DATA_HEX

MD14508 \$MN_MAXNUM_USER_DATA_FLOAT

Dem Anwender werden diese Einstellungen in den GP-Parametern "UDInt", "UDHex" und "UDReal" angegeben.

Die Datenablage im DB20 erfolgt durch das GP bündig in der Reihenfolge:

1. Integer-MD
2. Hexa-Felder-MD
3. Real-MD

Die Integer- und Realwerte werden im S7-Format im DB20 abgelegt.

Die Hexa-Daten werden in der Reihenfolge der Eingabe (Nutzung als Bitfelder) im DB20 gespeichert.

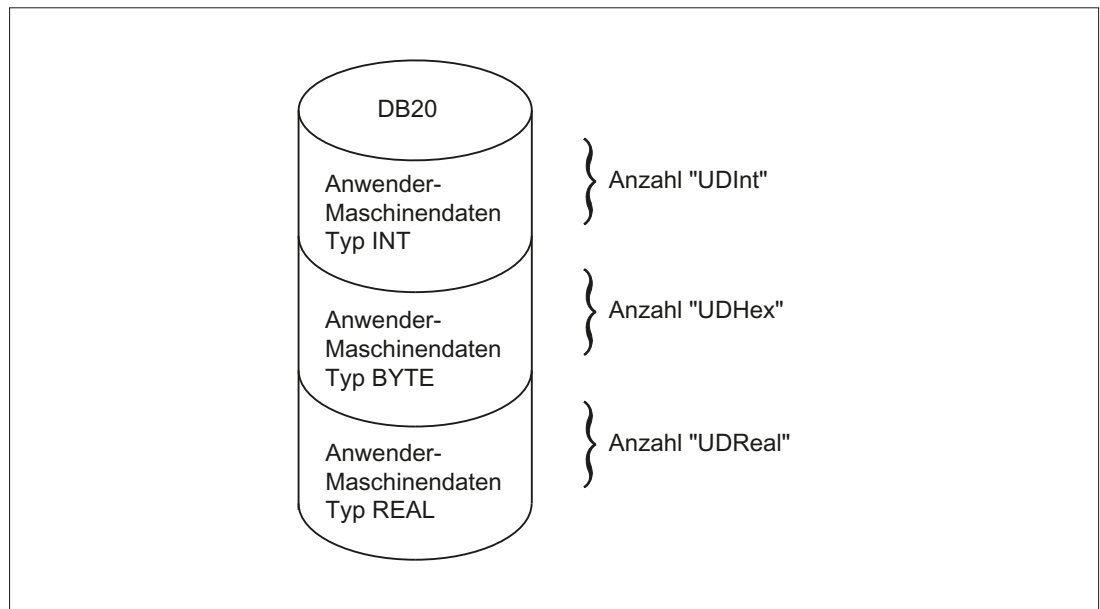


Bild 14-13 DB20

Hinweis

Soll die Anzahl der genutzten PLC-Maschinendaten später erhöht werden, ist es notwendig, den DB20 vorher zu löschen. Damit solche Erweiterungen keine Auswirkungen auf das bestehende Anwenderprogramm haben, sollten die Zugriffe auf die Daten im DB20 möglichst symbolisch erfolgen, z. B. über eine Strukturbeschreibung im UDT.

Alarmer	
400120	DB20 im PLC löschen und Neustart
Erläuterung	DB-Länge ist unterschiedlich zur geforderten DB-Länge
Reaktion	Alarmanzeige und PLC-Stop
Abhilfe	DB20 löschen und anschließender RESET
Fortsetzung	nach Neustart

Beispiel

Für das Projekt im Beispiel werden 4 INTEGER-Werte, 2 Hexa-Felder mit Bitinformationen und 1 Real-Wert benötigt.

Maschinendaten:

MD14510 \$MN_USER_DATA_INT[0]	123
MD14510 \$MN_USER_DATA_INT[1]	456
MD14510 \$MN_USER_DATA_INT[2]	789
MD14510 \$MN_USER_DATA_INT[3]	1011

```

...
MD14512 $MN_USER_DATA_HEX[0]      12
MD14512 $MN_USER_DATA_HEX[1]      AC
...
MD14514 $MN_USER_DATA_FLOAT[0]    123.456

```

GP-Parameter (OB100):

```

CALL FB1, DB7(
    MCPNum := 1,
    MCP1In := P#E0.0,
    MCP1Out := P#A0.0,
    MCP1StatSend := P#A8.0,
    MCP1StatRec := P#A12.0,
    MCP1BusAdr := 6,
    MCP1Timeout := S5T#700MS,
    MCP1Cycl := S5T#200MS,
    NCCyclTimeout := S5T#200MS,
    NCRunupTimeout := S5T#50S;

```

GP-Parameter (zur Laufzeit abfragen):

```

1 gp_par.UDInt;           //=4,
1 gp_par.UDHex;           //=2,
1 gp_par.UDReal;          //=1 )

```

Im Hochlauf der PLC wurde der DB20 mit einer Länge von 28 Byte erstellt:

DB20	
Adresse	Daten
0.0	123
2.0	456
4.0	789
6.0	1011
8.0	b#16#12
9.0	b#16#AC
10.0	1.234560e+02

Die Struktur der genutzten Maschinendaten wird in einem UDT angegeben:

```

TYPE UDT20
STRUCT
    UDInt : ARRAY [0 .. 3] OF INT;
    UDHex0 : ARRAY [0 .. 15] OF BOOL;
    UDReal : ARRAY [0 .. 0] OF REAL; //Beschreibung als Feld, für
                                     //spätere Erweiterungen

```

```
END_STRUCT;  
END_TYPE
```

Hinweis

ARRAY OF BOOL werden immer an geradzahligen Adressen ausgerichtet. Deshalb ist in der Definition des UDT generell ein Array-Bereich von 0 bis 15 zu wählen bzw. alle Booleschen Variablen sind einzeln aufzuführen.

Obwohl im Beispiel zunächst nur ein REAL-Wert genutzt wird, wurde für die Variable ein Feld (mit einem Element) angelegt. Dies sichert eine einfache Erweiterung zu einem späteren Zeitpunkt, ohne dass sich die symbolische Adresse ändert.

Symbolische Zugriffe

Für den symbolischen Zugriff erfolgt ein Eintrag in der Symbolliste:

Symbol	Operand	Datentyp
UData	DB20	UDT20

Zugriffe im Anwenderprogramm (nur symbolische Lesezugriffe dargestellt):

```
...  
  
L      "UData".UDInt[0];  
L      "UData".UDInt[1];  
L      "UData".UDInt[2];  
L      "UData".UDInt[3];  
  
U      "UData".UDHex0[0];  
U      "UData".UDHex0[1];  
U      "UData".UDHex0[2];  
U      "UData".UDHex0[3];  
U      "UData".UDHex0[4];  
U      "UData".UDHex0[5];  
U      "UData".UDHex0[6];  
U      "UData".UDHex0[7];  
...  
U      "UData".UDHex0[15];  
  
L      "UData".UDReal[0];  
...
```

14.10.11 Projektierung von Maschinensteuertafel, Bedienhandgerät, Direktasten

Allgemeines

Es ist ein gleichzeitiger Betrieb von maximal 2 Maschinensteuertafeln und einem Bedienhandgerät möglich. Für die Maschinensteuertafel (MCP) und Bedienhandgerät (BHG) gibt es verschiedene Anschlussmöglichkeiten (Ethernet/PROFINET, PROFIBUS). Hierbei ist es möglich, dass 2 MCP an unterschiedlichen Bussystemen (nur bei Ethernet und PROFIBUS ist Mischbetrieb zulässig) angeschlossen werden können. Erreicht werden kann dies durch den FB1-Parameter "MCPBusType". Bei diesem Parameter ist die rechte Dekade (1er-Stelle) für die 1. MCP und die linke Dekade (10er-Stelle) für die 2. MCP zuständig.

Die Parametrierung der Komponenten erfolgt grundsätzlich über den Aufruf des Grundprogrammbausteins FB1 im OB100. Der FB1 speichert seine Parameter im zugehörigen Instanz-DB (DB7, symbolisch "gp_par"). Hierzu existieren für jede Maschinensteuertafel und das Bedienhandgerät eigene Parametersätze. In diesen Parametersätzen ist die Ein-, Ausgangsadresse vom Anwender zu definieren. Diese Ein- und Ausgangsadressen werden auch im FC19, FC24, FC25, FC26 und FC13 verwendet. Weiterhin sind noch Adressen für Statusinformationen, PROFIBUS bzw. Ethernet/PROFINET zu definieren. Die Zeiteinstellungen für Timeout und zyklische Zwangsnachtriggenung sollen auf dem voreingestellten Wert bleiben. Weitere Informationen zu den MCP, BHG-Komponenten sind im Handbuch Bedienkomponenten zu finden.

Aktivierung

Die jeweilige Komponente wird entweder über die Anzahl der Maschinensteuertafeln (Parameter "MCPNum") oder beim Bedienhandgerät über den Parameter "BHG" aktiviert. Die Festlegung über den Anschluss der MCP, BHG wird über die FB1-Parameter "MCPMPI", "MCPBusType" bzw. "BHG", "BHGMPPI" vorgenommen.

Bedienhandgerät (HT 2)

Beim Bedienhandgerät wird die Adressierung über einen Parameter des GD-Parametersatzes vorgenommen. Dieses war aus Kompatibilitätsgründen der Parameternamen erforderlich.

Projektierung

Grundsätzlich existieren verschiedene Kommunikationsmechanismen, bedingt durch den Bus-Anschluss von MCP und BHG, zur Übertragung der Daten zwischen MCP/BHG und PLC. In einem Fall (Ethernet) werden die Daten durch das "CP 840D sl" transportiert.

Die Parametrierung erfolgt dabei vollständig über die MCP/BHG Parameter im FB1.

Im anderen Fall erfolgt die Übertragung durch das PLC-Betriebssystem durch die PROFIBUS-Konfiguration.

Die Parametrierung erfolgt über STEP 7 in HW-Config. Für den Zugriff des Grundprogramms auf diese Daten und Ausfallüberwachung von MCP/BHG müssen die eingestellten Adressen in FB1-Parametern dem Grundprogramm bekannt gemacht werden.

Im Weiteren folgt eine Übersicht über die verschiedenen Kopplungsmöglichkeiten. Hierbei kann auch ein Mischbetrieb projektiert werden.

Wird ein Fehler aufgrund einer Zeitüberwachung erkannt, erfolgt ein Eintrag im Alarmpuffer der PLC-CPU (Alarm 400260 bis 400262). In diesem Fall werden die Eingangssignale von der MCP bzw. vom Bedienhandgerät (MCP1In/MCP2In bzw. BHGIn) mit 0 initialisiert. Sollte eine Neusynchronisation zwischen PLC und MCP/BHG möglich sein, wird die Kommunikation automatisch wieder aufgenommen und die Fehlermeldung vom GP gelöscht.

Hinweis

Die Abkürzung "(n.r.)" in den unten angegebenen Tabellen bedeutet "nicht relevant".

Ethernet-Ankopplung (MCPBusType = 5)

Kommunikation erfolgt ohne weitere Projektierung direkt vom PLC-GP aus über die CP 840D sl. Die Parametrierung erfolgt über die unten genannten Parameter des FB1.

In "MCP1 BusAdr", "MCP2 BusAdr" bzw. "BHGRecGDNo" (entspricht der Busadresse des Teilnehmers) ist der numerische Teil des logischen Namen der Komponente einzutragen. Der logische Name wird über die Schalter an der MCP bzw. Anschlussbox eingestellt.

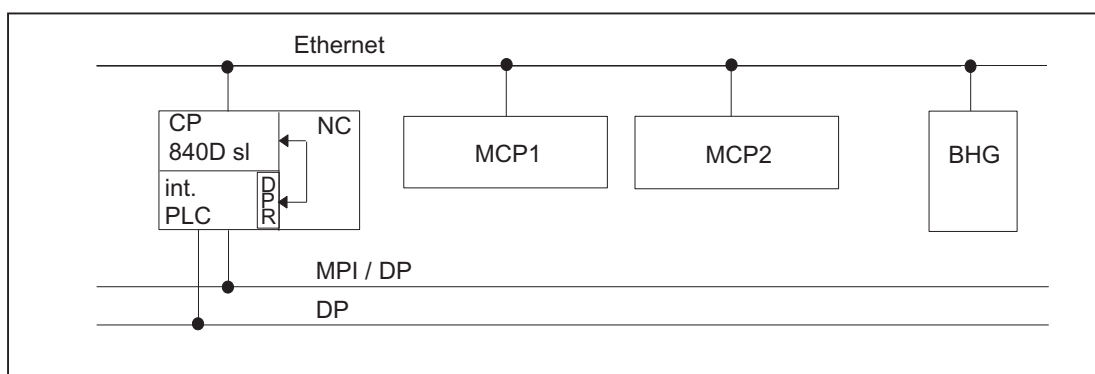


Bild 14-14 Ethernet-Ankopplung

Relevante Parameter (FB1)		
MCP		BHG
MCPNum=1 oder 2 (Anzahl MCP)		BHG = 5 (über CP 840D sl)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOOut
MCP1StatSend	MCP2StatSend	BHGStatSend
MCP1StatRec (n.r.)	MCP2StatRec (n.r.)	BHGStatRec
MCP1BusAdr	MCP2BusAdr	BHGInLen (n.r.)
MCP1Timeout (n.r.)	MCP2Timeout (n.r.)	BHGOOutLen (n.r.)
MCP1Cycl (n.r.)	MCP2Cycl (n.r.)	BHGTimeout (n.r.)
MCPMPI = FALSE		BHGCycl (n.r.)
MCP1Stop	MCP2Stop	BHGRecGDNo
MCP1NotSend	MCP2NotSend	BHGRecGBZNo (n.r.)
		BHGRecObjNo (n.r.)

Relevante Parameter (FB1)	
MCP	BHG
MCPBusType = b#16#55 (über CP 840D sl)	BHGSendGDNo (n.r.)
	BHGSendGBZNo (n.r.)
MCPsDB210 = FALSE	BHGSendObjNo (n.r.)
MCPCopyDB77 = FALSE	BHGMPI = FALSE
	BHGStop
	BHG NotSend

Für die Zeitüberwachungen wird ein Fehlereintrag im Alarmpuffer der PLC generiert. Daraus entstehen am HMI die Fehlermeldungen:

- 400260: MSST 1 ausgefallen
oder
- 400261: MSST 2 ausgefallen
- 400262: BHG ausgefallen

Ein Ausfall von MCP oder BHG wird auch gleich nach dem Neustart erkannt, auch dann, wenn noch keine Daten zwischen MCP/BHG und PLC ausgetauscht wurden.

Die Überwachungsfunktion wird aktiviert, sobald alle Komponenten nach dem Hochlauf "Ready" melden.

Beispiel: OP mit Direkttasten

Die Direkttasten der OPs am Ethernet-Bus sollen in die PLC übertragen werden. Bisher sind die Direkttasten über den PROFIBUS bzw. über eine spezielle Kabelverbindung zwischen OP und MCP an die PLC geführt worden.

Für die Ankopplung der Direkttasten über Ethernet, dies betrifft z. B. das "OP 08T", gibt es eine Parametrierung im Grundprogramm für die Aktivierung des Datentransports. Die zugehörigen Parameter liegen im Instanz-DB des FB 1 (OpKeyNum bis OpKeyBusType, siehe Datentabelle). Die Parameter werden im Anlauf-OB100 durch das Beschalten der Parameter beim FB1-Aufruf durch den Anwender versorgt. Die Busadresse und Op1/2KeyStop können auch im zyklischen Betrieb über Beschreiben des FB1-Instanz-DB DB7 verändert werden.

Der Transport der Nutzdaten der Direkttasten verläuft gleichartig wie bei der Ethernet-MCP. Der Datentransport kann über das Beschreiben des DB7-Parameters "Op1/2KeyStop" auch gestoppt werden und wieder neu gestartet werden. Während der Stopp-Phase kann auch die Adresse des Direkttasten-Moduls (TCU-Index bzw. MCP-Adresse) geändert werden.

Nach Rücksetzen des Stopp-Signals wird eine Verbindung zu der neuen Adresse aufgebaut.

Der Status der jeweiligen Direkttasten-Schnittstelle ist ablesbar im Nahtstellensignal:

DB10.DBX104.3 (OP1Key bereit)

bzw.

DB10.DBX104.4 (OP2Key bereit)

Adresse Direkttasten

Für den Parameter Op1/2KeyBusAdr ist im Regelfall der TCU-Index zu verwenden. Dieses trifft für die OPs wie z. B. OP 08T, OP 12T zu, die für die Direkttasten **keine** spezielle Kabel-Verbindung zu einer Ethernet-MCP haben.

Verfügen OPs mit Direkttasten über eine spezielle Kabelverbindung und sind diese mit einer Ethernet-MCP verbunden, so ist für den Parameter Op1/2KeyBusAdr die Adresse der MCP (DIP-Schalterstellung der MCP) zu verwenden. Über die Direkttasten-Schnittstelle wird nur der Datenstrom der Direkttasten (2 Bytes) übertragen.

Alarm Direkttasten

Für die Zeitüberwachungen wird ein Fehlereintrag im Alarmpuffer der PLC generiert. Daraus entstehen am HMI die Fehlermeldungen:

- 400274: Direkttasten 1 ausgefallen
oder
- 400275: Direkttasten 2 ausgefallen

Bedieneinheitenumschaltung bei Direkttasten

Der Anwender beschaltet Op1/2KeyBusAdr mit 0xFF und Stopp = TRUE im Anlaufbaustein OB100. Über den M-zu-N-Baustein FB9 wird die Direkttasten-Adresse der M-zu-N-Schnittstelle an den Parameter "Op1KeyBusAdr" gelegt.

Relevante Parameter (FB1)		
Direkttasten		z. B. Direkttasten OP 08T
OpKeyNum = 1 oder 2 (Anzahl OPs mit Direkttasten)		
Op1KeyIn	Op2KeyIn	
Op1KeyOut	Op2KeyOut	
OpKey1BusAdr	Op2KeyBusAdr	Adresse: TCU-Index
Op1KeyStop	Op2KeyStop	
Op1KeyNotSend	Op2KeyNotSend	
OpKeyBusType = b#16#55 (über CP 840D sl)		

MCP-Identifizierung

Über die Identify-Schnittstelle im DB7 ist es möglich, mit den relevanten Parametern am Ein-/Ausgang den Typ einer Ethernet-Komponente (MCP, HT 2, HT 8 oder Direkttasten) im zyklischen Betrieb zu erfragen:

- Relevante Parameter am Eingang:
"IdentMcpBusAdr", "IdentMcpProfilNo", "IdentMcpBusType", "IdentMcpStrobe"
- Relevante Parameter am Ausgang:
"IdentMcpType", "IdentMcpLengthIn", "IdentMcpLengthOut"

Hierbei wird die DIP-Device-Adresse bzw. TCU-Index am Parameter "IdentMcpBusAdr" vom Anwenderprogramm in Verbindung mit dem Setzen des Strobe-Signals angeschaltet.

Der Eingangsparameter "IdentMcpProfilNo" ist im Regelfall auf den Wert "0" zu setzen. Nur bei der Identifikation von Direkttasten ist dieser Parameter auf den Wert "1" zu setzen. Der Parameter "IdentMcpBusType" hat für ein Anwenderprogramm derzeit keine Bedeutung und ist auf dem Standardwert zu lassen.

Nach dem Rücksetzen des Strobe-Signals durch das Grundprogramm stehen gültige Ausgangsinformationen für den Anwender zur Verfügung. Das Rücksetzen des Strobe-Signals vom Grundprogramm kann mehrere PLC-Zyklen (bis zu 2 Sekunden) dauern.

Die Ausgangsparameter sollen dem Anwender anzeigen, wie groß die Datenbereiche für das angesprochene Gerät sind. Weiterhin kann hierüber z. B. festgestellt werden, ob an einer Anschlussbox ein HT 2 oder ein HT 8 oder kein Gerät angeschlossen ist. Mit der Information kann dann der MCP-Kanal oder BHG-Kanal aktiviert werden. Im zyklischen Betrieb können die Parameter vom Anwenderprogramm symbolisch beschrieben und gelesen werden über den Symbolnamen des DB7 (gp_par).

Relevante Parameter (FB1)		
MCP-Geräte Identifizierung		Eingangsparameter z. B. OP 08T
Eingang	Ausgang	Werte bei Direktasten
IdentMcpBusAdr	IdentMcpType	IdentMcpBusAdr = TCU-Index
IdentMcpBusProfilNo	IdentMcpLengthIn	IdentMcpBusProfilNo = Wert 1
IdentMcpBusType	IdentMcpLengthOut	IdentMcpBusType = Defaultwert
IdentMcpStrobe		
IdentMcpBusProfilNo		Wert
MCP, BHG, HT 8, HT 2		B#16#0
Direktasten wie z.B. OP 08T, OP 12T		B#16#1
IdentMcpType (Mcp-Type)		
kein Gerät verbunden		0
MCP 483C IE (Compact)		B#16#80
MCP 483C IE		B#16#81
MCP 310		B#16#82
MCP OEM		B#16#83
MCP DMG		B#16#84
HT 8		B#16#85
TCU_DT (Direktasten)		B#16#86
MCP_MPP		B#16#87
HT 2		B#16#88
OP 08T (Direktasten)		B#16#89

PROFIBUS-Ankopplung am DP-Anschluss (MCPBusType = 3)

Bei der PROFIBUS-Ankopplung der MCP muss diese Komponente in der Hardware-Projektierung von STEP 7 berücksichtigt werden. Die MCP wird auf dem Standard-DP-Bus der PLC angekoppelt (**nicht am MPI/DP**). Die Adressen sind in den Eingangs- und Ausgangs-Abbildbereich zu legen. Diese Anfangs-Adressen sind auch in den Pointer-Parametern des FB1 zu hinterlegen. Die weitere Parametrierung erfolgt über die unten aufgeführten Parameter des FB1.

Eine PROFIBUS-Variante des BHG existiert nicht. Deshalb ist in diesem Bild ein Ethernet-Anschluss für das BHG dargestellt. In den Parameter "MCP1BusAdr" und "MCP2BusAdr" muss die PROFIBUS-Slave-Adresse hinterlegt werden. "MCPxStatRec" ist der Pointer auf die projektierte Diagnoseadresse (z. B. P#A8190.0) einzutragen.

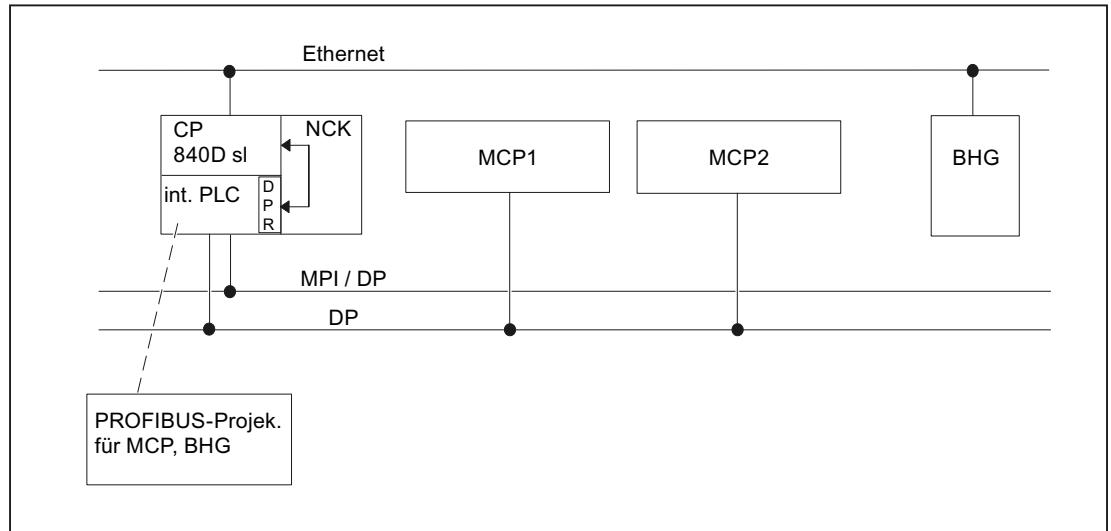


Bild 14-15 PROFIBUS-Ankopplung

Relevante Parameter (FB1)		
MCP		BHG
MCPNum = 1 oder 2 (Anzahl MCP)		BHG = 5 (über CP 840D sl)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOut
MCP1StatSend (n.r.)	MCP2StatSend (n.r.)	BHGStatSend
MCP1StatRec	MCP2StatRec	BHGStatRec
MCP1BusAdr	MCP2BusAdr	BHGInLen
MCP1Timeout	MCP2Timeout	BHGOutLen
MCP1Cycl (n.r.)	MCP2Cycl (n.r.)	BHGTimeout (n.r.)
MCPMPI = FALSE		BHGCycl (n.r.)
MCP1Stop	MCP2Stop	BHGRecGDNo
MCPBusType = b#16#33		BHGRecGBZNo (n.r.)
		BHGRecObjNo (n.r.)
MCPSDB210 = FALSE		BHGSendGDNo (n.r.)
MCPCopyDB77 = FALSE		BHGSendGBZNo (n.r.)
		BHGSendObjNo (n.r.)
		BHGMPI = FALSE
		BHGStop

Ein Ausfall der MCP führt die PLC normalerweise in den Stopp-Zustand. Falls dieses nicht gewünscht ist, dann kann durch OB82, OB86 ein Stopp vermieden werden. Das Grundprogramm bringt standardmäßig den OB82 und OB86 Aufruf mit. In diesen OBs wird der FC5 aufgerufen. Dieser FC5 prüft, ob der ausgefallene Slave eine MCP ist. Falls dies

zutrifft wird kein Stopp der PLC ausgelöst. Durch "MCPxStop" = TRUE wird die MCP vom Grundprogramm über SFC12 als Slave abgeschaltet. Falls die PLC bei Ausfall oder Störung der MCP nicht in Stopp-Zustand gebracht wird, wird eine Alarmmeldung durch das Grundprogramm erzeugt. Bei Stationswiederkehr wird der Alarm gelöscht.

PROFIBUS-Ankopplung am MPI/DP-Anschluss (MCPBusType = 4)

Bei der PROFIBUS-Ankopplung der MCP muss diese Komponente in der Hardwareprojektierung von STEP 7 berücksichtigt werden. Die MCP wird auf dem MPI/DP-Bus der PLC angekoppelt.

Die Adressen sind in den Eingangs- und Ausgangs-Abbildbereich zu legen. Diese Anfangsadressen sind auch in den Pointer-Parametern des FB1 zu hinterlegen. Die weitere Parametrierung erfolgt über die unten aufgeführten Parameter des FB1. Eine PROFIBUS-Variante des BHG existiert nicht. Deshalb ist in diesem Bild ein Ethernet-Anschluss für das BHG gezeichnet. Im MCP1BusAdr und MCP2BusAdr muss die PROFIBUS-Slave Adresse hinterlegt werden. In MCPxStatRec ist der Pointer auf die projektierte Diagnoseadresse einzutragen (z. B. P#A8190.0).

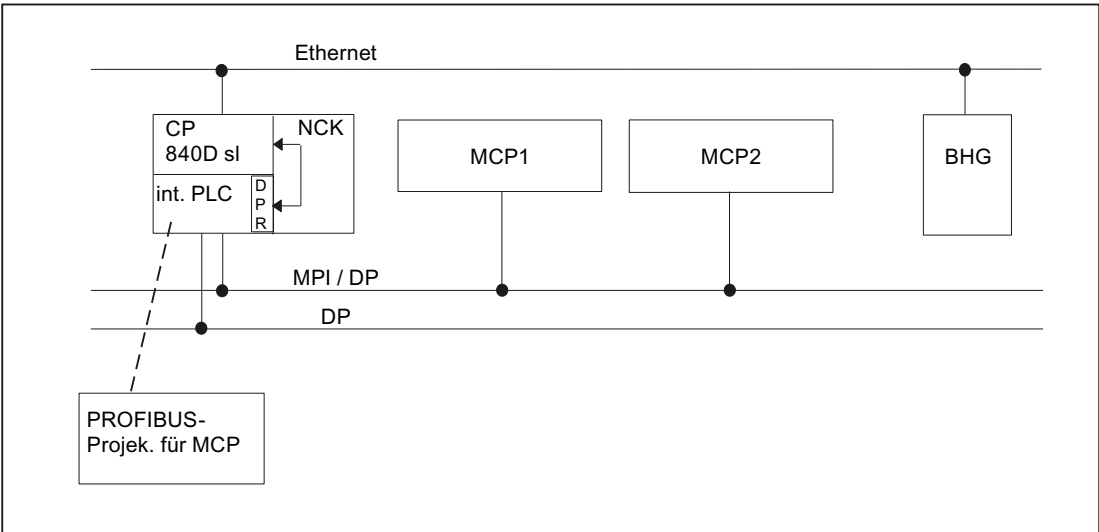


Bild 14-16 PROFIBUS-Ankopplung am MPI/DP-Anschluss

Relevante Parameter (FB1)		
MCP		BHG
MCPNum = 1 oder 2 (Anzahl MCP)		BHG = 5 (über CP 840D sl)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOut
MCP1StatSend (n.r.)	MCP2StatSend (n.r.)	BHGStatSend
MCP1StatRec	MCP2StatRec	BHGStatRec
MCP1BusAdr	MCP2BusAdr	BHGInLen
MCP1Timeout	MCP2Timeout	BHGOutLen
MCP1Cycl (n.r.)	MCP2Cycl (n.r.)	BHGTimeout (n.r.)
MCPMPI = FALSE		BHGCycl (n.r.)

Relevante Parameter (FB1)		
MCP		BHG
MCP1Stop	MCP2Stop	BHGRecGDNo
MCPBusType = b#16#44		BHGRecGBZNo (n.r.)
		BHGRecObjNo (n.r.)
MCPSDB210 = FALSE		BHGSendGDNo (n.r.)
MCPCopyDB77 = FALSE		BHGSendGBZNo (n.r.)
		BHGSendObjNo (n.r.)
		BHGMPI = FALSE
		BHGStop

Ein Ausfall der MCP führt die PLC normalerweise in den Stopp-Zustand. Falls dieses nicht gewünscht ist, kann durch OB82, OB86 ein PLC-Stopp vermieden werden. Das Grundprogramm bringt standardmäßig den OB82 und OB86 Aufruf mit. In diesen OBs wird der FC5 aufgerufen. Dieser FC5 prüft, ob der ausgefallene Slave eine MCP ist. Falls dieses zutrifft, wird kein Stopp der PLC ausgelöst. Durch MCPxStop:=True wird die MCP vom Grundprogramm über SFC12 als Slave abgeschaltet. Falls die PLC bei Ausfall oder Störung der MCP nicht in den Stopp-Zustand gebracht wird, wird eine Alarmmeldung durch das Grundprogramm erzeugt. Bei Stations-Wiederkehr wird der Alarm gelöscht.

PROFINET-Ankopplung (MCPBusType = 6)

Bei der PROFINET-Ankopplung der MCP muss diese Komponente in der Hardware-Projektierung von STEP 7 parametrisiert werden. Die MCP wird gekoppelt mit dem PROFINET-Modul der CPU.

Bei der Parametrierung der MCP in HW-Konfig sind die Adressen in den Eingangs- und Ausgangs-Abbildbereich zu legen. Diese Anfangsadressen sind auch in den Pointer-Parametern (MCPxIn und MCPxOut) des FB1 zu hinterlegen. Denn der Signaltransfer zwischen MCP und Grundprogramm erfolgt über diese Parameter. Über den Parameter MCPxIn wird die MCP auch überwacht. Deshalb ist der Parameter MCPxBusAdr bei dieser MCP-Variante nicht relevant.

In MCPxStatRec ist der Pointer auf die projektierte Diagnoseadresse einzutragen (z. B. P#A8190.0).

Die PROFINET-MCP hat einen eigenen Typ, der beim Parameter MCPBusType anzutragen ist.

Die weitere Parametrierung erfolgt über die unten aufgeführten Parameter des FB1. Eine Profibus-Variante des BHG existiert nicht. Im Bild ist ein Ethernet-Anschluss für das BHG gezeichnet.

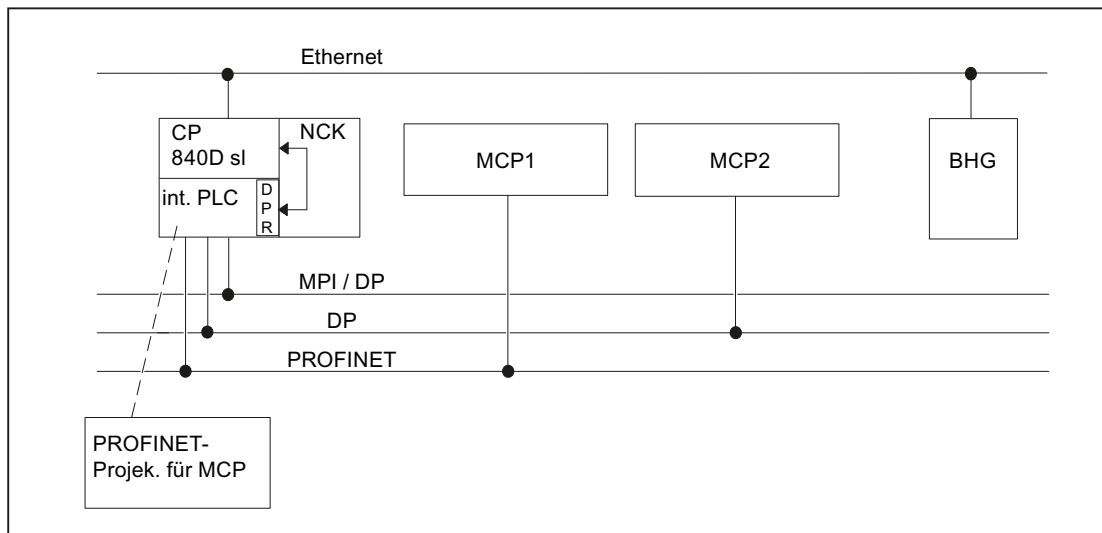


Bild 14-17 PROFINET-Ankopplung

Relevante Parameter (FB1)		
MCP		BHG
MCPNum = 1 oder 2 (Anzahl MCP)		BHG = 5 (über CP 840D sl)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOut
MCP1StatSend (n.r.)	MCP2StatSend (n.r.)	BHGStatSend
MCP1StatRec	MCP2StatRec	BHGStatRec
MCP1BusAdr (n.r.)	MCP2BusAdr (n.r.)	BHGInLen
MCP1Timeout	MCP2Timeout	BHGOutLen
MCP1Cycl	MCP2Cycl	BHGTimeout (n.r.)
MCPMPI = FALSE		BHGCycl (n.r.)
MCP1Stop	MCP2Stop	BHGRecGDNo
MCPBusType = b#16#36 (wie im Bild als Beispiel) (6 = PROFINET für MCP1) (3 = PROFIBUS für MCP2)		BHGRecGBZNo (n.r.)
		BHGRecObjNo (n.r.)
MCPSDB210 = FALSE		BHGSendGDNo (n.r.)
MCPCopyDB77 = FALSE		BHGSendGBZNo (n.r.)
		BHGSendObjNo (n.r.)
		BHGMPI = FALSE
		BHGStop

Ein Ausfall der MCP führt die PLC normalerweise in den Stopp-Zustand. Falls dieses nicht gewünscht ist, dann kann durch OB82, OB86 ein PLC-Stop vermieden werden. Das Grundprogramm bringt standardmäßig den OB82- und OB86-Aufruf mit. In diesen OBs wird der FC5 aufgerufen. Dieser FC5 prüft, ob der ausgefallene Slave eine MCP ist. Falls dies

zutrifft wird kein PLC-Stop ausgelöst. Für die Überwachung bei einem MCP-Ausfall ist die Eingangsadresse am Parameter `MCPxIn` von Bedeutung.

Durch `MCPxStop := True` wird die MCP vom Grundprogramm über SFC12 als Slave abgeschaltet. Falls die PLC bei Ausfall oder Störung der MCP nicht in Stopp-Zustand gebracht wird, wird eine Alarmmeldung durch das Grundprogramm erzeugt. Bei Stations-Wiederkehr wird der Alarm gelöscht.

14.10.12 Umschaltung von Maschinensteuertafel, Bedienhandgerät

Die Umschaltung bzw. Abschaltung einer Bedienkomponente (MCP oder BHG) wird vom Standard nur bei Ethernet-Varianten unterstützt.

PROFIBUS-Variante

Bei PROFIBUS-Varianten ist diese Funktionalität nur in einem begrenzten Umfang mit zusätzlichem Anwenderaufwand möglich.

Bei der PROFIBUS-Variante der MCP kann der für MCP1, MCP2 oder BHG spezifizierte Datenbereich des DB77 für die Pointer der MCP am FB1 verwendet werden. Die Slave-Busadresse der MCP muss unter `MCPxBusAdr` korrekt gesetzt sein, da hierüber die Überwachung realisiert ist. Durch eine Kopieroutine des Anwenderprogramms müssen die Signale der aktiven MCP von dem in HW-Config projektierten E/A-Bereich in den DB77 kopiert werden. Somit können mehrere MCPs am PROFIBUS signalmäßig umgeschaltet werden. Für die Umschaltphase von einer zur anderen MCP ist der Parameter `MCPxStop` auf TRUE zu setzen.

Steuersignale

Mit Hilfe der Parameter `MCP1Stop`, `MCP2Stop` und `BHGStop` ist ein Anhalten der Kommunikation zu den einzelnen Komponenten möglich (Wertzuweisung = 1). Dieses Stoppen bzw. auch Aktivieren der Kommunikation ist im laufenden Zyklus möglich. Allerdings darf die Wertänderung nicht über einen erneuten Aufruf des FB1 erfolgen, sondern durch die symbolische Schreibweise der Parameter.

Beispiel: Stop der Übertragung der 1. Maschinensteuertafel:

```
SET;  
S gp_par.MCP1Stop;
```

Bei gesetzten Parametern `MCP1Stop`, `MCP2Stop`, `BHGStop` erfolgt auch eine Unterdrückung bzw. Löschung der Alarmer 400260 bis 400262.

Umschalten der Bus-Adresse

Soll die bestehende Kommunikationsverbindung zu einer Bedienkomponente (MCP oder BHG) abgebaut und eine neue Kommunikationsverbindung zu einer anderen Bedienkomponente (MCP oder BHG) mit einer anderen Kommunikationsadresse aufgebaut werden, ist folgendermaßen vorzugehen:

1. Anhalten der Kommunikation der abzukoppelnden Bedienkomponente: Parameter `MCP1Stop`, `MCP2Stop` oder `BHGStop` = 1
2. Die Kommunikation ist gestoppt, wenn gilt: `DB10, DBX104.0, .1` oder `.2` == 0
3. Ändern der Busadresse:
 - MCP: FB1-Parameter `MCP1BusAdr` oder `MCP2BusAdr` = <Bus-Adresse der neuen Bedienkomponente>
 - BHG (Ethernet-Variante): FB1-Parameter `BHGRecGDNo` = <Bus-Adresse der neuen Bedienkomponente>
4. Freigabe der Kommunikation (im gleichen PLC-Zyklus wie Punkt 3. möglich): Parameter `MCP1Stop`, `MCP2Stop` oder `BHGStop` = 0
5. Die Kommunikation mit der neuen Komponente ist aktiv, wenn gilt: `DB10, DBX104.0, .1` oder `.2` == 1

Abschalten des LED-Blinkens einer Ethernet-MCP

Nach einem Power On zeigen MCPs standardmäßig durch Blinken ihrer LEDs den Abschluss des Hochlaufs und das Warten auf den Verbindungsaufbau an. Das Blinken der LEDs kann wie nachfolgende beschrieben abgeschaltet werden. Eine remanente Speicherung dieses Verhaltens auf dem MCP ist derzeit **nicht** möglich.

Voraussetzung

MCP-Firmware ab V02.02.04

Einstellung zum Abschalten des Blinkens

Das Setzen des Sende-Status in `MCPxStop` muss vor dem Start der Kommunikation mit der MCP erfolgen. Vor dem Start der Kommunikation bedeutet entweder im Hochlauf (OB100) oder im zyklischen Betrieb (OB1) vor dem Setzen des DB7-Parameters `MCPxStop` = FALSE

Setzen des Sende-Status: FB1-Parameter `MCPxStatSend`, Bit30 = 0 und Bit31 = 1

Es erfolgt keine Rückmeldung des aktuellen Status.

Beispiel

Auszug aus OB100: (am Beispiel fuer MCP1)

```
CALL "RUN_UP" , "gp_par"
...
MCP1StatSend  := P#A 8.0
...
// MCP-Blinken deaktivieren
SET
R    A 11.6
```

S A 11.7

...

14.11 SPL für Safety Integrated

Die SPL ist keine Funktion des Grundprogramms, sondern des Anwenders. Das Grundprogramm stellt einen Datenbaustein DB18 für Safety-SPL-Signale zur Verfügung und sichert über einen Datenvergleich die Information gegenüber dem SPL-Programm in der NC ab.

Literatur:

Funktionshandbuch Safety Integrated

14.12 Belegungsübersicht

14.12.1 Belegung: NCK/PLC-Nahtstelle

Die Belegung der NC/PLC-Nahtstelle für SINUMERK 840D sl ist ausführlich beschrieben in:

Literatur:

Listen sl (Buch2)

14.12.2 Belegung: FB/FC

Nummer	Bedeutung
FB15	Basis-Grundprogramm
FB1, FC2, FC3, FC5	Basis-Grundprogramm
FC0 ... 29	reserviert für Siemens
FB0 ... 29	reserviert für Siemens
FC30 ... 999 ¹⁾	frei für Anwender
FB30 ... 999 ¹⁾	frei für Anwender
FC1000 ... 1023	reserviert für Siemens
FB1000 ... 1023	reserviert für Siemens
FC1024 ... Obergrenze	frei für Anwender
FB1024 ... Obergrenze	frei für Anwender

¹⁾ Die tatsächliche Obergrenze der Baustein-Nummer (FB/FC) ist abhängig von der PLC-CPU, die in der gewählten NCU enthalten ist.

Hinweis

Belegung der FC, FB siehe " Speicherbedarf des PLC-Grundprogramms (Seite 962)".

14.12.3 Belegung: DB

Hinweis

Es werden nur so viele Datenbausteine eingerichtet, wie aufgrund der Projektierung in den NC-Maschinendaten erforderlich sind.

Übersicht der Datenbausteine			
DB-Nr.	Bezeichnung	Name	Paket
1		reserviert für Siemens	GP
2 ... 5	PLC-MELD	PLC-Meldungen	GP
6 ... 8		Grundprogramm	
9	NC-COMPILE	Nahtstelle für NC-Compile-Zyklen	GP
10	NC-NAHTSTELLE	Zentrale NC-Nahtstelle	GP
11	BAG 1	Nahtstelle BAG	GP
12		Nahtstelle Rechnerkopplung und Transportsystem	
13 ... 14		reserviert für Grundprogramm	
15		Grundprogramm	
16		PI-Dienst Definition	
17		Versionskennung	
18		reserviert für Grundprogramm	
19		Nahtstelle HMI	
20		PLC-Maschinendaten	
21 ... 30	KANAL 1 ... n	Nahtstelle NC-Kanäle	GP
31 ... 61	ACHSE 1 ... m	Nahtstellen für Achsen/Spindeln bzw. frei für Anwender	GP
62 ... 70		frei für Anwender	
71 ... 74		Werkzeugverwaltung	GP
75 ... 76		M-Gruppen-Dekodierung	
77		DB für MCP Signale	
78 ... 80		reserviert für Siemens	
81 ... 999 ¹⁾		siehe unten: ShopMill, ManualTurn	
1000 ... 1099		reserviert für Siemens	
1100 ... Obergrenze		frei für Anwender	

¹⁾ Die tatsächliche Obergrenze der Baustein-Nummer (DB) ist abhängig von der PLC-CPU, die in der gewählten NCU enthalten ist. Datenbausteine von nicht aktivierten Kanälen, Achsen/Spindeln, Werkzeugverwaltung sind frei für den Anwender.

Hinweis

Datenbausteine von nicht aktivierten Kanälen, Achsen/Spindeln, Werkzeugverwaltung können frei verwendet werden.

14.12.4 Belegung: Timer

Timer-Nr.	Bedeutung
T 0 ... T 512 ¹⁾	Anwenderbereich

¹⁾ Die tatsächliche Obergrenze der Timer-Nummer (DB) ist abhängig von der PLC-CPU, die in der gewählten NCU enthalten ist.

14.13 PLC-Funktionen für HMI (DB19)

14.13.1 Kanalanwahl

Funktion

Der am HMI angezeigte Kanal, z.B. im Maschinengrundbild, kann vom PLC-Anwenderprogramm aus über die HMI/PLC-Nahtstelle angewählt werden.

Voraussetzung

In der NC ist mehr als ein Kanal parametrisiert.

Auftrags- und Quittierungsschnittstelle

DB19	Bedeutung	
DBX32.0 - .5	PLC → HMI	Funktionsnummer: 1 = Kanalanwahl
DBX32.6	PLC ↔ HMI	Funktionsanforderung
DBX32.7	HMI → PLC	Status: 1 = "Funktion wird ausgeführt"
DBB33	PLC → HMI	Kanalnummer: 1, 2, 3, ... max. Anzahl Kanäle Nächster Kanal: FF _H
DBB36	HMI → PLC	Fehlererkennung: <ul style="list-style-type: none"> • 0: Kein Fehler • 1: Funktionsnummer ungültig (DBX32.0 - .5) • 2: Parameter ungültig (DBB33 - DBB35) • 3: Das Schreiben der HMI-internen Variable ist fehlgeschlagen. • 10: Kanal nicht vorhanden (DBB33)

Funktionsablauf

PLC → HMI

Vom PLC-Anwenderprogramm ist folgender Ablauf einzuhalten:

1. Prüfen, ob die Schnittstelle für einen neuen Auftrag frei ist:
 - DB19.DBX32.6 == 0 (Funktionsanforderung)
 - DB19.DBX32.7 == 0 (Status)
2. Ist die Schnittstelle frei, sind die Auftragsdaten einzutragen und die Funktionsanforderung zu setzen:
 - DB19.DBB33 = <Kanalnummer>
 - DB19.DBX32.0 - .5 = 1 (Funktionsnummer)
 - DB19.DBX32.6 = 1 (Funktionsanforderung)

HMI → PLC

Vom HMI erfolgen bei **fehlerfreier** Parametrierung folgende Reaktionen:

1. Nachdem HMI die Funktionsanforderung zur Kanalwahl erkennt hat, wird der Status auf "Funktion wird ausgeführt" gesetzt und die Funktionsanforderung zurückgesetzt:
 - DB19.DBX32.7 = 1 (Status)
 - DB19.DBX32.6 = 0 (Funktionsanforderung)
2. Nachdem die Kanalwahl durchgeführt wurde, wird der Status wieder zurückgesetzt und als Fehlerkennung der Wert 0 gesetzt:
 - DB19.DBX32.7 = 0 (Status)
 - DB19.DBX36 = 0 (Fehlerkennung)

Vom HMI erfolgen bei **fehlerhafter** Parametrierung folgende Reaktionen:

- Die Funktionsanforderung wird zurückgesetzt und die entsprechende Fehlerkennung gesetzt:
 - DB19.DBX32.6 = 0 (Funktionsanforderung)
 - DB19.DBX36 = <Fehlerkennung>

14.13.2 Programmanwahl

Funktion

Über die PLC/HMI-Nahtstelle können voreingestellte Programme / Werkstücke zur Abarbeitung durch die NC angewählt werden.

Die Voreinstellung erfolgt über den Eintrag der Programme / Werkstücke in Dateien, den sogenannten PLC-Programmlisten (*.ppl).

Voraussetzungen

Zur Freigabe der Auftragsbearbeitung durch den HMI muss folgendes Maschinendatum gesetzt werden:

MD9106 \$MM_SERVE_EXTCALL_PROGRAMS

Zur Aktivierung einer bereichsspezifischen PLC-Programmliste müssen das jeweilige Maschinendatum und mindestens das Kennwort der Schutzstufe gesetzt werden:

- Bereich **Anwender (user)**
 - MD51041 \$MN_ENABLE_PROGLIST_USER = 1
 - Kennwort der Schutzstufe: 3 (Anwender)
 - Programmliste: /user/sinumerik/hmi/plc/programlist/plc_proglis_user.ppl
- Bereich **Hersteller (oem)**
 - MD51043 \$MN_ENABLE_PROGLIST_MANUFACT = 1
 - Kennwort der Schutzstufe: 1 (Hersteller)
 - Programmliste: /oem/sinumerik/hmi/plc/programlist/plc_proglis_manufacturer.ppl

Aufbau einer Programmliste

Eine Programmliste ist eine Textdatei. Sie enthält pro Zeile folgende Informationen:

<Programmnummer> <Programmpfad><Programmname> [CH=<Kanalnummer>]

- **Programmnummer**
Die Programmnummern, die in einer Programmliste verwendet werden dürfen, sind abhängig vom Bereich:
 - Anwender (user): 1 - 100
 - Individual (oem_i): 101 - 200 (nur SINUMERIK 828D)
 - Hersteller (oem): 201 - 255
- **Kanalnummer**
Die Angabe der Kanalnummer "CH=<Kanalnummer>" ist optional. Sie ist nur erforderlich, wenn die NC über mehr als einen Kanal verfügt.
- **Programmpfad**
Der Programmpfad muss vollständig und absolut angegeben werden.
Für die Angabe des Programmpfades siehe:
Literatur
Programmierhandbuch Arbeitsvorbereitung, Kapitel "Datei- und Programmverwaltung" > "Programmspeicher" > "Adressierung von Dateien des Programmspeichers"
- **Kanalnummer**
Die Kanalnummer (1, 2, 3, ...) muss nur angegeben werden, wenn in der NC mehr als ein Kanal projektiert ist.

Den Aufbau der Anwender-Programmliste zeigt der folgende beispielhafte Ausschnitt:

Programmliste: plc_proglis_user.ppl

```
1 //DEV2:/MPFDir/PROG_01.MPF CH=1
2 //DEV2:/MPFDir/PROG_01.MPF CH=2
```

Einträge in einer Programmliste erstellen

Die Einträge in einer Programmliste (*.ppl) können direkt in der Datei editiert werden oder über Masken der Bedienoberfläche vorgenommen werden.

- Über die Bedienoberfläche für den Bereich **Anwender**
Bedienbereich "Programm Manager" > "ETC-Taste (>)" > "Prog.Liste"
- Über die Bedienoberfläche für den Bereich **Hersteller**
Bedienbereich "Inbetriebnahme" > "Systemdaten" > "ETC-Taste (>)" > "Prog.Liste"

Programmanwahl: Auftragsschnittstelle

Hinweis

Ein neuer Auftrag darf von der PLC nur angefordert werden, wenn der letzte Auftrag von HMI quittiert worden ist: DB19.DBB26 == 0

Programmliste

DB19.DBB16 = <Nummer der Programmliste>

Nummer	Programmliste
129	/user/sinumerik/hmi/plc/programlist/plc_proglist_user.ppl
130	/oem_i/sinumerik/hmi/plc/programlist/plc_proglist_individual.ppl (nur SINUMERIK 828D)
131	/oem/sinumerik/hmi/plc/programlist/plc_proglist_manufacturer.ppl

Programmnummer

Die Programmnummer bezieht sich auf die in der angewählten Programmliste enthaltenen Programme.

DB19.DBB17 = <Programmnummer>

- Bereich user: 1 - 100
- Bereich individual: 101 - 200 (nur SINUMERIK 828D)
- Bereich oem: 201 - 255

Programmanwahl anfordern

DB19.DBX13.7 = 1

Programmanwahl: Quittierungsschnittstelle

Auftragsquittierung

- DB19.DBX26.7 == 1 (Anwahl erkannt)
- DB19.DBX26.3 == 1 (Programmanwahl wird durchgeführt)
- DB19.DBX26.2 == 1 (Fehler bei der Programmanwahl, siehe Fehlerkennung DB19.DBB27)
- DB19.DBX26.1 == 1 (Auftrag abgeschlossen)

Fehlerkennung

DB19.DBB27 == <Fehlerkennung>

Fehlerkennung	
Wert	Bedeutung
0	Kein Fehler
1	Ungültige Programmlisten-Nummer (DB19.DBB16)
3	Anwenderspez. Programmliste plc_proglist_main.ppl nicht gefunden (nur bei DB19.DBB16 ≠ 1, 2, 3)
4	Ungültige Programmnummer (DB19.DBB17)
5	Die Jobliste im angewählten Werkstück konnte nicht geöffnet werden.
6	Fehler in der Jobliste (Joblisten-Interpreter meldet Fehler).
7	Joblisten-Interpreter meldet leere Auftragsliste.

Programmanwahl: Auftragsablauf

Ein Auftrag zur Programmanwahl ist folgendermaßen durchzuführen:

1. Prüfen des Quittungsbytes: DB19.DBB26 == 0
Ist das Quittungsbyte nicht 0, ist der letzte Auftrag noch nicht abgeschlossen.
2. Vorgabe der Programmliste: DB19.DBB16
3. Vorgabe der Programmnummer: DB19.DBB17
4. Setzen der Anforderung zur Programmanwahl: DB19.DBX13.7 = 1
5. Auswerten der Quittierungs- und Fehlerschnittstelle: DB19.DBB26 und DBB27
Der Auftrag ist HMI-seitig noch nicht abgeschlossen, solange: DB19.DBX26.3 == 1 (Aktiv)
Der Auftrag ist HMI-seitig abgeschlossen, wenn eines der beiden Signale gesetzt ist:
- DB19.DBX26.1 == 1 (OK)
- DB19.DBX26.2 == 1 (Error)
6. Zum Abschluss des Auftrags muss die Anforderung zur Programmanwahl zurückgesetzt werden: DB19.DBX13.7 = 0
7. HMI signalisiert durch Rücksetzen des Quittungsbytes die Bereitschaft einen neuen Auftrag entgegenzunehmen: DB19.DBB26 == 0

Siehe auch

A2: Diverse NC/PLC-Nahtstellensignale und Funktionen (Seite 39)

14.13.3 Tastensperre aktivieren

Die Tastatur der Bedientafel sowie eine direkt am HMI angeschlossenen Tastatur kann durch folgendes Nahtstellensignal gesperrt werden

- 1. HMI: DB19.DBX0.2 = <Wert>
- 2. HMI: DB19.DBX50.2 = <Wert>

Wert	Bedeutung
0	Tastensperre inaktiv
1	Tastensperre aktiv

14.13.4 Bedienbereichsnummern

Die Nummer des aktiven Bedienbereichs wird standardmäßig angezeigt in: DB19.DBB21

Wenn der HMI-Monitor aktiv ist, wird die Nummer des aktiven Bedienbereichs nicht mehr in DB19.DBB21 sondern im anwenderspezifisch projektierten Bereich des HMI-Monitor (Seite 959) angezeigt.

Bedienbereich	Nummer
Maschine	201
Parameter	205
Programmierung	203
Programm-Manager	202
Diagnose	204
Inbetriebnahme	206

14.13.5 Bildnummern

Die aktuelle Bildnummer wird standardmäßig angezeigt in: DB19.DBW24

Wenn der HMI-Monitor aktiv ist, wird die aktuelle Bildnummer nicht mehr in DB19.DBW24 sondern im anwenderspezifisch projektierten Bereich des HMI-Monitor (Seite 959) angezeigt.

Bildnummernbereiche

Es gibt folgende Bildnummernbereiche:

- JOG, manuelle Maschine (Seite 951)
- Referenzpunktfahren (Seite 956)
- MDA (Seite 956)
- AUTOMATIK (Seite 956)
- Bedienbereich Parameter (Seite 957)
- Bedienbereich Programm (Seite 958)
- Bedienbereich Prog., Programm-Manager (Seite 958)
- Bedienbereich Diagnose (Seite 959)

14.13.5.1 Bildnummern: JOG, manuelle Maschine

Betriebsart JOG

Bild	Nummer
Technologie Drehen	
Zyklusstartmaske für alle Masken die übernommen werden können	81
Technologie Fräsen	
Zyklusstartmaske für alle Masken die übernommen werden können	3
Technologie Drehen / Fräsen	
Grundbild	19
T,S,M	2
NPV setzen	21
Positionieren	4
Planfräsen	18
Abspanen	80
Zyklusstartmaske für alle Anwendermasken	91
Einstellungen Allgemein	1
Einstellungen Mehrkanalfunktion	106
Einstellungen Kollisionsvermeidung	107
Einstellungen Messprotokoll	108
Schwenken	60
Alle G-Befehle	100
Zoom Istwert (MKS/WKS)	101
Gewinde Synchronisieren	102
Rückziehen	103
Handrad	104
Synchronisation Aktionen	105
Technologie Drehen: Werkstück, Nullpunkt	
Nullpunkt Werkstück (Grundmenü)	30
Anwendermaske	31
Anwendermaske	34
Anwendermaske	35
Anwendermaske	36
Anwendermaske	37
Anwendermaske	38
Anwendermaske	40
Messen Kante Z	5
Technologie Drehen: Werkstück, Messen	
Messen Werkzeug (Grundmenü)	50
Manuell X oder Anwendermaske	51
Manuell Y	71
Manuell Z oder Anwendermaske	52

Bild	Nummer
Lupe oder Anwendermaske	53
Anwendermaske	54
Anwendermaske	55
Abgleich Messtaster X oder Anwendermaske	56
Abgleich Messtaster Z oder Anwendermaske	57
Länge Automatisch in Z	58
Länge Automatisch in Y	73
Länge Automatisch in X	59
Technologie Fräsen: Werkstück, Nullpunkt	
Nullpunkt Werkstück (Grundmenü)	30
messen Kante X	5
messen Kante Y	22
messen Kante Z	23
Anwendermaske	7
Kante ausrichten oder Anwendermaske	31
Abstand 2 Kanten oder Anwendermaske	32
Rechtwinklige Ecke	33
Beliebige Ecke oder Anwendermaske	8
Rechtecktasche	34
1 Bohrung oder Anwendermaske	9
2 Bohrungen	35
3 Bohrungen	36
4 Bohrungen	37
Rechteckzapfen	38
1 Kreiszapfen oder Anwendermaske	10
2 Kreiszapfen	39
3 Kreiszapfen	40
4 Kreiszapfen	41
Ebene einrichten	42
Abgleich Taster Länge oder Anwendermaske	11
Abgleich Taster Radius	12
Technologie Fräsen: Werkstück, Messen	
Messen Werkzeug (Grundmenü)	50
- Länge messen Manuell (mit Fräswerkzeug) oder - Länge messen in X Manuell (mit Drehwerkzeug) oder - Anwendermaske	16
Länge messen in Y Manuell (mit Drehwerkzeug)	74
Länge messen in Z Manuell (mit Drehwerkzeug)	24
Durchmesser messen Manuell oder Anwendermaske	17
- Länge messen Automatisch (mit Fräswerkzeug) oder - Länge messen in X Automatisch (mit Drehwerkzeug) oder - Anwendermaske	13
Länge messen in Y Automatisch (mit Drehwerkzeug)	75
Länge messen in Z Automatisch (mit Drehwerkzeug)	25

Bild	Nummer
Durchmesser messen Automatisch oder Anwendermaske	14
Anwendermaske	51
Abgleich Messtaster oder Anwendermaske	15
Abgleich Festpunkt oder Anwendermaske	52
RunMyScreens (nur bei gesetzter JobShopIntegration)	
Anwendermaske für 1. horizontalen Softkey	96
Anwendermaske für 2. horizontalen Softkey	98
Anwendermaske für 3. horizontalen Softkey	99
Anwendermaske für 4. horizontalen Softkey	94
Anwendermaske für 5. horizontalen Softkey	95
Anwendermaske für 6. horizontalen Softkey	92
Anwendermaske für 7. horizontalen Softkey	97
Anwendermaske für 8. horizontalen Softkey	90
Anwendermaske für 9. horizontalen Softkey	83
Anwendermaske für 10. horizontalen Softkey	82
Anwendermaske für 11. horizontalen Softkey	93
Anwendermaske für 12. horizontalen Softkey	84
Anwendermaske für 13. horizontalen Softkey	85
Anwendermaske für 14. horizontalen Softkey	86
Anwendermaske für 15. horizontalen Softkey	87
Anwendermaske für 16. horizontalen Softkey	88

Betriebsart JOG, manuelle Maschine

DB19.DBB24	
Bild	Bildnummer
Technologie Drehen / Fräsen	
Kegel Drehen	61
Winkel Fräsen	62
Anschlag	63
Gerade	1300
Gerade alle Achsen	1330
Gerade X alpha	1340
Gerade Z alpha	1350
Kreis	1360
Bohren	1400
Bohren Mittig	1410
Bohren Gewinde Mittig	1420
Bohren Zentrieren	1433
Bohren Bohren	1434
Bohren Reiben	1435
Bohren Ausdrehen	1436

DB19.DBB24	
Bild	Bildnummer
Bohren Tieflochbohren	1440
Bohren Tieflochbohren 2	1441
Bohren Gewindebohren	1453
Bohrgewindefräsen	1455
Positionen	1473
Positionsreihe	1474
Positionsgitter	1477
Positionsrahmen	1478
Positionskreis	1475
Positionsteilkreis	1479
Hindernis	1476
Drehen	1500
Drehen Abspannen1	1513
Drehen Abspannen2	1514
Drehen Abspannen3	1515
Drehen Einstich1	1523
Drehen Einstich2	1524
Drehen Einstich3	1525
Drehen Freistich Form E	1533
Drehen Freistich Form F	1534
Drehen Freistich Gewinde DIN	1535
Drehen Freistich Gewinde	1536
Drehen Gewinde Längs	1543
Drehen Gewinde Kegel	1544
Drehen Gewinde Plan	1545
Drehen Gewinde Kette	1546
Drehen Abstich	1550
Fräsen	1600
Fräsen Planfräsen	1610
Fräsen Rechtecktasche	1613
Fräsen Kreistasche	1614
Fräsen Rechteckzapfen	1623
Fräsen Kreiszapfen	1624
Fräsen Längsnut	1633
Fräsen Kreisnut	1634
Fräsen offene Nut	1635
Fräsen Mehrkant	1640
Fräsen Gewindefräsen	1454
Fräsen Gravur	1670
Konturdrehen	1200
Konturdrehen Neue Kontur/Letzte Kontur	1210
Konturdrehen Konturabspannen	1220

DB19.DBB24	
Bild	Bildnummer
Konturdrehen Konturstechen	1230
Konturdrehen Konturstechdrehen	1240
Konturfräsen	1100
Konturfräsen Neue Kontur/Letzte Kontur	1110
Konturfräsen Bahnfräsen	1120
Konturfräsen Zentrieren	1130
Konturfräsen Vorbohren	1140
Konturfräsen Konturtasche	1150
Technologie Drehen: Simulation	
Seitenansicht	1740
Stirnansicht	1750
3D-Ansicht	1760
2 Fenster Ansicht	1770
Halbschnitt	1780
Technologie Drehen: Mitzeichnen	
Seitenansicht	1741
Stirnansicht	1751
3D-Ansicht	1761
2 Fensteransicht	1771
Maschinenraum	1791
Halbschnitt	1781
Technologie Fräsen: Simulation	
Draufsicht	1742
3D-Ansicht	1760
Von vorne	1744
Von hinten	1746
Von links	1748
Von rechts	1752
Halbschnitt	1780
Drehansicht	1782
Technologie Fräsen: Mitzeichnen	
Draufsicht	1743
3D-Ansicht	1761
Von vorne	1745
Von hinten	1747
Von links	1749
Von rechts	1753
Maschinenraum	1791
Halbschnitt	1781
Drehansicht	1783

14.13.5.2 Bildnummern: Referenzpunktfahren

Bild	Nummer
Zoom Istwert MKS/WKS	101

14.13.5.3 Bildnummern: MDA

Bild	Nummer
MDA	20
Alle G-Befehle	100
Zoom Istwert MKS/WKS	101
Handrad	104
Synchronisation Aktionen	105
Programmbeeinflussung	210
Einstellungen	250

14.13.5.4 Bildnummern: AUTOMATIK

Bild	Nummer
Automatik	200
Überspeichern	202
Programmbeeinflussung	210
Satzsuchlauf	220
Einstellungen Allgemein	250
Einstellungen Mehrkanalfunktion	106
Einstellungen Kollisionsvermeidung	107
Alle G-Befehle	100
Zoom Istwert MKS/WKS	101
Handrad	104
Synchronisation Aktionen	105
Technologie Drehen: Mitzeichnen	
Seitenansicht	243
Stirnansicht	244
3D-Ansicht	245
2 Fensteransicht	246
Maschinenraum	247
Halbschnitt	253
Technologie Fräsen: Mitzeichnen	
Draufsicht	242
3D-Ansicht	244
Von vorne	248
Von hinten	249

Bild	Nummer
Von links	251
Von rechts	252
Maschinenraum	247
Halbschnitt	253
Drehansicht	254

14.13.5.5 Bildnummern: Bedienbereich Parameter

Bild	Nummer
Werkzeugliste	600
Werkzeugverschleiß	610
Anwenderwerkzeugliste	620
Magazin	630
Nullpunktverschiebung	
Nullpunktverschiebung Aktiv	642
Nullpunktverschiebung Übersicht	643
Nullpunktverschiebung Basis	644
Nullpunktverschiebung G54 - G509	645
Details von Nullpunktverschiebung Aktiv, Übersicht, Basis oder G54 - G509	647
Anwendervariable	
R-Parameter	650
Globale GUD 1 (SGUD)	660
Globale GUD 2 (MGUD)	661
Globale GUD 3 (UGUD)	662
Globale GUD 4	663
Globale GUD 5	664
Globale GUD 6	665
Globale GUD 7	666
Globale GUD 8	667
Globale GUD 9	668
Kanal GUD 1 (SGUD)	690
Kanal GUD 2 (MGUD)	691
Kanal GUD 3 (UGUD)	692
Kanal GUD 4	693
Kanal GUD 5	694
Kanal GUD 6	695
Kanal GUD 7	696
Kanal GUD 8	697
Kanal GUD 9	698
Lokale LUD	681
Lokale LUD / PUD	684
Settingdaten	

Bild	Nummer
Arbeitsfeldbegrenzung	671
Spindeldaten	670
Spindelfutterdaten	672
CTRL-Energie	
CTRL-Energie Grundmenü	6170
CTRL-Energie Analyse	6171
CTRL-Energie Profile	6172
CTRL-Energie Analyse Grafik	6176
CTRL-Energie Analyse Langzeitmessung	6177
CTRL-Energie Analyse Details	6179
CTRL-Energie Messungen vergleichen	6178

14.13.5.6 Bildnummern: Bedienbereich Programm

Bild	Nummer
Technologie Drehen: Simulation	
Seitenansicht	413
Stirnansicht	414
3D-Ansicht	415
2 Fenster Ansicht	416
Halbschnitt	423
Technologie Fräsen: Simulation	
Draufsicht	412
3D-Ansicht	414
Von vorne	418
Von hinten	419
Von links	421
Von rechts	422
Halbschnitt	423
Drehansicht	424

14.13.5.7 Bildnummern: Bedienbereich Prog., Programm-Manager

Bild	Nummer
Verzeichnis NC	300
Lokales Laufwerk	325
USB / Projektirtes Laufwerk1	330
Projektirtes Laufwerk2	340
Projektirtes Laufwerk3	350
Projektirtes Laufwerk4	360
Projektirtes Laufwerk5	383

Bild	Nummer
Projektiertes Laufwerk6	384
Projektiertes Laufwerk7	385
Projektiertes Laufwerk8	386

14.13.5.8 Bildnummern: Bedienbereich Diagnose

Bild	Nummer
Alarmliste	500
Meldungen	501
Alarmprotokoll	502
NC/PLC-Variable	503

14.13.6 HMI-Monitor

Funktion

Der HMI-Monitor ist ein 8 Byte langer Datenbereich in einem frei wählbaren Datenbaustein, in dem vom HMI folgenden Daten dem PLC-Anwenderprogramm zur Verfügung gestellt werden:

- Bedienbereichsnummern (Seite 950)
- Bildnummern (Seite 950)

Parametrierung

Der Datenbereich wird über folgendes Anzeige-Maschinendatum parametriert:

MD9032 \$MM_HMI_MONITOR = "String"

mit "String" = "DB<DB-Nummer>.DDB<Byte-Adresse>"

Hinweis

Gerade Byte-Adresse

Der Datenbereich muss auf einer geraden Byte-Adresse beginnen.

Struktur des Datenbereichs

Byte	Bedeutung
EB n + 0	Aktiver SINUMERIK-Bedienbereich
EB n + 1	reserviert
EB n + 2	Aktuelle Bildnummer
EB n + 3	
EB n + 4	reserviert

Byte	Bedeutung
...	...
EB n + 7	reserviert

Randbedingungen

Bei aktivem HMI-Monitor, werden folgende PLC/HMI-Nahtstellensignale nicht mehr bearbeitet:

- DB19.DBB10 (PLC Hardkeys)
- DB19.DBB21 (Aktiver SINUMERIK -Bedienbereich)
- DB19.DBW24 (Aktuelle Bildnummer)

Beispiel

Annahmen

- Aktueller Bedienbereich: "Maschine", Nummer: 201
- Aktuelles Bild: "AUTOMATIK-Grundbild", Nummer: 200
- PLC-Datenbereich: DB60.DBB10

Parametrierung

- MD9032 \$MM_HMI_MONITOR = "DB60.DBB10"

Werte im Datenbereich

- DB60.DBB10: 201
- DB60.DBW12: 200

14.14 PLC-Funktionen für Antriebskomponenten am integrierten PROFIBUS

14.14.1 Überblick

Über die nachfolgend beschriebene Funktion können vom PLC-Anwenderprogramm der **Hardware-PLC** aus, Ein-/Ausgangsdaten von Antriebskomponenten am integrierten PROFIBUS zyklisch konsistent gelesen bzw. geschrieben werden. Dabei sind folgende Randbedingungen zu beachten:

- Zum Lesen/Schreiben von Ein-/Ausgangsdaten müssen die Systemfunktionen **SFC14/ SFC15** verwendet werden.
- Das Lesen/Schreiben von Ein-/Ausgangsdaten erfolgt immer über die **gesamte** Slotlänge.
- Die Funktion unterstützt ausschließlich zyklische, **nicht äquidistante** Datenübertragung.
- Ein Ausgangsslot darf nicht bereits NC-seitig belegt sein(z. B. Ausgangsslots von Antrieben).

14.14.2 Inbetriebnahme

Voraussetzungen

Bevor die Funktion NC-seitig in Betrieb genommen wird, müssen folgende Voraussetzungen erfüllt sein:

- Die Antriebskomponenten am integrierten PROFIBUS der NCU müssen mittels SIMATIC STEP 7, HW-Konfig vollständig konfiguriert sein.
- Die Konfiguration ist in die PLC geladen.

NC-Maschinendaten

In die nachfolgenden Maschinendaten sind die Anfangsadressen der zyklisch zu übertragenden Slots einzutragen:

- Eingangsslots:
MD10520 \$MN_PLCINTERN_LOGIC_ADDRESS_IN[<Index>] = <Slotadresse>
- Ausgangsslots:
MD10525 \$MN_PLCINTERN_LOGIC_ADDRESS_OUT[<Index>] = <Slotadresse>

<Index>: 0, 1, 2, ... (max. Slotanzahl) - 1

<Slotadresse>: Die in HW-Konfig parametrisierte Slotadresse

Hinweis

Maximale Anzahl von Daten

Die Summe aller zyklisch in Ein- und Ausgangsrichtung zu übertragenden Daten darf aktuell **2048** Byte nicht überschreiten.

14.14.3 Beispiel

Ermittlung der Slotadressen

Nach Anwahl des DP-Slave "SINAMICS_Integrated" am integrierten PROFIBUS "PROFIBUS Integrated: DP master system (3)" im Stationsfenster von HW-Konfig, werden in der Detailansicht dessen PROFIdrive-Telegramme und die dazugehörigen Slotadressen angezeigt:

- Telegramm 136: Antrieb
- Telegramm 391: Control Unit
- Telegramm **370**: Einspeisung

Die zur Paramtrierung in den NC-Maschinen erforderlichen Slotadressen der Einspeisung sind:

- Eingangsslot: Slot 31, Adresse **6514**
- Ausgangsslot: Slot 32, Adresse **6514**

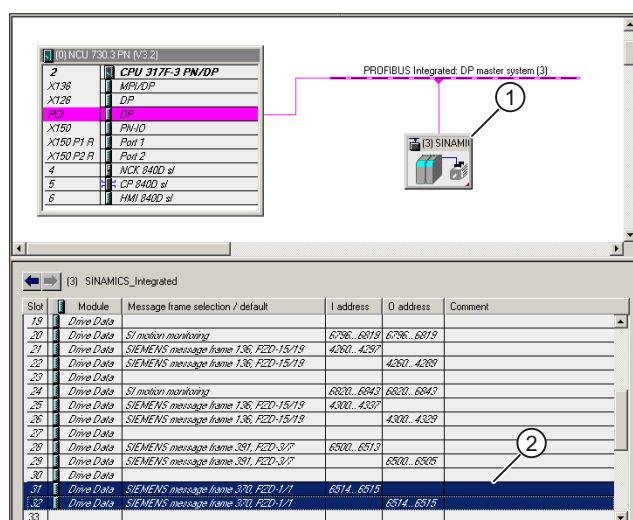


Bild 14-18 PROFIBUS-Telegramme der Einspeisung

NC-Maschinendaten einstellen

- Eingangsslots:
MD10520 \$MN_PLCINTERN_LOGIC_ADDRESS_IN[0] = 6514
- Ausgangsslots:
MD10525 \$MN_PLCINTERN_LOGIC_ADDRESS_OUT[0] = 6514

Steuern der ALM mittels FB390 "ALM_Control"

Die SINUMERIK Hardware-PLC ist über den internen PROFIBUS mit der CU320 des SINAMICS S120 verbunden. Der ALM ist über Drive-CLIQ mit der CU verbunden.

Im SINAMICS S120 kann für jedes Modul (CU, ALM, Motormodule, etc.) ein Steuer- und Status-Telegramm definiert werden. Wird dies für einen ALM gemacht, ist dieses vom PLC-Anwenderprogramm aus ein- und ausschaltbar.

Der SIMATIC S7 Baustein FB390 "ALM_Control" prüft den Status des ALM und ermöglicht es dem Anwender ihn ein- oder auszuschalten.

Über den nachfolgenden Link zum Industry Online Support steht die Beschreibung des Bausteins sowie ein Beispielprojekt zum Download zur Verfügung:

<http://support.automation.siemens.com/WW/view/de/49515414>

14.15 Speicherbedarf des PLC-Grundprogramms

Das Grundprogramm besteht aus Basisfunktionen und optionalen Funktionen. Zu den **Basisfunktionen** gehört der zyklische Signalaustausch NC ↔ PLC. Zu den **Optionen** gehören z. B. die FCs, die bei Bedarf eingesetzt werden können.

In der folgenden Tabelle ist der Speicherbedarf für die Basisfunktionen und die Optionen aufgelistet. Die Angaben stellen Richtwerte dar, sie sind vom jeweils aktuellen Softwarestand abhängig.

Speicherbedarf der Bausteine bei SINUMERIK 840D sl			
Baustein Typ, Nr.	Funktion	Bemerkung	Baustein-Größe (Byte)
Arbeitsspeicher			
Grundprogramm-Basisfunktionen			
FB1, FB15		muss geladen werden / auf CompactFlash Card	52182
FC2, 3, 5, 12		müssen geladen werden	470
DB4, 5, 7, 8		müssen geladen werden	1006
DB2, 3, 17		werden vom GP erzeugt	632
OB1, 40, 100, 82, 86		müssen geladen werden	398
		Summe	55698
Nahtstelle PLC / NC, PLC / HMI			
DB10	Signale PLC / NC	muss geladen werden	262
DB11	Signale PLC / BAG	wird vom GP erzeugt	56
DB19	Signale PLC / HMI	wird vom GP erzeugt	434
DB21 bis 30	Signale PLC / Kanal	werden vom GP abhängig von NC-MD erzeugt: Je DB	416
DB31 bis 61	Signale PLC / Achse, Spindel	werden vom GP abhängig von NC-MD erzeugt: je DB	148
Grundprogramm-Optionen			
Maschinensteuertafel			
FC19	Transfer MCP-Signale, M-Variante	muss bei M-Variante der MCP geladen werden	92
FC25	Transfer MCP-Signale, T-Variante	muss bei T-Variante der MCP geladen werden	92
FC24	Transfer MCP-Signale, schmale Variante	muss bei schmaler-Variante der MCP geladen werden	100
FC26	Transfer MCP-Signale, HT8-Variante	muss bei HT8 geladen werden	68
Bedienhandgerät			
FC13	Display-Steuerung BHG	kann bei Bedienhandgeräten geladen werden	144
Fehler-/Betriebsmeldungen			
FC10	Erfassung FM/BM	Laden bei Anwendung von FM / BM	66
ASUP			
FC9	ASUP-Start	Laden bei Anwendung von ASUPs von PLC	128

Grundprogramm-Optionen			
Stern- / Dreieck-Umschaltung			
FC17	Stern-/Dreieck-Umschaltung für HSA	Laden bei Stern-/Dreieck-Umschaltung	114
Spindelsteuerung			
FC18	Spindelsteuerung	Laden bei Spindelsteuerung von PLC	132
PLC-/NC-Kommunikation			
FB2	NC-Variable lesen	Laden bei NC-Variable lesen	76
DBn	NC-Variable lesen	ein Instanz-DB je FB2-Aufruf	270
FB3	NC-Variable schreiben	Laden bei NC-Variable schreiben	76
DBm	NC-Variable schreiben	ein Instanz-DB je FB3-Aufruf	270
FB4	PI-Dienste	Laden bei PI-Diensten	76
DBo	PI-Dienste	ein Instanz-DB je FB4-Aufruf	130
DB16	PI-Dienste Beschreibung	Laden bei PI-Diensten	618
FB5	GUD Variablen lesen	Laden bei PI-Diensten	76
DBp	GUD Variablen lesen	ein Instanz-DB je FB5-Aufruf	166
DB15	Allgemeine Kommunikation	Global-DB für Kommunikation	146
FB7	PI-Dienste 2	Laden bei PI-Diensten	76
DBo	PI-Dienste 2	ein Instanz-DB je FB4-Aufruf: je	144
FC21	Transfer	Laden bei DualPortRAM, ...	164
M zu N			
FB9	Umschaltung M zu N	Laden bei M zu N	58
Safety Integrated			
FB10	Sicherheitsrelais	Laden bei Option Safety	74
FB11	Bremsentest	Laden bei Option Safety	76
DB18	Safety Daten	DB für Safety	308
Werkzeug-Verwaltung			
FC7	Transfer-Funktion Revolver	Laden bei Option Werkzeugverwaltung	84
FC8	Transfer-Funktion	Laden bei Option Werkzeugverwaltung	132
FC22	Richtungsauswahl	Laden, wenn Richtungsauswahl benötigt	138
DB71	Beladestellen	wird vom GP abhängig von NC-MD erzeugt	40+30*B
DB72	Spindeln	wird vom GP abhängig von NC-MD erzeugt	40+48*Sp
DB73	Revolver	wird vom GP abhängig von NC-MD erzeugt	40+44*R
DB74	Basis-Funktion	wird vom GP abhängig von NC-MD erzeugt	100+(B+Sp+R)*22
Compile-Zyklen			
DB9	Nahtstelle PLC-Compile-Zyklen	wird vom GP abhängig von NC-Option erzeugt	436

Beispiel:

Anhand des in der vorigen Tabelle angegebenen Speicherbedarfs wurde für zwei Musterkonfigurationen der Speicherbedarf ermittelt (siehe folgende Tabelle).

Baustein Typ, Nr.	Funktion	Bemerkung	Baustein-Größe (Byte)
Arbeitsspeicher			
Minimal-Konfiguration (1 Spindel, 2 Achsen und T-MCP)			
s. o.	Grundprogramm, Basis		54688
	Nahtstellen-DBs		1612
	MCP		92
		Summe	56392
Maximal-Konfiguration (2 Kanäle, 4 Spindeln, 4 Achsen, T-MCP)			
s. o.	Grundprogramm, Basis		54688
s. o.	Nahtstellen-DBs		2768
s. o.	MCP		92
s. o.	Fehler- / Betriebsmeldungen		66
s. o.	ASUPs	1 ASUP-Anstoß	128
s. o.	Konkurrierende Achse	für 2 Revolver	132
s. o.	PLC- / NC-Kommunikation	1 x Variable lesen und 1 x Variable schreiben	838
s. o.	Werkzeug-Verwaltung	2 Revolver mit je 1 Beladestelle	674
s. o.	Compile-Zyklen		436
		Summe	59822

14.16 Rahmenbedingungen und NC-VAR-Selector

14.16.1 Rahmenbedingungen

14.16.1.1 Programmier- und Parametrierwerkzeuge

Hardware

Für die bei SINUMERIK 840D sl eingesetzten PLCs ist bei den Programmiergeräten oder PCs folgende Ausstattung erforderlich:

	Minimal	Empfehlung
Prozessor	Pentium	Pentium
RAM (MB)	256	512 oder mehr
Festplatte, freier Speicherplatz (MB)	500	> 500
Schnittstellen	MPI, Ethernet inkl. Kabel Memory-Card	
Grafik	SVGA (1024*768)	
Maus	ja	
Betriebssystem	Windows 2000 /XP Professional, ab STEP 7 Version 5.3 SP2	

Auf Geräten, die den oben aufgeführten Randbedingungen entsprechen, kann das erforderliche **STEP 7** installiert werden, sofern es nicht bereits zum Lieferumfang des PG gehört.

Mit diesem Paket sind folgende Funktionen möglich:

- Programmieren
 - Editoren und Compiler für AWL (kompletter Sprachumfang incl. SFB-/SFC-Aufrufe), KOP, FUP
 - Erstellen und Bearbeiten von Zuordnungslisten (Symbol-Editor)
 - DB-Editor
 - Ein- und Ausgabe von Bausteinen ON-/OFF-Line
 - Einfügen von Änderungen und Ergänzungen sowohl ON- als auch OFF-Line
 - Übertragen der Bausteine von PG nach PLC und zurück
- Parametrieren
 - Parametrierwerkzeug **HW Config** für CPU- und Peripherie-Parametrierung
 - Parametrierwerkzeug **NetPro** für Einstellung der CPU-Kommunikations-Parameter
 - Ausgabe von Systemdaten wie Hardware-, Softwarestand, Speicherausbau, Peripherieausbau/-belegung

- Test und Diagnose (ON-LINE)
 - Status/Steuern Variable (Ein-/Ausgänge, Merker, DB Inhalte, etc.)
 - Status einzelner Bausteine
 - Anzeige von Systemzuständen (USTACK, BSTACK, SZL)
 - Anzeige von Systemmeldungen
 - PLC-Stop/Neustart/Urlöschen auslösen von PG
 - PLC Komprimieren
- Dokumentation
 - Ausdruck einzelner oder aller Bausteine
 - Vergabe von symbolischen Namen (auch für Variable in DBs)
 - Ein- und Ausgabe von Kommentaren innerhalb jedes Bausteines
 - Ausdruck der Test und Diagnoseanzeigen
 - Hardcopy-Funktion
 - Querverweisliste
 - Programmübersicht
 - Belegungsplan E/A/M/T/Z/D
- Archivieren Dienstprogramme
 - Vergabe von Ausgabeständen einzelner Bausteine
 - Vergleichen von Bausteinen
 - Umverdrahten
 - Umsetzer STEP 5 → STEP 7
- Optionspakete
 - Programmierung in S7-HIGRAPH, S7-GRAPH, SCL.
Diese Pakete sind über den SIMATIC-Vertrieb bestellbar.
 - Zusatzpakete für die Projektierung von Baugruppen (z. B CP3425 → NCM-Paket)

Hinweis

Weitergehende Informationen zu den möglichen Funktionen sind den SIMATIC-Katalogen und der STEP 7-Dokumentation zu entnehmen.

14.16.1.2 Notwendige SIMATIC-Dokumentation

Literatur:

- Systembeschreibung SIMATIC S7
- Operationsliste S7-300
- Programmierung mit STEP 7
- Benutzerhandbuch STEP 7

- Programmierhandbuch STEP 7; Entwerfen von Anwenderprogrammen
- Referenzhandbuch STEP 7; Anweisungsliste AWL
- Referenzhandbuch STEP 7; Kontaktplan KOP
- Referenzhandbuch STEP 7; Standard- und Systemfunktionen
- Handbuch STEP 7: Konvertieren von STEP 5 Programmen
- STEP 7 Gesamtindex
- Handbuch CPU 317-2DP

14.16.1.3 Relevante SINUMERIK-Dokumente

Literatur:

- Inbetriebnahmehandbuch IBN CNC: NC, PLC, Antrieb
- Gerätehandbuch Bedienkomponenten und Vernetzung
- Funktionshandbuch Grundfunktionen
- Funktionshandbuch Erweiterungsfunktionen
- Funktionshandbuch Sonderfunktionen
- Listen sl (Buch1)
- Listen sl (Buch2)

14.16.2 NC-VAR-Selector

14.16.2.1 Übersicht

Allgemeines

Mit der PC-Applikation "NC-VAR-Selector" werden Adressen von benötigten NC-Variablen beschafft und für den Zugriff im PLC-Programm (FB2/FB3) aufbereitet. Damit wird es einem PLC-Programmierer ermöglicht, NC- und Antriebs-Variablen aus dem Gesamtangebot der NC- und Antriebs-Variablen auszuwählen, diese Auswahl an Variablen abzuspeichern und sie mittels eines Code-Generators für den STEP 7-Compiler aufzubereiten, um sie dann als ASCII-Datei (*.AWL) im Maschinen-CPU-Programm abzuspeichern. Das Bild "NC-VAR-Selector" gibt eine Vorstellung dieser Bearbeitung.

Zur Ablage der vom NC-VAR-Selector erzeugten Dateien ist ein Katalog über den Windows-Explorer mit beliebigem Katalognamen einzurichten. In diesen angelegten Katalog werden die selektierten Daten des NC-VAR-Selectors abgespeichert (Dateien Daten.VAR und Daten.AWL). Anschließend ist über den Menüpunkt "Code" → "in STEP 7 Projekt" die AWL-Datei zu übertragen und zu übersetzen. Alternativ ist über den STEP 7 Manager ein "Einfügen", "externe Quelle" für die Datei "Daten.AWL" in das STEP 7-Maschinenprojekt durchzuführen. Hierzu ist im Manager der Quell-Container anzuwählen. Hiermit wird diese Datei in die

Projektstruktur abgelegt. Nach der Übertragung der Datei sind diese AWL-Dateien mit STEP 7 zu übersetzen.

Hinweis

Für jeden NC-Softwarestand (auch ältere Versionen) kann der neueste NC-VAR-Selector verwendet werden. Für ältere NC-Software-Stände können die Variablen auch aus der neuesten Gesamtliste selektiert werden. Der Informationsinhalt im DB120 (Standard-DB für Variablen) ist nicht abhängig vom Softwarestand. D. h. selektierte Variablen in einem älteren Softwarestand müssen bei einer Software-Hochrüstung nicht neu selektiert werden.

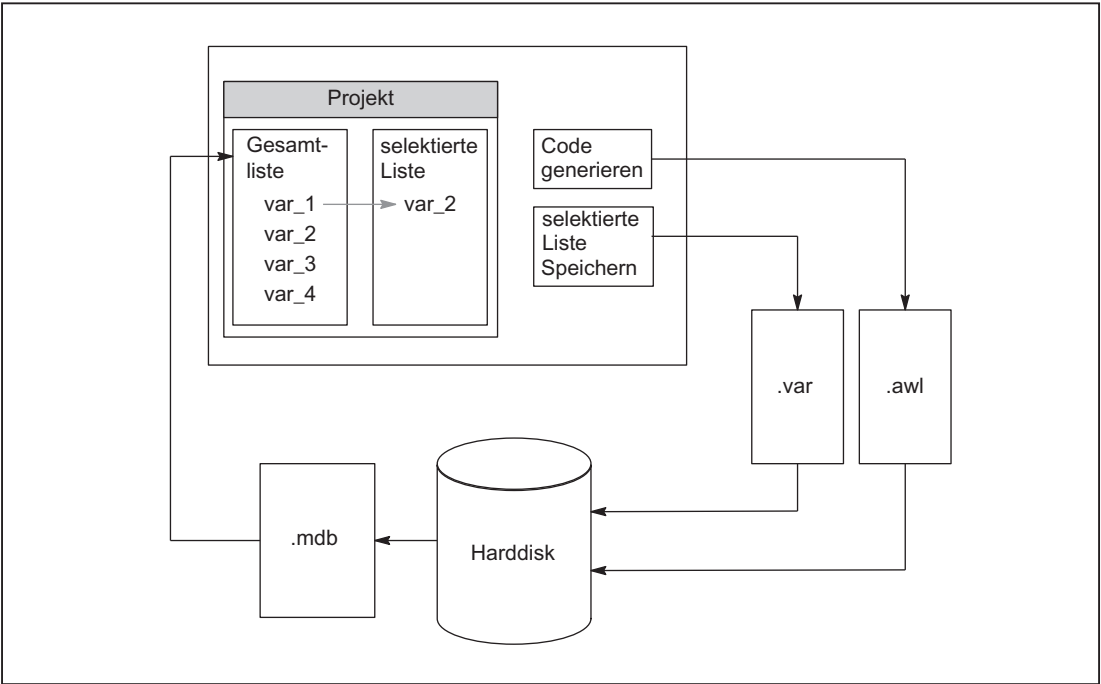


Bild 14-19 NC-VAR-Selector

Nach dem Starten der Applikation "NC-VAR-Selector" werden, nach Auswahl einer Variablenliste einer NC-Variante (Harddisk → File Ncv.mdb), alle in dieser Liste verfügbaren Variablen in einem Fenster angezeigt.

Es gibt folgende Variablen-Listen ncv*.mdb:

Variablen	Liste
Variablen der NC inklusive Maschinen-, Setting-Daten:	ncv_NcData.mdb
Parameter des Antriebs:	ncv_SinamicsServo.mdb

Es können vom Bediener Variablen in eine zweite Liste (weiteres Fenster) übernommen werden. Diese selektierten Variablen können anschließend in einer ASCII-Datei abgelegt sowie als STEP 7-Source-Datei (.awl) aufbereitet und abgespeichert werden.

Mit der erzeugten STEP 7-Datei ist der PLC-Programmierer in der Lage, nach Generierung eines PLC-Datenbausteins durch den STEP 7-Compiler, NC-Variablen zu lesen bzw. zu schreiben über die Grundprogramm-Funktionsbausteine "PUT" und "GET".

Die Liste der selektierten Variablen wird ebenfalls als ASCII-Datei abgelegt (Datei-Extension .var).

Die mit dem Werkzeug "NC-VAR-Selector" mitgelieferte Variablenliste ist passend zu dem aktuellen NC-Softwarestand. In dieser Liste sind keine vom Anwender definierten Variablen (GUD-Variablen) enthalten. Diese Variablen werden durch das Grundprogramm durch den Funktionsbaustein FB5 behandelt.

Hinweis

Mit der neuesten Version des "NC-VAR-Selectors" können alle vorherigen NC-Software-Stände bearbeitet werden. Es ist somit keine parallele Installation von verschiedenen Versionen des "NC-VAR-Selectors" nötig.

Systemmerkmale, Rahmenbedingungen

Die PC-Applikation "NC-VAR-Selector" setzt Windows 2000 oder ein höheres Betriebssystem voraus.

Die Namensgebung der Variablen ist beschrieben in:

Literatur:

Listen sl (Buch1); Kapitel: Variablen

bzw. auch in der Hilfedatei der Variablen (integriert im NC-VAR-Selector)

14.16.2.2 Funktionsbeschreibung

Übersicht

Folgendes Bild verdeutlicht den Umfang des NC-VAR-Selectors beim Einsatz in der STEP 7-Umgebung.

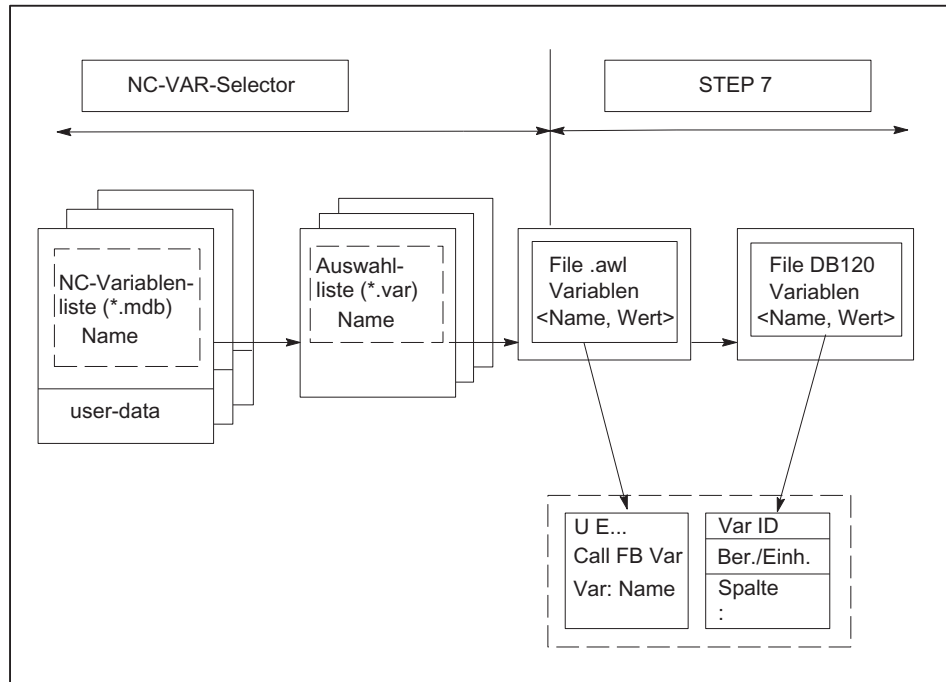


Bild 14-20 Einsatz des NC-VAR-Selectors in der STEP 7-Umgebung

Mit dem NC-VAR-Selector wird aus einer Variablenliste eine Liste selektierter Variablen erstellt und anschließend eine **.awl**-Datei erzeugt, die vom STEP 7-Compiler übersetzt werden kann.

- Ein *.awl-File enthält sowohl die Namen bzw. ALIAS-Namen als auch die Informationen für die Adressparameter der NC-Variablen. Ein daraus erzeugter Datenbaustein enthält nur noch die Adressparameter (10 Byte je Parameter).
- Die erzeugten Datenbausteine sind stets in der maschinenspezifischen Dateiablage entsprechend der STEP 7-Festlegungen zu speichern.
- Damit die Parametrierung der Bausteine GET/PUT (FB2/3) bezüglich der NC-Adressen symbolisch erfolgen kann, ist der frei vergebbare symbolische Name des erzeugten Datenbausteins in der STEP 7-Symbolliste aufzunehmen.

Grundbild / Grundmenü

Nach Anwahl (Starten) des NC-VAR-Selectors wird das Grundbild mit allen Bedienoptionen (obere Menüleiste) eingeblendet. Alle weiteren Fenster, die aufgeblendet werden, werden innerhalb des Gesamtfensters platziert.

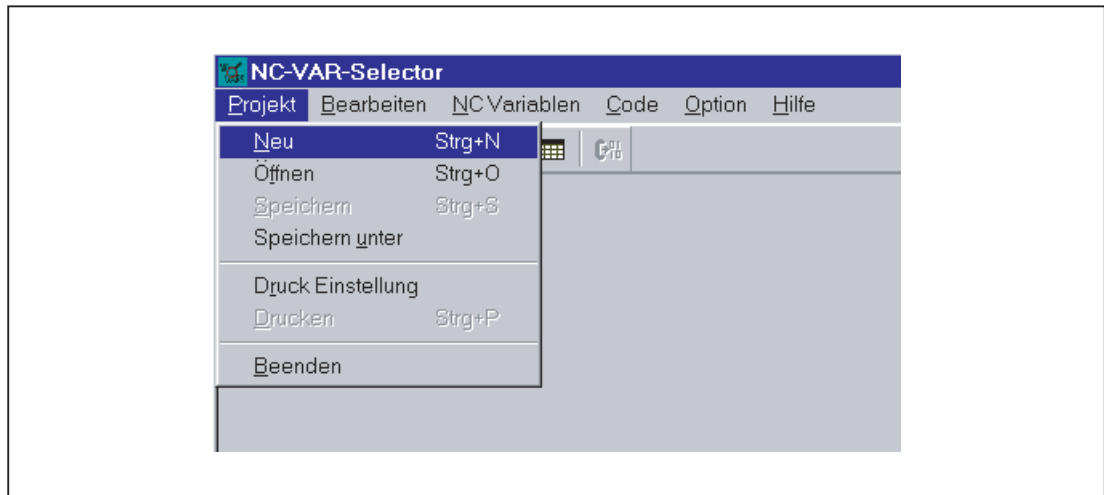


Bild 14-21 Grundbild mit Grundmenü

Menüpunkt Projekt

Unter diesem Menüpunkt werden sämtliche Bedienhandlungen durchgeführt, die mit dem Projektfile (File der selektierten Variablen) zusammenhängen.

Beenden der Applikation

Unter dem Menüpunkt "Projekt" wird über die Anwahl "Beenden" die Applikation beendet.

Anlegen eines neuen Projekts

Über den Menüpunkt "Projekt" wird ein neues Projekt (neue Datei für selektierte Variablen) erstellt.

Die Anwahl "NEU" blendet ein Fenster für die selektierten Variablen auf. Danach wird nach einer Abfrage die Dateiauswahl für die NC-Variablen-Liste angeboten (nur wenn die NC-Variablen-Liste noch nicht geöffnet ist).

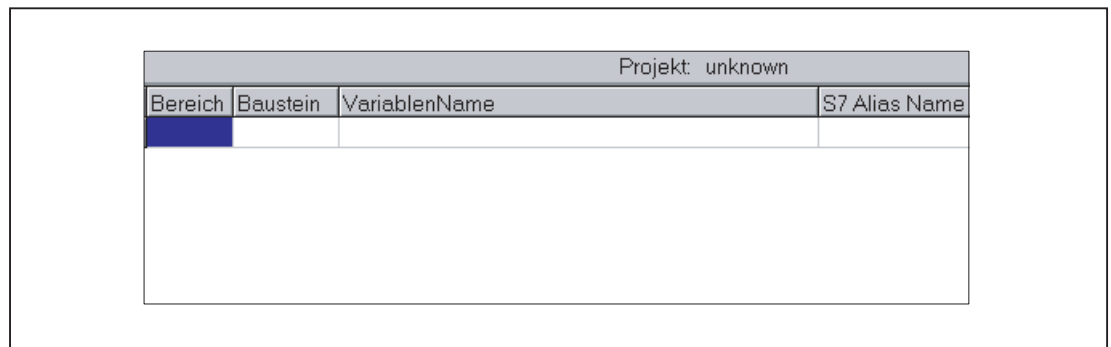


Bild 14-22 Fenster für selektierte Variable bei neuem Projekt

Die selektierten Variablen werden in einem Fenster dargestellt.

Öffnen eines bereits existierenden Projekts

Unter dem Menüpunkt "Projekt" kann über die Anwahl "Öffnen" ein bereits existierendes Projekt (bereits selektierte Variable) geöffnet werden. Es erscheint ein Datei-Auswahlfenster, mit dem das entsprechende Projekt mit der Erweiterung ".var" ausgewählt werden kann.

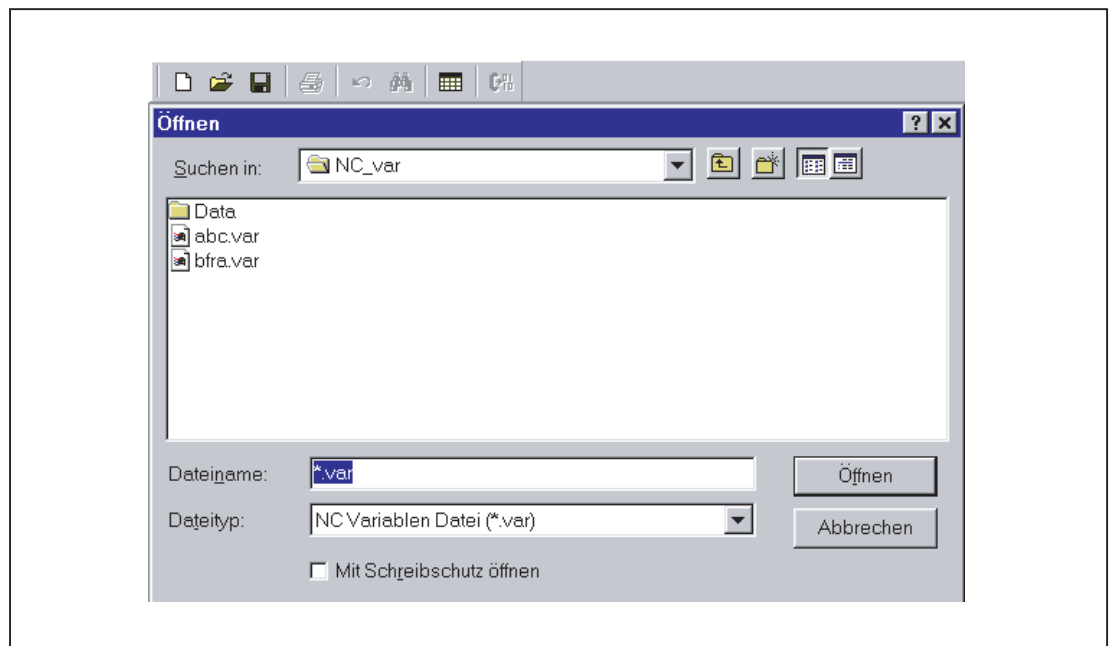


Bild 14-23 Auswahlfenster für vorhandene Projekte

Nach Auswahl des Projekts muss, falls Variable neu aufgenommen werden sollen, wiederum eine Gesamtliste von NC Variablen angewählt werden. Sollen nur Variable gelöscht werden, kann dies ohne Anwahl einer Gesamtliste erfolgen.

Speichern eines Projekts

Über die Menüpunkte "Projekt" > "Speichern" bzw. "Speichern unter ..." wird die Variablen-Liste gespeichert.

"Speichern" legt die Variablen-Liste unter einem bereits bekannten Pfad ab. Ist der Projektpfad nicht bekannt, ist das Verhalten wie bei "Speichern unter ...".

"Speichern unter ..." blendet ein Fenster auf, um den Pfad für das zu speichernde Projekt angeben zu können.

Drucken eines Projekts

Unter dem Menüpunkt "Projekt" kann über die Anwahl "Drucken" eine Projektdatei ausgedruckt werden. Die Zeilen je Seite werden durch den Menüpunkt "Druck Einstellung" vorgenommen. Standardeinstellung sind 77 Zeilen.

Menüpunkt Bearbeiten

Hier können z. B. folgende Bedienaktionen direkt durchgeführt werden:

- Variablen übernehmen
- Variablen löschen
- Aliasnamen ändern
- Variablen suchen

Weiterhin können diese Aktionen rückgängig gemacht werden.

Rückgängig machen

Bedienaktionen, die sich auf die Erstellung der Projektdatei beziehen (Variablen übernehmen, Variablen löschen, Aliasnamen ändern), können hier rückgängig gemacht werden.

Menüpunkt NC Variablen

Die Ablage der Basisliste aller Variablen erfolgt unter dem NC-Var-Selector-Pfad Data\Swxy (xy steht für SW-Stand-Nr., z. B. SW 5.3:=xy=53). Diese Liste kann als NC-Variablen-Liste angewählt werden. Bei SINUMERIK 840D sl sind die Basislisten unter dem Pfad Data\Swxy_sl enthalten.

Wählen einer NC-Variablen-Liste

Mit dem Menüpunkt "NC Variablen Liste", "Wählen" wird nun eine Liste der NC-Variablen einer NC-Version ausgewählt und angezeigt.

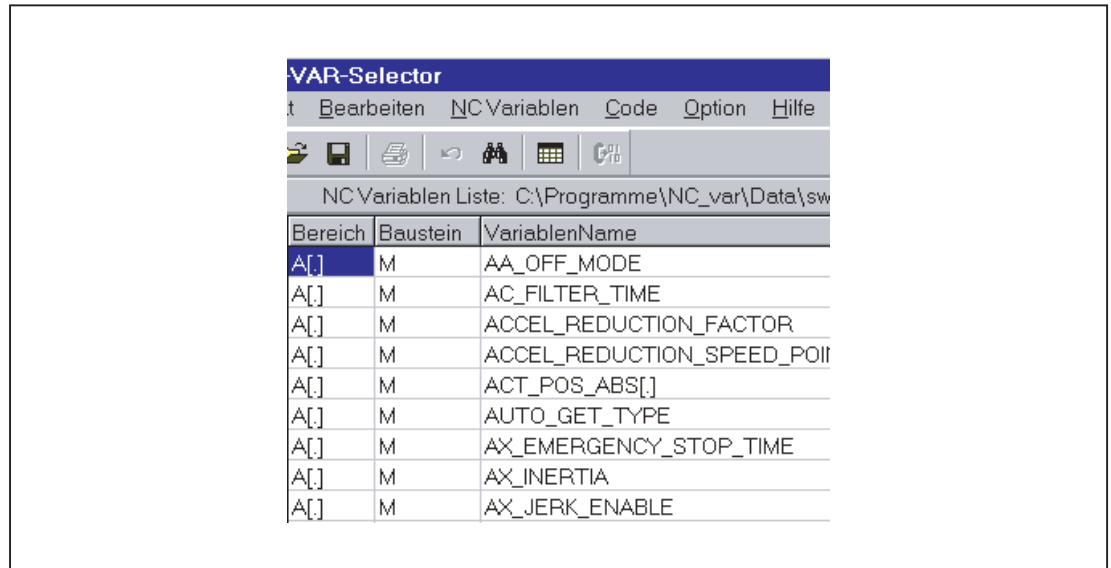


Bild 14-24 Fenster mit angewählter Gesamtliste

Die Feldvariablen (z. B. Achsbereich, T-Bereichsdaten usw.) werden mit Klammern ([.]) angedeutet. An dieser Stelle ist eine Zusatzinformation notwendig. Bei Übernahme der Variablen in die Projektliste wird die benötigte Zusatzinformation abgefragt.

Teilmengen anzeigen

Mit Doppelklick auf ein beliebiges Tabellenfeld (Ausnahme: Variablenfeld!) wird ein Fenster eingeblendet, in dem Filterkriterien vorgegeben werden können.

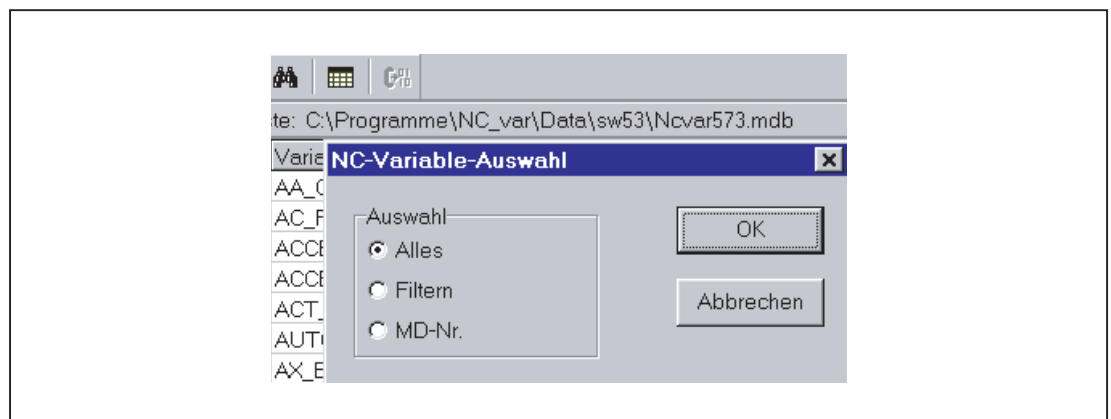


Bild 14-25 Fenster mit Filterkriterien für Anzeige der Variablen-Liste

Es stehen drei Optionen zur Verfügung:

- Alles anzeigen
- Bereich, Baustein und Name vorgeben (auch kombiniert)
- MD/SE-Daten-Nummer anzeigen

Es besteht auch die Möglichkeit, folgende Wildcards zu benutzen:

*	für eine beliebig lange Ergänzung des Suchkriteriums
---	--

Beispiel für Suchkriterien

Suchkriterium Name: CHAN* gefunden wird:

CHAN_NAME
chanAlarm
chanStatus
channelName
chanAssignment

- Variablen selektieren
Eine Variable wird mittels einfachem Mausklick selektiert und mit einem Doppelklick in das Fenster der selektierten Variablen übernommen. Unter dem Menüpunkt "Bearbeiten" kann diese Aktion auch wieder rückgängig gemacht werden.

Alias-Name

Die angebotenen Variablennamen können bis zu 32 Zeichen lang sein. Um Variablen im zu erzeugenden Datenbaustein eindeutig zu machen, wird der gewählte Name weiterhin um einige ASCII-Zeichen ergänzt. Der STEP 7-Compiler erkennt jedoch maximal 24 ASCII-Zeichen als eindeutige S7-Variable. Da nicht ausgeschlossen werden kann, dass Variablennamen sich nur in den letzten 8 Stellen unterscheiden, werden für die zu langen Namen **Alias**-Namen verwendet. Bei der Selektion einer Variablen wird deshalb geprüft, wie lang der zu verwendende S7-Name wird. Wird dieser länger als 24 Zeichen, muss über eine Zusatzeingabe ein vom Anwender vorzugebender Name (Alias-Name) eingesetzt werden.

Hierbei muss der Anwender darauf achten, dass dieser eindeutig ist.

Die Alias-Eingabe kann vom Anwender im Menü "Option" auch immer aktiviert werden. Die Alias-Eingabe ist dann bei jeder Übernahme einer Variablen möglich.

Weiterhin kann durch Doppelklick auf das S7-Variablennamen-Feld dieser Name nachträglich editiert werden. Unter dem Menüpunkt "Bearbeiten" kann diese Aktion auch wieder rückgängig gemacht werden.

7	A[.]	M	AX_EMERGENCY_STOP_TIME
8	A[.]	M	AX_INERTIA
9	A[.]	M	AX_JERK_ENABLE
10	A[.]	M	AX_JERK_TIME
Proje			
	Bereich	Baustein	VariablenName
1	A[1]	M	AA_OFF_MODE

Bild 14-26 Bild mit Gesamtliste und selektierten Variablen

Scrollen

Können nicht alle Variablen im Fenster angezeigt werden, wird ein Scrollbar eingeblendet. Mit Scrollen (Page-Up/Down) können die restlichen Variablen erreicht werden.

Variablen in mehrdimensionalen Strukturen

Werden Variable aus mehrdimensionalen Strukturen ausgewählt, so werden für die Adressierung dieser Variablen die Eingabe der Spalten- und/oder Zeilennummer sowie die Bereichs-Nummer abgefragt. Die erforderlichen Nummern können der NC-Variablen-Dokumentation entnommen werden.

Literatur:

Listen sl (Buch1); Variablen

Die Eingabe einer Null (0) als Bereichs-Nr., Zeilen- oder Spaltenindex bewirkt, dass die Variable im S7-AG als Zeiger auf diese Daten benutzt werden kann. Beim Lesen bzw. Schreiben dieser Daten über die Funktion "PUT" und "GET" müssen dann die optionalen Parameter "UnitX", "ColumnX" und "LineX" mit den notwendigen Informationen aufgefüllt werden.

Bild 14-27 Eingabefeld für Zeile, Spalte und Baustein-Nr.

Löschen von Variablen

Das Löschen von Variablen im Fenster der selektierten Variablen wird durch Auswahl (einfacher Mausklick) der Variablen und anschließendem Betätigen der Taste "Delete" ausgeführt. Für die Funktion Doppelklick gibt es keine Aktion. Eine Auswahl von mehreren Variablen zum Löschen ist möglich (siehe Abschnitt "Beispiel für Suchkriterien > Variablen selektieren").

Unter dem Menüpunkt "Bearbeiten" kann diese Aktion auch wieder rückgängig gemacht werden.

Hinweis

Beim Löschen von Variablen ergibt sich eine Veränderung der Absolutadressen der Zeiger-Strukturen auf die Variablen. Es ist bei der Änderung der Variablen-Selektion also zwingend erforderlich, **vor der Änderung** eine oder mehrere **Textdateien aller Anwender-Bausteine zu erzeugen**. Nur so kann nach erneuter Kompilierung sichergestellt werden, dass die Zuordnung der Variablen im FB "GET" oder FB "PUT" noch übereinstimmt.

Selektierte Liste speichern

Nach erfolgter Variablen Auswahl können diese unter einem Projektnamen angelegt werden. Die Ablage der Dateien erfolgt projektspezifisch.

Für die abzulegende Datei wird ein Fenster aufgeblendet, in welchem der Projektpfad und Name für diese Datei auszuwählen ist.

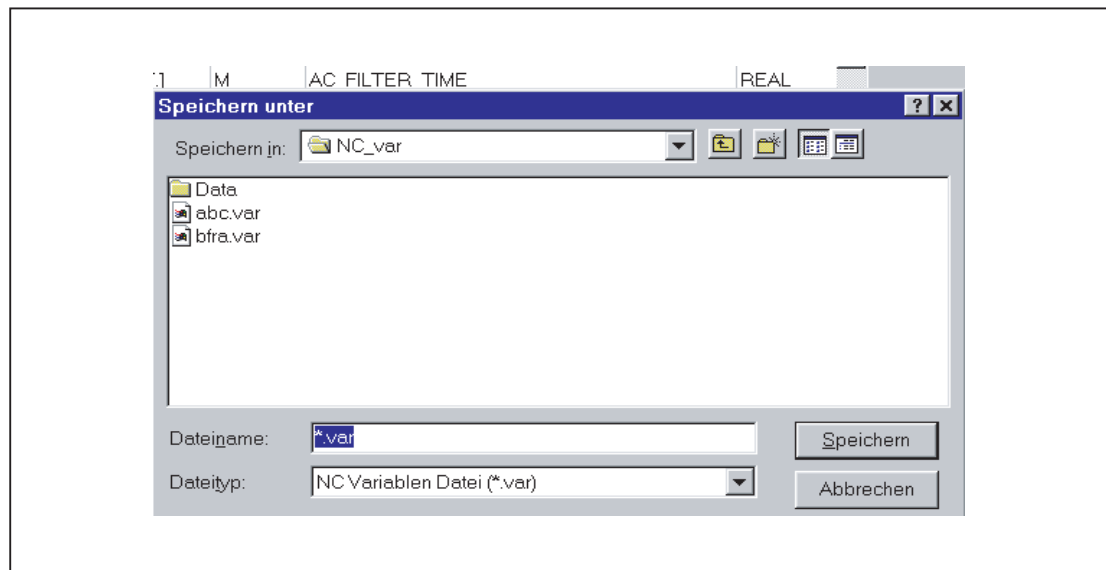


Bild 14-28 Fenster für Projektpfad und Name der abzulegenden Datei

Code-Generierung

Unter diesem Menüpunkt stehen drei Unterpunkte zur Auswahl:

1. Einstellungen (Vorgabe der zu erzeugenden Datenbausteinnummer) und weitere Einstellungen
2. Generieren (Datenbaustein generieren)
3. In STEP 7 Projekt (Übernahme des Datenbausteins in ein STEP 7-Projekt)

Einstellungen

Unter diesem Menüpunkt erfolgt die Eingabe der DB-Nummer und des Symbols für diese DB-Nummer, für den der Code erzeugt wird.

Unter dem Tabulator "Mass System" kann eine Auswahl getroffen werden, wie Maßsystem-behaftete Variablen in der PLC verrechnet werden.

Unter dem Tabulator "Generierung" kann die Erzeugung für das jeweilige Zielsystem festgelegt werden.

Generieren

Unter diesem Menüpunkt erfolgt die Einstellung der STEP 7-Datei aus der selektierten Variablen Liste mit der Erweiterung ".awl".

Mit "Anwahl" wird eine Datei erzeugt:

Eine .awl-Datei, die als Input für den STEP 7-Compiler genutzt werden kann.

Für die zu speichernde Datei wird ein Fenster eingeblendet, in welchem Pfad und Name für die zu erzeugende .awl-Datei anzugeben ist.

In STEP 7-Projekt

Die generierte AWL-Datei wird in ein auswählbares SIMATIC-Projekt (Programmpfad) übertragen und compiliert. Zusätzlich kann auch das Symbol übernommen werden. Diese Funktion ist ab STEP 7 Version 5.1 verfügbar. Dieser Vorgang dauert eine längere Zeit wegen Aufruf von STEP 7. Vor Übertragen einer neuen AWL-Datei ist im KOP/FUP/AWL-Editor das Dateifenster der AWL-Datei zu schließen.

Menüpunkt Option

Unter dem Menüpunkt "Option" kann gewählt werden:

- Die aktuelle Sprache
- Der Modus für die Alias-Eingabe (immer / > 24 Zeichen)

Menüpunkt Hilfe

Hier kann durch Auswahl des entsprechenden Untermenüpunktes nachgelesen werden:

- Die Bedienungsanleitung
- Die Variablenbeschreibung

Weiterhin werden hier das Copyright und die Versionsnummer angezeigt.

14.16.2.3 Inbetriebnahme, Installation

Die Installation der Windows-Applikation "NC-VAR-Selector" erfolgt über das mitgelieferte SETUP-Programm.

14.17 Bausteinbeschreibungen

14.17.1 FB1: RUN_UP - Grundprogramm, Anlaufteil

Funktion

Im Anlauf erfolgt die Synchronisation zwischen NC und PLC. Es werden die Datenbausteine für die NC/PLC-Anwendernahtstelle anhand der über Maschinendaten festgelegten NC-Konfiguration erzeugt und Grundprogramm-Parameter auf Plausibilität geprüft. Im Fehlerfall übergibt der FB1 eine Fehlerkennung in den Diagnosepuffer und überführt die PLC in den Stop-Zustand.

Anlaufart "Neustart"

Die integrierte PLC kennt nur die Anlaufart "Neustart". Dabei wird nach der grundlegenden Systeminitialisierung immer zuerst der Organisationsbaustein OB100 "Neustart" und anschließend der Organisationsbaustein OB1 "Zyklischer Betrieb" durchlaufen.

Eingangs-Parameter

Zur Parametrierung des Grundprogramms müssen nur die jeweils relevanten Parameter des Funktionsbausteins FB1 mit anwenderspezifischen Werte beschrieben werden. Die voreingestellten Werte im Instanz-Datenbaustein DB7 des FB1 brauchen nicht zugewiesen zu werden. Der Funktionsbaustein FB1 darf nur im Organisationsbaustein OB100 aufgerufen werden.

Ausgangs-Parameter

Die Ausgangsparameter des Funktionsbausteins FB1 können auch vom zyklischen Programmteil des Grundprogramms aus gelesen werden. Dazu stehen folgende zwei Möglichkeiten zu Verfügung:

1. Direktzugriff in den Instanz-Datenbaustein DB7 des FB1 in symbolischer Form.
Beispiel: "L gp_par.MaxChan", mit "gp_par" als symbolischer Name des DB7
2. Bei der Parametrierung des FB1 wird einem Ausgangsparameter ein Merker zugewiesen.
Im Grundprogramm wird dann der Merker gelesen, um den Wert des Ausgangsparameters zu ermitteln.
Beispiel: "MaxChan":= MW 20

Hinweis

Zur Belegung der FB1-Parameter für MCP und BHG siehe "Projektierung von Maschinensteuertafel, Bedienhandgerät, Direkttasten (Seite 932) ".

Deklaration der Funktion

```

FUNCTION_BLOCK FB1
VAR_INPUT
    MCPNum :          INT:=1;           // 0: keine MCP
                                           // 1: 1 MCP (default)
                                           // 2: 2 MCP

    MCP1In :          POINTER;          // Anfangsadr. Eingangssignale
                                           MCP1

    MCP1Out :         POINTER;          // Anfangsadr. Ausgangssignale
                                           MCP1

    MCP1StatSend :    POINTER;          // Status-DW für Senden MCP1
    MCP1StatRec :    POINTER;          // Status-DW für Empfangen MCP1
    MCP1BusAdr :      INT:=6;           // default
    MCP1Timeout :     S5TIME:=S5T#700MS;
    MCP1Cycl :        S5TIME:=S5T#200MS;
    MCP2In :          POINTER;          // Anfangsadr. Eingangssignale
                                           MCP2

    MCP2Out :         POINTER;          // Anfangsadr. Ausgangssignale
                                           MCP2

    MCP2StatSend :    POINTER;          // Status-DW für Senden MCP2
    MCP2StatRec :    POINTER;          // Status-DW für Empfangen MCP2
    MCP2BusAdr :      INT;
    MCP2Timeout :     S5TIME:=S5T#700MS;
    MCP2Cycl :        S5TIME:=S5T#200MS;
    MCPMPI :          BOOL:=FALSE;
    MCP1Stop :        BOOL:=FALSE;
    MCP2Stop :        BOOL:=FALSE;
    MCP1NotSend :     BOOL:=FALSE;
    MCP2NotSend :     BOOL:=FALSE;

```

14.17 Bausteinbeschreibungen

```

MCP SDB210 :      BOOL:=FALSE;
MCP CopyDB77 :    BOOL:=FALSE;
MCP BusType :     BYTE=B#16#0;
BHG :             INT:=0;           // Bedienhandgerät-Schnittstelle
                                   // 0: kein BHG
                                   // 1: BHG an MPI
                                   // 2: BHG an BTSS

BHGIn :           POINTER;          // Sendedaten des
                                   // Bedienhandgerätes

BHGOut :          POINTER;          // Empfangsdaten des
                                   // Bedienhandgerätes

BHGStatSend :     POINTER;          // Status-DW für Senden BHG
BHGStatRec :      POINTER;          // Status-DW für Empfangen BHG
BHGInLen :        BYTE:=B#16#6;     // Input 6 Byte
BHGOutLen :       BYTE:=B#16#14;    // Output 20 Byte
BHGTimeout :      S5TIME:=S5T#700MS;
BHGCycl :         S5TIME:=S5T#100MS;
BHGRecGDNo :      INT:=2;
BHGRecGBZNo :     INT:=2;
BHGRecObjNo :     INT:=1;
BHGSendGDNo :     INT:=2;
BHGSendGBZNo :    INT:=1;
BHGSendObjNo :    INT:=1;
BHGMPI :          BOOL:=FALSE;
BHGStop :         BOOL:=FALSE;
BHGNotSend :      BOOL:=FALSE;
NCCyclTimeout :   S5TIME:=S5T#200MS;
NCRunupTimeout :  S5TIME:=S5T#50S;
ListMDecGrp :     INT:=0;
NCKomm :          BOOL:=FALSE;
MMCToIF :         BOOL:=TRUE;
HWheelMMC :       BOOL:=TRUE;      // Handradanwahlen durch HMI
ExtendAlMsg :     BOOL;
MCP_IF_TCS :      BOOL;
ExtendChanAxMsg : BOOL;
MsgUser :         INT:=10;         // Anzahl der Userbereiche im DB2
UserIR :          BOOL:=FALSE;     // Anwenderprogramme im OB40,
                                   // Lokaldatenerweiterung beachten!

IRAuxfuT :        BOOL:=FALSE;     // T-Funktion in OB40 auswerten
IRAuxfuH :        BOOL:=FALSE;     // H-Funktion in OB40 auswerten
IRAuxfuE :        BOOL:=FALSE;     // DL-Funktion in OB40 auswerten
UserVersion :     POINTER;          // Zeiger auf String-Variable, die
                                   // in der Versionsanzeige der
                                   // Bedienoberfläche angezeigt wird

OpKeyNum :        INT;
Op1KeyIn :        POINTER;

```

```

Op1KeyOut :      POINTER;
Op1KeyBusAdr :   INT;
Op2KeyIn :       POINTER;
Op2KeyOut :      POINTER;
Op2KeyBusAdr :   INT;
Op1KeyStop :     BOOL;
Op2KeyStop :     BOOL;
Op1KeyNotSend :  BOOL;
Op2KeyNotSend :  BOOL;
OpKeyBusType :   BYTE;
IdentMcpBusAdr : INT;
IdentMcpProfilNo : BYTE;
IdentMcpBusType : BYTE;
IdentMcpStrobe : BOOL;
END_VAR

VAR_OUTPUT
MaxBAG :          INT;
MaxChan :         INT;
MaxAxis :         INT;
ActivChan :       ARRAY[1..10] OF
                  BOOL;
ActivAxis :       ARRAY[1..31] OF
                  BOOL;
UDInt :          INT;
UDHex :          INT;
UDReal :         INT;
IdentMcpType :    BYTE;
IdentMcpLengthIn : BYTE;
IdentMcpLengthOut : BYTE;
:
END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
MCPNum:	E	INT	0, 1, 2	Anzahl der aktiven MCP 0: keine MCP vorhanden
MCP1In: MCP2In:	E	POINTER	E0.0 ... E120.0 oder M0.0 ... M248.0 oder DBn DBX0.0 ... DBXm.0	Anfangsadresse für die Eingangssignale der betr. Maschinensteuertafel

Signal	Art	Typ	Wertebereich	Bedeutung
MCP1Out: MCP2Out:	E	POINTER	A0.0 ... A120.0 oder M0.0 ... M248.0 oder DBn DBX0.0 ... DBXm.0	Anfangsadresse für die Ausgangssignale der betr. Maschinensteuertafel
MCP1StatSend: MCP2StatSend:	E	POINTER	A0.0 ... A124.0 oder M0.0 ... M252.0 oder DBn DBX0.0 ... DBXm.0	nur bei Ethernet-MCP: Abschalten des Blinkens (siehe Kapitel "Umschaltung von Maschinensteuertafel, Bedienhandgerät (Seite 941)")
MCP1StatRec: MCP2StatRec:	E	POINTER	A0.0 ... A124.0 oder M0.0 ... M252.0 oder DBn DBX0.0 ... DBXm.0	Derzeit keine Bedeutung
MCP1BusAdr: MCP2BusAdr:	E	INT	1, 2, 3 ... 126	DP-Slave: PROFIBUS-Adresse
			192, 193, 194 .. 223	Ethernet MCP: DIP-Einstellung
MCP1Timeout: MCP2Timeout:	E	S5time	Empfehlung: 700 ms	Zyklische Lebenszeichen-Überwachung für die Maschinensteuertafel
MCP1Cycl: MCP2Cycl:	E	S5time	Empfehlung: 200 ms	Nur bei PROFIBUS relevant
MCPMPI:	E	BOOL	FALSE	Aus Kompatibilitätsgründen vorhanden
MCP1Stop: MCP2Stop:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Übertragung der Maschinensteuertafel-Signale starten
				1: Übertragung der Maschinensteuertafel-Signale anhalten
				DP-Slave: Slave wird deaktiviert
MCP1NotSend : MCP2NotSend:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Sende- und Empfangsbetrieb aktiviert 1: Nur Empfang der Maschinensteuertafel-Signale
MCPsDB210:	E	BOOL	false	Wegen Kompatibilität vorhanden
MCPCopyDB77:	E	BOOL	false	Wegen Kompatibilität vorhanden
MCPBusType:	E	BYTE	3, 4, 5, 6	b#16#yx: <ul style="list-style-type: none"> Bus-Typ MCP1: unteres Nibble (x) Bus-Typ MCP2: oberes Nibble (y) 3: PROFIBUS 4: PROFIBUS am MPI/DP-Anschluss 5: Ethernet 6: PROFINET Mischbetrieb ist möglich (siehe Kapitel "Projektierung von Maschinensteuertafel, Bedienhandgerät, Direktasten (Seite 932)")
BHG:	E	INT	0, 5	Bedienhandgerät-Schnittstelle: 0: kein BHG 5: BHG an Ethernet

Signal	Art	Typ	Wertebereich	Bedeutung
BHGIn:	E	POINTER	E0.0 ... E124.0 oder M0.0 ... M252.0 oder DBn DBX0.0 ... DBXm.0	Anfangsadresse Empfangsdaten der PLC vom Bedienhandgerät
BHGOut:	E	POINTER	A0.0 ... A124.0 oder M0.0 ... M252.0 oder DBn DBX0.0 ... DBXm.0	Anfangsadresse Sendedaten der PLC zum Bedienhandgerät
BHGStatSend:	E	POINTER	A0.0 ... A124.0 oder M0.0 ... M252.0 oder DBn DBX0.0 ... DBXm.0	Wegen Kompatibilität vorhanden
BHGStatRec:	E	POINTER	A0.0 ... A124.0 oder M0.0 ... M252.0 oder DBn DBX0.0 ... DBXm.0	Wegen Kompatibilität vorhanden
BHGInLen:	E	BYTE	BHG default: B#16#6 (6 Byte)	Wegen Kompatibilität vorhanden
BHGOutLen:	E	BYTE	BHG default: B#16#14 (20 Byte)	Wegen Kompatibilität vorhanden
BHGTimeout:	E	S5time	Empfehlung: 700 ms	Wegen Kompatibilität vorhanden
BHGCycl:	E	S5time	Empfehlung: 100 ms	Wegen Kompatibilität vorhanden
BHGRecGDNo:	E	INT	BHG default: 2	Ethernet DIP-Schalter
BHGRecGBZNo:	E	INT	BHG default: 2	Wegen Kompatibilität vorhanden
BHGRecObjNo:	E	INT	BHG default: 1	Wegen Kompatibilität vorhanden
BHGSendGDNo:	E	INT	BHG default: 2	Wegen Kompatibilität vorhanden
BHGSendGBZNo:	E	Int	BHG default: 1	Wegen Kompatibilität vorhanden
BHGSendObjNo:	E	INT	BHG default: 1	Wegen Kompatibilität vorhanden
BHGMPI:	E	BOOL	0 (FALSE)	Wegen Kompatibilität vorhanden
BHGStop:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Übertragung der Bedienhandgerät-Signale starten 1: Übertragung der Bedienhandgerät-Signale anhalten
BHGNotSend:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Sende und Empfangsbetrieb aktiviert 1: Nur Empfang der Bedienhandgerät-Signale
NCCyclTimeout:	E	S5time	Empfehlung: 200 ms	Zyklische Lebenszeichen-Überwachung NC
NCRunupTimeout:	E	S5time	Empfehlung: 50 s	Hochlauf-Überwachung NC
ListMDecGrp:	E	INT	0, 1, 2 ... 16	Aktivierung der erweiterten M-Gruppen-Decodierung 0: nicht aktiv 1...16: Anzahl der M-Gruppen

Signal	Art	Typ	Wertebereich	Bedeutung
NCKomm:	E	BOOL	0 (FALSE), 1 (TRUE)	PLC-NC-Kommunikationsdienste FB2, 3, 4, 5, 7 1: aktiv
MMCToIF:	E	BOOL	0 (FALSE), 1 (TRUE)	Übertragung der HMI-Signale an die Nahtstelle (Betriebsarten, Programmbeeinflussung, usw.) 1: aktiv
HWheelMMC:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Handradanwahl über HMI 1: Handradanwahl durch Anwenderprogramm
ExtendAIMsg:	E	BOOL	0 (FALSE), 1 (TRUE)	Aktivierung Erweiterungen Fehler- und Betriebsmeldungen des FC10 (siehe Abschnitt "Erweiterungen der PLC-Alarme über den Baustein FC10" im Kapitel "Nahtstelle PLC/HMI (Seite 905)")
MCP_IF_TCS	E	BOOL	0 (FALSE), 1 (TRUE)	FC19: Auswertung Taste R11 0: keine Auswertung 1: Taste R11 wirkt als manuelles Verfahren in Werkzeugorientierung (siehe Abschnitt "Kartesisches manuelles Verfahren" im Kapitel "FC19: MCP_IFM - Übertragung der MCP-Signale an die Nahtstelle (Seite 1102)")
ExtendChanAxMsg	E	BOOL	0 (FALSE), 1 (TRUE)	Aktivierung aller Bereiche für Fehler- und Betriebsmeldungen des FC10 (siehe Kapitel "FC10: AL_MSG - Fehler- und Betriebsmeldungen (Seite 1080)")
MsgUser:	E	INT	0, 1, 2 ... 64	Anzahl Userbereiche (DB2) – abhängig vom Parameter "ExtendAIMsg"
UserIR:	E	BOOL	0 (FALSE), 1 (TRUE)	Lokaldatenerweiterung OB40 notwendig zur Verarbeitung der dortigen Signale vom Anwender
IRAuxfuT:	E	BOOL	0 (FALSE), 1 (TRUE)	T-Funktion in OB40 auswerten
IRAuxfuH:	E	BOOL	0 (FALSE), 1 (TRUE)	H-Funktion in OB40 auswerten
IRAuxfuE:	E	BOOL	0 (FALSE), 1 (TRUE)	DL-Funktion in OB40 auswerten
UserVersion:	E	POINTER	P#DBn.DBXx.0	Zeiger auf String-Variable, die in der Versionsanzeige der Bedienoberfläche angezeigt wird Für die String-Variable (max. 41 Zeichen) gilt folgende Schreibweise: "<Name> <Version xx.xx.xx> <Datum yy/mm/dd>" Beispiel: "Test-Version 07.06.02 13/06/04"
OpKeyNum:	E	INT	0, 1, 2	Anzahl der aktiven Direkttasten-Module 0: keine Ethernet-Direkttasten vorhanden

Signal	Art	Typ	Wertebereich	Bedeutung
Op1KeyIn: Op2KeyIn:	E	POINTER	P#Ex.0 oder P#Mx.0 oder P#DBn.DBXx.0.	Anfangsadresse für die Eingangssignale der betreffenden Direktasten-Module
Op1KeyOut: Op2KeyOut:	E	POINTER	P#Ax.0 oder P#Mx.0 oder P#DBn.DBXx.0.	Anfangsadresse für die Ausgangssignale der betreffenden Direktasten-Module
Op1KeyBusAdr: Op2KeyBusAdr:	E	INT	1, 2, 3 ... 191	Direktasten über Ethernet: TCU-Index
Op1KeyStop: Op2KeyStop:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Übertragung der Direktasten-Signale starten 1: Übertragung der Direktasten-Signale anhalten
Op1KeyNotSend: Op2KeyNotSend:	E	BOOL	0 (FALSE), 1 (TRUE)	0: Sende und Empfangsbetrieb aktiviert 1: Nur Empfang der Direktasten-Signale
OpKeyBusType:	E	BYTE	b#16#55	Ethernet
IdentMcpBusAdr:	E	INT	1, 2, 3 ... 254	nur IE-Geräte
IdentMcpProfilNo:	E	BYTE	0, 1	Profil eines Geräts 0: vollständiges Gerät 1: nur Direktasten
IdentMcpBusType:	E	BYTE	b#16#5	nur IE-Geräte
IdentMcpStrobe:	E	BOOL	0 (FALSE), 1 (TRUE)	1: Abfrage aktivieren
MaxBAG:	A	INT	1, 2, 3 ... 10	Anzahl BAGs
MaxChan:	A	INT	1, 2, 3 ... 10	Anzahl Kanäle
MaxAxis:	A	INT	1, 2, 3 ... 31	Anzahl Achsen
ActivChan:	A	ARRAY [1..10] OF BOOL	0 (FALSE), 1 (TRUE)	Bitleiste aktive Kanäle
ActivAxis:	A	ARRAY [1..31] OF BOOL	0 (FALSE), 1 (TRUE)	Bitleiste aktive Achsen
UDInt:	A	INT	0,	Anzahl INTEGER-Maschinendaten im DB20
UDHex:	A	INT	---	Anzahl Hexadezimal-Maschinendaten im DB20
UDReal:	A	INT	---	Anzahl REAL-Maschinendaten im DB20
IdentMcpType:	A	BYTE	0, B#16#80, B#16#81, B#16#82 ... B#16#89	MCP-Type
IdentMcpLengthIn:	A	BYTE	---	Länge der MCP-Eingangsdaten (MCP → PLC)
IdentMcpLengthOut:	A	BYTE	---	Länge der MCP-Ausgangsdaten (PLC → MCP)

Tritt ein Fehler bei der Kommunikation mit einer Maschinensteuertafeln (MCP) oder Bedienhandgerät (BHG) auf, werden am HMI folgende Alarmer angezeigt und die Eingangssignale (MCP1In, MCP2In bzw. BHGIn) auf den Wert Null gesetzt:

Fehlerfall: MCP / BHG

- Alarm 400260: "MSTT 1 ausgefallen"
- Alarm 400261: "MSTT 2 ausgefallen"
- Alarm 400262: "BHG ausgefallen"

Ist eine Neusynchronisation zwischen PLC und MCP / BHG möglich, wird die Kommunikation wieder aufgenommen, die Fehlermeldung am HMI durch das Grundprogramm gelöscht und in die Eingangssignalen (MCP1In, MCP2In bzw. BHGIn) wieder Prozesswerte übertragen.

Beispiel: FB1-Aufruf im OB100

```

ORGANIZATION_BLOCK OB100
VAR_TEMP
    OB100_EV_CLASS :          BYTE;
    OB100_STRTUP   :          BYTE;
    OB100_PRIORITY :          BYTE;
    OB100_OB_NUMBR :          BYTE;
    OB100_RESERVED_1 :        BYTE;
    OB100_RESERVED_2 :        BYTE;
    OB100_STOP     :          WORD;
    OB100_RESERVED_3 :        WORD;
    OB100_RESERVED_4 :        WORD;
    OB100_DATE_TIME :        DATE_AND_TIME;
END_VAR
BEGIN
    CALL FB1, DB7(           // FB1-Aufruf, Instand-DB: DB7
        MCPNum :=            1,
        MCP1In :=            P#E0.0,
        MCP1Out :=           P#A0.0,
        MCP1StatSend :=      P#A8.0,
        MCP1StatRec :=       P#A12.0,
        MCP1BusAdr :=        6,
        MCP1Timeout :=       S5T#700MS,
        MCP1Cycl :=          S5T#200MS,
        NC-CyclTimeout :=    S5T#200MS,
        NC-RunupTimeout :=   S5T#50S);
    // HIER ANWENDERPROGRAMM EINFÜGEN
END_ORGANIZATION_BLOCK

```


14.17.2 FB2: GET - NC-Variable lesen

Funktion

Der Funktionsbaustein FB2 "GET" dient zum Lesen von Variablen aus dem Bereich NC.

Um die NC-Variablen zu referenzieren, werden diese zunächst mit dem Tool "NC-VAR-Selector" ausgewählt und in einen Datenbaustein als AWL-Quelle generiert. Für diesen Datenbaustein muss anschließend in der S7-Symbolliste ein Name vergeben werden. Beim Aufruf des FB2 werden die Variablenadresse in folgender Form übergeben: Parameter "Addr1" bis "Addr8" = "<DB-Name>.<S7-Name>"

Anforderung zum Lesen von NC-Variablen

Aufruf des FB2 mit positivem Flankenwechsel, Parameter "Req" = 0 → 1

S7-Namen der NC-Variablen: Parameter "Addr1" bis "Addr8" = "NCVAR".<S7-Name>"

Zeiger zum Schreiben der Variablenwerte: Parameter "RD1" bis "RD8" = "P#<Adresse>"

Abschluss der Lese-Anforderung

Lese-Anforderung erfolgreich beendet: Parameter "Done" == 1.

Lese-Anforderung mit Fehler beendet: Parameter "Error" == 1, Fehlerursache in Parameter "State"

Voraussetzungen

- Freigabe der NC/PLC-Kommunikation durch OB100, FB1-Parameter "NCKomm" = 1
- Für den Datenbaustein DB120 (Datenschnittstelle) muss mit dem S7-Symbol Editor in der S7-Symbolliste des S7-Projekts ein Symbol (Standard: NCVAR) vergeben werden. Über dieses Symbol wird dann in den FB2-Parametern "Addr<x>" die NC-Variable spezifiziert z.B. "ADDR1: = "NCVAR".<NC-Variable>"

Randbedingungen

- Der FB2 ist multiinstanzfähig.
- Jedem Aufruf des FB2 muss ein separater Instanz-DB aus dem Anwenderbereich zugeordnet werden.
- Beim Lesen von **kanalspezifischen** Variablen dürfen bei einem Aufruf des FB2 über "Addr1" bis "Addr8" nur Variablen von genau **einem** Kanal adressiert werden.
- Beim Lesen von **antriebsspezifischen** Variablen dürfen in einem Aufruf des FB2 über "Addr1" bis "Addr8" nur Variablen von genau **einem** Antriebsobjekt SERVO adressiert werden. Das Antriebsobjekt SERVO muss einer Maschinenachse der NC zugeordnet sein. Der Zeilenindex entspricht der logischen Antriebsnummer.
- In einem Leseauftrag dürfen nur Variablen des gleichen Bereiches, Kanal oder Antriebsobjekt, gelesen werden.

Hinweis**Fehlerfall**

Beim Lesen von Variablen aus unterschiedlichen Kanälen oder Antriebsobjekten, oder gleichzeitig von einem Kanal und einem Antriebsobjekt, erfolgt eine Fehlermeldung:

- "Error" == TRUE
- "State" == W#16#02

Variable Adressierung

Für einige NC-Variable ist es notwendig, im NC-VAR-Selector "Bereichs-Nr." und/oder "Zeile" bzw. "Spalte" auszuwählen. Dabei ist es möglich, einen Basistyp auszuwählen, d. h. "Bereichs-Nr.", "Spalte" und "Zeile" werden mit "0" vorbelegt. Im FB2 werden die Werte der vom NC-VAR-Selector vorgegebenen "Bereichs-Nr.", "Zeile" und "Spalte" auf "0" geprüft. Ist ein NC-VAR-Selector Wert == "0", wird der entsprechende Wert des FB2- Parameters übernommen. Vor dem Aufruf des FB2 müssen dazu die FB2-Parameter "Unit<x>", "Column<x>" und "Line<x>" mit <x> = 1 - 8, geschrieben werden.

Tabelle 14-1 Parameter-Entsprechung

FB2-Paramter	NC-VAR-Selector
Unit	Bereichs-Nr
Column	Spalte
Line	Zeile

In einem Auftrag können Variablen innerhalb **einer** Gruppe kombiniert werden:

Gruppe	Bereich				
1	C[1]	N	B	A	T
2	C[2]	N	B	A	T
3	V[.]	H[.]	---	---	---

Für Kanal 3 bis 10 gelten die gleichen Regeln, wie für Gruppe 1 und Gruppe 2 beispielhaft dargestellt.

Hinweis

Beim gleichzeitigen Lesen mehrerer Variablen vom Typ "String" kann die Anzahl der nutzbaren Variablen kleiner 8 sein.

Deklaration der Funktion

```
FUNCTION_BLOCK FB2
VAR_INPUT
```

```
Req :          BOOL;
NumVar :       INT;
Addr1 :        ANY;
Unit1 :        BYTE;
Column1 :      WORD;
Line1 :        WORD;
Addr2 :        ANY;
Unit2 :        BYTE;
Column2 :      WORD;
Line2 :        WORD;
Addr3 :        ANY;
Unit3 :        BYTE;
Column3 :      WORD;
Line3 :        WORD;
Addr4 :        ANY;
Unit4 :        BYTE;
Column4 :      WORD;
Line4 :        WORD;
Addr5 :        ANY;
Unit5 :        BYTE;
Column5 :      WORD;
Line5 :        WORD;
Addr6 :        ANY;
Unit6 :        BYTE;
Column6 :      WORD;
Line6 :        WORD;
Addr7 :        ANY;
Unit7 :        BYTE;
Column7 :      WORD;
Line7 :        WORD;
Addr8 :        ANY;
Unit8 :        BYTE;
Column8 :      WORD;
Line8 :        WORD;
END_VAR
VAR_OUTPUT
  Error :       BOOL;
  NDR :        BOOL;
  State :      WORD;
END_VAR
```

VAR_IN_OUT

```

RD1 : ANY;
RD2 : ANY;
RD3 : ANY;
RD4 : ANY;
RD5 : ANY;
RD6 : ANY;
RD7 : ANY;
RD8 : ANY;

```

END_VAR

Erläuterung der Formalparameter

Parameter	Art	Typ	Wertebereich	Bedeutung
Req:	E	BOOL	---	Auftragsstart mit positiver Flanke
NumVar:	E	INT	1 ... 8	Anzahl zu lesender Variablen: "Addr1" - "Addr8"
Addr1 - Addr8:	E	ANY	"DBName".<VarName>	Variablenbezeichner aus NC-VAR-Selector
Unit1 - Unit8:	E	BYTE	---	Bereichsadresse, optional für variable Adressierung
Column1 - Column8:	E	WORD	---	Spaltenadresse, optional für variable Adressierung
Line1 - Line8:	E	WORD	---	Zeilenadresse, optional für variable Adressierung
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Auftrag wurde negativ quittiert bzw. konnte nicht ausgeführt werden
NDR:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Auftrag wurde erfolgreich ausgeführt. Daten stehen zur Verfügung
State:	A	WORD	---	siehe Absatz "Fehlerkennungen"
RD1 - RD8:	E/A	ANY	P#Mm.n BYTE x ... P#DBnr.dbxm.n BYTE x	Zielbereich für gelesene Daten

Fehlerkennungen

NC-Variablen

State		Bedeutung	Hinweis
High-Byte ¹⁾	Low-Byte		
1 - 8	1	Zugriffsfehler	---
0	2	Fehler im Auftrag	falsche Zusammenstellung von Variablen in einem Auftrag

State		Bedeutung	Hinweis
High-Byte 1)	Low-Byte		
0	3	negative Quittung, Auftrag nicht ausführbar	interner Fehler, eventuelle Abhilfe: <ul style="list-style-type: none"> • Auftragsdaten überprüfen • NC-Reset
1 - 8	4	nicht ausreichend lokaler Anwenderspeicher zur Verfügung	Datentyp der gelesene Variable ist größer als in "RD1" - "RD8" angegeben
0	5	Formatwandlungsfehler	Fehler bei Wandlung vom Var.-Typ double: Variable liegt nicht im Bereich von S7-REAL
0	6	FIFO voll	Auftrag muss wiederholt werden, da die Warteschlange voll ist
0	7	Option nicht gesetzt	FB1-Parameter "NCKomm" ist nicht gesetzt
1 - 8	8	falscher Zielbereich (RD)	"RD1" - "RD8" dürfen keine Lokaldaten sein
0	9	Übertragung belegt	Auftrag muss wiederholt werden
1 - 8	10	Fehler bei variabler Adressierung	"Unit" oder "Column" / "Line" enthält den Wert 0
0	11	Variablenadresse ungültig	"Addr" (bzw. Variablenname), "Area", "Unit" überprüfen
0	12	NumVar == 0	Parameter "NumVar" prüfen
1 - 8	13	Referenz des ANY Datums falsch	Gefordertes "NcVar"-Datum ist nicht parametrisiert
1) High-Byte > 0 ⇒ Nummer der Variablen, bei der der Fehler auftrat			

Antriebsspezifische Variablen

Tritt beim Lesen/Schreiben einer antriebsspezifischen Variablen ein Fehler auf (DB1200.DBX3000.1 == 1) wird im Zugriffsergebnis eine Fehlernummer angezeigt, die auf den im PROFIdrive-Profil definierten Fehlernummern basiert.

State	Bedeutung
x	<Fehlernummer des PROFIdrive-Profils> + 20 _H bzw. 36 _D

Ermittlung der Bedeutung des Zugriffsergebnisses:

1. Berechnung der Fehlernummer des PROFIdrive-Profils
<Fehlernummer des PROFIdrive-Profils> = Zugriffsergebnis - 20_H bzw. 36_D.

2. Ermittlung der Bedeutung der Fehlernummer des PROFIdrive-Profils
Die Fehlernummern des PROFIdrive-Profils sind beschrieben in:

Literatur

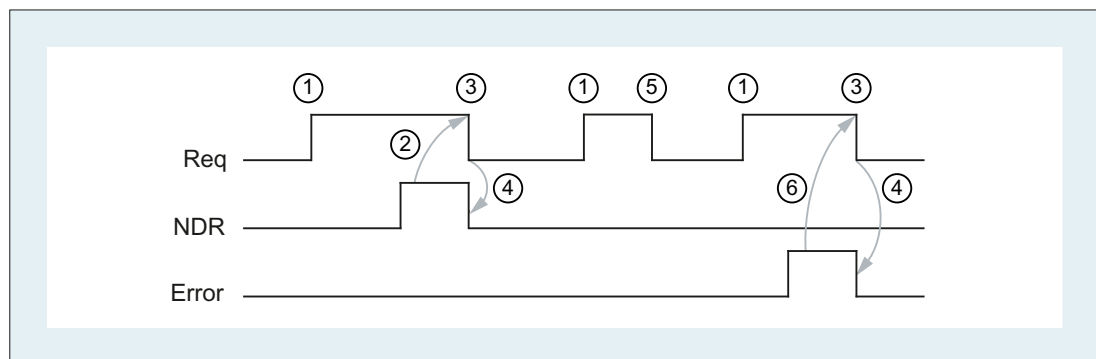
Funktionshandbuch Antriebsfunktionen SINAMICS S120; Kapitel "Kommunikation" > "Kommunikation nach PROFIdrive" > "Azyklische Kommunikation" > "Aufbau der Aufträge und Antworten" > Absatz "Fehlerwerte in Parameterantworten"

Projektierungsschritte

Für das Lesen von NC-Variablen sind folgende Projektierungsschritte nötig:

- Auswahl der Variablen mit dem NC-VAR-Selector
- Speichern der ausgewählten Variablen in einer Datei *.VAR
- Erzeugen einer STEP 7-Quelldatei *.AWL
- Erzeugen eines DBs mit den zugehörigen Adressangaben
- Eintrag des Symbols für den generierten DB in die Symboltabelle, damit im Anwenderprogramm symbolisch auf die Adressparameter zugegriffen werden kann
- Parametrierung des FB2

Impulsdiagramm



- ① Anwender: Anforderung setzen, Req = 0 → 1
- ② FB2 erfolgreich beendet, NDR = 1
Anwender: Anforderung zurücksetzen, IF NDR == 1 THEN Req = 0
- ③ Anwender: IF NDR == 1 THEN Anforderung zurücksetzen: 1 → 0
- ④ FB2 Auftragsbestätigung zurücksetzen, NDR = 0
- ⑤ Anwender: IF NDR == 0 UND Error == 0 THEN Anforderung zurücksetzen Req = 1 → 0 **nicht zulässig**
- ⑥ FB2 mit Fehler beendet, Error = 1
Anwender: Anforderung zurücksetzen, IF NDR == 1 OR Error == 1 THEN Req = 0, eventuell weitere Fehlerbehandlung

Aufrufbeispiel

Lesen von drei kanalspezifischen Maschinendaten aus Kanal 1, deren Adressangaben im DB120 hinterlegt werden.

Spezifikation der Daten

Die Daten werden mit dem NC-VAR-Selector ausgewählt und in der Datei DB120.VAR gespeichert. Anschließend wird daraus die Datei DB120.AWL erzeugt.

Es werden S7-(ALIAS-)Namen gewählt,

Um die Kanal-Bezeichnung in den Variablennamen aufzunehmen und die Zeichen "[" und "]" zu entfernen, die in einem STEP 7-Symbol nicht zulässig sind, werden neue S7-Namen gewählt:

Bereich	Baustein	Name	Typ	Nr.	Byte	S7-Name
C[1]	M	MD20070 \$MC_AXCONF_MACHAX_USED[1]	CHAR	20070	1	C1AxConfMachAxUsed1
C[1]	M	MD20070 \$MC_AXCONF_MACHAX_USED[2]	CHAR	20070	1	C1AxConfMachAxUsed2
C[1]	M	MD20090 \$MC_SPIND_DEF_MASTER_SPIND	INT	20090	1	C1SpindDefMasterSpind

S7-Symbol-Tabelle

Als symbolischer Name für den Datenbaustein DB120 wird "NCVAR" in die S7-Symboltabelle eingetragen:

Symbol	Operand	Datentyp
NCVAR	DB120	DB120

Die Datei DB120.AWL muss kompiliert und in die PLC übertragen werden.

Parametrierung des FB2 mit Instanz-DB110:

```

DATA_BLOCK DB110          // freier Anwender-DB, als Instanz für FB2
FB2
BEGIN
END_DATA_BLOCK
Function FC "VariablenCall" : VOID
    U      E 7.7;           // freie Taste Maschinensteuertafel
    S      M 100.0;         // Req aktivieren
    U      M 100.1;         // Fertigmeldung NDR
    R      M 100.0;         // Auftrag beenden
    U      E 7.6;           // Fehlerquittierung von Hand
    U      M 102.0;         // Fehler steht an
    R      M 100.0;         // Auftrag beenden
    CALL FB2, DB110(
        Req :=          M 100.0,
        NumVar :=        3,           // 3 Variablen lesen
        Addr1 :=         "NCVAR".C1AxConfMachAxUsed1,
        Addr2 :=         "NCVAR".C1AxConfMachAxUsed2,
        Addr3 :=         "NCVAR".C1SpindDefMasterSpind,
        Error :=         M102.0,
        NDR :=           M100.1,
        State :=         MW104,
        RD1 :=           P#DB99.DBX0.0 BYTE 1,
        RD2 :=           P#DB99.DBX1.0 BYTE 1,
        RD3 :=           P#M110.0 INT 1);

```

Beispiel: Variable Adressierung

Lesen von zwei R-Parametern aus Kanal 1, deren Adressangaben im DB120 als Basistyp hinterlegt werden. Die R-Parameter-Nummer wird über den Parameter "Line<x>" parametrieret.

```

DATA_BLOCK DB120
VERSION : 0.0
STRUCT
  C1_RP_rpa0_0 :
  STRUCT
    SYNTAX_ID :          BYTE := B#16#82;
    bereich_u_einheit :  BYTE := B#16#41;
    spalte :            WORD := W#16#1;
    zeile :             WORD := W#16#0;
    bausteintyp :       BYTE := B#16#15;
    ZEILENANZAHL :      BYTE := B#16#1;
    typ :               BYTE := B#16#F;
    laenge :            BYTE := B#16#8;
  END_STRUCT;
END_STRUCT;
BEGIN
END_DATA_BLOCK
CALL FB2 , DB110(
  Req :=          M 0.0,
  NumVar :=       2,
  Addr1 :=        "NCVAR".C1_RP_rpa0_0,
  Line1 :=        W#16#1,
  Addr2 :=        "NCVAR".C1_RP_rpa0_0,
  Line2 :=        W#16#2,
  Error :=        M 1.0,
  NDR :=          M 1.1,
  State :=        MW 2,
  RD1 :=          P#M 4.0 REAL 1,
  RD2 :=          P#M 24.0 REAL 1);

```

Zuordnung der Datentypen

Tabelle 14-2 Zuordnung der Datentypen

NC-Datentyp	S7-Datentyp
DOUBLE	REAL
DOUBLE	REAL2
FLOAT	REAL
LONG	DINT
INTEGER	DINT
UINT_32	DWORD

NC-Datentyp	S7-Datentyp
INT_16	INT
UINT_16	WORD
UNSIGNED	WORD
CHAR	CHAR oder BYTE
STRING	STRING
BOOL	BOOL
DATETIME	DATE_AND_TIME

Beispiel

Um z. B. eine Variable der NC vom Typ DOUBLE ohne Formatanpassung lesen zu können, muss im Zielbereich "RDx" ein ANY-Pointer mit Typ REAL2 angegeben werden (z. B.: P#M100.0 REAL2). Erkennt das PLC-Grundprogramm beim Lesen einer Variable vom Typ DOUBLE als Zieltyp REAL2, werden die Daten als 64-Bit Gleitkommazahl in den PLC-Datenbereich übernommen.

14.17.3 FB3: PUT - NC-Variable schreiben**Funktion**

Der Funktionsbaustein FB3 "PUT" dient zum Schreiben von Variablen aus dem Bereich NC.

Um die NC-Variablen zu referenzieren, werden diese zunächst mit dem Tool "NC-VAR-Selector" ausgewählt und in einen Datenbaustein als AWL-Quelle generiert. Für diesen Datenbaustein muss anschließend in der S7-Symbolliste ein Name vergeben werden. Beim Aufruf des FB3 werden die Variablenadressen in folgender Form übergeben: Parameter "Addr1" bis "Addr8" = "<DB-Name>.<S7-Name>"

Anforderung zum Schreiben von NC-Variablen

Aufruf des FB3 mit positivem Flankenwechsel, Parameter "Req" = 0 → 1

S7-Namen der NC-Variablen: Parameter "Addr1" bis "Addr8" = "NCVAR".<S7-Name>"

Zeiger zum Schreiben der Variablenwerte: Parameter "RD1" bis "RD8" = "P#<Adresse>"

Abschluss der Schreib-Anforderung

Schreib-Anforderung erfolgreich beendet: Parameter "Done" == 1.

Schreib-Anforderung mit Fehler beendet: Parameter "Error" == 1, Fehlerursache in Parameter "State"

Voraussetzungen

- Freigabe der NC/PLC-Kommunikation durch OB100, FB1-Parameter "NCKomm" = 1
- Für den Datenbaustein DB120 (Datenschnittstelle) muss mit dem S7-Symbol Editor in der S7-Symbolliste des S7-Projekts ein Symbol (Standard: NCVAR) vergeben werden. Über dieses Symbol wird dann in den FB3-Parametern "Addr<x>" die NC-Variable spezifiziert z.B. "ADDR1":= "NCVAR".<NC-Variable>"

Randbedingungen

- Der FB3 ist multiinstanzfähig.
- Jedem Aufruf des FB3 muss ein separater Instanz-DB aus dem Anwenderbereich zugeordnet werden.
- Um Maschinendaten und GUD ohne Passwort beschreiben zu können, muss die Schutzstufe des jeweiligen Datums auf die niedrigste Stufe undefiniert werden.

Literatur:

- Inbetriebnahmehandbuch; Kapitel: "Schutzstufenkonzept"
- Programmierhandbuch Arbeitsvorbereitung; Kapitel: "Schutzstufen für Anwenderdaten definieren"
- Beim Schreiben von **kanalspezifischen** Variablen dürfen bei einem Aufruf des FB2 über "Addr1" bis "Addr8" nur Variablen von genau **einem** Kanal adressiert werden.
- Beim Schreiben von **antriebsspezifischen** Variablen dürfen in einem Aufruf des FB2 über "Addr1" bis "Addr8" nur Variablen von genau **einem** Antriebsobjekt SERVO adressiert werden. Das Antriebsobjekt SERVO muss einer Maschinenachse der NC zugeordnet sein. Der Zeilenindex entspricht der logischen Antriebsnummer.
- In einem Schreibauftrag dürfen nur Variablen des gleichen Bereiches, Kanal oder Antriebsobjekt, geschrieben werden.

Hinweis**Fehlerfall**

Beim Schreiben von Variablen aus unterschiedlichen Kanälen oder Antriebsobjekten, oder gleichzeitig von einem Kanal und einem Antriebsobjekt, erfolgt eine Fehlermeldung:

- "Error" == TRUE
- "State" == W#16#02

Variable Adressierung

Für einige NC-Variable ist es notwendig, im NC-VAR-Selector "Bereichs-Nr." und/oder "Zeile" bzw. "Spalte" auszuwählen. Dabei ist es möglich, einen Basistyp auszuwählen, d. h. "Bereichs-Nr.", "Spalte" und "Zeile" werden mit "0" vorbelegt. Im FB2 werden die Werte der vom NC-VAR-Selector vorgegebenen "Bereichs-Nr.", "Zeile" und "Spalte" auf "0" geprüft. Ist ein NC-VAR-Selector Wert == "0", wird der entsprechende Wert des FB3- Parameters übernommen. Vor dem Aufruf des FB3 müssen dazu die FB3-Parameter "Unit<x>", "Column<x>" und "Line<x>" mit <x> = 1 - 8, geschrieben werden.

FB3-Paramter	NC-VAR-Selector
Unit<x>	Bereichs-Nr
Column<x>	Spalte
Line<x>	Zeile

In einem Auftrag können NC-Variablen innerhalb **einer** Gruppe kombiniert werden:

Gruppe	Bereich				
1	C[1]	N	B	A	T
2	C[2]	N	B	A	T
3	V[.]	H[.]	---	---	---
Für Kanal 3 bis 10 gelten die gleichen Regeln, wie für Gruppe 1 und Gruppe 2 beispielhaft dargestellt.					

Hinweis

Beim gleichzeitigen Schreiben mehrerer Variablen vom Typ "String" kann die Anzahl der nutzbaren Variablen kleiner 8 sein.

Deklaration der Funktion

```
FUNCTION_BLOCK FB3
VAR_INPUT
```

14.17 Bausteinbeschreibungen

```
Req :          BOOL;
NumVar :       INT;
Addr1 :        ANY;
Unit1 :        BYTE;
Column1 :      WORD;
Line1 :        WORD;
Addr2 :        ANY;
Unit2 :        BYTE;
Column2 :      WORD;
Line2 :        WORD;
Addr3 :        ANY;
Unit3 :        BYTE;
Column3 :      WORD;
Line3 :        WORD ;
Addr4 :        ANY;
Unit4 :        BYTE;
Column4 :      WORD;
Line4 :        WORD;
Addr5 :        ANY;
Unit5 :        BYTE;
Column5 :      WORD;
Line5 :        WORD;
Addr6 :        ANY;
Unit6 :        BYTE;
Column6 :      WORD;
Line6 :        WORD;
Addr7 :        ANY;
Unit7 :        BYTE;
Column7 :      WORD;
Line7 :        WORD;
Addr8 :        ANY;
Unit8 :        BYTE;
Column8 :      WORD;
Line8 :        WORD;
END_VAR
VAR_OUTPUT
  Error :       BOOL;
  Done :       BOOL;
  State :      WORD;
END_VAR
VAR_IN_OUT
```

```

SD1 :      ANY;
SD2 :      ANY;
SD3 :      ANY;
SD4 :      ANY;
SD5 :      ANY;
SD6 :      ANY;
SD7 :      ANY;
SD8 :      ANY;
END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Req:	E	BOOL	-	Auftragsstart mit positiver Flanke
NumVar:	E	INT	1 ... 8	Anzahl zu schreibender Variablen: "Addr1" - "Addr8"
Addr1 - Addr8:	E	ANY	"DBName".<VarName>	Variablenbezeichner aus NC-VAR-Selector
Unit1 - Unit8:	E	BYTE	-	Bereichsadresse, optional für variable Adressierung
Column1 - Column8:	E	WORD	-	Spaltenadresse, optional für variable Adressierung
Line1 - Line8:	E	WORD	-	Zeilenadresse, optional für variable Adressierung
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	Auftrag wurde negativ quittiert, bzw. konnte nicht ausgeführt werden
Done:	A	BOOL	0 (FALSE), 1 (TRUE)	Auftrag wurde erfolgreich ausgeführt.
State:	A	WORD	-	siehe Absatz "Fehlerkennungen"
SD1 - SD8:	E/A	ANY	P#Mm.n BYTE x... P#DBNr.dbxm.n BYTE x	zu schreibende Daten

Fehlerkennungen

NC-Variablen

State		Bedeutung	Hinweis
High-Byte ¹⁾	Low-Byte		
1 - 8	1	Zugriffsfehler	---
0	2	Fehler im Auftrag	falsche Zusammenstellung von Variablen in einem Auftrag
0	3	negative Quittung, Auftrag nicht ausführbar	interner Fehler, evtl. Abhilfe: <ul style="list-style-type: none"> • Auftrag überprüfen • NC-Reset

State		Bedeutung	Hinweis
High-Byte 1)	Low-Byte		
1 - 8	4	Datenbereiche oder Datentypen stimmen nicht überein oder String ist leer	Daten in "SD1" - "SD8" überprüfen
0	6	FIFO voll	Auftrag muss wiederholt werden, da die Warteschlange voll ist
0	7	Option nicht gesetzt	FB1-Parameter "NCKomm" ist nicht gesetzt
1 - 8	8	falscher Zielbereich (SD)	"SD1" - "SD8" dürfen keine Lokaldaten sein
0	9	Übertragung belegt	Auftrag muss wiederholt werden
1 - 8	10	Fehler bei variabler Adressierung	"Unit" oder "Column" / "Line" enthält den Wert 0
0	11	Variablenadresse ungültig oder Variable nur lesbar	"Addr" (bzw. Variablenname), "Area", "Unit" überprüfen
0	12	NumVar == 0	Parameter "NumVar" prüfen
1 - 8	13	Referenz des ANY-Datums falsch	Gefordertes "NcVar"-Datum ist nicht parametrisiert
1 - 8	15	Nutzdaten zu lang	Abhilfe: Weniger Variable pro Auftrag schreiben oder kürzere Stringvariablen verwenden
1) High-Byte > 0 ⇒ Nummer der Variablen, bei der der Fehler auftrat			

Antriebsspezifische Variablen

Tritt beim Lesen/Schreiben einer antriebsspezifischen Variablen ein Fehler auf (DB1200.DBX3000.1 == 1) wird im Zugriffsergebnis eine Fehlernummer angezeigt, die auf den im PROFIdrive-Profil definierten Fehlernummern basiert.

State	Bedeutung
x	<Fehlernummer des PROFIdrive-Profiles> + 20 _H bzw. 36 _D

Ermittlung der Bedeutung des Zugriffsergebnisses:

1. Berechnung der Fehlernummer des PROFIdrive-Profiles
 <Fehlernummer des PROFIdrive-Profiles> = Zugriffsergebnis - 20_H bzw. 36_D.
2. Ermittlung der Bedeutung der Fehlernummer des PROFIdrive-Profiles
 Die Fehlernummern des PROFIdrive-Profiles sind beschrieben in:
Literatur
 Funktionshandbuch Antriebsfunktionen SINAMICS S120; Kapitel "Kommunikation" >
 "Kommunikation nach PROFIdrive" > "Azyklische Kommunikation" > "Aufbau der Aufträge
 und Antworten" > Absatz "Fehlerwerte in Parameterantworten"

Projektierungsschritte

Für das Schreiben von NC-Variablen sind die gleichen Projektierungsschritte wie für das Lesen von NC-Variablen notwendig. Es ist zweckmäßig, die Adressangaben aller NC-Variablen, die gelesen oder geschrieben werden sollen, in einem DB zu hinterlegen.


```

BEGIN
Function FC "VariablenCall" : VOID
END_DATA_BLOCK
  U      E 7.7;          // freie Taste Maschinensteuertafel
  S      M 100.0;        // Req aktivieren
  U      M 100.1;        // Fertigmeldung Done
  R      M 100.0;        // Auftrag beenden
  U      E 7.6;          // Fehlerquittierung von Hand
  U      M 102.0;        // Fehler steht an
  R      M 100.0;        // Auftrag beenden
  CALL FB3, DB111(
    Req :=                M 100.0,
    NumVar :=              3,                // 3 Variablen schreiben
    Addr1 :=               "NCVAR".rpa_5C1RP,
    Addr2 :=               "NCVAR".rpa_11C1RP,
    Addr3 :=               "NCVAR".rpa_14C1RP,
    Error :=               M102.0,
    Done :=                M100.1,
    State :=               MW104,
    SD1 :=                 P#DB99.DBX0.0 REAL 1,
    SD2 :=                 P#DB99.DBX4.0 REAL 1,
    SD3 :=                 P#M110.0 REAL 1);

```

Beispiel: Variable Adressierung

Schreiben von zwei R-Parametern von Kanal 1, deren Adressangaben im DB120 als Basistyp hinterlegt werden. Die R-Parameter-Nummer wird über den Parameter LineX parametrisiert.

```

DATA_BLOCK DB120
VERSION : 0.0
STRUCT
  C1_RP_rpa0_0:
  STRUCT
    SYNTAX_ID :          BYTE := B#16#82;
    bereich_u_einheit :  BYTE := B#16#41;
    spalte :            WORD := W#16#1;
    zeile :             WORD := W#16#0;
    bausteintyp :       BYTE := B#16#15;
    ZEILENANZAHL :      BYTE := B#16#1;
    typ :               BYTE := B#16#F;
    laenge :           BYTE := B#16#8;
  END_STRUCT ;
END_STRUCT ;
BEGIN
END_DATA_BLOCK
CALL FB3, DB122(

```



```

Req :=                M 10.0,
NumVar :=             2,
Addr1 :=              "NCVAR".C1_RP_rpa0_0,
Line1 :=              W#16#1,
Addr2 :=              "NCVAR".C1_RP_rpa0_0,
Line3 :=              W#16#2
Error :=              M 11.0,
Done :=               M 11.1,
State :=              MW 12,
SD1 :=                P#M 4.0 REAL 1,
SD2 :=                P#M 24.0 REAL 1);

```

Zuordnung der Datentypen

Siehe Tabelle "Zuordnung der Datentypen" im Kapitel "FB2: GET - NC-Variable lesen (Seite 989)".

14.17.4 FB4: PI_SERV - PI-Dienst anfordern

Funktion

Der Funktionsbaustein FB4 "PI_SERV" dient zum Starten von PI-Diensten.

Die zur Verfügung stehenden PI-Dienste sind mit ihren spezifischen Parametern in den nachfolgenden Kapitel beschrieben. Ein Überblick über die vorhandenen PI-Dienste findet sich in: Liste verfügbarer PI-Dienste (Seite 1009)

Hinweis

Aufgrund der größeren Anzahl an Parameter "WVar", wird empfohlen anstelle des FB4 den Funktionsbaustein FB7 zu verwenden. Siehe Kapitel "FB7: PI_SERV2 - PI-Dienst anfordern (Seite 1043)".

Start eines PI-Dienstes

Anforderung zum Start eines PI-Dienstes: Aufruf des FB4 mit positivem Flankenwechsel, Parameter "Req" = 0 → 1

Abschluss eines PI-Dienstes

Auftrag bzw. PI-Dienst erfolgreich beendet: Parameter "Done" == 1.

Auftrag bzw. PI-Dienstes mit Fehler beendet: Parameter "Error" == 1, Fehlerursache in Parameter "State"

Voraussetzungen

- Freigabe der NC/PLC-Kommunikation durch OB100, FB1-Parameter "NCKomm" = 1
- Für den Datenbaustein DB16 (Datenschnittstelle der PI-Dienste) muss mit dem S7-Symbol Editor in der S7-Symbolliste des S7-Projekts ein Symbol (Standard: PI) vergeben werden. Über dieses Symbol wird dann im FB4-Parameter "PIService" der angeforderte PI-Dienst spezifiziert z. B. "PIService:= "PI".<PI-Dienst>"

Randbedingungen

- Jedem Aufruf des FB4 muss ein separater Instanz-DB aus dem Anwenderbereich zugeordnet werden.
- Der Start eines PI-Dienstes (FB4-Aufruf mit "Req" = 1) ist nur im zyklischen Teil des PLC-Grundprogramms (OB1) zulässig.
Wird der PI-Dienst nicht gestartet (FB4-Aufruf mit "Req" = 0), können die Parameter auch schon im Anlaufteil des PLC-Grundprogramms (OB100) geschrieben werden. Der Start des PI-Dienstes kann dann mit den bereits geschriebenen Parametern im im zyklischen Teil des PLC-Grundprogramms (OB1) durch Aufruf des FB4 mit "Req" = 1 durchgeführt werden.
- Die Ausführung eines PI-Dienstes erstreckt sich in der Regel über mehrere PLC-Zyklen.

Deklaration der Funktion

```
FUNCTION_BLOCK FB4
VAR_INPUT
w   Req :          BOOL;
    PIService :    ANY;
    Unit :        INT;
    Addr1 :       ANY;
    Addr2 :       ANY;
    Addr3 :       ANY;
    Addr4 :       ANY;
    WVar1 :       WORD;
    WVar2 :       WORD;
    WVar3 :       WORD;
    WVar4 :       WORD;
    WVar5 :       WORD;
    WVar6 :       WORD;
    WVar7 :       WORD;
    WVar8 :       WORD;
    WVar9 :       WORD;
    WVar10 :      WORD;
END_VAR
VAR_OUTPUT
```

```

Error :          BOOL;
Done  :          BOOL;
State :          WORD;

END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Req:	E	BOOL	0 (FALSE), 1 (TRUE)	Auftragsanforderung
PIService:	E	ANY	"<DBName>".<PI-Dienst>	Angeforderter PI-Dienst <ul style="list-style-type: none"> <DBName>: Symbolname für DB16, Standard: "PI" <PI-Dienst>: Liste verfügbarer PI-Dienste (Seite 1009)
Unit:	E	INT	1, 2, 3 ...	Bereichsnummer
Addr1 bis Addr4:	E	ANY	"<DBName>".<VarName>	Referenz auf einen String Bedeutung abhängig vom angewählten PI-Dienst
WVar1 bis WVar10:	E	WORD	-32768 ... 32767 8000 _H - 7FFF _H	INTEGER- oder WORD-Variable Bedeutung abhängig vom angewählten PI-Dienst
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	Fehlerstatus 1: Der Auftrag wurde negativ quittiert bzw. konnte nicht ausgeführt werden
Done:	A	BOOL	0 (FALSE), 1 (TRUE)	Auftragsstatus 1: Der Auftrag wurde erfolgreich ausgeführt
State:	A	WORD	¹⁾	Fehlerkennung Nur relevant bei "Error" == 1
1)	State	Bedeutung	Hinweis	
	3	negative Quittung, Auftrag nicht ausführbar	Interner Fehler, mögliche Abhilfe durch NC-RESET	
	6	FIFO voll	Auftrag wiederholen	
	7	Option nicht gesetzt	FB1-Parameter "NCKomm" ist nicht gesetzt	
	9	Übertragung belegt	Auftrag wiederholen	
	13	Addr1.. Addr4: Referenz ungültig	Fehlenden String angeben	
	14	PI-Dienst ist unbekannt	Der im Parameter "PIService" angegebene PI-Dienst ist unbekannt. → Schreibweise überprüfen.	
	15	Addr1.. Addr4: String zu lang	Stringlängen überprüfen	

Aufrufbeispiel

Funktion: Programmanwahl im Kanal 1 (Hauptprogramm und Werkstückprogramm)

Eintrag PI-Dienst für DB16 und STR für DB124 mit dem S7-SYMBOL-Editor:

Parametrierung		
Symbol	Operand	Datentyp
PI	DB16	DB16
STR	DB124	DB124

```

DATA_BLOCK DB126          // Instanz für FB4, freier Anwender-DB
FB4
BEGIN
END_DATA_BLOCK
// -----
DATA_BLOCK DB124
STRUCT
    PName:      string[32] := '_N_TEST_MPF'; // Hauptprogramm
    Path:       string[32] := '/'           // Hauptprogramm-Pfad
                _N_MPF_DIR/';
    PName_WST:  string[32] := '_N_ABC_MPF'; // Werkstückprogramm
    Path_WST:   string[32] := '/_N_WKS_DIR/ // Werkstückprogramm-Pfad
                _N_ZYL_WPD';
END_STRUCT
BEGIN
END_DATA_BLOCK
// -----
Function FC "PICall" : VOID
    U E 7.7;      // freie Taste Maschinensteuertafel
    S M 0.0;      // Req aktivieren
    U M 1.1;      // Fertigmeldung Done
    R M 0.0;      // Auftrag beenden
    U E 7.6;      // Fehlerquittierung von Hand
    U M 1.0;      // Fehler steht an
    R M 0.0;      // Auftrag beenden

    CALL FB4, DB126 (
        Req:=      M0.0,
        PIService:= "PI".SELECT,           // PI-Dienst: SELECT
        Unit:=      1,                     // Kanal 1
        Addr1:=     "STR".Path,             // Hauptprogramm: Pfad
        Addr2:=     "STR".PName,            // Hauptprogramm: Programm
        // Addr1:=   "STR".Path_WST,        // Werkstück: Pfad
        // Addr2:=   "STR".PName_WST,       // Werkstück: Programm
        Error:=     M1.0,

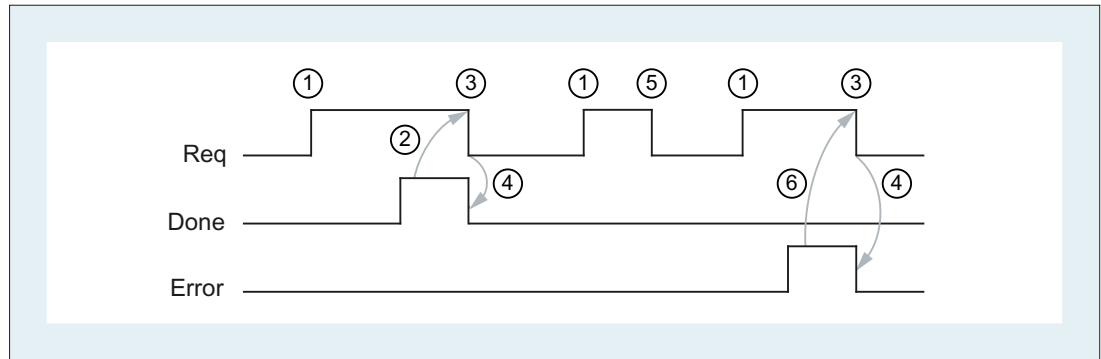
```

```

Done:=      M1.1,
State:=     MW2
);

```

Ablaufdiagramm



- ① Anwender: Anforderung setzen, Req = 0 → 1
- ② FB4: PI-Dienst erfolgreich beendet, Done = 1
Anwender: Anforderung zurücksetzen, IF Done == 1 THEN Req = 0
- ③ Anwender: IF Done == 1 THEN Anforderung zurücksetzen: 1 → 0
- ④ FB4: Auftragsbestätigung zurücksetzen, Done = 0
- ⑤ Anwender: IF Done == 0 UND Error == 0 THEN Anforderung zurücksetzen Req = 1 → 0 **nicht zulässig**
- ⑥ FB4: PI-Dienst mit Fehler beendet, Error = 1
Anwender: Anforderung zurücksetzen, IF Done == 1 OR Error == 1 THEN Req = 0, eventuell weitere Fehlerbehandlung

14.17.4.1 Liste verfügbarer PI-Dienste

Allgemeine PI-Dienste

PI-Dienst	Funktion
ASUP (Seite 1010)	Interrupt zuordnen
CANCEL (Seite 1011)	Cancel durchführen
CONFIG (Seite 1011)	Umkonfiguration von gekennzeichneten Maschinendaten
DIGION (Seite 1012)	Digitalisieren ein
DIGIOF (Seite 1012)	Digitalisieren aus
FINDBL (Seite 1012)	Satzsuchlauf aktivieren
LOGIN (Seite 1013)	Kennwort aktivieren
LOGOUT (Seite 1013)	Kennwort rücksetzen
NCRES (Seite 1013)	NC-RESET auslösen
SELECT (Seite 1014)	Programm zur Abarbeitung für einen Kanal anwählen
SETUDT (Seite 1014)	Setzt die aktuellen User-Daten aktiv

PI-Dienst	Funktion
SETUFR (Seite 1015)	Anwender Frame aktivieren
RETRAC (Seite 1015)	Freifahren des Werkzeugs in Werkzeugrichtung

PI-Dienste der Werkzeugverwaltung

PI-Dienst	Funktion
CRCEDN (Seite 1016)	Werkzeugschneide anlegen unter Eingabe der T-Nummer
CREACE (Seite 1017)	Werkzeugschneide anlegen mit der nächsthöheren/freien D-Nummer
CREATO (Seite 1017)	Anlegen eines Werkzeuges unter Vorgabe einer T-Nummer
DELECE (Seite 1017)	Löschen einer Werkzeugschneide
DELETO (Seite 1018)	Werkzeug löschen
MMCSEM (Seite 1018)	Semaphoren für verschiedene PI-Dienste
TMCRT0 (Seite 1019)	Anlegen eines Werkzeuges unter Angabe eines Namens, einer Duplo-Nummer
TMFDPL (Seite 1021)	Leerplatzsuche zum Beladen
TMFPBP (Seite 1022)	Leerplatzsuche
TMGETT (Seite 1023)	T-Nummer zum vorgegebenen Werkzeugnamen mit Duplo-Nummer
TMMVTL (Seite 1024)	Magazinplatz zum Beladen bereitstellen, Werkzeug entladen
TMPOSM (Seite 1025)	Magazinplatz oder Werkzeug positionieren
TMPCIT (Seite 1026)	Inkrementwert für Stückzahlzähler setzen
TMRASS (Seite 1027)	Rücksetzen des Aktiv-Status
TRESMO (Seite 1027)	Rücksetzen der Überwachungswerte
TSEARC (Seite 1028)	Komplexes Suchen über Suchmasken
TMCRMT (Seite 1031)	Multitool erzeugen
TMDLMT (Seite 1032)	Multitool löschen
POSMT (Seite 1032)	Multitool positionieren
FDPLMT (Seite 1033)	Leerplatz innerhalb des Multitools suchen/prüfen

14.17.4.2 PI-Dienst: ASUP

Funktion: Interrupt zuordnen

Ein auf der NC abgelegtes Programm wird einem Interrupt-Signal eines Kanals zugeordnet. Das Programm muss ausführbar sein. Pfad- und Programmname müssen vollständig angegeben werden.

Ausführliche Informationen zu Programmverwaltung, Pfad- und Dateinamen siehe:

Literatur:

Programmierhandbuch Arbeitsvorbereitung; Kapitel: "Datei- und Programmverwaltung" > "Programmspeicher"

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".ASUP	Interrupt zuordnen ¹⁾
Unit:	INT	1, 2, 3 ... 10	Kanal

Signal	Typ	Wertebereich	Bedeutung
Addr1:	STRING		Pfadname
Addr2	STRING		Programmname
WVar1:	WORD	1, 2, 3 ... 8	Interrupt-Nummer
WVar2:	WORD	1, 2, 3 ... 8	Priorität
WVar3:	WORD	0, 1	LIFTFAST ²⁾
WVar4:	WORD	0, 1	BLSYNC ³⁾
¹⁾ Alternativ kann für die Interrupt-Zuordnung der Befehl SETINT verwendet werden. Siehe 2) ²⁾ Literatur: Programmierhandbuch Arbeitsvorbereitung, Kapitel: "Flexible NC-Programmierung" > "Interruptroutine (ASUP)" > "Schnellabheben von der Kontur (SETINT, LIFTFAST, ALF)" ³⁾ Literatur: Programmierhandbuch Arbeitsvorbereitung, Kapitel: "Flexible NC-Programmierung" > "Interruptroutine (ASUP)" > "Interruptroutine zuordnen und starten (SETINT, PRIO, BLSYNC)"			

Hinweis

Der PI-Dienst ASUP darf nur im RESET-Zustand des angegebenen Kanals ausgeführt werden.

Literatur:

Programmierhandbuch Arbeitsvorbereitung, Kapitel: "Flexible NC-Programmierung" > "Interruptroutine (ASUP)"

14.17.4.3 PI-Dienst: CANCEL**Funktion: Cancel durchführen**

Löst die Funktion "Cancel" gleichwertig der entsprechenden "Alarm Cancel"-Taste der Bedienoberfläche (Bedientafelfront) aus.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".CANCEL	Cancel

14.17.4.4 PI-Dienst: CONFIG**Funktion: Umkonfiguration**

Bewirkt, dass Maschinendaten, die sequentiell vom Bediener oder auch PLC eingegeben wurden, quasi parallel aktiviert werden.

Das Kommando kann nur im RESET-Zustand der Steuerung bzw. bei Programmunterbrechung (NC-Stop an Satzgrenze) aktiviert werden. Bei Nichteinhaltung dieser Bedingung erfolgt eine Fehlerrückmeldung des FB4 (State = 3).

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".CONFIG	Umkonfiguration
Unit:	INT	1	
WVar1:	INT	1	Klassifizierung

14.17.4.5 PI-Dienst: DIGION**Funktion: Digitalisieren ein**

Digitalisieren im parametrierten Kanal anwählen.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".DIGION	Digitalisieren ein
Unit:	INT	1, 2, 3 ... 10	Kanal

14.17.4.6 PI-Dienst: DIGIOF**Funktion: Digitalisieren aus**

Digitalisieren im parametrierten Kanal ausschalten.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".DIGIOF	Digitalisieren aus
Unit:	INT	1, 2, 3 ... 10	Kanal

14.17.4.7 PI-Dienst: FINDBL**Funktion: Satzsuchlauf aktivieren**

Ein Kanal wird auf Satzsuchlaufmodus geschaltet und daraufhin die Quittung gesendet. Der Satzsuchlauf wird danach von der NC sofort ausgeführt. Der Satzsuchlaufzeiger muss sich zu diesem Zeitpunkt bereits in der NC befinden. Der Satzsuchlauf kann jederzeit durch NC-RESET abgebrochen werden. Ist der Satzsuchlauf abgeschlossen, wird automatisch wieder der normale Abarbeitungsmodus aktiviert. NC-Start wirkt dann ab dem gefundenen Suchziel.

Es liegt ausschließlich in der Verantwortung des Bedieners, einen kollisionsfreien Anfahrweg sicherzustellen.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".FINDBL	Satzsuchlauf
Unit:	INT	1 bis 10	Kanal
WVar1:	WORD	1, 2, 3	Vorlaufmodus 1: ohne Berechnung 2: mit Berechnung 3: mit Hauptsatzbetrachtung

14.17.4.8 PI-Dienst: LOGIN**Funktion: Kennwort anlegen**

Übergibt das parametrisierte Kennwort an die NC. Das Kennwort besteht generell aus 8 Zeichen. Gegebenenfalls müssen in dem String des Kennworts Leerzeichen ergänzt werden.

Beispiel

Kennwort : STRING[8] := 'SUNRISE';

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".LOGIN	Kennwort anlegen
Unit:	INT	1	NC
Addr1:	STRING	8 Zeichen	Kennwort

14.17.4.9 PI-Dienst: LOGOUT**Funktion: Kennwort rücksetzen**

Das zuletzt an die NC übergebene Kennwort wird zurückgesetzt.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".LOGOUT	Kennwort rücksetzen
Unit:	INT	1	NC

14.17.4.10 PI-Dienst: NCRES**Funktion: NC-RESET auslösen**

Löst NC-RESET aus. Parameter "Unit" und "WVar1" sind immer mit 0 zu versorgen.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".NCRES	NC-RESET auslösen
Unit:	INT	0	-
WVar1:	WORD	0	-

14.17.4.11 PI-Dienst: SELECT**Funktion: Abarbeitung für einen Kanal anwählen**

Ein auf der NC abgelegtes Programm wird für einen Kanal zur Abarbeitung angewählt. Dies ist nur möglich, wenn das File ausgeführt werden darf. Die Pfadnamen und Programmnamen sind in korrekter Schreibweise einzugeben. Ausführliche Informationen dazu finden sich in:

Literatur

Programmierhandbuch Arbeitsvorbereitung; Kapitel: "Datei- und Programmverwaltung" > "Programmspeicher"

Mögliche Bausteintypen

Bausteintypen	
Werkstückverzeichnis	WPD
Hauptprogramm	MPF
Unterprogramm	SPF
Zyklen	CYC
Asynchrone Unterprogramme	ASP
Binärfiles	BIN

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".SELECT	Programm-Anwahl
Unit:	INT	1, 2, 3 ... 10	Kanal
Addr1:	STRING	-	Pfad-Name
Addr2:	STRING	-	Programm-Name

14.17.4.12 PI-Dienst: SETUDT**Funktion: Aktuelle User-Daten aktiv setzen**

Die aktuellen User-Daten wie Werkzeugkorrekturen, Basisframes und einstellbare Frames werden nur im Stopp-Zustand zum nächsten NC-Satz gesetzt.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".SETUDT	User Daten aktivieren
Unit:	INT	1, 2, 3 ... 10	Kanal
WVar1:	WORD	1, 2, 3, 4, 5	User Data Type 1: aktive Werkzeug-Korrektur 2: aktiver Basis-Frame 3: aktiver einstellbarer Frame 4: aktiver globaler Basis-Frame 5: aktiver globaler einstellbarer Frame
WVar2:	WORD	0	Reserve
Wvar3:	WORD	0	Reserve

14.17.4.13 PI-Dienst: SETUFR**Funktion: Anwender-Frames aktivieren**

User-Frames werden in die NC geladen. Alle notwendigen Werte der Frames müssen vorher mit FB3 "Variablen schreiben" an die NC übermittelt worden sein.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".SETUFR	User-Frames aktivieren
Unit:	INT	1, 2, 3 ... 10	Kanal

14.17.4.14 PI-Dienst: RETRAC**Funktion: JOG-Retract anwählen**

Wählt die Betriebsart JOG-Retract an. Die Rückzugsachse d. h. die Geometrieachse, mit der der Rückzug ausgeführt wird, kann von der NC automatisch ermittelt oder explizit vorgegeben werden.

Die Betriebsart bleibt aktiv, bis sie mit RESET beendet wird.

Hinweis

Der PI-Dienst RETRAC kann nur in der Betriebsart JOG im Zustand "Reset" aktiviert werden.

Automatischen Ermittlung

Bei der automatischen Ermittlung wird die Geometrieachse als Rückzugsachse ausgewählt, die senkrecht (orthogonal) zur aktuell angewählten Arbeitsebene steht:

- G17: Rückzugsachse ⇒ 3. Geometrieachse (Z)
- G18: Rückzugsachse ⇒ 2. Geometrieachse (Y)
- G19: Rückzugsachse ⇒ 1. Geometrieachse (X)

Hinweis**BTSS-Variable retractState**

Die aktive Rückzugsachse kann über die BTSS-Variable retractState.Bit 2/3 gelesen werden.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".RETRAC	Betriebsart JOG-Retract anwählen
Unit:	INT	1, 2, 3 ... 10	Kanal
WVar1:	WORD	0, 1, 2, 3	Rückzugsachse 0: Automatische Ermittlung der Rückzugsachse durch die NC. 1: Rückzugsachse ist 1. Geometrieachse des WKS 2: Rückzugsachse ist 2. Geometrieachse des WKS 3: Rückzugsachse ist 3. Geometrieachse des WKS
WVar2:	WORD	0	Reserviert. Der Wert ist mit 0 vorzubelegen.

14.17.4.15 PI-Dienst: CRCEDN**Funktion: Legt neue Schneide an**

Wird in dem PI-Dienst unter dem Parameter "T-Nummer" die T-Nummer eines existierenden Werkzeugs angegeben, wird die Schneide zu diesem Werkzeug angelegt (in diesem Fall hat der Parameter "D-Nummer" - die Nummer der anzulegenden Schneide - einen Wertebereich von 1, 2, 3 ... 9).

Ist eine positive T-Nummer als Parameter angegeben und das Werkzeug zu der angegebenen T-Nummer existiert nicht, schlägt der PI-Dienst fehl.

Wird für die T-Nummer der Wert 0 angegeben (Modell der absoluten D-Nummern), kann sich der Wertebereich der D-Nummer von 1, 2, 3 ... 31999 erstrecken. Die neue Schneide wird mit der vorgegebenen D-Nummer erzeugt.

Existiert die angegebene Schneide schon, schlägt der PI-Dienst in beiden Fällen fehl.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".CRCEDN	Lege neue Schneide an
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		T-Nummer des Werkzeuges, zu dem die Werkzeugschneide angelegt werden soll. Ein Wert 00000 besagt, dass kein Bezug zu einem Werkzeug existieren soll (absolute D-Nummer)
WVar2:	INT	1 ... 9	Schneiden-Nummer der Werkzeugschneide
		1 - 31999	

14.17.4.16 PI-Dienst: CREAM**Funktion: Werkzeugschneide anlegen**

Erzeugen der Schneide mit der nächst höheren/freien D-Nummer zu dem Werkzeug mit der übergebenen T-Nummer in TO, TS (sofern vorhanden), im TUE-Baustein (sofern vorhanden) wird die Schneide für die OEM-Schneidendaten mit angelegt.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".CREAM	Werkzeugschneide anlegen
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		T-Nummer

14.17.4.17 PI-Dienst: CREATO**Funktion: Werkzeug anlegen**

Anlegen eines Werkzeuges unter Vorgabe einer T-Nummer. Das Werkzeug wird im Bereich TV (Werkzeug-Directory) als vorhanden eingetragen, im TO-Baustein wird die erste Schneide "D1" (mit Null-Inhalt) für die Korrekturen angelegt, im TUE-Baustein - sofern vorhanden - wird die erste Schneide "D1" (mit Null-Inhalt) für die OEM-Schneidendaten angelegt. Falls ein TU-Baustein vorhanden ist, ist dort der Datensatz für das Werkzeug bereitgestellt.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".CREATO	Werkzeug anlegen
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		T-Nummer

14.17.4.18 PI-Dienst: DELECE**Funktion: Löschen einer Werkzeugschneide**

Wird in dem PI-Dienst unter dem Parameter T-Nummer die T-Nummer eines existierenden Werkzeugs angegeben, so wird die Schneide zu diesem Werkzeug gelöscht (in diesem Fall hat der Parameter D-Nummer - die Nummer der anzulegenden Schneide - einen Wertebereich von 00001 - 00009. Ist eine positive T-Nummer als Parameter angegeben und das Werkzeug zu der angegebenen T-Nummer existiert nicht, so schlägt der PI-Dienst fehl. Wird für die T-Nummer der Wert 00000 angegeben (Modell der absoluten D-Nummern), so kann sich der Wertebereich der D-Nummer von 00001 - 31999 erstrecken. Existiert die angegebene Schneide nicht, so schlägt der PI-Dienst in beiden Fällen fehl.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".DELECE	Lösche Schneide
Unit:	INT	1, 2, 3 ... 10	TOA

Signal	Typ	Wertebereich	Bedeutung
WVar1:	INT		T-Nummer des Werkzeuges, zu dem die Werkzeugschneide angelegt werden soll. Ein Wert 00000 besagt, dass kein Bezug zu einem Werkzeug existieren soll (absolute D-Nummer)
WVar2:	INT	1 ... 9	Schneiden-Nummer der Werkzeugschneide, welche gelöscht werden soll
		1 - 31999	

14.17.4.19 PI-Dienst: DELETO

Funktion: Werkzeug löschen

Löscht das Werkzeug zu der übergebenen T-Nummer mit allen Schneiden (in TO, ggf. TU, TUE und TG (Typ 4xx), TD und TS Bausteinen).

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".DELETO	Werkzeug löschen
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		T-Nummer

14.17.4.20 PI-Dienst: MMCSEM

Funktion

Der PI-Dienst dient zum Test / Setzen und Rücksetzen von kanalspezifischen Semaphoren von HMI und PLC aus. Pro Kanal stehen 10 Semaphoren zum Schützen von kritischen Datenbereichen zur Verfügung.

Den Semaphoren 1 bis 6 sind Funktionen (PI-Dienste) zugeordnet. Die Semaphoren 7 bis 10 können frei verwendet werden.

FB4 Rückgabewerte

- "Done" == TRUE
Die Semaphore wurde gesetzt, die kritische Funktion kann aufgerufen werden
- "Error" == TRUE UND "State" == 3
Die Semaphore war bereits gesetzt, die kritische Funktion kann zur Zeit nicht aufgerufen werden.

Schematischer Ablauf bei freier Semaphore

```

Test und Setzen der Semaphore
IF Semaphore == FREI
THEN
    Schreiben / Lesen der kritischen Daten
    Rücksetzen der Semaphore
ELSE // Semaphore ist gesperrt
    ...
ENDIF

```

ACHTUNG**Rücksetzen der Semaphore**

Nach dem Sperren des kritischen Datenbereichs durch Setzen der Semaphore und dem anschließenden Lesen / Schreiben der Daten, muss der kritische Datenbereich durch Rücksetzen der Semaphore unbedingt wieder freigegeben werden.

Schematischer Ablauf bei gesperrter Semaphore

Test und Setzen der Semaphore

```
IF Semaphore == FREI
```

```
THEN
```

```
...
```

```
ELSE // Semaphore ist gesperrt
```

```
    Merker setzen für "Funktion konnte nicht ausgeführt werden,  
    Wiederholung ist notwendig"
```

```
ENDIF
```

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".MMCSEM	Semaphore bearbeiten
Unit:	INT	1, 2, 3 ... 10	Kanal
WVar1:	INT	¹⁾	PI-Dienst-spezifische Nummer der Semaphore
WVar2:	WORD	0, 1	Auftragsart 0: Semaphore zurücksetzen 1: Semaphore testen und setzen
1)	Nummer	PI-Dienst	
	1	TMCRTO (Werkzeug anlegen)	
	2	TMFDPL (Leerplatzsuche zum Beladen)	
	3	TMMVTL (Magazinplatz zum Beladen bereitstellen, Werkzeug entladen)	
	4	TMFPBP (Suche Platz)	
	5	TMGETT (Suche Werkzeugnummer)	
	6	TSEARC (Werkzeug suchen)	
	7 ... 10	frei verwendbar	

14.17.4.21 PI-Dienst: TMCRT0**Funktion Werkzeug anlegen:**

Anlegen eines Werkzeuges unter Angabe eines Namens, einer Duplo-Nummer, z. B. mit \$TC_TP1[y] = Duplo-Nummer oder \$TC_TP2[y] = "Werkzeugname". Oder optional durch einer T-Nummer, z. B. mit y = T-Nummer

Das Werkzeug wird im Bereich TV (Werkzeug-Verzeichnis) als vorhanden eingetragen.

Im TO-Baustein wird die erste Schneide "D1" (mit Null-Inhalt) für die Korrekturen angelegt.

Im TS-Baustein wird die erste Schneide "D1" (mit Null-Inhalt) für die Überwachungsdaten angelegt.

Im TUE-Baustein - sofern vorhanden - wird die Schneide "D1" für die OEM-Schneidendaten mit angelegt.

Im TD-Baustein stehen Bezeichner, Duplonummer und Anzahl der Schneiden (=1) zu der optional vorgegebenen oder von der NC vergebenen T-Nummer.

Ist ein TU-Baustein vorhanden, ist dort der Datensatz für das Werkzeug bereitgestellt.

Nach Ausführung des PI-Dienstes steht im TV-Baustein unter **TnumWZV** die T-Nummer zu dem angelegten Werkzeug.

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" mit der Funktionsnummer 1 für TMCRT0 aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMCRT0	Werkzeug anlegen
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT	¹⁾	T-Nummer
WVar2:	INT		Duplo-Nummer
Addr1:	STRING	max. 32 Zeichen	Werkzeugname
¹⁾ T-Nummer > 0 ⇒ die angegebene T-Nummer wird vorgeben T-Nummer = -1 ⇒ die NC vergibt die T-Nummer			

14.17.4.22 PI-Dienst: TMFDPL

Funktion

Leerplatzsuche zum Beladen, in Abhängigkeit der Parameterbelegung:

- Platz-Nummer_Wohin = -1 UND Magazin-Nummer_Wohin = -1
Sucht unter allen Magazinen zum angegebenen Bereich (= Kanal) einen Leerplatz für das mit T-Nummer spezifizierte Werkzeug. Die gefundene Magazin- und Platznummer steht nach Ausführung des PI-Dienstes im Konfigurationsblock des Kanals (Komponente **magCMCmdPar1** (Magazinnummer) und **magCMCmdPar2** (Platznummer)). Platz-Nummer_Bez und Magazin- Nummer_Bez können als Suchkriterium besetzt sein oder nicht (= -1). Der PI-Dienst wird in Abhängigkeit vom Suchergebnis positiv oder negativ quittiert.
- Platz-Nummer_Wohin = -1 UND Magazin-Nummer_Wohin = Magazin-Nummer
Es wird in dem angegebenen Magazin ein Leerplatz für das mit T-Nummer spezifizierte Werkzeug gesucht. Platz-Nummer_Bez und Magazin- Nummer_Bez können als Suchkriterium besetzt sein oder mit -1. Der PI-Dienst wird in Abhängigkeit vom Suchergebnis positiv oder negativ quittiert.
- Platz-Nummer_Wohin = Platz-Nummer UND Magazin-Nummer_Wohin = Magazin-Nummer
Der angegebene Platz wird geprüft, ob er zum Beladen mit dem angegebenen Werkzeug frei ist. Platz-Nummer_Bez und Magazin-Nummer_Bez können als Suchkriterium besetzt sein oder -1. Der PI-Dienst wird in Abhängigkeit vom Prüfergebnis positiv oder negativ quittiert.

Die Parameter "WVar1" und "WVar2" stehen bei der Quelle.

Laden: Wenn Quelle ein internes Belademagazin ist, dann stehen die Parameter beim Ziel (einem realen Magazin).

Entladen: Quelle ist immer reales Magazin.

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" mit der Funktionsnummer 2 für TMFDPL aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMFDPL	Leerplatz zum Beladen
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		T-Nummer
WVar2:	INT		Platz-Nummer_Wohin
WVar3:	INT		Magazin-Nummer_Wohin
WVar4:	INT		Platz-Nummer_Bez
WVar5:	INT		Magazin-Nummer_Bez

14.17.4.23 PI-Dienst: TMFPBP**Funktion: Leerplatzsuche**

Der PI-Dienst sucht in den angegebenen Magazin(en) einen Leerplatz, welcher den angegebenen Kriterien wie Werkzeuggröße und Platztyp genügt.

Das Ergebnis kann bei erfolgreicher Suche aus folgenden BTSS-Variablen gelesen werden:

- magCMCmdPar1 (Magazin-Nummer)
- magCMCmdPar2 (Platznummer)

Hinweis

Der PI-Dienst kann nur mit FB7 angefordert werden. Siehe Kapitel "FB7: PI_SERV2 - PI-Dienst anfordern (Seite 1043)".

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" mit der Funktionsnummer 4 für TMFPBP aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMFPBP	Leerplatzsuche
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		Magazinnummer des Magazins, ab dem gesucht werden soll
WVar2:	INT		Platznummer des Platzes im Magazin aus "WVar1"
WVar3:	INT		Magazinnummer des Magazins, bis zu dem gesucht werden soll
WVar4:	INT		Platznummer des Platzes im Magazin aus "WVar3"
WVar5:	INT		Magazinnummer Bezug
WVar6:	INT		Platznummer Bezug
WVar7:	INT	0, 1, 2 ... 7	Anzahl der benötigten Halbplätze nach links
WVar8:	INT	0, 1, 2 ... 7	Anzahl der benötigten Halbplätze nach rechts
WVar9:	INT	0, 1, 2 ... 7	Anzahl der benötigten Halbplätze nach oben
WVar10:	INT	0, 1, 2 ... 7	Anzahl der benötigten Halbplätze nach unten
WVar11:	INT		Nummer des gewünschten Platztyps
WVar12:	INT	0, 1, 2, 3, 4	spezifiziert die gewünschte Suchrichtung 0: Suchstrategie wie in \$TC_MAMP2 eingestellt 1: vorwärts 2: rückwärts 3: symmetrisch

Beispiele: Einstellen des Suchbereichs

ab	Platz	bis	Platz	Beschreibung
WVar1	WVar2	WVar3	WVar4	
#M1	#P1	#M1	#P1	Es wird nur der Platz #P1 im Magazin #M1 überprüft
#M1	#P1	#M2	#P2	Es werden die Plätze beginnend von Magazin #M1, Platz #P1 bis Magazin #M2, Platz #P2 durchsucht
#M1	-1	#M1	-1	Es werden alle Plätze von Magazin #M1 - und nur diese - durchsucht
#M1	-1	-1	-1	Es werden alle Plätze beginnend ab Magazin #M1 durchsucht
#M1	#P1	-1	-1	Es werden alle Plätze beginnend ab Magazin #M1 und darin Platz #P1 durchsucht
#M1	#P1	#M1	-1	Es werden die Plätze in Magazin #M1 beginnend ab Magazin #M1 und darin Platz #P1 durchsucht
#M1	#P1	#M2	-1	Es werden die Plätze beginnend ab Magazin #M1 und darin Platz #P1 bis einschließlich Magazin #M2 durchsucht
#M1	-1	#M2	#P2	Es werden die Plätze beginnend ab Magazin #M1 bis einschließlich Magazin #M2 und darin Platz #P2 durchsucht
#M1	-1	#M2	-1	Es werden die Plätze beginnend ab Magazin #M1 bis einschließlich Magazin #M2 durchsucht
-1	-1	-1	-1	Es werden alle Magazinplätze durchsucht

14.17.4.24 PI-Dienst: TMGETT**Funktion: T-Nummer zum vorgegebenen Werkzeugname mit Duplonummer bestimmen**

Der PI-Dienst dient zum Ermitteln der T-Nummer eines Werkzeugs über Werkzeugname und Duplo-Nummer.

Das Ergebnis wird in BTSS-Variablen im Baustein TF (Parametrierung, Rückgabeparameter von TMGETT, TSEARC) geschrieben:

- resultNrOfTools
 - resultNrOfTools == 0: das angegebene Werkzeug wurde nicht gefunden
 - resultNrOfTools == 1: das angegebene Werkzeug wurde gefunden
- resultToolNr: T-Nummer des angegebenen Werkzeugs bei "resultNrOfTools" == 1

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" mit der Funktionsnummer 5 für TMGETT. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMGETT	T-Nummer ermitteln
Unit:	INT	1, 2, 3 ... 10	TOA

Signal	Typ	Wertebereich	Bedeutung
Addr1:	STRING	max. 32 Zeichen	Werkzeugname
WVar1:	INT		Duplo-Nummer

14.17.4.25 PI-Dienst: TMMVTL

Funktion: Magazinplatz zum Beladen bereitstellen, Werkzeug entladen

Der PI-Dienst dient zum Be- und Entladen, sowie zum Umsetzen von Werkzeugen:

1. Be- und Entladen: Beladestelle ↔ Magazin)
2. Be- und Entladen: Beladestelle ↔ Zwischenspeicher, z. B. Spindel
3. Umsetzen innerhalb eines Magazins
4. Umsetzen zwischen verschiedenen Magazinen
5. Umsetzen zwischen Magazin und Zwischenspeicher
6. Umsetzen innerhalb des Zwischenspeichers

Fall 1, 3, 4 und 5: folgende BTSS-Variablen des Bausteins TM (Magazindaten: Allgemeine Daten) werden geschrieben:

- magCmd (BereichsNr = TO-Einheit, Zeile = Magazinnummer)
- magCmdState ← "Quittung"

Fall 2 und 6: folgende BTSS-Variablen des Bausteins TMC (Magazindaten: Konfigurationsdaten) werden geschrieben:

- magCBCmd (BereichsNr = TO-Einheit)
- magCBCmdState ← "Quittung"

Beladen

- WVar2 "Platz-Nummer_Von", WVar3 "Magazin-Nummer_Von"
Der Werkzeugplatz des Magazin wird zum Beladen an die Beladestation/-platz verfahren und das Werkzeug eingewechselt.
- WVar4 "Platznummer_Zu" == -1
Es wird im Magazin zuerst ein Leerplatz für das Werkzeug gesucht. Anschließend wird der Leerplatz des Magazin zum Beladen an die Beladestation/-platz verfahren und das Werkzeug eingewechselt.
Die gefundene Leerplatznummer steht nach Ausführung des PI-Dienstes in "magCMCmdPar2" (BTSS-Bausten TM), **reales** Magazin des Kanals.
- WVar4 "Platznummer_Zu" == -2
Das Werkzeug wird in den aktuellen Werkzeugplatz des Magazin beladen. Die Platznummer steht nach Ausführung des PI-Dienstes in "magCMCmdPar2" (BTSS-Bausten TM), **reales** Magazin des Kanals.

Entladen

- WVar4 "Platz-Nummer_Zu", WVar5 "Magazin-Nummer_Zu"
Der Werkzeugplatz des Magazin wird an die Beladestelle/-station verfahren und das Werkzeug ausgewechselt.

Im BTSS-Baustein TP (Magazindaten: Platzdaten) wird der Magazinplatz des ausgewechselten Werkzeugs als frei gekennzeichnet.

Adressierung des Werkzeugs

Das Werkzeug kann wahlweise über T-Nummer oder über Platz- und Magazinnummer adressiert werden. Einem nicht benutzte Parameter ist der Wert -1 zuzuweisen.

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" mit der Funktionsnummer 3 für TMMVTL aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMMVTL	Magazinplatz zum Beladen bereitstellen, WZ entladen
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		T-Nummer
WVar2:	INT		Platz-Nummer_Von
WVar3:	INT		Magazin-Nummer_Von
WVar4:	INT		Platz-Nummer_Zu
WVar5:	INT		Magazin-Nummer_Zu

14.17.4.26 PI-Dienst: TMPOS

Funktion: Magazinplatz oder Werkzeug positionieren, in Abhängigkeit der Parameterbelegung

Über den PI-Dienst wird ein Magazinplatz, der entweder direkt oder über ein darauf befindliches Werkzeug angegeben ist, an eine vorgegebene Position (z. B. vor einen Beladeplatz) verfahren.

Die Angabe des Ziels, z.B. des Beladeplatzes, erfolgt in den Parameter:

- WVar5 "Platz-Nummer_Bezug"
- WVar6 "Magazin-Nummer_Bezug"

Die Angabe des zu positionierenden Magazinplatzes erfolgt, abhängig von der jeweiligen Spezifikation, in den Parametern:

- WVar1 "T-Nummer" des Werkzeugs
Folgenden Parameter sind hierbei irrelevant:
 - Addr1 "Werkzeugname" = ""
 - WVar2 "Duplo-Nummer" = -1
 - WVar3 "Platz-Nummer_Von" = -1
 - WVar4 "Magazin-Nummer_Von" = -1
- Addr1 "Werkzeugname", WVar2 "Duplo-Nummer"
Folgenden Parameter sind hierbei irrelevant:
 - WVar1 "T-Nummer des Werkzeugs" = -1
 - WVar3 "Platz-Nummer_Von" = -1
 - WVar4 "Magazin-Nummer_Von" = -1
- WVar3 "Platz-Nummer_Von", WVar4 "Magazin-Nummer_Von"
Folgenden Parameter sind hierbei irrelevant:
 - Addr1 "Werkzeugname" = ""
 - WVar1 "T-Nummer des Werkzeugs" = -1
 - WVar2 "Duplo-Nummer" = -1

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMPOSM	Magazinplatz oder Werkzeug positionieren
Unit:	INT	1, 2, 3 ... 10	TOA
Addr1:	STRING	max 32. Zeichen	Werkzeugname
WVar1:	INT		T-Nummer des Werkzeugs
WVar2:	INT		Duplo-Nummer
WVar3:	INT		Platz-Nummer_Von
WVar4:	INT		Magazin-Nummer_Von
WVar5:	INT		Platz-Nummer_Bezug
WVar6:	INT		Magazin-Nummer_Bezug

14.17.4.27 PI-Dienst: TMPCIT

Funktion: Inkrementwert für Stückzahlzähler setzen

Inkrementieren des Stückzahlzählers des Spindelwerkzeugs.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMPCIT	Inkrementwert für Stückzahlzähler setzen
Unit:	INT	1, 2, 3 ... 10	TOA

Signal	Typ	Wertebereich	Bedeutung
WVar1:	WORD	0, 1, 2 ... max.	Spindelnummer; entspricht dem Artindex bei den Platzdaten mit Platzart "Spindel" des Zwischenspeichermagazins im Kanal.000 = Hauptspindel
WVar2:	WORD	0 ... max.	Inkrementwert; gibt an, nach wie vielen Umdrehungen der Spindel der Stückzahlzähler inkrementiert wird.

14.17.4.28 PI-Dienst: TMRASS

Funktion: Rücksetzen des Werkzeugstatus "aktiv"

Der PI-Dienst setzt für alle Werkzeuge mit Werkzeugstatus "aktiv" oder "gesperrt" den Status auf "nicht aktiv".

Sinnvolle Zeitpunkte für das Rücksetzen des Werkzeugstatus sind folgende Ereignisse:

- eine negative Flanke des NC/PLC-Nahtstellensignals "Werkzeug-Sperre unwirksam"
- Programmende
- Kanal-Reset.

Der PI-Dienst ist hauptsächlich für die PLC gedacht, da hier bekannt ist, wann das gesperrte Werkzeug endgültig nicht mehr verwendet werden soll.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI". TMRASS	Rücksetzen des Aktiv-Status
Unit:	INT	1, 2, 3 ... 10	TO-Bereich

14.17.4.29 PI-Dienst: TRESMO

Funktion: Rücksetzen der Überwachungswerte

Mit diesem PI-Dienst können die Überwachungswerte der bezeichneten Schneiden der bezeichneten Werkzeuge auf die Soll- (Ausgangs-) werte zurückgesetzt werden. Dies erfolgt nur für Werkzeuge mit aktiver Überwachung.

Siehe dazu auch den Befehl RESETMON.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI". TRESMO	Rücksetzen der Überwachungswerte
Unit:	INT	1, 2, 3 ... 10	TO-Bereich
WVar1:	WORD	- max ... max	Werkzeugnummer 0: alle Werkzeuge bearbeiten > 0: angegebenes Werkzeug bearbeiten < 0: alle Schwesterwerkzeuge bearbeiten

Signal	Typ	Wertebereich	Bedeutung
WVar2:	WORD	0, 1, 2 ... max.	D-Nummer < 0: Zurücksetzen der Überwachung der angegebenen Schneide 0: Zurücksetzen der Überwachung aller Schneiden
Wvar3:	WORD		Überwachungsart, die zurückgesetzt werden soll (alle Kombinationen sind möglich): Bit 0 = 1: Standzeit-Überwachung Bit 1 = 1: Stückzahl-Überwachung Bit 2 = 1: Verschleiß-Überwachung Bit 3 = 1: Summenkorrektur-Überwachung 0: Alle aktiven Überwachungen des Werkzeugs (\$TC_TP9).

14.17.4.30 PI-Dienst: TSEARCH

Funktion: Komplexes Suchen über Suchmaske, in Abhängigkeit der Parameterbelegung

Der PI-Dienst dient zum Suchen von Werkzeugen mit bestimmten Eigenschaften innerhalb eines Suchbereichs in einem oder mehreren Magazine, beginnend ab einem bestimmten Platz bis zu einem bestimmten Platz

Hinweis

Aktive Werkzeugverwaltung

Der PI-Dienst ist nur bei aktiver Werkzeugverwaltung verfügbar.

Vorgabemöglichkeiten

- Suchrichtung
- Suche das nächstes Werkzeug mit der angegebenen Eigenschaft
- Suche alle Werkzeuge mit der angegebenen Eigenschaft

Ergebnis

Als Ergebnis wird eine Liste mit den internen T-Nummern der gefundenen Werkzeuge erzeugt.

Verknüpfungsmöglichkeiten

Für Filterung von Eigenschaften steht als Verknüpfungsmöglichkeit nur eine UND-Verknüpfung zur Verfügung. Ein ODER-Verknüpfung muss anwenderseitig über mehrere Aufrufe des PI-Dienstes und anschließender Auswertung der Einzelergebnisse realisiert werden.

Parametrierung der Werkzeugeigenschaften

Das Einstellen der Eigenschaften der gesuchten Werkzeuge erfolgt im BTSS-Baustein TF (Parametrierung, Rückgabeparameter von TMGETT, TSEARCH) über folgende Variablen:

- "parMask<X>" (Parametrierungsmaske)
- "parData<X>" (Vergleichswerte)

Mit <X> = TAD, TAO, TAS, TD

Ergebnisliste

Nach dem fehlerfreien Abschluss des PI-Dienstes steht das Suchergebnis im BTSS-Baustein TF (Parametrierung, Rückgabeparameter von TMGETT, TSEARC) in folgenden Variablen:

- "resultCuttingEdgeNrUsed" (D-Nummern eingesetzter Schneiden seit der letzten Stückzahlzählung)
- "resultNrOfCutEdgesUsed" (Anzahl Schneiden seit letzter Stückzahlzählung)
- "resultNrOfTools" (Anzahl gefundener Werkzeuge)
- "resultToolNr" (T-Nummern der gefundenen Werkzeuge)
- "resultToolNrUsed" (T-Nummern der eingesetzten Schneiden seit der letzten Stückzahlzählung)

Wurde kein Werkzeug gefunden, ist die Anzahl gefundener Werkzeuge Null ("resultNrOfTools" == 0)

Suchbereichsangaben

von Mag.- Nummer wVar1	von Platz- nummer wVar2	bis Mag.- Nummer wVar3	bis Platz- nummer wVar4	Beschreibung Es werden folgende Magazinplätze durch- sucht:
#M<a>	#P	#M<x>	#P<y>	von: Magazin #M<a>, Platz #P bis: Magazin #M<x>, Platz #P<y>
#M<a>	-1	#M<a>	-1	von: Magazin #M<a>, erster Platz bis: Magazin #M<a>, letzter Platz
#M<a>	-1	-1	-1	von: Magazin #M<a>, erster Platz bis: letztes Magazin, letzter Platz
#M<a>	#P	-1	-1	von: Magazin #M<a>, Platz #P bis: letztes Magazin, letzter Platz
#M<a>	#P	#M<a>	-1	von: Magazin #M<a>, Platz #P bis: Magazin #M<a>, letzter Platz
#M<a>	#P	#M<x>	-1	von: Magazin #M<a>, Platz #P bis: Magazin #M, letzter Platz
#M<a>	-1	#M<x>	#P<y>	von: Magazin #M<a>, erster Platz bis: Magazin #M, Platz #P<y>
#M<a>	-1	#M<x>	-1	von: Magazin #M<a>, erster Platz bis: Magazin #M<x>, letzter Platz
-1	-1	-1	-1	von: erstes Magazin, erster Platz bis: letztes Magazin, letzter Platz

Symmetrische Suche

Damit bezogen auf einen Werkzeugplatz eine symmetrische Suche durchgeführt werden kann, müssen folgende Voraussetzungen erfüllt sein:

- Der Suchbereich darf nur ein Magazin umfassen: "WVar1" (von: Magazin-Nummer) == "WVar3" (bis: Magazin-Nummer)
- Angabe eines Referenzplatz, d.h. eines Werkzeugplatzes um den symmetrisch gesucht wird: "WVar5" (Nummer des Referenz-Magazins) und "WVar6" (Nummer des Referenz-Platzes)
- Für den Referenzplatz muss eine Mehrfachzuordnung zu dem zu durchsuchenden Magazin im Baustein TPM konfiguriert sein.
- "WVar7" (Suchrichtung) = 3

Der Referenzplatz ist ein Zwischenspeicher-Platz, d.h. ein Werkzeugplatz aus dem Magazin-Zwischenspeicher, z.B. Wechselstelle, Greifer etc., oder ein Beladeplatz/-stelle, d.h. ein Werkzeugplatz aus dem internen Belademagazin. Die symmetrische Suche erfolgt bezüglich des Werkzeugplatzes, der vor dem Referenzplatz liegt.

Liegt der Werkzeugplatz vor dem Referenzplatz außerhalb des Suchbereichs, verhält sich der PI-Dienst so, als ob kein passender Platz gefunden worden wäre.

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit entsprechendem Parameter "WVar1" für diesen PI-Dienst aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TSEARC	Komplexes Suchen über Suchmasken
Unit:	INT	1, 2, 3 ... 10	TOA
WVar1:	INT		von:Magazin-Nummer
WVar2:	INT		von: Platz-Nummer
WVar3:	INT		bis: Magazin-Nummer
WVar4:	INT		bis: Platz-Nummer
WVar5:	INT		Nummer des Referenz-Magazins (nur relevant bei symmetrischer Suche: Suchrichtung == 3)
WVar6:	INT		Nummer des Referez-Platzes (nur relevant bei symmetrischer Suche: Suchrichtung == 3)

Signal	Typ	Wertebereich	Bedeutung
WVar7:	INT	1, 2, 3	Suchrichtung: 1: vorwärts vom ersten Platz des Suchbereichs 2: rückwärts vom letzten Platz des Suchbereichs 3: symmetrisch zum realen Magazinplatz, der vor dem unter "WVar5" (Nummer des Referenz-Magazins) und "WVar6" (Nummer des Referenz-Platzes) angegebenen Platz liegt
WVar8:	INT	0, 1, 2, 3	Suchart Suche nach der Eigenschaft im Suchbereich: 0: alle Werkzeuge, schneidenspezifisch 1: erstes Werkzeug, schneidenspezifisch 2: alle Werkzeuge, über alle Schneiden 3: erstes Werkzeug, über alle Schneiden

14.17.4.31 PI-Dienst: TMCRT

Funktion: Multitool erzeugen

Der PI-Dienst dient zum Anlegen eines neuen Multitools mit einem definierten Bezeichner, einer optional vorgebbaren Multitool-Nummer, der Anzahl der Werkzeugplätze und die Art der Abstandskodierung.

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" =1 (TMCRT0) aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMCRT	Multitool anlegen
Unit:	INT	1, 2, 3 ... 10	TOA
Addr1:	STRING		Multitool-Bezeichner (max. 32 Zeichen)
WVar1:	INT	0	reserviert
WVar2:	INT	-1, 1, 2 ... 32000	Multitool-Nummer -1: Automatische Vergabe der Multitool-Nummer durch NC 1, 2, 3 ... 32000: Multitool-Nummer, Hinweis: Die Multitool-Nummer muss systemweit eindeutig sein
WVar3:	INT	2, 3, 4 ... MAX	Anzahl der Werkzeugplätze MAX = Parametrierte Anzahl in MD17504 \$MN_MAX_TOOLS_PER_MULTITool
WVar4:	INT	1, 2, 3	Art der Abstandskodierung

14.17.4.32 PI-Dienst: TMDLMT**Funktion: Multitool löschen**

Der PI-Dienst dient zum Löschen eines Multitools in allen Datenbausteinen, in denen es gespeichert ist. Im Multitool bestückte Werkzeuge sind anschließend nicht mehr bestückt und nicht mehr beladen, aber weiterhin definiert.

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".TMDLMT	Multitool löschen
Unit:	INT	1, 2, 3 ... 10	TOA
Addr1:	STRING		Multitool-Bezeichner (max. 32 Zeichen)
WVar1:	INT	0	reserviert
WVar2:	INT	-1, 1, 2 ... 32000	Multitool-Nummer -1: lösche das Multitool mit dem in "Addr1" angegebenen Namen 1, 2, 3 ... 32000: lösche das Multitool mit der angegebene Multitool-Nummer; Hinweis: Parameter "Addr1" wird nicht ausgewertet
WVar3:	INT	0, 1	Im Multitool enthaltene Werkzeuge: 0: nicht löschen 1: löschen

14.17.4.33 PI-Dienst: POSMT**Funktion: Multitool positionieren**

Der PI-Dienst dient zum Positionieren eines Multitools auf den programmierten Platz oder alternativ auf das programmierte Werkzeug, das sich auf einem der Plätze des Multitools befindet. Das Werkzeug selbst kann entweder mit seiner T-Nummer angegeben werden oder mit seinem Namen plus Duplonummer. Ein Multitool kann nur positioniert werden, wenn es sich auf einem Werkzeughalterplatz befindet und wenn keine Werkzeugkorrektur bezüglich dieses Werkzeughalterplatzes aktiv ist.

Positionsangabe

Die Positionsangabe kann alternativ vorgegeben werden über:

Nr.	Addr1	WVar1	WVar2	WVar3	WVar4
1	Leerstring	Nummer des Werkzeughalters	Werkzeugnummer	-1	
2	Werkzeugname	Nummer des Werkzeughalters	-1	Duplonummer	
3	Leerstring	Nummer des Werkzeughalters	-1	-1	Multitool-Platznummer

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".POSMT	Multitool positionieren
Unit:	INT	1, 2, 3 ... 10	TOA
Addr1:	STRING		Werkzeugname des im Multitool zu positionierenden Werkzeugs (max. 32 Zeichen)
WVar1:	INT	1, 2, 3 ... 999	Nummer des Werkzeughalters
WVar2:	INT	-1, 1, 2, 3 ... 32000	Werkzeugnummer (T-Nummer) des im Multitool zu positionierenden Werkzeugs 1, 2, 3 ... 32000: Werkzeugnummer -1: Die Werkzeugnummer ist irrelevant, "Addr1" und "WVar3" müssen programmiert werden
WVar3:	INT	-1, 1, 2, 3 ... 32000	Duplonummer des im Multitool zu positionierenden Werkzeugs 1, 2, 3 ... 32000: Duplonummer -1: Duplonummer ist irrelevant, "WVar2" muss programmiert werden
WVar4:	INT	1, 2, 3 ... 999	Multitool-Platznummer des Werkzeughalterplatzes, auf den positioniert werden soll

14.17.4.34 PI-Dienst: FDPLMT**Funktion: Leerplatz innerhalb des Multitools suchen / prüfen**

Der PI-Dienst dient zum Suchen eines freien Werkzeugplatzes innerhalb eines Multitools zur Aufnahme des angegebenen Werkzeugs bzw. zum Prüfen ob der angegebene Werkzeugplatz im Multitool zur Aufnahme des angegebenen Werkzeugs frei ist. Das Werkzeug kann über seine T-Nummer oder mit seinem Bezeichner und seiner Duplonummer angegeben werden.

Hinweis

Vor und nach diesem PI-Dienst ist der PI-Dienst MMCSEM mit Parameter "WVar1" =x (FDPLMT) aufzurufen. Siehe Kapitel "PI-Dienst: MMCSEM (Seite 1018)".

Positionsangabe

Das im Multitool zu positionierenden Werkzeug kann alternativ vorgegeben werden über:

- "Addr1" = <Leerstring>, "WVar1" = <Werkzeugnummer> und "WVar2" = <Duplonummer>
- "Addr1" = <Werkzeugname>, "WVar1" = -1 und "WVar2" = <Duplonummer>
- "Addr1" = <Leerstring>, "WVar1" = <Werkzeugnummer> und "WVar2" = -1

Die Positionsangabe kann alternativ vorgegeben werden über:

Erläuterung der Formalparameter

Signal	Typ	Wertebereich	Bedeutung
PIService:	ANY	"PI".FDPLMT	freien Werkzeugplatz innerhalb eines Multitools suchen / prüfen
Unit:	INT	1, 2, 3 ... 10	TOA
Addr1:	STRING		Werkzeugname des im Multitool zu positionierenden Werkzeugs (max. 32 Zeichen)
WVar1:	INT	-1, 1, 2, 3 ... 32000	Werkzeugnummer (T-Nummer) des im Multitool zu positionierenden Werkzeugs -1: Die Werkzeugnummer ist irrelevant, "Addr1" und "WVar2" müssen programmiert werden 1, 2, 3 ... 32000: Werkzeugnummer
WVar2:	INT	-1, 1, 2, 3 ... 32000	Duplonummer des im Multitool zu positionierenden Werkzeugs 1, 2, 3 ... 32000: Duplonummer -1: Die Duplonummer ist irrelevant, "WVar1" muss programmiert werden
WVar3:	INT	-1, 1, 2, 3 ... 32000	Nummer des Multitools 1, 2, 3 ... 32000: Nummer des Multitools -1: suche über alle Multitools hinweg nach einem Leerplatz bzw. prüfe über alle Multitools hinweg, ob in einem davon der angegebene Werkzeugplatz zur Aufnahme des Werkzeugs frei ist
WVar4:	INT	-1, 1, 2, 3 ... 999	Multitool-Platznummer des Werkzeugplatzes, auf den positioniert werden soll 1, 2, 3, ... 999: Multitool-Platznummer -1: suche einen beliebigen leeren Werkzeugplatz innerhalb des Multitools

14.17.5 FB5: GETGUD - GUD-Variable lesen

Funktion

Der Funktionsbaustein FB5 "GETGUD" dient zum Lesen von globalen Anwenderdaten (GUD) im Bereich NC- oder Kanal.

Anforderung zum Lesen von NC-Variablen

Aufruf des FB5 mit positivem Flankenwechsel, Parameter "Req" = 0 → 1

Parameter "Addr": Zeiger auf den Namen der GUD-Variablen, symbolisch mit "<Datenbaustein>.<VariablenName>"

Parameter "Area", "Unit", "Index1" und "Index2": Zusatz-Informationen zur Adressierung der Variablen

Mit Aktivierung des Parameters "CnvtToken" erhält man ein Token (Variablenzeiger) für die zu lesende GUD-Variable. Mit diesem kann dann über FB2 und FB3 mit Parameter "Addr1" ... "Addr8" = "<Token>" die GUD-Variable gelesen bzw. geschrieben werden. Zum **Schreiben**

von GUD-Variablen ist die Adressierung mittels Token zwingend erforderlich. Bei Verwendung des Tokens zur Adressierung von GUD Variablenfeldern ist zusätzlich der Parameter "Line1" ... "Line8" = "<Feldindex>" des FB2/FB3 zu versorgen.

Um eine GUD-Variable von Typ DOUBLE ohne Formatanpassung zu lesen, muss im Zielbereich ein ANY-Pointer mit Typ REAL2 programmiert werden. Z. B. P#M100.0 REAL 2. Der Wert der GUD-Variablen vom Typ DOUBLE wird dann als 64-Bit Gleitkommazahl in den PLC-Datenbereich übernommen.

Abschluss der Lese-Anforderung

Lese-Anforderung erfolgreich beendet: Parameter "Done" == 1.

Lese-Anforderung mit Fehler beendet: Parameter "Error" == 1, Fehlerursache in Parameter "State"

Voraussetzungen

- Freigabe der NC/PLC-Kommunikation durch OB100, FB1-Parameter "NCKomm" = 1
- Für den Datenbaustein, der den String mit dem Namen der GUD-Variable aufnimmt, muss mit dem S7-Symbol-Editor in der S7-Symbolliste des S7-Projekts ein Symbol vergeben werden (vgl. Aufrufbeispiel 1 unten: DB_GUDVAR). Im Datenbaustein muss ein String passender Länge angelegt werden, der den Namen der GUD-Variablen aufnimmt (vgl. Aufrufbeispiel 1 unten: "DB_GUDVAR".GUDVar1). Dieses Symbol wird dann an den Parameter "Addr" des FB5 übergeben, z. B. Addr := "DB_GUDVAR".GUDVar1. Falls der Token für die GUD-Variable für Folgeaufrufe des FB2/FB3 benötigt wird, muss in einem DB, z. B. dem gleichen DB, die 10-Byte-Tokenstruktur angelegt werden (vgl. Aufrufbeispiel 1 unten: "DB_GUDVAR".GUDVar1Token). Diese Struktur wird an den Parameter VarToken des FB5 spezifiziert, vgl. Aufrufbeispiel 1 unten: FB5, VarToken := "DB_GUDVAR".GUDVar1Token. Dieser Token wird dann beim Aufruf von FB2/FB3 an den Parametern "Addr1" ... "Addr8" spezifiziert, vgl. Aufrufbeispiel 1 unten: FB3, Addr1 := "GUDVar1Token".

Randbedingungen

- Der FB5 ist multiinstanzfähig.
- Jedem Aufruf des FB5 muss ein separater Instanz-DB aus dem Anwenderbereich zugeordnet werden.
- Das Lesen einer GUD-Variablen (FB5-Aufruf mit Req = 1) ist nur im zyklischen Teil des PLC-Grundprogramms (OB1) zulässig. Wird der Auftrag nicht gestartet (FB5-Aufruf mit Req = 0), können die Parameter auch schon im Anlaufteil des PLC-Grundprogramms (OB100) geschrieben werden. Der Auftrag kann dann mit den bereits geschriebenen Parametern im im zyklischen Teil des PLC-Grundprogramms (OB1) durch Aufruf des FB5 mit Req = 1 durchgeführt werden.
- Im Namen der GUD-Variablen sind nur Großbuchstaben zulässig.
- Das Lesen einer GUD-Variablen erstreckt sich in der Regel über mehrere PLC-Zyklen.

Hinweis**Fehlerfall**

Beim Lesen von Variablen aus unterschiedlichen Kanälen erfolgt folgende Rückmeldung:

- "Error" == TRUE
 - "State" == W#16#02
-

Hinweis

Um eine Variable von Typ DOUBLE von NC ohne Formatanpassung zu lesen, muss im Zielbereich ein ANY-Pointer mit Typ REAL2 angegeben werden (z. B.: P#M100.0 REAL 2). Erkennt das Grundprogramm beim Lesen einer Variablen von Typ DOUBLE als Zieltyp REAL 2, werden die Daten als 64-Bit Gleitkommazahl in den PLC-Datenbereich übernommen.

Deklaration der Funktion

```

FUNCTION_BLOCK FB5           // Name-Server
    KNOW_HOW_PROTECT
    VERSION: 3.0
VAR_INPUT
    Req :          BOOL;
    Addr :         ANY;    // Variablenname
    Area  :        BYTE;    // Bereich: NCK = 0, Kanal = 2
    Unit :         BYTE;
    Index1 :       INT;    // Feldindex 1
    Index2 :       INT;    // Feldindex 2
    CnvtToken :    BOOL;    // Umsetzung in 10 Byte Token
    VarToken :     ANY;    // Struct mit 10 Byte für den variablen Token
END_VAR

VAR_OUTPUT
    Error :        BOOL;
    Done :         BBOL;
    State :        WORD;
END_VAR

VAR_IN_OUT
    RD :           ANY;
END_VAR

BEGIN
END_FUNCTION_BLOCK

```


Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Req:	E	BOOL		Auftragsstart mit positiver Flanke
Addr:	E	ANY	"<DBName>".<VarName>	Variablenname in einer Variable vom-Typ STRING
Area:	E	BYTE	0, 2	Bereich: 0: NC 2: Kanal
Unit:	E	BYTE	1, 2, 3, ... 10	Bereich == NC: Unit:= 1 Bereich == Kanal: Kanalnummer
Index1:	E	INT	0, 1, 2, ... <max. Feldindex>	Feldindex 1 Bei Variablen ohne Feldindex: Index1 = 0
Index2:	E	INT	0, 1, 2, ... <max. Feldindex>	Feldindex 2 Bei Variablen ohne 2. Feldindex: Index2 = 0
CnvtToken:	E	BOOL	0 (FALSE), 1 (TRUE)	Aktivierung der Generierung eines 10 Byte Variablen-Token
VarToken:	E	ANY	"<DBName>".<VarName>	Adresse eines 10 Byte Tokens (siehe Beispiel)
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	Auftrag wurde negativ quittiert, bzw. konnte nicht ausgeführt werden
Done:	A	BOOL	0 (FALSE), 1 (TRUE)	Auftrag wurde erfolgreich ausgeführt.
State:	A	WORD	---	siehe Absatz "Fehlerkennungen"
RD:	E/A	ANY	P#Mm.n BYTE x ... P#DBnr.dbxm.n BYTE x	Zielbereich für gelesene Daten

Fehlerkennungen

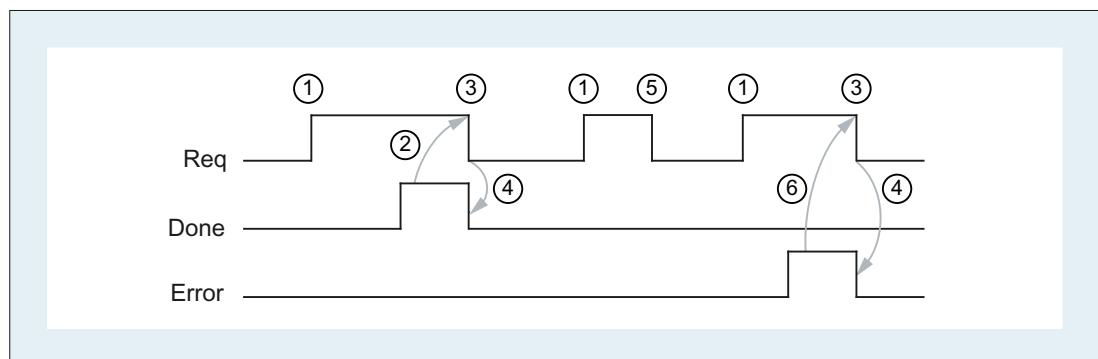
State		Bedeutung	Hinweis
WORT-H ¹⁾	WORT-L		
0	1	Zugriffsfehler	
0	2	Fehler im Auftrag	falsche Zusammenstellung von Var. in einem Auftrag
0	3	negative Quittung, Auftrag nicht ausführbar	interner Fehler, evtl. Abhilfe: NC-RESET
0	4	Datenbereiche oder Datentypen stimmen nicht überein	zu lesendes Datum in RD überprüfen;
1	4	nicht ausreichend lokaler Anwenderspeicher zur Verfügung	gelesene Var. ist länger als in RD angegeben
0	6	FIFO voll	Auftrag muss wiederholt werden, da die Warteschlange voll ist
0	7	Option nicht gesetzt	GP-Parameter "NCKomm" ist nicht gesetzt
0	8	falscher Zielbereich (SD)	RD dürfen keine Lokaldaten sein

State		Bedeutung	Hinweis
WORT-H 1)	WORT-L		
0	9	Übertragung belegt	Auftrag muss wiederholt werden
0	10	Fehler bei Adressierung	Unit enthält Wert 0
0	11	Variablenadresse ungültig	Addr. (bzw. Variablenname), Area, Unit überprüfen
1 - 8	13	Referenz des ANY Datums falsch	Gefordertes String bzw. NcVar-Datum ist nicht parametrierbar
0	15	String mehr als 32 Zeichen	GUD Variablenname zu lang
1) High-Byte > 0 ⇒ Nummer der Variablen, bei der der Fehler auftrat			

Projektierungsschritte

Für das Lesen einer GUD-Variablen ist der Name der GUD-Variablen in einer String-Variablen zu hinterlegen. Der Datenbaustein mit dieser String-Variablen ist in der Symbolliste zu definieren, so dass eine symbolische Zuweisung des Parameters "Addr" für den FB GETGUD erfolgen kann. Optional ist eine Strukturvariable in einem beliebigen Datenbereich der PLC für den Empfang des Variablenzeigers zu definieren, siehe Vorgabe im folgenden Beispiel.

Impulsdiagramm



- ① Anwender: Anforderung setzen, Req = 0 → 1
- ② FB5 erfolgreich beendet, Done = 1
Anwender: Anforderung zurücksetzen, IF Done == 1 THEN Req = 0
- ③ Anwender: IF Done == 1 THEN Anforderung zurücksetzen: 1 → 0
- ④ FB5 Auftragsbestätigung zurücksetzen, Done = 0
- ⑤ Anwender: IF Done == 0 UND Error == 0 THEN Anforderung zurücksetzen Req = 1 → 0 **nicht zulässig**
- ⑥ FB5 mit Fehler beendet, Error = 1
Anwender: Anforderung zurücksetzen, IF Done == 1 OR Error == 1 THEN Req = 0, eventuell weitere Fehlerbehandlung

Aufrufbeispiel 1

Lesen einer GUD-Variablen aus Kanal 1:

- Namen "GUDVAR1"
- Typ: INTEGER

Konvertierung in einen 10 Byte Variablen-Zeiger. Siehe Tabelle "Zuordnung der Datentypen" im Kapitel "FB2: GET - NC-Variable lesen (Seite 989)".

Lesen der GUD-Variablen: FB5 mit Instanz-DB111

```
// Datenbaustein für GUD-Variable
DATA_BLOCK DB_GUDVAR    // Zuordnung in Symbolliste vornehmen

STRUCT
    GUDVar1 : STRING[32] := 'GUDVAR1';           // Name ist vom
                                                    Anwender def.
    GUDVar1Token :
        STRUCT
            SYNTAX_ID : BYTE;
            bereich_u_einheit : BYTE;
            spalte : WORD;
            zeile : WORD;
            bausteintyp : BYTE;
            ZEILENANZAHL : BYTE;
            typ : BYTE;
            laenge : BYTE;
        END_STRUCT ;
    END_STRUCT;

BEGIN
END_DATA_BLOCK

// freier Anwender DB, als Instanz für FB5
DATA_BLOCK DB111

    FB5
    BEGIN
    END_DATA_BLOCK

// freier Anwender DB, als Instanz für FB3
DATE_BLOCK DB112

    FB3
    BEGIN
    END_DATA_BLOCK
```

```

// Lesen einer kanalspezifischen GUD-Variablen aus Kanal 1, mit
// Konvertierung in einen Variablen-Zeiger
Function FC "VariablenCall" : VOID
U      E 7.7;           // freie Taste Maschinensteuertafel
S      M 100.0;         // Req aktivieren
U      M 100.1;         // Fertigmeldung Done
R      M 100.0;         // Auftrag beenden
U      E 7.6;           // Fehlerquittierung von Hand
U      M 102.0;         // Fehler steht an
R      M 100.0;         // Auftrag beenden
CALL FB5, DB111(
    Req      := M 100.0,           // Startflanke für Lesen
    Addr     := "DB_GUDVAR".GUDVar1, // Name des GUDs
    Area     := B#16#2,           // Kanal-Variable
    Unit     := B#16#1,           // Kanal 1
    Index1   := 0,                // kein Feldindex 1
    Index2   := 0,                // kein Feldindex 2
    CnvtToken := TRUE,            // Anforderung: Umsetzung in
                                // 10 Byte Token
    VarToken  := "DB_GUDVAR".GUDVar1Token, // Adresse des Tokens
    Error     := M 102.0,
    Done      := M 100.1,
    State     := MW 104,
    RD        := P#DB99.DBX0.0 DINT 1 // freier Speicherplatz
);

```

Schreiben der GUD-Variablen: FB3 mit Instanz-DB112

GUD-Variablen-Token aus FB5, Parameter: "VarToken" zum Schreiben mit FB3, Parameter "Addr1"

```

CALL FB3, DB112(
    Req      := M 200.0,
    NumVar   := 1,                // eine GUD-Variable
    Addr1    :=                  // Token
    "DB_GUDVAR".GUDVar1Token,
    Error     := M 102.0,
    Done      := M 100.1,
    State     := MW 104,
    SD1      := P#DB99.DBX0.0 DINT 1
);

```

Aufrufbeispiel 2

Lesen einer GUD-Variablen aus Kanal 1:

- Namen "GUD_STRING"
- Typ: STRING[32]

Konvertierung in einen 10 Byte Variablen-Zeiger.

Lesen der GUD-Variablen: FB5 mit Instanz-DB111

```
// Datenbaustein für GUD-Variable
DATA_BLOCK DB_GUDVAR    // Zuordnung in Symbolliste vornehmen

STRUCT
    GUDVarS : STRING[32] := 'GUD_STRING';    // Name vom Anwender definiert
    GUDVarSToken :
    STRUCT
        SYNTAX_ID : BYTE;
        bereich_u_einheit : BYTE;
        spalte : WORD;
        zeile : WORD;
        bausteintyp : BYTE;
        ZEILENANZAHL : BYTE;
        typ : BYTE;
        laenge : BYTE;
    END_STRUCT ;

    string_of_GUD : STRING[30];    // muss mindestens so lang sein wie
                                   // die Definition von 'GUD_STRING'!
    new_name : STRING[30] := 'GUD_123';
END_STRUCT;

BEGIN
END_DATA_BLOCK

// freier Anwender DB, als Instanz für FB5
DATA_BLOCK DB111

    FB5

BEGIN
END_DATA_BLOCK

// freier Anwender DB, als Instanz für FB3
DATE_BLOCK DB112

    FB3

BEGIN
END_DATA_BLOCK

// Lesen einer kanalspezifischen GUD-Variablen aus Kanal 1, mit
Konvertierung in einen Variablen-Zeiger
Function FC "VariablenCall" : VOID
```

```

U      E 7.7;           // freie Taste Maschinensteuertafel
S      M 100.0;         // Req aktivieren
U      M 100.1;         // Fertigmeldung Done
R      M 100.0;         // Auftrag beenden
U      E 7.6;           // Fehlerquittierung von Hand
U      M 102.0;         // Fehler steht an
R      M 100.0;         // Auftrag beenden
CALL FB5, DB111(
    Req      := M 100.0,    // Startflanke für Lesen
    Addr     := "DB_GUDVAR".GUDVarS,
    Area     := B#16#2,    // Kanal-Variable
    Unit     := B#16#1,    // Kanal 1
    Index1   := 0,         // kein Feldindex
    Index2   := 0,         // kein Feldindex
    CnvtToken := TRUE,     // Anforderung: Umsetzung in 10 Byte
                        Token
    VarToken  := "DB_GUDVAR".GUDVarSToken, // Adresse des Tokens
    Error     := M 102.0,
    Done      := M 100.1,
    State     := MW 104,
    RD        := "DB_GUDVAR".string_of_GUD
);

```

Schreiben der GUD-Variablen: FB3 mit Instanz-DB112

GUD-Variablen-Token aus FB5, Parameter: "VarToken" zum Schreiben mit FB3, Parameter "Addr1"

```

CALL FB3, DB112(
    Req      := M 200.0,
    NumVar   := 1,         // eine GUD-Variable
    Addr1    := "DB_GUDVAR".GUDVarSToken, // Token
    Error     := M 102.0,
    Done      := M 100.1,
    State     := MW 104,
    SD1      := "DB_GUDVAR".new_name
);

```

Zuordnung der Datentypen

Siehe Tabelle "Zuordnung der Datentypen" im Kapitel "FB2: GET - NC-Variable lesen (Seite 989)".

14.17.6 FB7: PI_SERV2 - PI-Dienst anfordern

Funktion

Der Funktionsbaustein FB7 hat bis auf die größere Anzahl an "WVar"-Parametern ("WVar11" - "WVar16") die gleiche Funktionalität wie der Funktionsbaustein FB4. Es wird empfohlen, den Funktionsbaustein FB7 anstelle des FB4 zu verwenden.

Zur ausführlichen Beschreibung siehe Kapitel "FB4: PI_SERV - PI-Dienst anfordern (Seite 1005)".

Deklaration der Funktion

```

FUNCTION_BLOCK FB7
Var_INPUT
    Req:          BOOL;
    PIService:    ANY;
    Unit:         INT;
    Addr1:        ANY;
    Addr2:        ANY;
    Addr3:        ANY;
    Addr4:        ANY;
    WVar1:        WORD;
    WVar2:        WORD;
    WVar3:        WORD;
    WVar4:        WORD;
    WVar5:        WORD;
    WVar6:        WORD;
    WVar7:        WORD;
    WVar8:        WORD;
    WVar9:        WORD;
    WVar10:       WORD;
    WVar11:       WORD;
    WVar12:       WORD;
    WVar13:       WORD;
    WVar14:       WORD;
    WVar15:       WORD;
    WVar16:       WORD;
END_VAR
VAR_OUTPUT
    Error :       BOOL;
    Done :       BOOL;
    State :       WORD;
END_VAR

```

14.17.7 FB9: MzuN - Bedieneinheitenumschaltung

Funktion

Der Funktionsbaustein FB9 "MzuN" dient zur Umschaltung von Bedienkomponenten (MCP/OP), die über ein Bussystem mit einer/mehreren Steuerungsbaugruppen NCU verbunden sind.

Die Schnittstelle zwischen den einzelnen Bedieneinheiten und der NCU (PLC) ist die M : N-Schnittstelle im Datenbaustein DB19. Der FB9 arbeitet mit den Signalen dieser Schnittstellen.

Folgende Grundfunktionen werden neben der Initialisierung, der Lebenszeichenüberwachung und der Fehlerrouinen vom Baustein für die Bedieneinheitenumschaltung ausgeführt:

Tabellarische Übersicht der Funktionen:	
Grundfunktion	Bedeutung
HMI klopft an	HMI hat den Wunsch an eine NCU online zu gehen
HMI kommt	HMI verbindet sich mit einer NCU
HMI geht	HMI trennt sich von einer NCU
Verdrängung	HMI muss Verbindung zu einer NCU lösen
Bedienfocuswechsel im Serverbetrieb	Bedienfocus von einer auf die andere NCU umschalten
Aktiver / Passiver Bedienmodus	Bedienen und beobachten / nur beobachten
MCP-Umschaltung	MCP kann optional mit dem HMI umgeschaltet werden

Hinweis

Der Baustein ist vom Anwenderprogramm aufzurufen. Hierbei ist ein Instanz-DB mit beliebiger Nummer vom Anwender beizustellen. Der Aufruf ist nicht Multi-Instanzfähig.

Kurzbeschreibung einiger wichtiger Funktionen

Aktiver-/ Passiver Bedienmodus:

Ein Online-HMI kann zwei verschiedene Bedienmodi haben:

Aktiv-Modus: Bediener kann bedienen und beobachten

Passiv-Modus: Bediener kann beobachten (nur Kopfzeile HMI)

Nach Umschaltung auf eine NCU fordert sie in der PLC der Online-NCU zuerst den aktiven Bedienmodus an. Falls zwei HMIs zu einer Zeit an einer NCU online sind, ist eine der beiden immer im aktiven, die andere im passiven Bedienmodus. Der Bediener kann den aktiven Bedienmodus an der passiven HMI per Tastendruck anfordern.

MCP-Umschaltung

Zusammen mit dem HMI kann optional eine ihm zugeordnete MCP mit umgeschaltet werden. Voraussetzung ist, dass die MCP-Adresse im Parameter "mstt_adress" der Konfigurationsdatei NETNAMES.INI der HMI 's eingetragen und "MCPEnable" = TRUE gesetzt ist. Die MCP des passiven HMI 's ist deaktiviert. Dadurch gibt es an einer NCU immer nur eine aktive MCP.

Hochlaufbedingung

Um zu verhindern, dass bei einem Neustart der NCU, die davor zuletzt angewählte MCP aktiviert wird, muss beim Aufruf von FB1 in OB100 der Eingangsparameter "MCP1BusAdr" = **255** (Adresse 1. MCP) und "MCP1Stop" = **TRUE** (1. MCP ausschalten) gesetzt werden.

Freigaben

Wenn von einer MCP auf eine andere umgeschaltet wird, dann bleiben eventuell eingeschaltete Vorschub- sowie Achsfreigaben erhalten.

Hinweis

Die zur Zeit der Umschaltung betätigten Tasten wirken bis zur Aktivierung der neuen MCP (vom HMI, der anschließend aktiviert wird) weiter. Auch die Override-Stellungen für Vorschub, Spindel bleiben erhalten. Um die betätigten Tasten zu deaktivieren, ist bei fallender Flanke des Signals DB10.DBX104.0 (MSTT 1 bereit) das Eingangsabbild der Maschinensteuer-Signale auf nicht betätigte Signalpegel zu legen. Die Override-Stellungen sollten unverändert bleiben. Maßnahmen zur Deaktivierung der Tasten sind im PLC-Anwenderprogramm zu realisieren (siehe Beispiel "Override-Umschaltung").

Der Aufruf ist nur im zyklischen Programm OB1 zulässig.

Deklaration der Funktion

```

FUNKTION_BLOCK FB9
VAR_INPUT
    Quit:          BOOL;          // Quittierung Alarme
    OPMixedMode:   BOOL:=FALSE;   // Mischbetrieb mit nicht M zu N fähigen
                                OP // deaktiviert

    ActivEnable:   BOOL:=TRUE;    // wird nicht unterstützt
    MCPEnable:     BOOL:=TRUE;    // MCP Umschaltung aktivieren
END_VAR
VAR_OUTPUT
    Alarm1:        BOOL;          // Alarm: Fehler HMI-Busadresse, Bustyp!
    Alarm2:        BOOL;          // Alarm: Keine Bestätigung HMI 1
                                offline!
    Alarm3:        BOOL;          // Alarm: HMI 1 geht nicht offline!
    Alarm4:        BOOL;          // Alarm: Keine Bestätigung HMI 2
                                offline!

```

```

Alarm5:      BOOL;           // Alarm: HMI 2 geht nicht offline!
Alarm6:      BOOL;           // Alarm: Anklopf- HMI geht nicht online!
Report:      BOOL;           // Meldung: Lebenszeichenüberwachung
ErrorMMC:    BOOL;           // Fehlererkennung HMI
END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Quit:	E	BOOL	0 (FALSE), 1 (TRUE)	Quittierung Alarme
OPMixedMode:	E	BOOL	0 (FALSE), 1 (TRUE)	Mischbetrieb mit einem nicht M zu N fähigen OP
ActivEnable:	E	BOOL	0 (FALSE), 1 (TRUE)	Funktion wird nicht unterstützt. Bedienfeldumschaltung Verriegelung über MMCx_SHIFT_LOCK im DB19
MCPEnable:	E	BOOL	0 (FALSE), 1 (TRUE)	MCP Umschaltung aktivieren 1: MCP wird mit Bedienfeld umgeschaltet. 0: MCP wird nicht mit Bedienfeld umgeschaltet. Hiermit kann eine MCP fest verbunden werden. Siehe auch MMCx_MSTT_SHIFT_LOCK im DB19.
Alarm1:	A	BOOL	0 (FALSE), 1 (TRUE)	Alarm: Fehler HMI -Busadresse, Bus-typ!
Alarm2:	A	BOOL	0 (FALSE), 1 (TRUE)	Alarm: Keine Bestätigung HMI 1 offline!
Alarm3:	A	BOOL	0 (FALSE), 1 (TRUE)	Alarm: HMI 1 geht nicht offline!
Alarm4:	A	BOOL	0 (FALSE), 1 (TRUE)	Alarm: Keine Bestätigung HMI 2 offline!
Alarm5:	A	BOOL	0 (FALSE), 1 (TRUE)	Alarm: HMI 2 geht nicht offline!
Alarm6:	A	BOOL	0 (FALSE), 1 (TRUE)	Alarm: Anklopf- HMI geht nicht online!
Report:	A	BOOL	0 (FALSE), 1 (TRUE)	Meldung: Lebenszeichenüberwachung HMI
ErrorMMC:	A	BOOL	0 (FALSE), 1 (TRUE)	Fehlererkennung HMI

Aufrufbeispiel für FB9

```

CALL FB9, DB109(
Quit           := Fehler_Quitt,    // z.B. MCP-RESET
OPMixedMode    := FALSE,
ActivEnable     := TRUE,
MCPEnable      := TRUE);          // Freigabe MCP Umschaltung

```

Hinweis

Der Eingangsparameter "MCPEnable" muss TRUE sein, um die MCP Umschaltung zu ermöglichen. Der Defaultwert dieser Parameter ist so geschaltet und muss beim Aufruf der Funktion nicht zusätzlich beschaltet werden.

Alarmer, Fehler

Die Ausgangsparameter "Alarm1" bis "Alarm6" und "Report" existieren als Information in der PLC, dass bei M zu N Fehler aufgetreten sind, die durch die Alarmer 410900 - 410906 am HMI sichtbar werden.

Konnte eine HMI-Funktion nicht ausgeführt werden (bei der die Fehlermeldung nicht angezeigt werden kann), wird dies am Zustandsparameter "ErrorMMC" mit logisch 1 angezeigt (z. B. Fehler im Hochlauf, wenn keine Verbindung aufgebaut wird).

Aufrufbeispiel für FB (Aufruf im OB100)

```
CALL "RUN_UP" , "gp_par" (
    MCPNum           := 1,
    MCP1In           := P#E 0.0,
    MCP1Out          := P#A 0.0,
    MCP1StatSend     := P#A 8.0,
    MCP1StatRec      := P#A 12.0,
    MCP1BusAdr       := 255,           // Adresse 1. MCP
    MCP1Timeout      := S5T#700MS,
    MCP1Cycl         := S5T#200MS,
    MCP1Stop         := TRUE,         // MCP ausgeschaltet
    NCCyclTimeout    := S5T#200MS,
    NCRunupTimeout   := S5T#50S);
```

Beispiel: Override-Umschaltung

```
// verwendete Hilfsmerker M100.0, M100.1, M100.2, M100.3
// Flanke positiv von MCP1Ready muss Override prüfen
// und Maßnahmen für Aktivierung
// MCP Baustein einleiten
// Dieses Beispiel gilt für Vorschub-Override;
// für Spindel-Override sind Nahtstellen- und Eingangsbyte auszutauschen.
U    DB10.DBX104.0;    // MCP1Ready
FN   M    100.0;       // Flankenmerker 1
SPBN weil;
S    M    100.2;       // Hilfsmerker 1 setzen
R    M    100.3;       // Hilfsmerker 2 rücksetzen
```

```

// Override speichern
    L DB21.DBB4;          // Nahtstelle Vorschub-Override
    T EB28;               // Zwischenspeicher (freies Eingangs-
                        // oder Merkerbyte)

weil:
U    M    100.2;          // Umschaltung erfolgt
O    DB10.DBX104.0;      // MCP1Ready
SPBN weil;
U    DB10.DBX104.0;      // MCP1Ready
FP    M    100.1;        // Flankenmerker 2
SPB weil;
U    M    100.2;          // Umschaltung erfolgt
R    M    100.2;          // Hilfsmerker 1 rücksetzen
SPB weil;
U    M    100.3;          // Vergleich ist erfolgt
SPB MCP;                  // MCP-Programm aufrufen
// gespeicherten Override auf Nahtstelle der umgeschalteten MCP lenken
// bis die Override-Werte übereinstimmen
    L EB28;               // Zwischenspeicher auf
    T DB21.DBB4;          // Nahtstelle Override lenken
    L EB3;                // Override-Eingangsbyte für Vorschub
    <>i;                  // Übereinstimmung?
SPB weil;                  // nein, Absprung
S    M    100.3;          // ja, Hilfsmerker 2 setzen
// nach Übereinstimmung der Override-Werte MCP-Programm wieder aufrufen
MCP: CALL "MCP_IFM" (     // FC19
    BAGNo      := B#16#1,
    ChanNo     := B#16#1,
    SpindleIFNo := B#16#0,
    FeedHold   := M 101.0,
    SpindleHold := M 101.1);
weil: NOP                0;

```

14.17.8 FB10: Sicherheits-Relais (SI-Relais)

Funktion

Der SPL-Funktionsbaustein FB10 "Sicherheitsrelais" für "Safety Integrated" ist das Äquivalent auf der PLC zu der gleichnamigen NC-Funktion. Der Standard-SPL-Baustein "Sicherheitsrelais" ist für die Anforderungen einer Not-Halt-Realisierung mit sicherer programmierbarer Logik ausgelegt. Er kann aber auch für weitere ähnliche Anforderungen, z. B. einer Schutztür-Ansteuerung, genutzt werden.

Die Funktion enthält 3 Eingangsparameter ("In1", "In2", "In3"). Bei Schalten eines dieser Parameter auf den Wert 0, wird der Ausgang "Out0" unverzüglich abgeschaltet und die Ausgänge "Out1", "Out2" und "Out3" über die parametrisierten Zeitwerte (Parameter "TimeValue1", "TimeValue2", "TimeValue3") verzögert abgeschaltet. Die Ausgänge werden unverzüglich wieder eingeschaltet, wenn die Eingänge "In1" bis "In3" den Wert 1 annehmen und an einem der Quittierungseingänge "Quit1", "Quit2" ein positiver Flankenwechsel erkannt wird.

Aufrufmerker

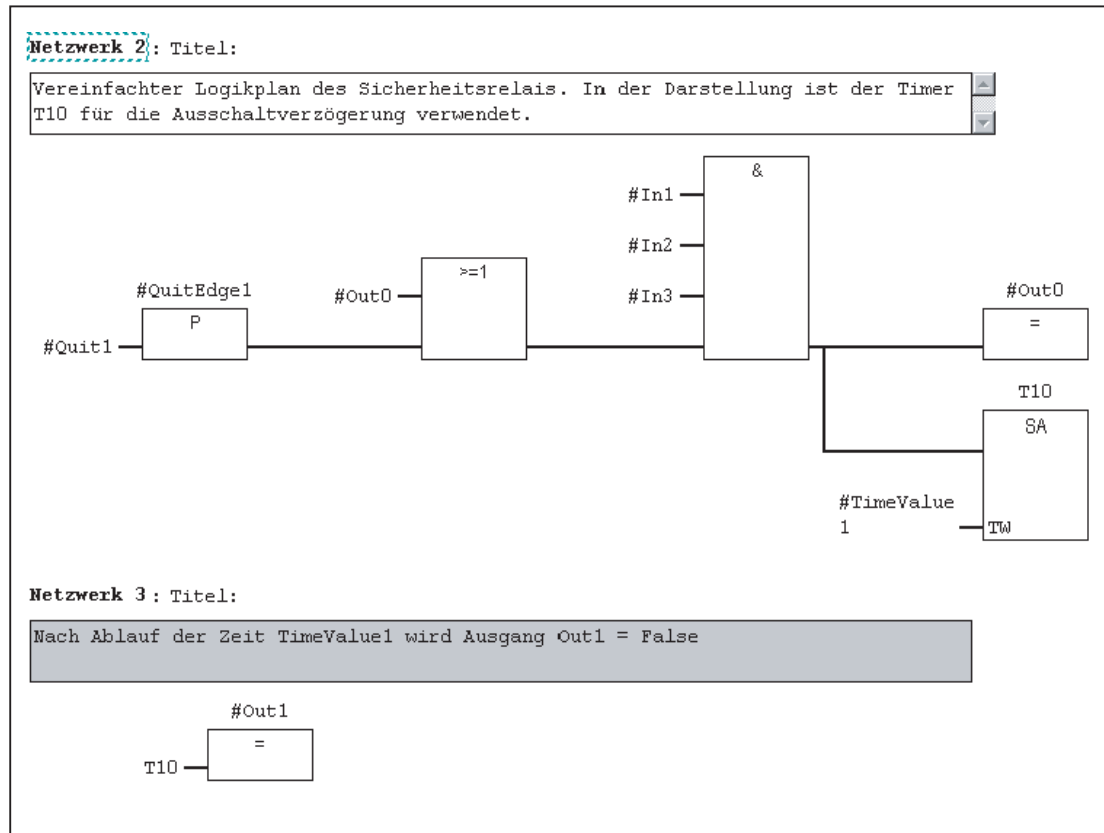
Um die Ausgänge in Grundstellung (Werte = 0) nach Neueinschalten zu bringen, ist der Parameter "FirstRun" wie folgt zu beschalten. Der Parameter "FirstRun" muss beim 1. Durchlauf nach Hochlauf der Steuerung über ein remanentes Datum (Merkerbit, Bit im Datenbaustein) auf den Wert TRUE geschaltet sein. Das Datum kann z. B. im OB100 vorbesetzt werden. Der Parameter "FirstRun" wird nach dem erstmaligen Durchlauf des FB10 auf den Wert FALSE zurückgesetzt. Für jeden Aufruf mit eigener Instanz ist für den Parameter "FirstRun" ein separates Datum zu verwenden.

Randbedingungen

- Der FB10 ist multiinstanzfähig
- Der FB10 ist ab dem Start des SPL-Programms einmalig pro SI-Relais im zyklischen Teil des PLC-Grundprogramms (OB1) aufzurufen.
- Jedem Aufruf des FB10 muss ein separater Instanz-DB aus dem Anwenderbereich zugeordnet werden.

Vereinfachtes Prinzipschaltbild im Funktionsplan

In der folgenden Darstellung sind nur ein Quittiereingang "Quit1" und ein ausschaltverzögerter Ausgang "Out1" dargestellt. Schaltung für "Quit2" und die weiteren ausschaltverzögerten Ausgänge sind identisch realisiert. In dem Funktionsplan fehlt auch der Parameter "FirstRun". Die Wirkungsweise ist oben beschrieben.



Deklaration der Funktion

```
FUNCTION_BLOCK FB10
VAR_INPUT
    In1 :      BOOL;
    In2 :      BOOL;
    In3 :      BOOL;
    Quit1 :    BOOL;
    Quit2 :    BOOL;
    TimeValue1 : TIME;
    TimeValue2 : TIME;
    TimeValue3 : TIME;
END_VAR
VAR_OUTPUT
    Out0 :     BOOL;
```

```

        Out1 :      BOOL;
        Out2 :      BOOL;
        Out3 :      BOOL;
    END_VAR
    VAR_INOUT
        FirstRun :    BOOL;
    END_VAR

```

Erläuterung der Formalparameter

Parameter	Art	Typ	Wertebereich	Bedeutung
In1:	E	BOOL	0 (FALSE), 1 (TRUE)	Eingang 1
In2:	E	BOOL	0 (FALSE), 1 (TRUE)	Eingang 2
In3:	E	BOOL	0 (FALSE), 1 (TRUE)	Eingang 3
Quit1:	E	BOOL	0 (FALSE), 1 (TRUE)	Quittier-Eingang 1
Quit2:	E	BOOL	0 (FALSE), 1 (TRUE)	Quittier-Eingang 2
TimeValue1:	E	S5TIME		Zeit-Wert 1 für Ausschaltverzögerung
TimeValue2:	E	S5TIME		Zeit-Wert 2 für Ausschaltverzögerung
TimeValue3:	E	S5TIME		Zeit-Wert 3 für Ausschaltverzögerung
Out0:	A	BOOL	0 (FALSE), 1 (TRUE)	Ausgang unverzögert
Out1:	A	BOOL	0 (FALSE), 1 (TRUE)	Ausgang verzögert mit TimeValue1
Out2:	A	BOOL	0 (FALSE), 1 (TRUE)	Ausgang verzögert mit TimeValue2
Out3:	A	BOOL	0 (FALSE), 1 (TRUE)	Ausgang verzögert mit TimeValue3
FirstRun:	E/A	BOOL	0 (FALSE), 1 (TRUE)	Aufrufmerker

14.17.9 FB11: Bremsentest

Funktion

Die Funktionsprüfung der Bremsenmechanik sollte für alle Achsen verwendet werden, die durch eine Haltebremse gegen Bewegungen im ungeregelten Betrieb gesichert werden müssen. Der Hauptanwendungsfall sind "Hängende Achsen".

In seinem PLC-Anwenderprogramm kann der Maschinenhersteller regelmäßig (Richtwert alle 8 Stunden) zu einem geeigneten Zeitpunkt die Bremse schließen und vom Antrieb zusätzlich zur Gewichtskraft der Achse ein Moment / Kraft aufbringen lassen. Im fehlerfreien Zustand kann die Bremse das notwendige Bremsmoment / die Bremskraft aufbringen. Die Achse wird sich dabei kaum bewegen.

Im Fehlerfall wird ein Verlassen des parametrierten Überwachungsfensters für den Positionswert erkannt. Ein Absacken der Achse wird durch den Lageregler verhindert. Die Funktionsprüfung der Bremsenmechanik negativ quittiert.

Literatur

Eine ausführliche Beschreibung der Parametrierung von NC und Antrieb im Rahmen der Funktion "Safety Integrated" findet sich in:

Funktionshandbuch Safety Integrated

Start des Bremsentests

Der Start des Bremsentests muss immer im Stillstand der Achse erfolgen. Für den gesamten Zeitraum des Bremsentests sind die Freigabesignale der parametrierten Achse auf Freigabe zu setzen (z. B. Reglersperre, Vorschubfreigabe). Weiterhin ist das Signal an Achse/Spindel DB31,DBX28.7 (PLC kontrolliert Achse) während der gesamten Dauer des Tests auf Zustand 1 vom Anwenderprogramm zu setzen.

Vor Aktivierung des NC/PLC-Nahtstellen DB31,DBX28.7 (PLC kontrolliert Achse) ist die Achse als "neutrale Achse" zu schalten, z. B. ist DB31,DBX8.0 - 8.3 (NC-Achse Kanal zuordnen) auf Kanal 0 zu setzen, sowie DB31,DBX8.4 (Aktiviersignal bei Veränderung dieses Bytes) zu setzen.

Die Rückmeldung:

- über den aktuellen Zustand kann abgefragt werden in DB31,DBB68.
- der NC über das Signal DB31,DBX63.1 (PLC kontrolliert Achse) ist abzuwarten bevor der Baustein gestartet wird. Die Richtung, in die der Antrieb sein Moment / seine Kraft aufbringt, wird von der PLC durch eine "Verfahrbewegung", z. B. über den FC18 vorgegeben.

Der Zielpunkt dieser Verfahrbewegung muss gefahrlos erreicht werden können (keine Kollision), falls die Bremse das notwendige Moment / die Kraft nicht aufbringen kann.

Hinweis

Hinweis zu FC18

Falls FC18 für die gleiche Achse im weiteren Verlauf des Anwenderprogramms aufgerufen wird, müssen die Aufrufe gegeneinander verriegelt werden. Dies kann z. B. über einen gemeinsamen Aufruf dieser Funktion mit einer verriegelten gemeinsamen Datenschnittstelle für die Parameter des FC18 erfolgen. Eine 2. Möglichkeit ist der Mehrfachaufruf des FC18, wobei der inaktive FC18 programmäßig nicht durchlaufen wird. Eine Verriegelung für Mehrfachnutzung ist vorzusehen.

Gliederung eines Bremsentests

	Schritt	Erwartete Rückmeldung	Überwachungs-Zeitwert
1	Bremsentest starten	DBX71.0 = 1	TV_BTactiv
2	Bremse schließen	Bclosed = 1	TV_Bclose
3	Fahrbehl ausgeben	DBX64.6 OR DBX64.7	TV_FeedCommand
4	Prüfung Fahrbehl ausgegeben	DBX62.5 = 1	TV_FXSreached
5	Haltezeit abwarten	DBX62.5 = 1	TV_FXShold

Schritt		Erwartete Rückmeldung	Überwachungs-Zeitwert
6	Abwahl Bremsen Test / Bremsen öffnen	DBX71.0 = 0	TV_BTactiv
7	Test ok ausgeben	---	---

Randbedingungen

- Der FB2 ist multiinstanzfähig.
- Jedem Aufruf des FB11 muss ein separater Instanz-DB aus dem Anwenderbereich zugeordnet werden.

Deklaration der Funktion

```

Function_BLOCK FB11
VAR_INPUT
    Start          BOOL;
    Quit :         BOOL;
    Bclosed :      BOOL;
    Axis :         INT;
    TimerNo :      TIMER;
    TV_BTactiv :   S5TIME;
    TV_Bclose :    S5TIM;
    TV_FeedCommand : S5TIME;
    TV_FXSreached : S5TIME;
    TV_FXShold :   S5TIME;
END_VAR
VAR_OUTPUT
    CloseBrake : BOOL;
    MoveAxis :   BOOL;
    Done :      BOOL;
    Error :     BOOL;
    State : :   BYTE;
END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Bedeutung
Start	E	BOOL	Start des Bremsentests
Quit	E	BOOL	Fehler Quittierung
Bclosed	E	BOOL	Rückmelde-Eingang ob Bremse schließen angesteuert ist (einkanlig - PLC)
Axis	E	INT	Achsnummer der zu testenden Achse
TimerNo	E	TIMER	Timer aus Anwenderprogramm

Signal	Art	Typ	Bedeutung
TV_BTactiv	E	S5TIME	Überwachungs-Zeitwert → Bremsentest aktiv, Prüfung des Achs-Signals DBX71.0
TV_Bclose	E	S5TIME	Überwachungs-Zeitwert → Bremse schließen. Prüfung des Eingangssignals Bclosed nachdem Ausgang CloseBrake gesetzt wurde
TV_FeedCommand	E	S5TIME	Überwachungs-Zeitwert → Fahrbefehl ausgegeben Prüfung Fahrbefehl nachdem MoveAxis gesetzt wurde
TV_FXSreched	E	S5TIME	Überwachungs-Zeitwert → Festanschlag erreicht
TV_FXShold	E	S5TIME	Überwachungs-Zeitwert → Bremse testen
CloseBrake	A	BOOL	Anforderung für Bremse schließen
MoveAxis	A	BOOL	Anforderung Verfahrbewegung anstoßen
Done	A	BOOL	Test erfolgreich beendet
Error	A	BOOL	Fehler aufgetreten
State	A	BYTE	siehe Absatz "Fehlerkennungen"

Fehlererkennungen

State	Bedeutung
0	Kein Fehler
1	Startbedingungen nicht gegeben, z. B. Achse nicht in Regelung / Bremse zu / Achsensperre steht an
2	Bei Anwahl Bremsentest erfolgt keine Rückmeldung der NC im Signal "Bremsentest aktiv"
3	Keine Rückmeldung "Bremse eingefallen" durch Eingangssignal Bclosed
4	Keine Fahrbefehlausgabe (z. B. Bewegung der Achse wurde nicht gestartet)
5	Festanschlag wird nicht erreicht → Achs-RESET wurde ausgelöst.
6	Verfahrsperr/Anfahrt zu langsam → Festanschlag kann nicht erreicht werden. Überwachungszeit TV FXSreached abgelaufen.
7	Bremse hält überhaupt nicht (Endposition wird erreicht) / Anfahrsgeschwindigkeit zu groß
8	Bremse bricht während der Haltezeit auf
9	Fehler bei Bremsentest-Abwahl
10	interner Fehler
11	Signal "PLC kontrolliert Achse" nicht eingeschaltet vom Anwenderprogramm

Aufrufbeispiel für FB11:

```

UN    M    111.1;    //Anforderung Bremse schließen Z-Achse von FB
=     A    85.0;     //Ansteuerung Bremse Z-Achse
AUF   A    Axis3";   //Bremsentest Z-Achse
O     E    73.0;     //Anstoß für den Bremsentest Z-Achse
O     M    110.7;    //Bremsentest läuft
FP    M    110.0;
UN    M    111.4;    //Fehler aufgetreten
S     M    110.7;    //Bremsentest läuft

```

```

S      M      110.6;  //nächster Schritt
SPBN   m001
L      DBB     68;
UW     W#16#F;
T      MB      115;   //Kanalzustand merken
L      B#16#10
T      DBB     8;     //Neutrale Achse anfordern

m001:  U      DBX     68.6;  //Rückmeldung Achse ist neutral
      U      M      110.6;
      FP     M      110.1;
      R      M      110.6;
      S      M      110.5;  //nächster Schritt
      S      DBX     28.7;  //PLC-kontrollierte Achse anfordern

      U      DBX     63.1;  //Rückmeldung Achse wird von PLC kontrolliert
      U      M      110.5;
      FP     M      110.2;
      R      M      110.5;
      S      M      111.0;  //Start Bremsentest für FB

CALL   FB11, DB211  (//Bremsentestbaustein
      Start      := M 111.0,  //Start des Bremsentests
      Quit       := E 3.7,    //Fehlerquittierung mit RESET-
                               Taste
      Bclosed    := E 54.0,    //Rückmeldung Bremse schließen
                               //angesteuert
      Axis       := 3,        //Achsnnummer der zu testenden
                               Achse
                               //Z-Achse
      TimerNo    := T 110,    //Timernummer
      TV_BTactiv := S5T#200MS, //Überwachungs-Zeitwert:
                               //Bremsentest aktiv DBX71.0
      TV_Bclose  := S5T#1S,   //Überwachungs-Zeitwert:
                               //Bremse geschlossen
      TV_FeedCommand := S5T#1S, //Überwachungs-Zeitwert:
                               //Fahrbefehl ausgegeben
      TV_FXSreache := S5T#1S, //Überwachungs-Zeitwert:
                               //Festanschlag erreicht
      TV_FXShold  := S5T#2S,   //Überwachungs-Zeitwert:
                               //Testzeit Bremse
      CloseBrake  := M 111.1,  //Anforderung Bremse schließen
      MoveAxis    := M 111.2,  //Anforderung Verfahrbewegung //
                               anstoßen
      Done        := M 111.3,  //Test erfolgreich beendet
      Error       := M 111.4,  //Fehler aufgetreten
      State       := MB 112);  //Fehler Status

```

```

      AUF          "Axis3"; //Bremsentest Z-Achse

      U      M      111.2;  //Moveaxis
      FP     M      111.5;  //FC18 Start
      S      M      111.7;  //Start FC18

      O      M      111.3;  //Test erfolgreich beendet
      O      M      111.4;  //Fehler aufgetreten
      FP     M      110.3;
      R      DBX     28.7;  //Anforderung PLC-kontrollierte Achse

      UN      DBX     63.1;  //Rückmeldung Achse wird von PLC kontrolliert
      U      M      111.0;  //Start Bremsentest für FB
      U      M      110.7;  //Bremsentest läuft
      FP     M      110.4;
      R      M      111.0;  //Start Bremsentest für FB
      R      M      110.7;  //Bremsentest läuft

//optional begin
      SPBN  m002;
      L      MB      115;    //alter Kanalzustand
      OW     W#16#10;
      T      DBB      8;     //Kanal Achse anfordern
m002:  NOP    0;
//optional end

      CALL "SpinCtrl" (//Z-Achse verfahren
      Start  := M      111.2,  //Start des Verfahrbewegung
      Stop   := FALSE,
      Funct  := B#16#5,        //Mode: Achsbetrieb
      Mode   := B#16#1,        //Verfahren: Inkrementell
      AxisNo := 3,             //Achsnnummer der zu verfahrenen
                                //Achse Z-Achse
      Pos    := -5.000000e+000, //Verfahrweg: minus 5 mm
      FRate  := 1.000000e+003,  //Vorschub: 1000 mm/min
      InPos  := M      113.0,  //Position erreicht
      Error  := M      113.1,  //Fehler aufgetreten
      State  := MB      114);  //Fehler Status

      AUF          "Axis3"; //Bremsentest Z-Achse
      U      M      113.0;  //Position erreicht
      O      M      113.1;  //Fehler aufgetreten

```

```

FP      M      113.2;
R       M      111.7;  //Start FC18

```

14.17.10 FB29: Diagnose Signalrekorder und Datentrigger

Funktion

Signalrekorder

Mit dem FB29 "Diagnose" besteht die Möglichkeit verschiedene Diagnosen am PLC-Anwenderprogramm durchzuführen. Ein Diagnosefall ist das Protokollieren von Signalzuständen und Signaländerungen. Bei diesem Diagnosefall wird die Funktionsnummer 1 dem Parameter "Func" zugeordnet. Bis zu 8 Signale der Parameter "Signal_1" bis "Signal_8" werden bei Änderung von einem dieser Signale in einem Ringpuffer aufgezeichnet. Zusätzlich werden noch die aktuellen Informationen der Parameter "Var1" als BYTE-Wert sowie "Var2", "Var3" als INTEGER-Werte im Ringpuffer mit abgelegt.

Weiterhin wird die Anzahl der vergangenen OB1-Zyklen als Zusatzinformation im Ringpuffer abgelegt. Hiermit ist eine graphische Auswertung von Signalen und Werten im OB1-Zyklusraster möglich.

Aufrufregel

Erster Aufruf des FB29 im OB1-Zyklus: Parameter "NewCycle" = 1

Alle weiteren Aufruf des FB29 im gleichen OB1-Zyklus: Parameter "NewCycle" = 0

Ringpuffer

Der vom Anwender zu definierende Ringpuffer muss eine ARRAY-Struktur wie im Quellcode angegeben haben. Die Anzahl der ARRAY-Elemente ist beliebig. Empfohlen wird eine Größe von 250 Elementen. Über den Parameter "ClearBuf" wird der Ringpuffer gelöscht und der Zeiger "BufAddr" auf den Anfang gesetzt. Der zugehörige Instanz-DB zum FB29 ist ein DB aus dem Anwenderbereich und ist dem FB Diagnose mit dem Parameter "BufDB" zu übergeben.

Datentrigger

Die Funktion "Datentrigger" soll eine Triggerung auf bestimmte Werte (auch Bits) an einer beliebigen zulässigen Speicherzelle ermöglichen. Hierbei wird die zu triggernde Zelle mit einer Bitmaske (Parameter "AndMask") vor dem Vergleich des Parameters "TestVal" in dem Diagnose-Baustein "verundet".

Hinweis

Die Funktion steht als Quelle im Quellcontainer der Grundprogramm-Bibliothek mit Namen "Diagnose.awl" zur Verfügung. In diesem Quellbaustein sind auch Instanz-DB und Ringpuffer-DB definiert. Weiterhin ist der Aufruf der Funktion dargestellt. Hierbei sind die DB-Nummern und auch der Aufruf anzupassen.

Deklaration der Funktion

```

FUNCTION_BLOCK FB29
VAR_INPUT
Func                : INT;
    Signal_1         : BOOL;
    Signal_2         : BOOL;
    Signal_3         : BOOL;
    Signal_4         : BOOL;
    Signal_5         : BOOL;
    Signal_6         : BOOL;
    Signal_7         : BOOL;
    Signal_8         : BOOL;
    NewCycle         : BOOL;
    Var1             : BYTE;
    Var2             : INT;
    Var3             : INT;
    BufDB            : INT;
    ClearBuf         : BOOL;
    DataAdr          : POINTER;
    TestVal          : WORD;
    AndMask          : WORD;
END_VAR
VAR_OUTPUT
    TestIsTrue       : BOOL;
END_VAR
VAR_IN_OUT
    BufAddr          : INT;
END_VAR

```

Struktur des Ringpuffers

```

TITLE =

                                                                    // Ring buffer-DB for FB29
VERSION : 1.0

STRUCT
    Feld : ARRAY [0 .. 249 ] OF      // can be any size of this struct
    STRUCT

        Cycle : INT;                // Delta cycle to last storage in buffer
        Signal_1 : BOOL;            // Signal names same as FB29
        Signal_2 : BOOL;
        Signal_3 : BOOL;

```

```

Signal_4 : BOOL;
Signal_5 : BOOL;
Signal_6 : BOOL;
Signal_7 : BOOL;
Signal_8 : BOOL;
Var1 : BYTE;
Var2 : WORD;
Var3 : WORD;
END_STRUCT;
END_STRUCT;
BEGIN
END_DATA_BLOCK

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Func:	E	INT	0, 1, 2	Funktion 0: Abschalten 1: Signalrekorder 2: Datentrigger
Parameter für Funktion 1				
Signal_1 ... Signal_8:	E	BOOL	0 (FALSE), 1 (TRUE)	Bit-Signale, die auf Änderung geprüft werden
NewCycle:	E	BOOL	0 (FALSE), 1 (TRUE)	Siehe oben Absatz "Funktion"
Var1:	E	BYTE		Zusatzwert
Var2:	E	INT		Zusatzwert
Var3:	E	INT		Zusatzwert
BufDB:	E	INT		Ringpuffer DB-Nr.
ClearBuf:	E	BOOL	0 (FALSE), 1 (TRUE)	Löschen des Ringpuffer DB mit Rücksetzen des Zeiger BufAddr
BufAddr:	E/A	INT		Zielbereich für gelesene Daten
Parameter für Funktion 2				
DataAdr:	E	POINTER		Zeiger auf zu prüfendes Wort
TestVal:	E	WORD		Vergleichswert
AndMask:	E	WORD		Siehe oben Absatz "Funktion"
TestIsTrue:	A	BOOL	0 (FALSE), 1 (TRUE)	Ergebnis des Vergleichs

Projektierungsschritte

- Funktion des Diagnosebausteins auswählen.
- Festlegung der geeigneten Daten für die Aufzeichnung als Signalrekorder bzw. der Datentriggerung.
- Geeignete Stelle(n) im Anwenderprogramm suchen für den Aufruf des FB Diagnose.

- Erzeugung eines Datenbausteins für den Ringpuffer, siehe Aufrufbeispiel.
- Aufruf des FB Diagnose im Anwenderprogramm mit Parametern.

Bei der Funktion 1 ist ein Löschen des Ringpuffers mit dem Parameter "ClearBuf" sinnvoll. Nach Beendigung der Aufzeichnung mit Funktion 1 wird der Ringpuffer-DB über STEP 7 mit der Funktion "Datenbaustein öffnen in Datenansicht" ausgelesen. Die Inhalte des Ringpuffer-DB können nun analysiert werden.

Aufrufbeispiel

```

FUNCTION FC99: VOID
TITLE =
VERSION : 0.0

BEGIN
NETWORK
TITLE = NETWORK

CALL FB29, DB80(
Func          := 1,
Signal_1      := M      100.0,
Signal_2      := M      100.1,
Signal_3      := M      100.2,
Signal_4      := M      100.3,
Signal_5      := M      10.4,
Signal_6      := M      100.5,
Signal_7      := M      100.6,
Signal_8      := M      100.7,
NewCycle      := TRUE,
Var1          := MB      100,
BufDB         := 81,
ClearBuf      := M      50.0);
END_FUNCTION

```

14.17.11 FC2: GP_HP - Grundprogramm, zyklischer Teil

Funktion

Die Bearbeitung der NC/PLC-Nahtstelle erfolgt durch das Grundprogramm im zyklischen Betrieb (OB1). Um die Laufzeit möglichst gering zu halten, werden nur die Steuer- und Statussignale zyklisch übertragen. Die Hilfs- und G-Befehlsübergabe wird nur auf Anforderung der NC bearbeitet.

Weiter werden die Daten für die Handradanwahl, Betriebsarten und Bediensignale von der Bedientafel (HMI) an die NC/PLC-Nahtstelle so übertragen, dass bei den Betriebsarten eine wahlweise Anwahl von der MCP oder von der HMI her möglich ist.

Die Übertragung der HMI-Signale an die NC/PLC-Nahtstelle kann durch Setzen des Parameters "MMCToIF" im FB1 (DB7) auf den Wert "FALSE" abgeschaltet werden.

Handrad-Anwahlsignale

Voraussetzung: FB1, Parameter "HWheelMMC == "TRUE"

Die Handrad-Anwahlsignale von HMI werden decodiert und in der Nahtstelle der entsprechende Maschinen- oder Geometrieachse des jeweiligen Handrades aktiviert.

Deklaration

```
FUNCTION FC2: VOID
// keine Parameter
```

Aufrufbeispiel

Das Grundprogramm muss - zeitlich gesehen - **vor** der Bearbeitung des Anwenderprogramms durchlaufen werden. Deshalb ist es als erstes im OB1 aufzurufen.

Das vorliegende Beispiel enthält die Standard-Deklarationen für den OB1 und die Aufrufe für das Grundprogramm (FC2), den Transfer der MCP-Signale (FC19) sowie die Erfassung von Fehler- und Betriebsmeldungen (FC10).

```
ORGANIZATION_BLOCK OB1
VAR_TEMP
  OB1_EV_CLASS :    BYTE;
  OB1_SCAN_1   :    BYTE;
  OB1_PRIORITY :    BYTE;
  OB1_OB_NUMBR :    BYTE;
  OB1_RESERVED_1 : BYTE;
  OB1_RESERVED_2 : BYTE;
  OB1_PREV_CYCLE : INT;
  OB1_MIN_CYCLE : INT;
  OB1_MAX_CYCLE : INT;
  OB1_DATE_TIME :  DATE_AND_TIME;
END_VAR
BEGIN
CALL FC2;                // Aufruf Grundprogramm als 1. FC
// HIER ANWENDERPROGRAMM EINFUEGEN
CALL FC19(                // MCP-Signale an Nahtstelle
  BAGNo :=                B#16#1,    // BAG Nr. 1
  ChanNo :=                B#16#1,    // Kanal Nr. 1
  SpindleIFNo :=          B#16#4,    // Spindel Interface Nummer = 4
                                   // (Nummer der zugehörigen
                                   // Maschinenachse)
  FeedHold :=             m22.0,     // Vorschub Halt Signal
                                   // selbsthaltend
  SpindleHold := db2.dbx151.0);      // Spindel Halt selbsthaltend
```

```

// in Meldungs-DB
CALL FC10 (
    ToUserIF := TRUE, // Fehler- und Betriebsmeldungen
                        // Signale aus DB2 an Nahtstelle
                        // übertragen
    Quit := E6.1);    // Quittierung der Fehlermeldungen
                        // über E6.1
END_ORGANIZATION_BLOCK

```

14.17.12 FC3: GP_PRAL - Grundprogramm, alarmgesteuerter Teil

Funktion

Im alarmgesteuerten Teil des Grundprogramms erfolgt die Bearbeitung der satzsynchronen Übergaben von NC an PLC mittels Hilfs- und G-Befehle. Bei den **Hilfsfunktionen** wird zwischen normalen und schnellen Hilfsfunktionen unterschieden.

Schnelle Hilfsfunktionen eines NC-Satzes werden zwischengespeichert und die Übernahme an die NC quittiert. Zu Beginn des nächsten OB1-Zyklus werden diese an die Anwendernahtstelle übergeben.

Schnelle Hilfsfunktionen, die sofort hintereinander programmiert sind, gehen nicht für das Anwenderprogramm verloren. Hierfür sorgt ein Mechanismus im Grundprogramm.

Normale Hilfsfunktionen werden erst dann an die NC quittiert, wenn an diesen eine Zyklusdauer angestanden hat. Damit hat der Anwender bei Bedarf die Möglichkeit, eine Einlesesperre an die NC zu geben.

Die **G-Befehle** werden sofort ausgewertet und an die Anwendernahtstelle übergeben.

NC-Prozessalarme

Bei Auslösung des Interrupts durch die NC (ist in jedem IPO-Takt möglich), wird ein Bit im Lokaldatum des OB40 ("GP_IRFromNCK") vom Grundprogramm nur dann gesetzt, wenn FB1 Parameter "UserIR": = TRUE ist. Bei anderen Ereignissen (Prozessalarme durch die Peripherie), ist dieses Datum nicht gesetzt. Mit dieser Information kann in die zugehörige Interruptroutine im Anwenderprogramm verzweigt werden, um die notwendigen Aktionen einzuleiten.

Einer schnellen auftragsgesteuerten Bearbeitung vom Anwenderprogramm für die Maschine, stehen folgende NC-Funktionen in der Interruptbearbeitung (OB40-Programmteil) für das PLC-Anwenderprogramm zur Verfügung:

- Ausgewählte **Hilfsfunktionen**
- **Werkzeugwechsel-Funktion** bei der Option Werkzeugverwaltung
- **Position erreicht** bei den Positionierachsen, Teilungsachsen und Spindeln bei Aktivierung über PLC

Diese oben aufgeführten Funktionen müssen durch das Anwenderprogramm im OB40 ausgewertet werden, um schnelle Reaktionen an der Maschine einzuleiten. Hier kann z. B. bei der Programmierung eines T-Befehls an einer Drehmaschine das Revolver-Schaltwerk aktiviert werden.

Weitere Einzelheiten zur Programmierung von Prozessalarmen, Zeitverzögerung, Unterbrechbarkeit, usw. sind in der entsprechenden SIMATIC-Dokumentation nachzulesen.

Hilfsfunktionen

Generell gilt, dass schnelle oder quittierende Hilfsfunktionen unabhängig von einer Zuordnung mit oder ohne Interruptsteuerung bearbeitet werden.

Durch Grundprogrammparameter im FB1 kann ausgewählt werden, welche Hilfsfunktionen (T, H, DL) vom Anwenderprogramm nur interruptgesteuert bearbeitet werden sollen.

Die nicht über Interrupt zugeordneten Funktionen werden wie bisher erst durch das zyklische Grundprogramm zur Verfügung gestellt. Hierbei stehen die Änderungssignale der Funktionen einen PLC-Zyklus an.

Wenn die Auswahl für die Hilfsfunktionsgruppen (T, H, DL) mit Interruptsteuerung getroffen ist, dann kann für die ausgewählten Funktionen nur noch eine Interruptbearbeitung durch das Anwenderprogramm erfolgen.

Für das Anwenderprogramm wird kanalorientiert ein Bit in dem Lokaldatum "GP_AuxFunction" gesetzt (wenn "GP_AuxFunction[1]" gesetzt ist, dann steht für den 1. Kanal eine Hilfsfunktion bereit).

Im zugehörigen Kanal-DB stehen Änderungssignal und Funktionswert für den Anwender zur Verfügung. Das Änderungssignal dieser interruptgesteuerten Funktion wird im zyklischen Grundprogrammteil nach Ablauf von mindestens einem vollem OB1-Zyklus (max. ca. zwei OB1-Zyklen) wieder auf Null zurückgesetzt.

Werkzeugwechsel

Bei der Option Werkzeugverwaltung werden der Werkzeugwechsel-Befehl für Revolver und der Werkzeugwechsel in die Spindel durch einen Interrupt unterstützt. Hierzu wird das Lokaldatenbit "GP_TM" im OB40 gesetzt. Damit kann das PLC-Anwenderprogramm den DB der Werkzeugverwaltung (DB72 bzw. DB73) auf die Werkzeugwechsselfunktion abprüfen und den Wechsellvorgang anstoßen.

Position erreicht

In der Bitstruktur "GP_InPosition" der Lokaldaten des OB40 ist maschinenachsspezifisch (jedes Bit entspricht einer Achse/Spindel, z. B. GP_InPosition[5] entspricht der 5. Achse) aufgebaut.

Wenn eine Funktion FC18 (Spindelsteuerung, Positionierachse, Teilungsachse) für eine Achse oder Spindel aktiviert wurde, kann das zugehörige "GP_InPosition"-Bit eine unverzögerte Auswertung des Signals "InPos" der oben aufgeführten FCs erreicht werden. Hierdurch können z. B. Klemmungen für eine Teilungsachse unverzögert aktiviert werden.

Deklaration

```
FUNCTION FC3: VOID  
//keine Parameter
```

Aufrufbeispiel

Das Grundprogramm muss - zeitlich gesehen - **vor** der Bearbeitung weiterer alarmgesteuerter Anwenderprogramme durchlaufen werden. Deshalb ist es als erstes im OB40 aufzurufen.

Das vorliegende Beispiel enthält die Standard-Deklarationen für den OB40 und den Aufruf für das Grundprogramm.

```

ORGANIZATION_BLOCK OB40
VAR_TEMP
    OB40_EV_CLASS :      BYTE;
    OB40_STRT_INF :      BYTE;
    OB40_PRIORITY :      BYTE;
    OB40_OB_NUMBR :      BYTE;
    OB40_RESERVED_1 :    BYTE;
    OB40_MDL_ID :         BYTE;
    OB40_MDL_ADDR :       INT;
    OB40_POINT_ADDR :     DWORD;
    OB40_DATE_TIME :      DATE_AND_TIME;

//Belegung durch das Grundprogramm
GP_IRFromNCK : BOOL;           //Interrupt durch NC für Anwender
GP_TM : BOOL;                  //Werkzeugverwaltung
GP_InPosition : ARRAY [1..3] OF BOOL; //Achsorientiert für Positionier.-,
                                     //Teilungsachsen, Spindeln
GP_AuxFunction : ARRAY [1..10] OF BOOL; //Kanalorientiert für Hilfsfunktionen
GP_FMBlock : ARRAY [1..10] OF BOOL;    //derzeit nicht genutzt
//ab hier dürfen weitere Lokaldaten des Anwenders definiert werden
END_VAR
BEGIN
    CALL FC3;
    //HIER ANWENDERPROGRAMM EINFUEGEN
END_ORGANIZATION_BLOCK

```

14.17.13 FC5: GP_DIAG - Grundprogramm, Diagnosealarm und Baugruppenausfall**Funktion**

Der Baustein FC5 "GP_DIAG" dient zur Erfassung von Baugruppenstörungen und -ausfällen. Über den Parameter "PlcStop" kann ein PLC-Stop ausgelöst werden. Der PLC-Stop wird nur bei kommenden Ereignissen ausgelöst. Ausgenommen sind die am FB1 parametrisierten PROFIBUS (DP1) angeschlossenen MCPs.

Deklaration

```

FUNCTION FC5: VOID
    VAR_INPUT
        PlcStop: BOOL :=TRUE;
    END_VAR

```

Aufrufbeispiel

Das Grundprogramm sollte nach der Bearbeitung der Anwenderprogramme durchlaufen werden. Das ist empfehlenswert, da durch den FC5 ein PLC-Stop ausgelöst werden kann.

Das Beispiel enthält die Standard-Deklaration für OB82 und OB86 und den Aufruf des FC5.

```

ORGANIZATION_BLOCK OB82
VAR_TEMP
    OB82_EV_CLASS : BYTE ;
    OB82_FLT_ID : BYTE ;
    OB82_PRIORITY : BYTE ;
    OB82_OB_NUMBR : BYTE ;
    OB82_RESERVED_1 : BYTE ;
    OB82_IO_FLAG : BYTE ;
    OB82_MDL_ADDR : INT ;
    OB82_MDL_DEFECT : BOOL ;
    OB82_INT_FAULT : BOOL ;
    OB82_EXT_FAULT : BOOL ;
    OB82_PNT_INFO : BOOL ;
    OB82_EXT_VOLTAGE : BOOL ;
    OB82_FLD_CONNCTR : BOOL ;
    OB82_NO_CONFIG : BOOL ;
    OB82_CONFIG_ERR : BOOL ;
    OB82_MDL_TYPE : BYTE ;
    OB82_SUB_NDL_ERR : BOOL ;
    OB82_COMM_FAULT : BOOL ;
    OB82_MDL_STOP : BOOL ;
    OB82_WTCH_DOG_FLT : BOOL ;
    OB82_INT_PS_FLT : BOOL ;
    OB82_PRIM_BATT_FLT : BOOL ;
    OB82_BCKUP_BATT_FLT : BOOL ;
    OB82_RESERVED_2 : BOOL ;
    OB82_RACK_FLT : BOOL ;
    OB82_PROC_FLT : BOOL ;
    OB82_EPROM_FLT : BOOL ;
    OB82_RAM_FLT : BOOL ;
    OB82_ADU_FLT : BOOL ;
    OB82_FUSE_FLT : BOOL ;
    OB82_HW_INTR_FLT : BOOL ;

```

```

        OB82_RESERVED_3 : BOOL ;
        OB82_DATE_TIME : DATE_AND_TIME ;
    END_VAR
    BEGIN
        CALL FC5
            (PlcStop := FALSE) ;
    END_ORGANIZATION_BLOCK

    ORGANIZATION_BLOCK OB86
    VAR_TEMP
        OB86_EV_CLASS : BYTE ;
        OB86_FLT_ID : BYTE ;
        OB86_PRIORITY : BYTE ;
        OB86_OB_NUMBR : BYTE ;
        OB86_RESERVED_1 : BYTE ;
        OB86_RESERVED_2 : BYTE ;
        OB86_MDL_ADDR : WORD ;
        OB86_RACKS_FLTD : ARRAY [0 .. 31 ] OF BOOL ;
        OB86_DATE_TIME : DATE_AND_TIME ;
    END_VAR
    BEGIN
        CALL FC5
            (PlcStop := TRUE) ;
    END_ORGANIZATION_BLOCK

```

14.17.14 FC6: TM_TRANS2 - Transfer-Baustein für Werkzeugverwaltung und Multitool

Funktion

Der Baustein FC6 "TM_TRANS2" wird bei Positionsänderungen der Werkzeuge, Statusänderungen und Multitool eingesetzt.

Der Baustein FC6 hat die gleiche Funktionalität wie der Baustein FC8 und zusätzlich die **Multitool**-Funktionalität.

Die Beschreibung des FC6 beinhaltet nur die **Multitool**-Funktionalität.

Die Funktionalität des FC8 ist beschrieben in "FC8: TM_TRANS - Transfer-Baustein für Werkzeugverwaltung (Seite 1071)".

Deklaration der Funktion

```

FUNCTION FC6: VOID
VAR_INPUT
    Start:          BOOL;
    TaskIdent:      BYTE;

```

```

    TaskIdentNo:    BYTE;
    NewToolMag:     INT;
    NewToolLoc:     INT;
    OldToolMag:     INT;
    OldToolLoc:     INT;
    Status:         INT;
    MtoolPlaceNum:  INT;
END_VAR
VAR_OUTPUT
    Ready:         BOOL;
    Error:         INT;
END_VAR
BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Start:	E	BOOL	0 (FALSE), 1 (TRUE)	siehe Bausteinbeschreibung FC8
TaskIdent:	E	BYTE		siehe Bausteinbeschreibung FC8
TaskIdentNo:	E	BYTE		siehe Bausteinbeschreibung FC8
NewToolMag:	E	INT		siehe Bausteinbeschreibung FC8
NewToolLoc:	E	INT		siehe Bausteinbeschreibung FC8
OldToolMag:	E	INT		siehe Bausteinbeschreibung FC8
OldToolLoc:	E	INT		siehe Bausteinbeschreibung FC8
Status:	E	INT		siehe Bausteinbeschreibung FC8
MtoolPlaceNum:	E	INT		Multitoolplatz-Nr.
Ready:	A	BOOL	0 (FALSE), 1 (TRUE)	siehe Bausteinbeschreibung FC8
Error:	A	INT		siehe Bausteinbeschreibung FC8

14.17.15 FC7: TM_REV - Transfer-Baustein für Werkzeugwechsel mit Revolver

Funktion

Der Anwender ruft nach erfolgtem Wechsel eines Revolvers den Baustein FC7 "TM_REV" auf. Hierzu ist in dem Parameter "ChgdRevNo" die Revolvernummer entsprechend der Schnittstellen-Nummer im DB73 mitzuteilen. Mit Aufruf dieses Bausteins wird das zugehörige "Schnittstelle aktiv"-Bit im Datenbaustein DB73.DBW0 vom FC7 zurückgesetzt, nachdem der Parameter "Ready" == TRUE zurückgemeldet wird.

Auftrag korrekt ausgeführt

Wurde der Auftrag korrekt ausgeführt, ist "Ready" == 1. Vom Anwender ist dann Parameter "Start" = 0 zu setzen bzw. der FC7 nicht mehr aufrufen.

Auftrag mit Fehler ausgeführt

Wurde der Auftrag mit Fehler ausgeführt, ist Parameter "Ready" == 0 und Parameter "Error" == 1. Der Auftrag ist im nächsten PLC-Zyklus zu wiederholen. Da der Parameter "Start" keine positive Flanke für einen Folgeauftrag benötigt, bleibt "Start" = 1, da der Auftrag noch nicht abgeschlossen wurde. Siehe unten "Aufrufbeispiel" und "Impulsdiagramm".

Randbedingungen

- Der Baustein FC7 darf nur mit Parameter "Start" = 1 gestartet werden, wenn für diesen Transfer eine Aktivierung der zugehörigen Schnittstelle (DB73.DBW0) durch die Werkzeugverwaltung vorliegt.
- Ein Abbruch eines Transfers, z. B. durch Kanal-Reset, ist nicht zulässig.
- Parameter "Start" = 1, bis Parameter "Ready" == 1 oder "Error" == 1

Literatur

- Ausführliche Informationen zur Werkzeugverwaltung siehe Funktionshandbuch Werkzeugverwaltung.
- PI-Dienste für die Werkzeugverwaltung siehe:
 - FB4: PI_SERV - PI-Dienst anfordern (Seite 1005)
 - FC8: TM_TRANS - Transfer-Baustein für Werkzeugverwaltung (Seite 1071)
 - FC22: TM_DIR - Richtungs Auswahl für Werkzeugverwaltung (Seite 1116)

Revolverschalten von Hand

Wird der Revolver über Handbedienung gedreht, ist damit weder ein Werkzeugwechsel noch eine Korrekturanwahl verbunden. Der erste Schritt ist das Auswechseln des Werkzeugs auf dem Werkzeughalter zurück auf seinen Revolverplatz. Dazu ist mit dem FC8 (alternativ FC6) ein asynchroner Transfer durchzuführen. Nachfolgend die zugehörige Parametrierung:

```
TaskIdent = 4
TaskIdentNo = Kanal-Nr.
NewToolMag = Magazinnummer des Revolvers
NewToolLoc = Ursprünglicher Platz des Werkzeugs
OldToolMag = Magazinnr. Zwischenspeicher (Spindel) = 9998
OldToolLoc = Zwischenspeichernummer der Spindel
Status = 1
```

Wird jetzt der Revolver auf eine beliebige Position gedreht, auf der ein Werkzeug sitzt, muss dieses Werkzeug aktiviert werden. Im einfachsten Fall geschieht das durch die neue T-Programmierung im Teileprogramm. Soll das jedoch z. B. mit Ende des Revolvertakts vom PLC-Anwenderprogramm aus erfolgen, muss dazu ein ASUP gestartet werden. Dem ASUP muss die aktuelle Revolverposition übergeben werden. Damit wird im ASUP das Werkzeug auf diesem Platz ermittelt und dieses ausgewählt (siehe Jobshop-Beispiel auf der Toolbox).

Deklaration der Funktion

```

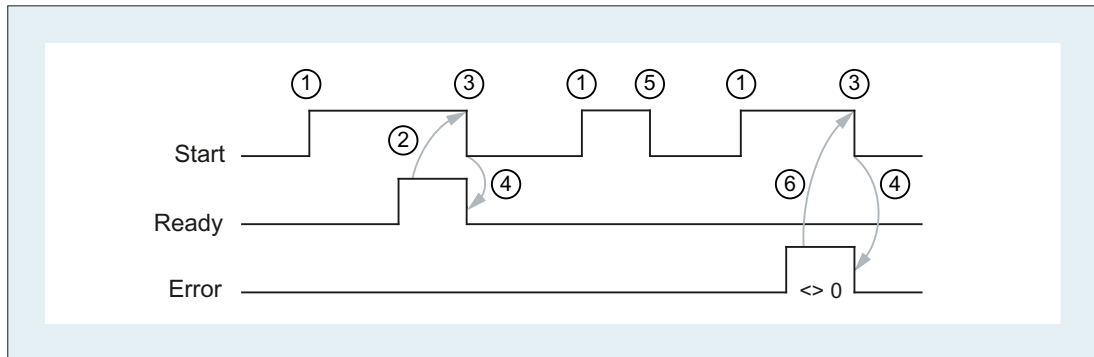
FUNCTION FC7:          VOID
//NAME :TM_REV
VAR_INPUT
    Start :          BOOL;
    ChgdRevNo :      BYTE;
END_VAR
VAR_OUTPUT
    Ready :          BOOL;
    Error :          INT;
END_VAR
BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Start:	E	BOOL	0 (FALSE), 1 (TRUE)	1 = Transfer starten
ChgdRevNo:	E	BYTE	1, 2, 3, ...	Nummer der Revolver-Schnittstelle
Ready:	A	BOOL	0 (FALSE), 1 (TRUE)	1 = Transfer abgeschlossen
Error:	A	INT	0, 1, 2, 3	Fehler-Rückmeldung 0: kein Fehler aufgetreten 1: kein Revolver vorhanden 2: unzulässige Revolvernummer in Parameter "ChgdRevNo" 3: unzulässiger Auftrag ("Signal "Schnittstelle (SS) aktiv" des angewählten Revolvers = "FALSE")

Impulsdiagramm



- ① Anwender: Anforderung setzen, Start = 0 → 1
- ② FB4: PI-Dienst erfolgreich beendet, Ready = 1
Anwender: Anforderung zurücksetzen, IF Ready == 1 THEN Start = 0
- ③ Anwender: IF Ready == 1 THEN Anforderung zurücksetzen: 1 → 0
- ④ FB4: Auftragsbestätigung zurücksetzen, Ready = 0
- ⑤ Anwender: IF Ready == 0 UND Error == 0 THEN Anforderung zurücksetzen Start = 1 → 0 **nicht zulässig**
- ⑥ FB4: PI-Dienst mit Fehler beendet, Error = 1
Anwender: Anforderung zurücksetzen, IF Ready == 1 OR Error == 1 THEN Start = 0, eventuell weitere Fehlerbehandlung

Aufrufbeispiel

```
CALL      // Werkzeugverwaltung: Transfer-Baustein für Revolver
FC7(
    Start :=          m 20.5,          // Start := "1 " => Anstoß des
                                   Transfers
    ChgdRevNo :=      DB61.DBB1,
    Ready :=         m 20.6,
    Error :=         DB61.DBW12
);
u      m 20.6;          // Ready abfragen
r      m 20.5;          // Start zurücksetzen
spb    m001;            // Springe, wenn alles in Ordnung
l      db61.dbw 12;     // Fehler-Information
ow      w#16#0;         // Fehler auswerten
spn     fehl;           // Springe auf Fehlerbehandlung, wenn
                                   <> 0
m001 :          // Beginn des weiteren Programms
fehl :
r      m 20.5;          // Start zurücksetzen, wenn Fehler
                                   vorliegt
```

14.17.16 FC8: TM_TRANS - Transfer-Baustein für Werkzeugverwaltung

Funktion

Der Anwender ruft diesen Baustein FC TM_TRANS bei Positionsänderungen der Werkzeuge oder bei Statusänderungen des Transfervorgangs auf. Mit dem Parameter "TaskIdent" wird für den Baustein FC8 der Transferauftrag an der Werkzeugverwaltungs-Schnittstelle angegeben:

- Für Be-/Entladestellen
- Für Spindel-Wechselstellen
- Für Revolver-Wechselstellen als Transferkennung
- Asynchroner Transfer
- Asynchroner Transfer mit Platzreservierung

Die Schnittstellen-Nummer wird im Parameter "TaskIdentNo" mitgeteilt.

Beispiel für Beladestelle 5:

Parameter "TaskIdent":= 1 und "TaskIdentNo":= 5.

Weiterhin werden für diesen Transfer die **aktuellen** Werkzeugpositionen und Statusinformationen, siehe Status-Auflistung des Parameters "Status" im nachfolgenden Text, übermittelt.

Hinweis

Der FC8 teilt dem NC die aktuellen Positionen vom alten Werkzeug mit.

Dem NC ist bekannt, wo sich das alte und das neue Werkzeug bis zur Positionsänderung befunden haben.

Bei einem Transfer ohne ein sogenanntes "altes Werkzeug" (z. B. beim Beladen), wird den Parametern "OldToolMag", "OldToolLoc" der Wert 0 zugeteilt.

Der Transfer-Baustein darf nur mit den Parameter "Start" = TRUE gestartet werden, wenn für diesen Transfer eine Aktivierung der zugehörigen Schnittstelle (DB71, DB72, DB73 im Wort 0) durch die Werkzeugverwaltung vorliegt.

Wenn dieser Auftrag korrekt ausgeführt wurde, enthält der Ausgangsparameter "Ready" den Wert TRUE.

Damit muss der Anwender den Parameter "Start" = FALSE setzen bzw. den Baustein nicht mehr aufrufen.

Falls der Parameter "Ready" = FALSE ist, muss der Fehlercode im Parameter "Error" interpretiert werden, siehe Beispiel FC8-Aufruf und Impulsdiagramm.

Wenn der Fehlercode = 0 ist, dann ist dieser Auftrag im nächsten PLC-Zyklus zu wiederholen ("Start" bleibt TRUE). Es bedeutet, dass der Transferauftrag noch nicht beendet ist.

Wenn der Parameter "Status" einen Wert kleiner 100 durch den Anwender erhält, wird die zugehörige Schnittstelle im Datenbaustein DB71 bzw. DB72 oder DB73, Wort 0 deaktiviert (Vorgang beendet). Das entsprechende Bit für die Schnittstelle wird auf 0 gesetzt durch den FC8.

Der Parameter "Start" benötigt keine Flanke für einen Folgeauftrag. Dies bedeutet, dass sofort bei Erhalten von "Ready" = TRUE neue Parameter zugeordnet werden können mit "Start" = TRUE.

Asynchroner Transfer

Für eine eigenständige Mitteilung der PLC von den Positionsänderungen eines Werkzeugs zur Werkzeugverwaltung (z. B. Spannungsunterbrechung bei einem aktiven Kommando oder selbständige Positionsveränderungen durch PLC) wird der FC8 mit "TaskIdent" = 4 oder 5 aufgerufen. Zu diesem Aufruf muss keine Schnittstellen-Aktivierung durch die Werkzeugverwaltung vorliegen.

Bei dem Parameter "TaskIdent" = 5 wird zusätzlich zur Positionsveränderung eine Platzreservierung von der Werkzeugverwaltung vorgenommen. Diese Platzreservierung erfolgt aber nur, wenn das Werkzeug aus einem realen Magazin in einen Zwischenspeicher transportiert wurde.

Im Parameter "TaskIdentNo" ist ein zugehöriger NC-Kanal zu parametrieren.

In den Parametern "OldToolMag" und "OldToolLoc" wird die bisherige Position des Werkzeugs angegeben. In den Parametern "NewToolMag", "NewToolLoc" wird die aktuelle Position dieses Werkzeugs angegeben. Hierbei ist "Status" = 1 anzugeben.

Bei "Status" = 5 verbleibt das angegebene Werkzeug auf dem Platz "OldToolMag", "OldToolLoc". Dieser Platz muss ein Zwischenspeicher sein (z. B. Spindel). In den Parametern "NewToolMag", "NewToolLoc" ist das reale Magazin mit Platz anzugeben, wobei sich der Platz an der Position des Zwischenspeichers befindet. Dieses Verfahren ist immer dann anzuwenden, wenn der Werkzeugverwaltung mitgeteilt werden soll, wo sich ein bestimmter Platz des Magazins befindet. Dieses Verfahren dient zum Abgleich bei Suchstrategien.

Randbedingungen

- Ein Abbruch eines Transfers, z. B. durch Kanal-Reset, ist nicht zulässig.
- Parameter "Start" = 1, bis Parameter "Ready" == 1 oder "Error" == 1

Literatur

- Ausführliche Informationen zur Werkzeugverwaltung finden sich in Funktionshandbuch Werkzeugverwaltung enthalten.
- PI-Dienste für die Werkzeugverwaltung
 - FB4: PI_SERV - PI-Dienst anfordern (Seite 1005)
 - FC7: TM_REV - Transfer-Baustein für Werkzeugwechsel mit Revolver (Seite 1067)
 - FC22: TM_DIR - Richtungsauswahl für Werkzeugverwaltung (Seite 1116)

Deklaration der Funktion

```
FUNCTION FC8: VOID
// NAME :TM_TRANS
VAR_INPUT
    Start :          BOOL;
    TaskIdent :      BYTE;
```

```

    TaskIdentNo :      BYTE;
    NewToolMag :      INT;
    NewToolLoc :      INT;
    OldToolMag :      INT;
    OldToolLoc :      INT;
    Status :          INT;

END_VAR
VAR_OUTPUT
    Ready :          BOOL;
    Error :          INT;

END_VAR
BEGIN
END_FUNCTION

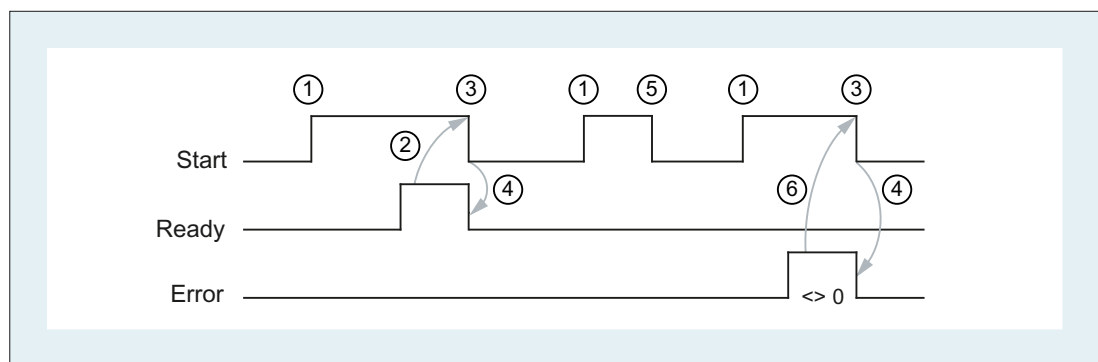
```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Start:	E	BOOL	0 (FALSE), 1 (TRUE)	1: Transfer starten
TaskIdent:	E	BYTE	1, 2, 3, 4, 5	Kennung der Schnittstelle bzw. des Auftrags 1: Be-/Entladestelle 2: Spindel-Wechselstelle 3: Revolver-Wechselstelle 4: asynchroner Transfer 5: asynchroner Transfer mit Platzreservierung
TaskIdentNo:	E	BYTE	1, 2, 3 ... 10	Nummer der zugehörigen Schnittstelle bzw. Kanalnummer.
NewToolMag:	E	INT	-1, 0, 1, 2 ...	aktuelle Magazinnummer des neuen Werkzeugs -1: Werkzeug verbleibt auf seinem bisherigen Platz. NewToolLoc = beliebiger Wert Nur bei TaskIdent = 2 zulässig.
NewToolLoc:	E	INT	0, 1, 2 ... max. Platznr.	aktuelle Platznummer des neuen Werkzeugs
OldToolMag:	E	INT	-1, 0 ...	aktuelle Magazinnummer des auszuwechselnden Werkzeugs -1: das Werkzeug verbleibt auf seinem bisherigen Platz. "OldToolLoc" = <beliebiger Wert>. Nur bei "TaskIdent" = 2 zulässig.
OldToolLoc:	E	INT	max. Platznummer	aktuelle Platznummer des auszuwechselnden Werkzeugs
Status:	E	INT	1, 2, 3 ... 7, 103, 104, 105	Status-Information des Transfervorgangs

Signal	Art	Typ	Wertebereich	Bedeutung
Ready:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Transfer abgeschlossen
Error:	A	INT	0 ... 65535	Fehler-Rückmeldung 0: kein Fehler aufgetreten 1: unbekannte "TaskIdent" 2: unbekannte "TaskIdentNo" 3: unzulässiger Auftrag, (Signal "Schnittstelle (SS) aktiv" des angewählten Revolvers == 0) anderer Wert: Die Zahl entspricht der Fehlermeldung der WZV der NC, die durch diesen Transfer verursacht wurde.

Impulsdiagramm



- ① Funktionsanstoß durch positive Flanke
- ② Positive Quittung: WZV Transfer wurde ausgeführt
- ③ Rücksetzen vom Funktionsanstoß nach Erhalt der Quittung
- ④ Signalwechsel durch FC
- ⑤ Dieser Signalverlauf ist unzulässig. Der Auftrag ist generell zu beenden, da die neuen Werkzeug-Positionen der Werkzeugverwaltung im NC mitzuteilen sind.
- ⑥ Negative Quittung: Fehler aufgetreten, Fehlercode im Ausgangsparameter Error

Status

Status	Beschreibung
1	<p>Der WZV-Auftrag wurde beendet</p> <p>Die Parameter "NewToolMag", "NewToolLoc", "OldToolMag", "OldToolLoc" des FC8-Bausteins sind auf die tatsächlichen Positionen der beteiligten Werkzeuge zu parametrieren. Außer beim Wechsel vorbereiten sind dies die in der Regel die angegebenen Werkzeug-Zielpositionen der zugehörigen WZV-Schnittstelle, siehe auch "Erläuterungen der Formalparameter".</p> <ol style="list-style-type: none"> 1. Beim Be-/Ent-/Umladen ist das Werkzeug auf der geforderten Zieladresse angekommen. Wenn das Bit in der Nahtstelle im DB71.DBX(n+0).3 "Positionieren zur Beladestelle" gesetzt ist, dann darf für den Abschluss der Funktion nicht der Status 1 verwendet werden. Für einen korrekten Abschluss ist der Status 5 zu verwenden. 2. Beim "Wechsel vorbereiten" ist die Bereitstellung des neuen Werkzeugs erfolgt. Dieses kann z. B. auf einem Zwischenspeicher (Greifer) abgelegt sein. Gegebenenfalls wurde auch das Ziel (Magazin, Platz) des alten Werkzeugs zur Wechselposition positioniert, nachdem das neue Werkzeug in einem Zwischenspeicher abgelegt wurde. Das alte Werkzeug verbleibt aber noch in der Spindel. Somit ist die Vorbereitung zum Wechseln abgeschlossen. Nach dieser Quittierung kann das Kommando "Wechseln" empfangen werden. Die Positionen in den Parametern "NewToolMag", "NewToolLoc", "OldToolMag", "OldToolLoc" entsprechen den aktuellen Positionen der Werkzeuge. 3. Beim "Wechseln" (in Spindel bzw. beim Revolver) sind die in der Schnittstelle angesprochenen Werkzeuge auf den geforderten Zieladressen angekommen. Der Vorgang des Werkzeugwechsels ist damit abgeschlossen.
2	<p>Das "neue" Werkzeug kann nicht bereitgestellt werden</p> <p>Dieser Status ist nur bei dem Befehl "Wechseln" zulässig. Wenn dieser Status angewendet wird, soll von der PLC ein Wechsel mit dem vorgeschlagenen Werkzeug verhindert werden. Durch die WZV im NC erfolgt das Sperren dieses vorgeschlagenen (neuen) Werkzeugs. Anschließendes erfolgt ein neues Kommando der Werkzeugverwaltung mit einem Duplo-Werkzeug. Die Positionen in den Parametern "NewToolMag", "NewToolLoc", "OldToolMag", "OldToolLoc" entsprechen den ursprünglichen Positionen der Werkzeuge.</p>
3	<p>Ein Fehler ist aufgetreten</p> <p>Es darf keine Änderung der Werkzeugpositionen erfolgt sein. Wenn sich die Magazinpositionen der Werkzeuge zwischenzeitlich verändert hatten, ist dieses z. B. vorher mit dem Status = 105 über den FC8-Transfer-Baustein mitzuteilen. Nur dann werden die Positionen von der Werkzeugverwaltung berücksichtigt.</p>
4	<p>Das "alte" Werkzeug sollte besser auf der in den Parametern "OldToolMag", "OldToolLoc" angegebenen Magazinposition untergebracht werden</p> <p>Dieser Status ist nur bei der Werkzeugwechsel-Vorbereitung (Wechsel in die Spindel) zulässig. Der für das "alte" Werkzeug angegebene Magazinplatz muss frei sein. Nachdem dieser Status an die Werkzeugverwaltung im NC gegeben wurde, wird ein neues Vorbereitekommando erzeugt (Status_4 = Endequittung) und an den DB72 ausgegeben, bei dem die angeforderte Magazinposition des Alt-Werkzeugs berücksichtigt wird.</p> <p>Es erfolgt explizit keine neue Werkzeugsuche, die Positionen für „NewToolMag“ und „NewToolLoc“ werden aus dem ursprünglichen Vorbereitekommando übernommen. Dieses erfolgt aber nur dann, wenn diese Position frei ist. Die Parameter "NewToolMag" und "NewToolLoc" werden nicht berücksichtigt.</p>

Status	Beschreibung
5	<p>Der Vorgang ist beendet worden</p> <p>Das "neue" Werkzeug ist an der in den Parametern "NewToolMag", "NewToolLoc" angegebenen Position. Hierbei befindet sich das angegebene Werkzeug nicht wirklich in dieser Position, sondern ist weiterhin im gleichen Magazinplatz. Dieser Magazinplatz wurde jedoch zu dieser Position (z. B. Wechselstelle) hin positioniert. Dieser Status darf nur bei Revolver, Ketten- und Scheibenmagazinen verwendet werden. Der Status dient der Werkzeugverwaltung zum Abgleich der aktuellen Position eines Magazins und zur Verbesserung der Suchstrategie für nachfolgende Befehle. Dieser Status ist nur beim Beladen, Entladen, Umladen und bei der Wechsel-Vorbereitung erlaubt. Die Parameter "OldToolMag" und "OldToolLoc" müssen mit den Daten eines Zwischenspeichers parametrisiert sein.</p> <ul style="list-style-type: none"> • Beladen, Umladen: Beim Beladen oder Umladen ist im NC schon ein Platz für das Werkzeug reserviert worden. Im Anschluss muss der Maschinen-Bediener das Werkzeug auf dem Zielplatz einsetzen. Achtung: Nach einem erneuten Einschalten der Steuerung ist die Platzreservierung aufgehoben. • Wechsel-Vorbereitung: Die noch ausstehenden Werkzeugbewegungen erfolgen erst mit dem Werkzeug-Wechsel. • Positionieren zur Beladestelle: Wenn das Bit in der Nahtstelle im DB71.DBX(n+0).3 "Positionieren zur Beladestelle" gesetzt ist, dann darf für den Abschluss der Funktion nur der Status 5 (nicht Status 1) verwendet werden.
6	<p>Der WZV-Auftrag ist beendet worden</p> <p>Dieser Status hat die gleiche Funktion wie der Status 1, aber zusätzlich wird eine Reservierung des Quellplatzes vorgenommen. Dieser Status ist nur beim Umladen erlaubt. Das Kommando wird beendet und der Quellplatz des Werkzeugs wird reserviert, falls der Zielplatz in einem Zwischenspeichermagazin liegt.</p>
7	<p>Wiederholung des "Werkzeug Vorbereiten" Befehls anstossen</p> <p>Dieser Status ist nur bei dem "Wechsel vorbereiten"-Befehl zulässig. Der Status soll angewendet werden, wenn das "neue" Werkzeug seine Position verändert hat (z. B. über einen asynchronen Befehl des "neuen" Werkzeugs). Nach "Ready = 1" vom FC8 erfolgt eine automatische Wiederholung des "Wechsel vorbereiten" mit dem gleichen Werkzeug. Für die automatische Wiederholung wird eine neue Werkzeugsuche durchgeführt. Die Positionen in den Parametern "NewToolMag", "NewToolLoc", "OldToolMag", "OldToolLoc" müssen den ursprünglichen Positionen der Werkzeuge entsprechen.</p>
103	<p>Das "neue" Werkzeug kann eingesetzt werden</p> <p>Dieser Status ist nur bei der Wechsel-Vorbereitung zulässig, wenn die PLC das neue Werkzeug ablehnen darf (z. B. bei MD20310 \$MC_TOOL_MANAGEMENT_MASK, Bit 4 = 1 für Möglichkeit, geänderte Parameter vom PLC noch einmal anzufordern). Die Positionen der Werkzeuge sind unverändert geblieben. Dieser Status ist damit notwendig, wenn der Vorlauf im NC ohne einen unnötigen Stopp der Bearbeitung fortgesetzt werden soll.</p>

Status	Beschreibung
104	<p>Das "neue" Werkzeug ist an der in den Parametern "NewToolMag", "NewToolLoc" angegebenen Position</p> <p>Dieser Status ist nur zulässig, wenn das Werkzeug sich noch im Magazin auf dem gleichen Platz befindet. Das "alte" Werkzeug ist auf der in den Parametern "OldToolMag", "OldToolLoc" angegebenen Position (Zwischenspeicher). Hierbei befindet sich das neue Werkzeug nicht wirklich in dieser Position, sondern ist weiterhin im gleichen Magazinplatz. Dieser Magazinplatz wurde jedoch zu dieser Position (z. B. Wechselstelle) hin positioniert. Dieser Status darf nur bei Revolver, Ketten- und Scheibenmagazinen beim "Werkzeug-Wechsel Vorbereiten" verwendet werden. Der Status dient der Werkzeugverwaltung zum Abgleich der aktuellen Position eines Magazins, zur Verbesserung der Suchstrategie für nachfolgende Befehle.</p>
105	<p>Der angegebene Zwischenplatz ist von allen beteiligten Werkzeugen erreicht</p> <p>Standardfall, wenn der Vorgang noch nicht beendet ist.</p> <p>Die Werkzeuge sind in den angegebenen Werkzeug-Positionen (Parameter "NewToolMag", "NewToolLoc", "OldToolMag", "OldToolLoc").</p>

Status-Definition

Generell gilt für den Quittungsstatus, dass die Statusinformationen 1 bis 7 zu einer Beendigung des Kommandos führen. Wenn eine dieser Statusinformationen dem FC8 mitgeteilt wird, wird das "Schnittstelle Aktiv-Bit" der im FC8 angegebenen Schnittstelle auf "0" zurückgesetzt (siehe auch Nahtstellen Listen DB71 bis DB73). Damit ist der Vorgang abgeschlossen. Anders verhält es sich bei den Statusinformationen 103 bis 105. Bei Übergabe einer dieser Statusinformationen zum FC8 bleibt das "Schnittstelle Aktiv-Bit" dieser Schnittstelle auf "1". Eine weitere Bearbeitung ist notwendig durch das Anwenderprogramm in der PLC (z. B. Fortführen der Magazinpositionierung). Diese Statusinformation dient in der Regel zum Übermitteln von veränderten Positionen von einem oder beiden Werkzeugen, ohne dass der Vorgang abgeschlossen ist.

Aufrufbeispiel

```

CALL FC8(           // Werkzeugverwaltungs Transfer Baustein
  Start :=          m 20.5,           // Start := "1 " => Anstoß des Transfers
  TaskIdent :=      DB61.DBB0,
  TaskIdentNo :=    DB61.DBB1,
  NewToolMag :=     DB61.DBW2,        // aktuelle Position neues Werkzeug
  NewToolLoc :=     DB61.DBW4,
  OldToolMag :=     DB61.DBW6,        // aktuelle Position altes Werkzeug
  OldToolLoc :=     DB61.DBW8,
  Status :=         DB61.DBW10,       // Status
  Ready :=          m 20.6,
  Error :=          DB61.DBW12);

u m 20.6;           // Ready abfragen
r m 20.5;           // Start zurücksetzen
spb m001;           // springe wenn alles in Ordnung
l DB61.dbw12;       // Fehler Information
ow w#16#0;          // Fehler auswerten
spn fehl;           // Springe auf Fehlerbehandlung

```

```

m001 :                               // normaler Zweig

fehl :                               // Fehlerbehandlung
r m 20.5 :                           // Start zurücksetzen

```

14.17.17 FC9: ASUP - Start von asynchronen Unterprogrammen

Funktion

Mit dem Baustein FC9 "ASUP" können beliebige Funktionen in der NC ausgelöst werden. Voraussetzung dafür, dass ein ASUP von der PLC gestartet werden kann, ist dessen Anwahl und Parametrierung durch ein NC-Programm oder durch den FB4 (PI-Dienst ASUP). Hierbei muss Kanal- und Interruptnummer übereinstimmen zu den Parametern im FC9.

Ein derart vorbereitetes ASUP kann von der PLC zu einem beliebigen Zeitpunkt gestartet werden. Das in dem betreffenden Kanal laufende NC-Programm wird durch das ASUP unterbrochen.

In einem Kanal kann zu einem Zeitpunkt nur ein ASUP gestartet werden. Werden in **einem** PLC-Zyklus mehrere ASUP gestartet, werden die ASUP in der NC in dieser Reihenfolge gestartet.

Parameter "Start" = 0 muss vom Anwender gesetzt werden, wenn das ASUP beendet ("Done" == 1) oder ein Fehleraufgetreten ist ("Error" == 1).

Zur Auftragsbearbeitung benötigt jeder FC9 einen eigenen Parameter "Ref" aus dem globalen Anwenderbereich. Dieser wird intern verwendet und darf vom Anwender nicht verändert werden. Der Parameter "Ref" wird im ersten OB1-Zyklus mit dem Wert 0 initialisiert, aus diesem Grund muss jeder FC9 **absolut** aufgerufen werden. Alternativ kann der Parameter "Ref" durch den Anwender im Anlauf mit dem Wert 0 initialisiert werden. Damit sind auch bedingte Aufrufe möglich. Ein bedingter Aufruf erfordert bei Aktivierung des FC9 solange Parameter "Start" = 1, bis am Parameter "Done" ein negativer Flankenwechsel aufgetreten ist (1 → 0).

Randbedingungen

- Der Funktionsbaustein FB4 muss beendet sein, bevor der Baustein FC9 gestartet wird.
- Der Baustein FC9 kann nicht gestartet werden, wenn DB10, DBX56.1 == 1 (Not-Halt).
- Der Baustein FC9 darf nicht gestartet werden, wenn im Kanal, in dem das ASUP gestartet werden soll, Kanal-Reset aktiv ist.

Deklaration der Funktion

```

FUNCTION FC9: VOID
//NAME :ASUP
VAR_INPUT
    Start :      BOOL;
    ChanNo :     INT;
    IntNo  :     INT;

```

```

END_VAR
VAR_OUTPUT
    Activ :    BOOL;
    Done  :    BOOL;
    Error  :    BOOL;
    StartErr :  BOOL;
END_VAR
VAR_IN_OUT
    Ref :      WORD;
END_VAR

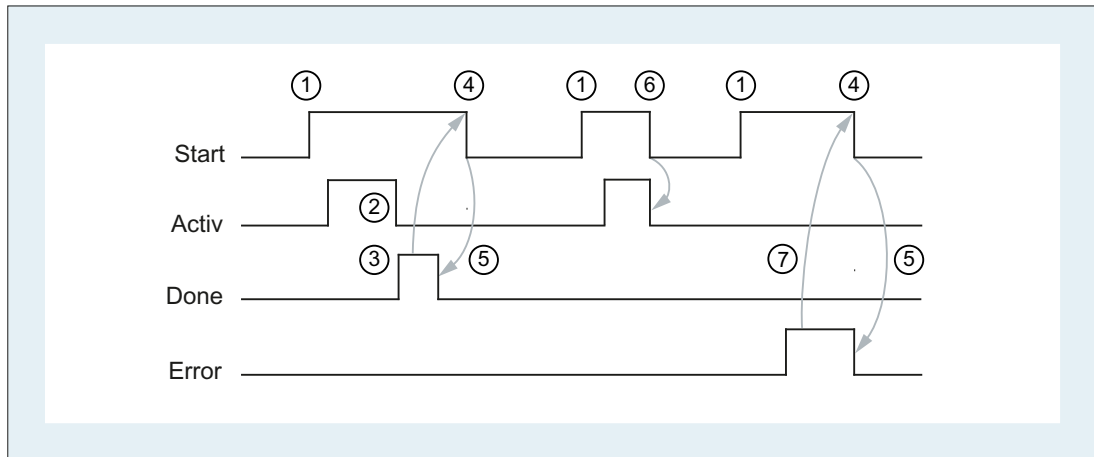
```

Erläuterung der Formalparameter

Die folgende Tabelle zeigt alle Formalparameter der Funktion ASUP:

Signal	Art	Typ	Wertebereich	Bedeutung
Start:	E	BOOL	0 (FALSE), 1 (TRUE)	Auftragsstart mit positiver Flanke
ChanNo:	E	INT	1, 2, 3 ... 10	Kanalnummer
IntNo:	E	INT	1, 2, 3 ... 8	Interrupt-Nummer
Activ:	A	BOOL	0 (FALSE), 1 (TRUE)	1: aktiv
Done:	A	BOOL	0 (FALSE), 1 (TRUE)	1: ASUP beendet
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Interrupt ist ausgeschaltet
StartErr:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Interruptnummer nicht vergeben oder gelöscht
Ref:	E/A	WORD	globale Variable (MW, DBW, ...)	1 Wort je FC9 (für interne Verwendung)

Impulsdiagramm



- (1) Funktionsanstoß
- (2) ASUP aktiv
- (3) Positive Quittung: ASUP beendet
- (4) Rücksetzen vom Funktionsanstoß nach Erhalt der Quittung
- (5) Signalwechsel durch FC
- (6) Nicht zulässig. Wird Funktionsanstoß vor Erhalt der Quittung rückgesetzt, werden die Ausgangssignale nicht aktualisiert, ohne Einfluss auf Ablauf der angestoßenen Funktion.
- (7) Negative Quittung: Fehler aufgetreten

Aufrufbeispiel

```
CALL FC9(                                     //Start eines asynchronen Unterprogramms
                                                //im Kanal 1 Interruptnummer 1

    Start :=      E 45.7,
    ChanNo :=     1,
    IntNo  :=     1,
    Activ  :=     M 204.0,
    Done   :=     M204.1,
    Error  :=     M 204.4,
    StartErr :=   M 204.5,
    Ref    :=     MW 200);
```

14.17.18 FC10: AL_MSG - Fehler- und Betriebsmeldungen

Funktion

Mit dem Baustein FC10 "AL_MSG" werden die in DB2 eingetragenen Signale ausgewertet und als kommende und gehende Fehler- und Betriebsmeldungen auf der Bedienoberfläche angezeigt.

Die kommenden Signale (positive Flanke) werden sowohl bei Fehler- als auch bei Betriebsmeldungen sofort zur Anzeige gebracht.

Gehende Signale (negative Flanke) werden nur bei Betriebsmeldungen sofort gelöscht. Bei Fehlermeldungen werden erst mit dem Parameter "Quit" die nicht mehr anstehenden Meldungen gelöscht, d.h. Fehleranzeigen bleiben, auch wenn die Signale nicht mehr anstehen auf der Bedienoberfläche solange erhalten, bis sie vom Anwender quittiert wurden.

Über den Parameter "ToUserIF" kann die Übertragung der Sammelsignale für Vorschub-, Einlese- und NC-Start-Sperre sowie Vorschub-Halt an vorhandene Achs-, Spindel- und Kanal-Nahtstellen veranlasst werden. Die Sammelsignale werden unabhängig von der Quittierung eines Alarms direkt aus den Zustandsinformationen des DB2 in die Anwendernahtstelle transferiert.

1. Bei Parameter "ToUserIF" = 0 erfolgt keine Übertragung der Signale an die Anwendernahtstelle. Der Anwender muss in diesem Fall durch sein PLC-Programm dafür sorgen, dass diese Signale in der Nahtstelle beeinflusst werden. Es erfolgt eine Auswertung des FB1-Parameters "ExtendChanAxMsg", so dass eine Begrenzung der nutzbaren Meldungsbereiche durch die Projektierung der NCK-Maschinendaten entfällt.
2. Bei Parameter "ToUserIF" = 1 erfolgt eine Übertragung aller oben aufgeführten Signale an die Anwendernahtstelle als jeweils ein Sammelsignal. Das PLC-Programm des Anwenders kann folglich die oben aufgeführten Signale nur über den DB2 beeinflussen in Verbindung mit einer Meldungs- oder Alarm-Ausgabe. Es erfolgt ein Überschreiben der jeweiligen Information in der Anwendernahtstelle.

Alternativ zu dem unter Punkt 2 beschriebenen Verhalten kann eine Beeinflussung des Sperr- und Halt-Signals ohne Meldungs Ausgabe vorgenommen werden, indem nach dem Aufruf des FC10 die Nahtstellensignale mit einem Sperr- oder Halt-Signalzustand beeinflusst werden

Folgende Programmsequenz soll das verdeutlichen:

```
CALL FC10(
    ToUserIF := TRUE,
    Quit := e 6.1);

u m 50.0;           // Vorschub Sperre für Kanal 1
auf db 21;
s dbx 6.0;          // Setzen der Sperrbedingung,
                    // Rücksetzen erfolgt über den FC AL_MSG,
                    // wenn M 50.0 das Signal "0" führt.
```

FB1-Parameter "ExtendAIMsg"

Mit FB1-Parameter "ExtendAIMsg" = TRUE wird eine neue Strukturierung des DB2 wirksam (siehe "Nahtstelle PLC/HMI (Seite 905)"). Bei Aktivierung stehen Bitfelder für die Sperr- und Haltsignale für 10 Kanäle, 31 Achsen und maximal 64 Anwenderbereiche (die Anzahl der Anwenderbereiche ist am FB1-Parameter "MsgUser" anzugeben) zur Verfügung. Durch einfaches Setzen / Rücksetzen von Signalen im DB2 ist die jeweilige Funktionalität automatisch gegeben.

Die Fehler- und Betriebsmeldungen im Datenbaustein DB2 sind anwenderspezifisch zu versorgen.

FB1-Parameter "ExtendChanAxMsg"

Mit dem Aktivieren dieses Parameters wird eine Kanal- bzw. Achsnummer unabhängige Erfassung von Alarmen und Meldungen wirksam. Alle DB2-Bereiche sind für den Anwender nutzbar. Es kann keine Übertragung von Sammelsignalen an die Anwendernahtstelle erfolgen. Die Auswertung des Parameters erfolgt nur, wenn der FC10-Parameter "ToUserIF" deaktiviert ist.

Anzeige am HMI

Im DB2 muss ein "1-Signal" für mehrere OB1-Zyklen anstehen, damit eine Meldung auch am HMI angezeigt werden kann.

Es gibt eine Obergrenze von gleichzeitig anstehenden Alarmen und Meldungen. Diese Obergrenze ist abhängig von der PLC-CPU. Bei der PLC317-2DP beträgt die Obergrenze 60 gleichzeitig anstehende Meldungen.

Literatur:

Listenhandbuch NC-Variable und Nahtstellensignale, Kapitel "PLC-Anwendernahtstelle" > "PLC- Alarme / Meldungen"

Deklaration der Funktion

```
FUNCTION FC10:      VOID
    // NAME:        AL_MSG
VAR_INPUT
    ToUserIF :      BOOL;
    Quit :        BOOL;
END_VAR
END_FUNCTION
```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
ToUserIF:	E	BOOL	0 (FALSE), 1 (TRUE)	1: Übertragung der Signale an Anwenderschnittstelle je Zyklus
Quit:	E	BOOL	0 (FALSE), 1 (TRUE)	1: Quittierung von Fehlermeldungen

Aufrufbeispiel

```
CALL FC10(
    ToUserIF := TRUE,    // Fehler- und Betriebsmeldungen
                          // Signale aus DB2 werden an Nahtstelle
                          // übertragen.
    Quit := E6.1         // Quittierung der Fehlermeldung über Eingang E6.1
);
```

14.17.19 FC12: AUXFU - Aufrufschnittstelle für Anwender bei Hilfsfunktionen

Funktion

Der Baustein FC12 "AUXFU" wird ereignisgesteuert im Grundprogramm aufgerufen, wenn neue Hilfsfunktionen für den im Eingangs-Parameter übergebenen Kanal bereitstehen. Dieser FC kann durch den PLC-Anwender für seine Hilfsfunktion-Bearbeitung mit Programm-Anweisungen erweitert werden, um das zyklische Polling der Kanal-DBs bei Hilfsfunktionen zu vermeiden. Über diesen Mechanismus wird eine auftragsgesteuerte Bearbeitung der Hilfsfunktionen ermöglicht. Dieser FC wird als leerer Baustein in kompilierter Form im Grundprogramm geliefert. Hierbei wird der Parameter "Chan" vom Grundprogramm mit der Kanalnummer versorgt. Somit kennt der PLC-Anwender den Kanal in dem neue Hilfsfunktionen bereitstehen. Über die Änderungssignale der Hilfsfunktionen in diesem Kanal können die neuen Hilfsfunktionen ermittelt werden.

Deklaration der Funktion

```
FUNCTION FC12: VOID           // Ereignissteuerung der Hilfsfunktionen
VAR_INPUT
    Chan :    BYTE;
END_VAR
BEGIN
    BE;
END_FUNCTION
```

Erläuterungen der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Chan:	E	BYTE	0, 1, 2 ... 9	Index des Kanals = Kanalnummer -1

Beispiel

```
FUNCTION FC12: VOID           // Ereignissteuerung der Hilfsfunktionen
VAR_INPUT
    Chan :    BYTE;           // Parameter wird vom Grundprogramm versorgt
END_VAR
VAR_TEMP
    ChanDB:    INT;
END_VAR

BEGIN
    L Chan;                    // Kanal-Index
    + 21;                      // Kanal DB Offset
    T ChanDB;                  // Kanal DB-Nr. speichern
    AUF DB[ChanDB];           // Kanal-DB wird indirekt geöffnet
```

```
// Jetzt werden die Änderungssignale der Hilfsfunktionen abgefragt, usw.  
    BE;  
END_FUNCTION
```

14.17.20 FC13: BHGDisp - Displaysteuerung für Bedienhandgerät

Funktion

Der Baustein FC13 "BHGDisp" übernimmt die Display-Ansteuerung des Bedienhandgeräts (BHG oder HT 2). Die Information, die auf dem Display erscheinen soll, ist in einer String-Variablen abzulegen. Der Zeiger auf den String wird im Parameter "ChrArray" angegeben. Hierzu ist beim Erstellen des Datenbausteins für diesen String eine Festtext-Zuweisung von 32 (BHG) bzw. 64 (HT 2) Zeichen notwendig.

Pro Auftrag werden 16 Zeichen zum Bedienhandgerät gesendet. Die Zuordnung der Zeichen im "ChrArray" zur jeweiligen Zeile ist eindeutig. Für die Zeile 1 werden die Zeichen 1 bis 16 und für die Zeile 2 die Zeichen 17 bis 32 des Stringdatums "ChrArray" übertragen. Zusätzlich werden beim HT 2 die Zeile 3 aus den Zeichen 33 bis 48 und die Zeile 4 aus den Zeichen 49 bis 64 angezeigt. Ein Auftrag dauert mehrere OB1-Zyklen.

Anzeige

Der Baustein FC13 überprüft, ob die notwendige Mindestlänge des "ChrArray" für den Betrieb des Bedienhandgeräts vorhanden ist. Sind weniger Zeichen in der String-Variablen vorhanden als angezeigt werden sollen, wird die Zeile mit Leerzeichen aufgefüllt. Sollen mehrere Variablen in einem oder mehreren PLC-Zyklen im String eingetragen werden ohne dass eine Display-Ausgabe erfolgt, kann durch Parameter "Row" = 0 die Display-Ausgabe unterdrückt werden. Die Übertragung der Zeichen zu den Zeilen dauert mehrere OB1-Zyklen. Sollen mehrere Zeilen "gleichzeitig" aktualisiert werden (Parameter "Row" > 1), werden die Zeilen nacheinander mit je 16 Zeichen pro Zeile aktualisiert.

Variable Anteile

Variable Anteile innerhalb des Strings können mit der optionalen Zahlenwandler-Funktionalität mit Parameter "Convert" = 1 eingefügt werden. Die anzuzeigende Variable wird über den Parameter "Addr" referenziert. Im Parameter "DataType" wird das Format der Variablen beschrieben. Die Anzahl der Bytes der Variablen ist an die Formatbeschreibung gekoppelt. Die rechtsbündige Adresse innerhalb des Strings wird durch den Parameter "StringAddr" angegeben. Die Anzahl der geschriebenen Zeichen ist aus der Parametertabelle ersichtlich.

Höhere Auflösung der Anzeige

Soll z. B. der Achswert mit einer höheren Auflösung dargestellt werden, ist folgendes zu beachten:

- Das Lesen der Variablen erfolgt wie bisher mit FB2 oder FB5. Statt Anypointer BYTE 8 als Kriterium zur Ausgabe als 64-Bit Gleitkommazahl wird REAL 2 verwendet (z. B.: P#M100.0 REAL 2).
- Bei der Angabe der 64-Bit Gleitkommazahl am BHG / HT 2 kann, anstelle von fest vorgegebenen Formaten, das Ausgabeformat mit bis zu max. 14 Stellen, frei auf Vor- und Nachkommastellen verteilt, gewählt werden.

BHG-Ausgangssignale

Von den BHG-Ausgangssignalen wird Byte 1 und die Zeichenvorgaben durch den Baustein FC13 verwendet. Diese dürfen nicht durch das PLC-Anwenderprogramm beschrieben werden.

Relevante FB1-Parameter**Bedienhandgerät BHG**

Im OB100 sind die FB1-Parameter für die Ein- und Ausgangsdaten des Bedienhandgeräts einzustellen:

- Parameter "BHGIIn" entspricht den Eingangsdaten der PLC vom Bedienhandgerät (Empfangsdaten der PLC)
- Parameter "BHGOOut" entspricht den Ausgangsdaten der PLC zum Bedienhandgerät (Sendedaten der PLC).

Die beiden Zeiger sind auf den jeweiligen Datenbereichs-Anfang einzustellen, der auch bei MPI-Kopplung im SDB210 parametrisiert ist.

Für den Betrieb eines BHG ist der FB1-Parameter "BHG" = 2 zu setzen.

Handheld Terminal HT 2

Beim Einsatz des HT 2 ist FB1-Parameter "BHG" = 5 einzustellen. Die Parameter der Ein- und Ausgangsdaten sind, wie oben im Absatz "Bedienhandgerät BHG" beschrieben, zu setzen.

Den Parametern "BHGRecGDNo" und "BHGRecGBZNo" ist der Wert zuzuweisen, der am S2 des DIP-Fix-Schalters (Drehcodierschalter) des Anschlussmoduls des HT 2 konfiguriert wurde.

Deklaration der Funktion

```
DATA_BLOCK "strdat"
STRUCT
    disp :      STRING [32] := 'character_line1 character_line2';
END_STRUCT;
BEGIN
END_DATA_BLOCK

FUNCTION FC13: VOID
    VAR INPUT
        Row :      BYTE;
        ChrArray :  STRING;
        Convert :   BOOL;
        Addr :      POINTER;
        DataType :  BYTE;
        StringAddr : INT;
        Digits :    BYTE;
    END VAR
    VAR OUTPUT
```

14.17 Bausteinbeschreibungen

```
Error :      BOOL;
END VAR
```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Row:	E	BYTE	0, 1, 2 ... 8, B#16#F	Display-Zeile "binäre" Auswertung 0: keine Display-Ausgabe 1: Zeile 1 2: Zeile 2 3: Zeile 1 und Zeile 2 im Wechsel 4: Zeile 3 5: Zeile 1 und Zeile 3 im Wechsel 8: Zeile 4 B#16#F: automatischer Wechsel aller 4 Zeilen
ChrArray:	E	STRING	"DBName".<VarName>	Display-Inhalt als Zeiger auf String string[32]: nicht HT 2 string[64]: HT 2
Convert:	E	BOOL	0 (FALSE), 1 (TRUE)	Aktivierung der Zahlenwandlung
Addr:	E	Pointer		Zeiger auf die zu konvertierende Variable
DataType:	E	BYTE	1, 2, 3 ... 8, B#16#13, B#16#30	Datentyp der Variable 1: BOOL, 1 Zeichen 2: BYTE, 3 Zeichen 3: CHAR, 1 Zeichen 4: WORD, 5 Zeichen 5: INT, 6 Zeichen 6: DWORD, 7 Zeichen 7: DINT, 8 Zeichen 8: REAL, 9 Zeichen (7 Ziffern plus Vorzeichen und Dezimalpunkt; Nachkommastellen siehe Parameter Digits) B#16#13: String, bis 32/64 Zeichen, "Addr" muss Pointer auf STRING sein. B#16#30: REAL64, (12 Zeichen: 10 Ziffern plus Vorzeichen und Dezimalpunkt; Nachkommastellen siehe Parameter Digits)
StringAddr:	E	INT	1 ... 32 / 64	Rechtsbündige Adresse innerhalb der Variablen aus "ChrArray"

Signal	Art	Typ	Wertebereich	Bedeutung
Digits:	E	BYTE	1, 2, 3 ... 9	Anzahl Nachkommastellen 1 ... 4: DataType REAL 1 ... 9: DataType REAL64
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	Fehler 1: Fehler aufgetreten

Wertebereiche

Wertebereiche von Datentypen	
Datentyp	Darstellbarer Zahlenbereich
BOOL	0, 1
BYTE	0 ... 255
WORD	0 ... 65535
INT	- 32768 ... 32767
DWORD	0 bis 9999999
DINT	-9999999 ... 9999999
REAL (Digits := 1)	-999999.9 ... 999999.9
REAL (Digits := 2)	-99999.99 ... 99999.99
REAL (Digits := 3)	-9999.999 ... 9999.999
...	...
REAL (Digits := 9)	-0.9999999 ... 0.9999999

Aufrufbeispiel

// DB mit Namen strdat in der Symboltabelle, Datenelement disp ist als String[32] (bei HT 2: // String[64]) deklariert und komplett zugewiesen mit Zeichen

```
CALL FC13(
    Row :=          MB 26,
    ChrArray :=     "strdat".disp,
    Convert :=      M 90.1,
    Addr :=         P#M 20.0,      // zu konvertierende Zahl
    DataType :=     MB 28,         // Datentyp der Variablen
    StringAddr :=   MW 30,
    Digits :=       B#16#3,        // 3 Nachkommastellen
    Error :=        M 90.2);
```

14.17.21 FC17: YDelta - Stern-Dreieck-Umschaltung

Funktion

Der Baustein FC17 dient zur Stern-Dreieck-Umschaltung bei digitalen Hauptspindelantrieben. Die Umschaltung kann in beiden Richtungen (Stern > Dreieck bzw. Dreieck > Stern) vorgenommen werden.

Voraussetzung

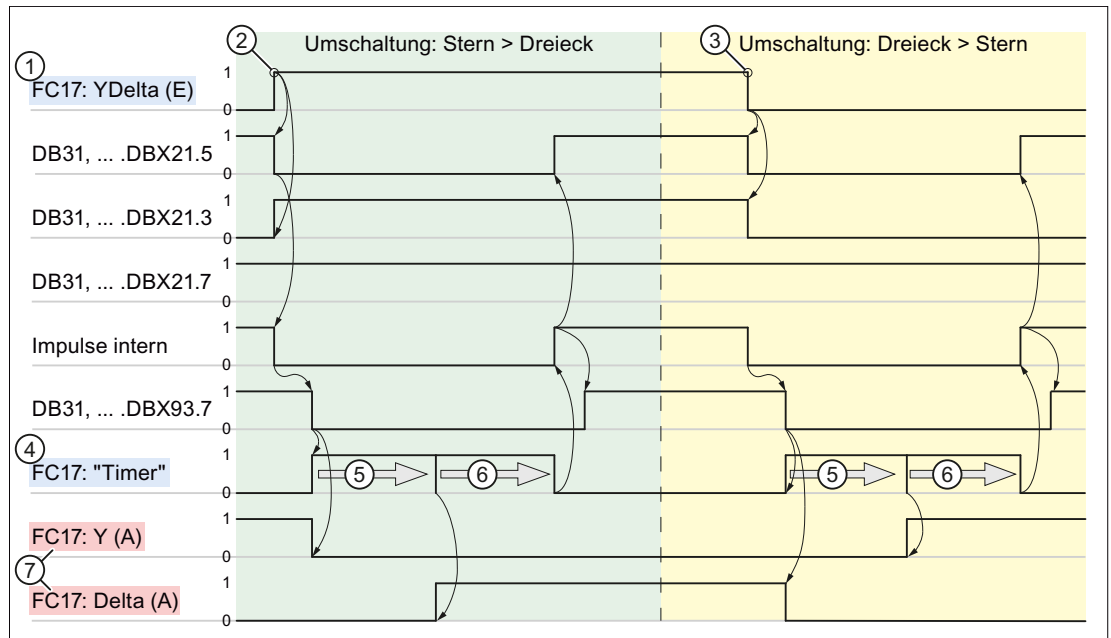
Voraussetzung sind zwei getrennte Schütze. Die Schütze werden über die an den FC17 Ausgängen: "Y" bzw. "Delta" projektierten Peripherie-Ausgangssignale angesteuert.

Interner Ablauf

Nachfolgend ist der interne Ablauf einer Stern-Dreieck-Umschaltung nach dem Umschalten des Ansteuersignals FC17: "YDelta" aufgezeigt.

1. DB31,DBX21.5 = 0 (Rückmeldung "Motoranwahl erfolgt" zurücksetzen)
DB31,DBX21.x = 1 (Anforderung "2. Motordatensatz" entsprechend der Schnittstellenparametrierung DB31,DBX130.0 - 4 (Seite 70) setzen)
2. DB31,DBX93.7 == 0 (Rückmeldung "Impulse freigegeben" wurde zurückgesetzt) ⇒
 - FC17-Timer starten
 - FC17: "Y" = 0 (Ausgang für Stern-Schütz zurücksetzen)
3. Nach Ablauf des FC17-Timers (FC17: "TimeVal") ⇒
 - FC17: "Y" = 1 (Ausgang für Dreieck-Schütz setzen)
4. Nach erneutem Ablauf des FC17-Timers (FC17: "TimeVal") ⇒
 - Die Impulse werden intern wieder gesetzt
 - DB31,DBX93.7 == 1 (Rückmeldung "Impulse freigegeben" wurden gesetzt)
 - DB31,DBX21.5 = 0 (Rückmeldung "Motoranwahl erfolgt" wurden gesetzt)

Signalverlauf



- ① FC17 Eingang: Signal zur Stern-Dreieck-Umschaltung
- ② - Umschaltung: Stern > Dreieck
- ③ - Umschaltung: Dreieck > Stern
- ④ FC17 Eingang: Parametrierbare Umschaltzeit "TimeVal"
- ⑤ - Wartezeit bis Ansteuerung des Ausgangssignals: "Y" bzw. "Delta"
- ⑥ - Wartezeit bis Impulsfreigabe
- ⑦ FC17 Ausgang: Signale zur Schützensteuerung

Literatur

Weitere Erläuterungen zu Motordrehzahlanpassungen siehe:

- Funktionshandbuch Grundfunktionen; Spindeln (S1); Kapitel "Projektierbare Getriebeanpassungen"
- Funktionshandbuch Grundfunktionen; Geschwindigkeiten, Soll-/Istwertsyst., Regelung (G2)

Hinweis

Antriebsparameter

Für eine Stern-Dreieck-Umschaltung sind folgende Antriebsparameter zu berücksichtigen:

- p833 (Datensatzumschaltung Konfiguration)
 - Bit 0 = **1** (Schützumschaltung über **Applikation**)
 - Bit 1 = **0** (Impulslöschung durch **Antrieb**)
- p826 (Motorumschaltung Motornummer)
- p827 (Motorumschaltung Zustandswort Bitnummer)

Randbedingungen

- Der Baustein FC17 muss absolut und für jede Spindel getrennt aufgerufen werden.
- Während einer Stern-Dreieck-Umschaltung werden die Impulse im Antrieb abgeschaltet. Die Rückmeldung an die PLC erfolgt über:
 - DB31,DBX93.7 == 0 (Impulse freigegeben)
 - DB31,DBX61.7 == 0 (Stromregler aktiv)
 - DB31,DBX61.6 == 0 (Drehzahlregler aktiv)
- Bei einer Spindel, die sich in einer Achs-Betriebsart wie z. B. M70 oder SPOS befindet, wird eine Stern-Dreieck-Umschaltung nicht durchgeführt.
- Bei einer lagegeregelten Spindel (DB31,DBX61.5 == 1 (Lageregler aktiv)) darf, während die Spindel verfährt, keine Stern-Dreieck-Umschaltung vorgenommen werden. Im Fehlerfall wird Alarm 25050 "Konturüberwachung" angezeigt und die Stern-Dreieck-Umschaltung nicht durchgeführt.
- Eine über FC17 angestoßene Stern-Dreieck-Umschaltung kann vom Anwender nicht, z. B. durch betriebsmäßiges Warten auf eine erfolgreich durchgeführte Umschaltung der Stern-Dreieck-Schütze, verzögert werden. Eine derartige Verzögerung muss vom Anwender im PLC-Anwenderprogramm realisiert werden.

Deklaration der Funktion

```

VAR_INPUT
    YDelta :      BOOL;
    SpindleIFNo : INT;
    TimeVal :     S5TIME;
    TimerNo :     INT;
END_VAR
VAR_OUTPUT
    Y :           BOOL;
    Delta :       BOOL;
END_VAR
VAR_IN_OUT
    Ref :         WORD;
END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung	
YDelta:	E	BOOL	0 (FALSE), 1 (TRUE)	Eingangssignal zur Stern-Dreieck-Umschaltung Eine Signaländerung löst die Umschaltung aus: <ul style="list-style-type: none">• 0 (FALSE): Stern• 1 (TRUE): Dreieck	
SpindleIFNo:	E	INT	1 ...	Nummer der Spindel-Schnittstelle (Nummer der zugehörigen Maschinenachse)	
TimeVal:	E	S5time	0, 50 ms ...	Umschaltzeit	
				Projektiert	Intern wirksam
				0	100
				< 50	50
TimerNo:	E	INT	10 ...	Nummer des verwendeten Timers	
Y:	A	BOOL	0 (FALSE), 1 (TRUE)	Peripherie-Ausgang zur Ansteuerung des Stern-Schützes	
Delta:	A	BOOL	0 (FALSE), 1 (TRUE)	Peripherie-Ausgang zur Ansteuerung des Dreieck-Schützes	
Ref:	E/A	WORD		Instanz für Zustandsinformationen. Interne Verwendung	

Aufrufbeispiel

```

CALL FC17(
    YDelta :=      E 45.7,           // Stern/Dreieck-Umschaltung von
                                   // Eingang 45.7 übernehmen

    SpindleIFNo :=  4,               // Spindel Interface Number: 4
                                   // (Nummer der zugehörigen
                                   // Maschinenachse)

    TimeVal :=      S5T#150ms,       // Umschaltzeit: 150 ms
    TimerNo :=      10,              // Timer: 10

    Y :=            A 52.3,           // Ansteuerung des Stern-Schütz:
                                   // Ausgang 52.3

    Delta :=        A 52.4,           // Ansteuerung des Dreieck-Schütz:
                                   // Ausgang 52.4

    Ref :=          MW 50             // Merkerwort 50
};

```

14.17.22 FC18: SpinCtrl - Spindelsteuerung

Funktion

Mit dem Baustein FC18 "SpinCtrl" können Spindeln und Achsen von der PLC gesteuert werden. Der Baustein unterstützt folgende Funktionen:

- Spindel positionieren
- Spindel drehen
- Spindel pendeln
- Teilungsachse verfahren
- Positionierachse verfahren

Jede Funktion wird durch die positive Flanke des entsprechenden Anstoß-Signals (Start, Stop) gestartet. Das Anstoß-Signal muss so lange auf logisch "1" bleiben, bis die Funktion durch InPos = "1" oder Error = "1" positiv oder negativ quittiert wurde. Die Ausgangsparameter werden mit Rücksetzen des jeweiligen Anstoß-Signals und dem Ende der Funktion gelöscht.

Um die Spindel/Achsen über die PLC zu beeinflussen, muss sie für die PLC aktiviert werden. Dieses kann z. B. durch Aufruf des Bausteins mit Aktivieren des Parameters "Start" bzw. "Stop" erreicht werden. Hierbei fordert der Baustein die Kontrolle der Spindel/Achse von der NC an.

Die NC meldet in der zugehörigen achsspezifischen Nahtstelle DB31, ... DBX68.4 - 7 den Status der Spindel/Achse. Nachdem die Achse / Spindel der PLC gehört, kann über die zugehörige achsspezifische Nahtstelle der Fahrbefehl für den Aktivzustand ausgewertet werden.

Nach Beendigung ("InPos" ist TRUE, "Start" wechselt auf Null) wird die Kontrolle der Achse/Spindel vom Baustein FC18 in einen neutralen Zustand geschaltet.

Alternativ kann auch das PLC-Anwenderprogramm vor Aufruf des FC18 die Kontrolle für die PLC anfordern.

Hierdurch kann ein besseres Reaktionsverhalten der Spindel/Achse bei mehrfach hintereinander folgenden Aufrufen dieser Funktion erreicht werden, da der Umschaltvorgang im FC entfällt.

Die Aktivierung durch das PLC-Anwenderprogramm wird in der zugehörigen Spindel-Nahtstelle im Byte 8 durchgeführt.

Nach Rückgabe der Kontrolle kann die Spindel vom NC-Programm wieder programmiert werden.

Literatur

- Funktionshandbuch Grundfunktionen; Spindeln (S1)
- Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2)
- Funktionshandbuch Erweiterungsfunktionen; Teilungsachsen (T1)

! WARNUNG**Verändertes Reaktionsverhalten der Achse/Spindel**

Wenn mehrere Bausteinaufrufe (FC18) für die gleiche Achse / Spindel im PLC-Anwenderprogramm programmiert wurden, dann ist eine Verriegelung dieser Funktionen durch bedingte Aufrufe im Anwenderprogramm notwendig. Der bedingte Aufruf eines gestarteten (Parameter Start oder Stop = TRUE) ist solange zyklisch aufzurufen bis ein Zustandswechsel des Ausgangs-Parameters "Activ" bzw. "InPos" von 1 nach 0 erfolgt.

Hinweis**Aufrufhinweis**

Der FC18 ist zyklisch aufzurufen, bis das Signal "InPos" , bzw. bei Fehler das Signal "Error", einen Flankenwechsel von 1 nach 0 liefert. Erst wenn das Signal "InPos"/"Error" den Wert 0 geliefert hat, ist ein weiterer "Start" oder "Stop" für diese Spindel/Achse möglich. (mindestens ein PLC-Zyklus muss mit dem nächsten "Start" oder "Stop" gewartet werden). Dies gilt auch bei Veränderung der Zuordnung im Datenbyte 8 der axialen Nahtstelle.

Abbruch

Ein Abbrechen der Funktion ist nicht möglich über den Parameter "Start" oder "Stop", sondern nur durch die axialen Nahtstellensignale (z. B. Restweglöschen). Ebenso liefert die axiale Nahtstelle Statussignale der Achse zurück, die gegebenenfalls auszuwerten sind (z. B. Genauhalt, Fahrbefehl).

Gleichzeitigkeit

Mehrere Achsen können zeitgleich oder zeitversetzt durch den Bausteine FC18 verfahren werden. Die Obergrenze ist nur über die maximale Achsanzahl der NC begrenzt.

Achssperre

Bei gesetzter Achssperre (DB31,DBX1.4 == 1) bewegt sich eine über FC18 gesteuerte Achse nicht. Es wird nur ein simulierter Istwert erzeugt. Gleiches Verhalten wie beim Verfahren der Achse bei Achsensperre durch die NC.

Funktionen**Funktion 1: Spindel positionieren**

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Funct :	1: Funktionsnummer für "Spindel positionieren"
Mode :	Positionier-Mode 1, 2, 3, 4 (siehe unten Absatz "Erläuterung der Formalparameter")
AxisNo :	Nummer der Maschinenachse
Pos :	Position
FRate :	FRate ≠ 0: Positioniergeschwindigkeit FRate = 0: Geschwindigkeit entsprechend MD35300 \$MA_SPIND_POSCTRL_VELO
InPos :	1: Position mit "Genauhalt Fein" erreicht

Parameter	Bedeutung
Error :	1: Positionierfehler
State :	Fehlercode

Funktion 2: Spindel drehen

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Stop :	0 → 1: Stop der Funktion
Funct :	2: Funktionsnummer für "Spindel drehen"
Mode :	Mode = 5: Drehrichtung M4 Mode ≠ 5: Drehrichtung M3
AxisNo :	Nummer der Maschinenachse
FRate :	Spindel-Drehzahl
InPos :	1: Solldrehzahl wird ausgegeben, siehe dazu auch DB31, ... DBX83.5 (Spindel im Sollbereich)
Error :	1: Positionierfehler
State :	Fehlercode

Funktion 3: Spindel pendeln

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Stop :	0 → 1: Stop der Funktion
Funct :	3: Funktionsnummer für "Spindel pendeln"
AxisNo :	Nummer der Maschinenachse
Pos :	Sollgetriebestufe
InPos :	1: Solldrehzahl wird ausgegeben, siehe dazu auch DB31, ... DBX83.5 (Spindel im Sollbereich)
Error :	1: Positionierfehler
State :	Fehlercode

Parametrierte Pendeldrehzahl: MD35400 \$MA_SPIND_OSCILL_DES_VELO

Die Funktion des Parameters "Pos" ist abhängig von der Einstellung in MD35010
\$MA_GEAR_STEP_CHANGE_ENABLE = <Wert>

<Wert>	Pos	Funktion
0	0, 1, 2, ... 5	Pendeln
1	0	Pendeln mit Getriebestufenwechsel M40
	1	Pendeln mit Getriebestufenwechsel M41
	2	Pendeln mit Getriebestufenwechsel M42
	3	Pendeln mit Getriebestufenwechsel M43
	4	Pendeln mit Getriebestufenwechsel M44
	5	Pendeln mit Getriebestufenwechsel M45

Funktion 4: Teilungsachsen verfahren**Hinweis**

Die Modulo-Wandlung ist vergleichbar mit Teilungsposition anfahren über POS[AX]=CIC(wert) im Teileprogramm.

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Funct :	4: Funktionsnummer für "Teilungsachse"
Mode :	Positionierungs-Mode 0, 1, 2, 3, 4
AxisNo :	Nummer der Maschinenachse
Pos :	Teilungs-Position
FRate :	FRate ≠ 0: Positioniergeschwindigkeit FRate = 0: Geschwindigkeit entsprechend MD32060 \$MA_POS_AX_VEL
InPos :	1: Position mit "Genauhalt Fein" erreicht
Error :	1: Positionierfehler
State :	Fehlercode

Funktion 5, 6, 7, 8: Achsen positionieren

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Funct :	5, 6, 7, 8: Funktionsnummer für "Achsen positionieren"
Mode :	Positionierung-Mode 0, 1, 2, 3, 4
AxisNo :	Nummer der Maschinenachse
Pos :	Position
FRate :	FRate ≠ 0: Positioniergeschwindigkeit FRate = 0: Geschwindigkeit entsprechend MD32060 \$MA_POS_AX_VELO
InPos :	1: Position mit "Genauhalt Fein" erreicht
Error :	1: Positionierfehler
State :	Fehlercode

Funktion 9: Spindel drehen mit automatischer Getriebestufenanwahl:

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Stop :	0 → 1: Stop der Funktion
Funct :	9: Funktionsnummer für "Spindel drehen mit Getriebestufenanwahl"
Mode :	Mode = 5: Drehrichtung M4 Mode ≠ 5: Drehrichtung M3
AxisNo :	Nummer der Maschinenachse
FRate :	Spindel-Drehzahl
InPos :	1: Solldrehzahl wird ausgegeben

Parameter	Bedeutung
Error :	1: Positionierfehler
State :	Fehlercode

Funktion 10, 11: Spindel drehen mit konstanter Schnittgeschwindigkeit

In der NC muss die Funktion "konstante Schnittgeschwindigkeit" (G96) aktiv sein.

Parameter	Bedeutung
Start :	0 → 1: Start der Funktion
Stop :	0 → 1: Stop der Funktion
Funct :	10: Funktionsnummer für "konstanter Schnittgeschwindigkeit (m/min)" 11: Funktionsnummer für "konstanter Schnittgeschwindigkeit (feet/min)"
Mode :	Mode = 5: Drehrichtung M4 Mode ≠ 5: Drehrichtung M3
AxisNo :	Nummer der Maschinenachse
FRate :	Schnittgeschwindigkeit
InPos :	1: Solldrehzahl wird ausgegeben
Error :	1: Positionierfehler
State :	Fehlercode

Deklaration der Funktion

```

FUNCTION FC18: VOID           //SpinCtrl
VAR_INPUT
    Start :                BOOL;
    Stop  :                BOOL;
    Funct  :                BYTE;
    Mode   :                BYTE;
    AxisNo :                INT;
    Pos    :                REAL;
    FRate  :                REAL;
END_VAR
VAR_OUTPUT
    InPos  :                BOOL;
    Error  :                BOOL;
    State  :                BYTE;
END_VAR

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
Start:	E	BOOL	0 (FALSE), 1 (TRUE)	0 → 1: Start der Funktion
Stop:	E	BOOL	0 (FALSE), 1 (TRUE)	0 → 1: Stop der Funktion
Funct:	E	BYTE	1, 2, 3, ... 11	1: Spindel positionieren 2: Spindel drehen 3: Spindel pendeln 4: Teilungsachse 5: Positionierachse metrisch 6: Positionierachse inch 7: PosAchse metrisch mit Handradüberlagerung 8: PosAchse inch mit Handradüberlagerung 9: Spindel drehen mit automatischer Getriebestufenanwahl 10: Spindel drehen mit konstanter Schnittgeschwindigkeit (m/min) 11: Spindel drehen mit konstanter Schnittgeschwindigkeit (feet/min)
Mode:	E	BYTE	0, 1, 2, ... 5	0: Positionieren auf Absolutposition 1: Positionieren inkrementell 2: Positionieren auf kürzestem Weg 3: Positionieren absolut, positive Anfahrriichtung 4: Positionieren absolut, negative Anfahrriichtung 5: Drehrichtung wie M4
AxisNo:	E	INT	1, 2, 3, ... 31	Nummer der zu verfahrenen Achse/Spindel
Pos:	E	REAL	± 0,1469368 E -38 bis ± 0,1701412 E +39	Rundachse: Grad Teilungsachse: Teilungsposition Linearachse: mm oder inch
FRate:	E	REAL	± 0,1469368 E -38 bis ± 0,1701412 E +39	Rundachse und Spindel: [Umdr./Min] Linearachsen: [m/min] bzw. [ft/min]
InPos:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Position erreicht bzw. Funktion ausgeführt
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Fehler
State:	A	BYTE	0, 1, 2, ... 255	Fehlerkennung

Fehlerkennungen

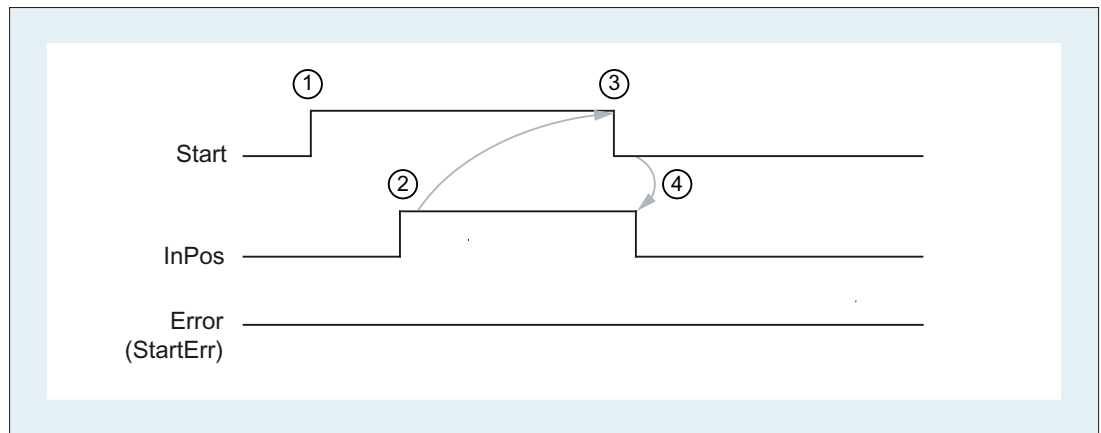
Ein Fehler liegt vor, wenn: Parameter "Error" == 1 (TRUE)

Die Fehlerursache wird angezeigt in: Parameter "State"

State	Bedeutung
Fehlerursache PLC-seitig	
1	es wurden mehrere Funktionen der Achse/Spindel gleichzeitig aktiviert
20	es wurde eine Funktion gestartet, ohne dass die Position erreicht wurde
30	Die Achse/Spindel wurde vor Ende der Bewegung an die NC abgegeben
40	Die Achse ist durch NC Programm programmiert, NC interner Fehler
50	fest zugeordnete PLC-Achse: verfährt (JOG) oder referenziert
60	fest zugeordnete PLC-Achse: Kanalzustand lässt keinen Start zu
Fehlerursache NC-seitig	
100	Falsche Position bei Achse/Spindel programmiert (entspricht Alarm 16830)
101	Programmierte Drehzahl ist zu hoch
102	Falscher Wertebereich konstante Schnittgeschwindigkeit (entspricht Alarm 14840)
104	Folgespindel: unerlaubte Programmierung (entspricht Alarm 22030)
105	Kein Messsystem vorhanden (entspricht Alarm 16770)
106	Positioniervorgang der Achse noch aktiv (entspricht Alarm 22052)
107	Referenzmarke nicht gefunden (entspricht Alarm 22051)
108	Kein Übergang von der Drehzahlregelung in die Lageregelung (entspricht Alarm 22050)
109	Referenzmarke nicht gefunden (entspricht Alarm 22051)
110	Geschwindigkeit/Drehzahl ist negativ
111	Soll-Drehzahl == Null
112	ungültige Getriebestufe
115	programmierte Position wurde nicht erreicht
117	in der NC ist G96/G961 nicht aktiv
118	in der NC ist G96/G961 noch aktiv
120	Achse ist keine Teilungsachse (entspricht Alarm 20072)
121	Teilungsposition- Fehler (entspricht Alarm 17510)
125	DC (kürzester Weg) nicht möglich (entspricht Alarm 16800)
126	Absolutwert Minus nicht möglich (entspricht Alarm 16820)
127	Absolutwert Plus nicht möglich (entspricht Alarm 16810)
128	Keine Planachse für Durchmesserprogrammierung vorhanden (entspricht Alarm 16510)
130	Software-Endschalter Plus (entspricht Alarm 20070)
131	Software-Endschalter Minus (entspricht Alarm 20070)
132	Arbeitsfeldbegrenzung Plus (entspricht Alarm 20071)
133	Arbeitsfeldbegrenzung Minus (entspricht Alarm 20071)
134	Frame für Teilungsachse unzulässig
135	Teilungsachse mit "Hirth-Verzahnung" aktiv (entspricht Alarm 17501)
136	Teilungsachse mit "Hirth-Verzahnung" aktiv und Achse nicht referenziert (entspricht Alarm 17503)
137	Spindelbetrieb für transformierte Spindel/Achse nicht möglich (entspricht Alarm 22290)
138	Achse: Koordinatensystem-spezifische Arbeitsfeldbegrenzung Plus verletzt (entspricht Alarm 20082)
139	Achse: Koordinatensystem-spezifische Arbeitsfeldbegrenzung Minus verletzt (entspricht Alarm 20082)

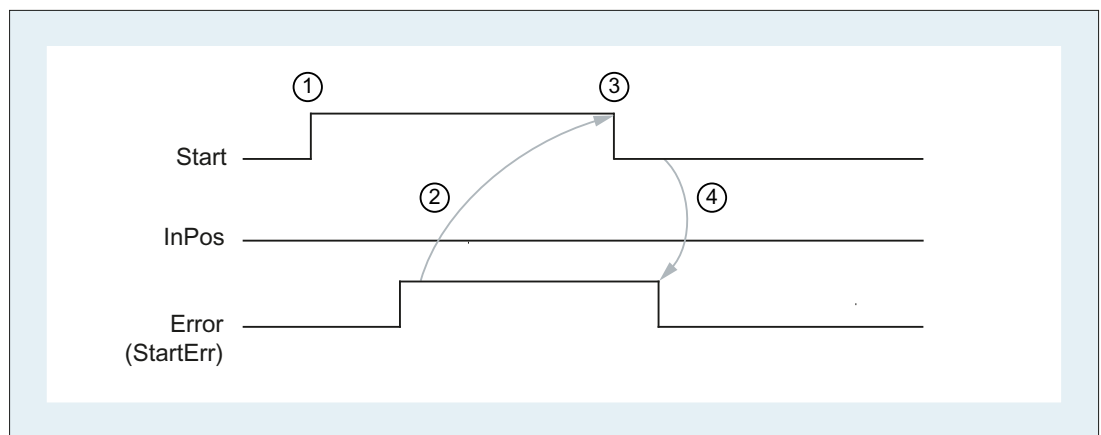
State	Bedeutung
Systemfehler	
200	entspricht Alarm 450007
Alarmnummern: Literatur Diagnosehandbuch	

Signalverlauf: Normalfall



- ① PLC-Anwenderprogramm: Funktionsstart durch positive Flanke: 0 → 1
- ② NC: Positive Quittierung, Funktion ausgeführt / Position erreicht
- ③ PLC-Anwenderprogramm: Rücksetzen nach Erkennen der positiven Quittierung
- ④ FC18: Rücksetzen der positiven Quittierung

Signalverlauf: Fehlerfall



- ① PLC-Anwenderprogramm: Funktionsstart durch positive Flanke: 0 → 1
- ② NC: Negative Quittierung, Fehler aufgetreten
- ③ PLC-Anwenderprogramm: Rücksetzen nach Erkennen der negativen Quittierung
- ④ FC18: Rücksetzen der negativen Quittierung

Aufrufbeispiele

Beispiel 1: Spindel positionieren:

```
//positive Quittung setzt Start zurück:
U M112.0;                //InPos
R M 100.0;                //Start
//negative Quittung , nach Fehlerauswertung (State: MB114) mit T12 Start rücksetzen
U M113.0;                //Error
U E 6.4;                 //Taste T12
R M 100.0;                //Start
//Starten mit T13
U E 6.3;                 //Taste T13
UN M 112.0;               //neuen Start erst wenn InPos bzw. Error = 0
UN M 113.0;
S M 100.0;

CALL FC18(
    Start := M100.0,
    Stop := FALSE,
    Funct := B#16#1,       //Spindel positionieren
    Mode := B#16#2,       //kürzester Weg
    AxisNo := 5,
    Pos := MD104,
    FRate := MD108,
    InPos := M112.0,
    Error := M113.0,
    State := MB114);
```

Beispiel 2: Start Spindel drehen:

```
CALL FC18(
    Start := M100.0,
    Stop := FALSE,
    Funct := B#16#2,       //Spindel drehen
    Mode := B#16#5,       //Drehrichtung wie M4
    AxisNo := 5,
    Pos := 0.0,
    FRate := MD108,
    InPos := M112.0,
    Error := M113.0,
    State := MB114);
```

Beispiel 3: Start Spindel pendeln

```
CALL FC18(
    Start := M100.0,
```



```

Stop := FALSE,
Funct := B#16#3,      //Spindel pendeln
Mode := B#16#0,
AxisNo := 5,
Pos := 0.0,
FRate := MD108,
InPos := M112.0,
Error := M113.0,
State := MB114);

```

Beispiel 4: Teilungsachse verfahren

```

CALL FC18(
  Start := M100.0,
  Stop := FALSE,      //nicht genutzt
  Funct := B#16#4,    //Teilungsachse verfahren
  Mode := B#16#0,     //absolut positionieren
  AxisNo := 4,
  Pos := MD104,       //Vorgabe in REAL: 1.0;2.0;..
  FRate := MD108,
  InPos := M112.0,
  Error := M113.0,
  State := MB114);

```

Beispiel 5: Achsen positionieren

```

CALL FC18(
  Start := M100.0,
  Stop := FALSE,      //nicht genutzt
  Funct := B#16#5,    //Achsen positionieren
  Mode := B#16#1,     //inkrementell positionieren
  AxisNo := 6,
  Pos := MD104,
  FRate := MD108,
  InPos := M112.0,
  Error := M113.0,
  State := MB114);

```

14.17.23 FC19: MCP_IFM - Übertragung der MCP-Signale an die Nahtstelle

Funktion

Der Baustein FC19 "MCP_IFM" (M-Variante z. B. MCP 483) dient zur Übertragung der Daten von der Maschinensteuertafel in die NC/PLC-Nahtstelle:

- Betriebsarten
- Achsanwahlen
- WKS/MKS-Umschaltung
- Verfahrstasten
- Overrides
- Schlüsselschalter

Für **Vorschub-Override**, **Achsfahr**- und **INC-Tasten** gelten, abhängig von der aktiven Betriebsart bzw. vom angewählten Koordinatensystem, folgende Festlegungen:

- **Vorschub-Override**
 - Der Vorschub-Override wird auf die Nahtstelle des angewählten Kanals und auf die Nahtstelle der Achsen transferiert.
 - Die Vorschub-Override-Signale werden zusätzlich zum Nahtstellenbyte "Eilgangkorrektur" (DBB5) an den NC-Kanal übergeben, wenn das HMI-Signal "Vorschub-Override für Eilgang wirksam" gesetzt ist (Ausnahme: Schalterstellung "Null"). Weiterhin wird mit diesem HMI-Signal auch "Eilgangkorrektur wirksam" gesetzt.
- **Maschinenfunktion INC- und Achsfahrtasten**
 - Bei angewähltem MKS werden die Signale auf die Nahtstelle der angewählten Maschinenachse transferiert.
 - Bei angewähltem WKS werden die Signale auf die Geo-Achs-Nahtstelle des parametrisierten Kanals transferiert.
 - Bei Umschaltung zwischen MKS und WKS erfolgt generell eine Abwahl der bis dahin angewählten Achse.

Die Ansteuerung der zugehörigen LED der Maschinensteuertafel wird aus der Rückmeldung entsprechender Anwahlen abgeleitet.

Vorschub- und Spindel-Start/Stopp werden nicht an die Nahtstelle übertragen, sondern als Signal "FeedHold" bzw. "SpindleHold" selbsthaltend ausgegeben. Der Anwender kann diese Signale mit weiteren Signalen verknüpfen, die zu Vorschub- oder Spindel-Halt führen sollen (dies kann z. B. über die entsprechenden Eingangssignale des FC10: AL_MSG erfolgen). Zusätzlich werden die zugehörigen LED mit angesteuert.

Bei Ausfall der Maschinensteuertafel werden die Signale, die von dieser kommen, mit Null vorbesetzt; ebenso die Ausgangssignale "FeedHold" und "SpindleHold".

In einem PLC-Zyklus sind Mehrfachaufrufe des FC19 bzw. auch FC24, FC25, FC26 zulässig. Hierbei steuert der erste Aufruf im Zyklus die LED-Anzeigen an. Weiterhin werden im ersten Aufruf alle Aktionen des parametrisierten Bausteins durchgeführt. Bei den weiteren Aufrufen findet nur noch eine reduzierte Bearbeitung von Kanal und BAG-Schnittstelle statt. Die

Geometrieachsen werden nur im ersten Aufruf des Bausteins im Zyklus mit Richtungsvorgaben versorgt.

Die Einzelsatz-An-/Abwahl wird nur durch den ersten Aufruf im Zyklus beeinflusst.

Die zweite Maschinensteuertafel kann bearbeitet werden, wenn der Parameter "BAGNo" um B#16#10 erhöht ist. Bei der Parametrierung ist die BAG Nummer im unteren Nibble enthalten.

"BAGNo" = 0 oder B#16#10 \Rightarrow keine Bearbeitung der BAG Signale.

"ChanNo" = 0 \Rightarrow keine Bearbeitung der Kanal-Signale.

Die INC-Anwahlen werden in die BAG-Schnittstelle übertragen. Die Aktivierung für diese Vorgabe findet über den DB10.DBX57.0 (INC-Eingänge im BAG-Bereich aktiv) durch diesen Baustein einmalig nach Hochlauf statt.

Weiterhin können zwei Maschinensteuertafeln parallel durch den Baustein FC19 bearbeitet werden. Hierbei ist der Aufruf des Bausteins für die 2. Maschinensteuertafel im OB1-Zyklus zeitlich hinter den Aufruf für die 1. Maschinensteuertafel zu setzen. Eine Unterstützung von 2 Maschinensteuertafeln ist in den Maschinensteuertafel-Bausteinen in Grenzen vorhanden. Nicht unterstützt werden gegenseitige Verriegelungen der Achsanwahlen bei gleich zugeordneten Achsen bei 2 Maschinensteuertafeln.

Kartesisches manuelles Verfahren

Über die Richtungstaste R11 der Maschinensteuertafel (links neben WCS/MCS) ist die Funktion "Manuelles Verfahren in Werkzeugorientierung" möglich. Dazu ist eine Aktivierung über den FB1-Eingangsparameter "MCP_IF_TCS" im DB7 notwendig.

Bei "MCP_IF_TCS" = TRUE findet über die Taste R11 eine Umschaltung auf "Manuelles Verfahren in Werkzeugorientierung" statt. Dabei wird die Z-Taste (R3) fest durch FC19 angewählt. Die Richtungstasten wirken auf die 3. Geometrieachse des jeweiligen Kanals.

Flexible Achskonfiguration

Es ist eine Flexibilität bezüglich der Zuordnung Achsanwahlen bzw. Richtungstasten von Maschinen-Achsnummern verfügbar.

Der Einsatz von 2 Maschinensteuertafeln, die gleichzeitig betrieben werden sollen, wird durch die MCP Bausteine insbesondere für den Anwendungsfall 2 Kanäle, 2 BAG besser unterstützt. Bei den Achs-Tabellen der jeweiligen MCP ist zu beachten, dass die Achsnummern auch in der parametrisierten BAG Nummer des MCP-Bausteins angegeben sind.

Für diese Flexibilität existieren Tabellen für Achsnummern im DB10.

Für die **erste** Maschinensteuertafel beginnt die Tabelle ab dem Byte 8 (symbolischer Name: MCP1AxisTbl[1..22]) und für die **zweite** Maschinensteuertafel ab dem Byte 32 (symbolischer Name: MCP2AxisTbl[1..22]). Hier sind die Maschinen-Achsnummern Byte-weise einzutragen.

Der Eintrag 0 ist in der Achstabelle zulässig. Eine Prüfung auf eine unzulässige Achsnummer findet nicht statt und kann bei Falscheintrag zum PLC-Stop führen.

Für den **FC19** kann auch eine Begrenzung der **maximal möglichen Achsanwahlen** erfolgen. Diese Obergrenze wird für die erste Maschinensteuertafel im DB10.DBW30 (symbolischer Name: MCP1MaxAxis) bzw. für die zweite Maschinensteuertafel im DB10.DBW54 (symbolischer Name: MCP2MaxAxis) eingestellt.

Der voreingestellte Wert ist 0, damit wirkt die konfigurierte maximale Achsanzahl. Die Achsnummern und die Begrenzung können auch dynamisch angepasst werden. Danach muss eine erneute Achsanwahl bei FC19 erfolgen. Während des Fahrens von Achsen über die jeweiligen Richtungstasten darf keine Umschaltung der Achsnummern erfolgen. Voreingestellt ist der Kompatibilitätsmodus mit den Achsnummern **1 bis 9** für beide MCP und die Begrenzung auf die konfigurierte Achsanzahl.

Beispiel

Es sollen mehr als 9 Achsen mit dem FC19 über Zusatzapplikation angesteuert werden. Folgende Vorgehensweise ist erforderlich:

- Freie Taste an MCP reservieren
- Diese Taste als Flipflop auswerten
- Ausgang des FlipFlops als positiver und negativer Flanke auswerten
- Bei positiver Flanke einen Satz Achsnummern in Achstabelle (DB10) schreiben und LED über dieser Taste einschalten
- Bei negativer Flanke einen anderen Satz Achsnummern in Achstabelle (DB10) schreiben und LED über dieser Taste ausschalten

Deklaration der Funktion

```

FUNCTION FC19: VOID                                // symbolischer Name: MCP_IFM

    VAR_INPUT
        BAGNo :      BYTE;
        ChanNo :      BYTE;
        SpindleIFNo :  BYTE;
    END_VAR

    VAR_OUTPUT
        FeedHold :     BOOL;
        SpindleHold :   BOOL;
    END_VAR

BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
BAGNo:	E	BYTE	B#16#00 - B#16#0A	1. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
			B#16#10 - B#16#1A	2. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden

Signal	Art	Typ	Wertebereich	Bedeutung
ChanNo:	E	BYTE	B#16#00 - B#16#0A	Nummer des Kanals in den die Kanalsignale übertragen werden
SpindleIFNo:	E	BYTE	0 - 31 (B#16#1F)	Nummer der Achse/Spindel in welche die Spindel-daten übertragen werden (Nummer der zugehörigen Maschinenachse)
FeedHold:	A	BOOL	0 (FALSE), 1 (TRUE)	Vorschub Halt von MCP, selbsthaltend
SpindleHold:	A	BOOL	0 (FALSE), 1 (TRUE)	Spindel Halt von MCP, selbsthaltend

Anwahlsignale der MCP auf die Anwender-Nahtstelle

Tabelle 14-3 Schlüsselschalter

Quelle: MCP - Schalter	Ziel: Nahtstellen-DB
Stellung 0	DB10.DBX56.4
Stellung 1	DB10.DBX56.5
Stellung 2	DB10.DBX56.6
Stellung 3	DB10.DBX56.7

Tabelle 14-4 Betriebsarten und Maschinenfunktionen

Quelle: MCP - Taster	Ziel: Nahtstellen-DB (Parameter BAGNo) Darstellung für BAG 1
AUTOMATIC	DB11.DBX0.0
MDA	DB11.DBX0.1
JOG	DB11.DBX0.2
REPOS	DB11.DBX1.1
REF	DB11.DBX1.2
TEACH IN	DB11.DBX1.0
INC 1 ... 10 000, INC Var.	DB11.DBX2.0 - 2.5

Tabelle 14-5 Richtungstasten Eilgangüberlagerung

Quelle: MCP - Taster	Ziel: Nahtstellen-DB (Parameter ChanNo)
Richtungstaste +	DB21,DBX12.7
Richtungstaste -	DB21,DBX12.6
Eilgangüberlagerung	DB21,DBX12.5
Richtungstaste +	DB21,DBX16.7
Richtungstaste -	DB21,DBX16.6
Eilgangüberlagerung	DB21,DBX16.5

Quelle: MCP - Taster	Ziel: Nahtstellen-DB (Parameter ChanNo)
Richtungstaste +	DB21,DBX20.7
Richtungstaste -	DB21,DBX20.6
Eilgangüberlagerung	DB21,DBX20.5

Quelle: MCP - Taster	Ziel: Nahtstellen-DB (alle Achs-DB)
Richtungstaste +	DB31,DBX4.7
Richtungstaste -	DB31,DBX4.6
Eilgangüberlagerung	DB31,DBX4.5

Die Übertragung ist abhängig von der angewählten Achse. Für nicht angewählte Achsen werden die zugehörigen Nahtstellenbits gelöscht.

Tabelle 14-6 Override

Quelle: MCP - Schalter	Ziel: Nahtstellen-DB (Parameter ChanNo)
Vorschuboverride	DB21,DBB4

Quelle: MCP - Schalter	Ziel: Nahtstellen-DB (alle Achs-DB)
Vorschuboverride	DB31,DBB0 (angewählte Achsnummer) Durch Vorschuboverride der 1. MCP werden alle Achsen beaufschlagt.
Spindeloverride	DB31,DBB19 (Parameter SpindleIFNo)

Tabelle 14-7 Kanalsignale

Quelle: MCP - Tasten	Ziel: Nahtstellen-DB (Parameter ChanNo)
NC-Start	DB21,DBX7.1
NC-Stop	DB21,DBX7.3
RESET	DB21,DBX7.7
Einzelsatz	DB21,DBX0.4

Tabelle 14-8 Vorschub, Spindelsignale

Quelle: MCP - Tasten	Ziel: Ausgangsparameter vom FC
Vorschub Halt Vorschub Freigabe	Parameter: "FeedHold" speichernd verknüpft, LED werden angesteuert
Spindel Halt Spindel Freigabe	Parameter: "SpindleHold" speichernd verknüpft, LED werden angesteuert

Tabelle 14-9 Kartesisches manuelles Verfahren

Quelle: MCP - Tasten	Ziel: Nahtstellen-DB (Parameter ChanNo)
Richtungstaste R11	DB21,DBB392
Richtungstaste +	DB21,DBX20.7
Richtungstaste -	DB21,DBX20.6

Rückmeldungen der Anwender-Nahtstelle zur Ansteuerung von Anzeigen

Tabelle 14-10 Betriebsarten und Maschinenfunktionen

Ziel: MCP - LED	Quelle: Nahtstellen-DB (Parameter BAGNo) Darstellung für BAG 1
AUTOMATIC	DB11.DBX6.0
MDA	DB11.DBX6.1
JOG	DB11.DBX6.2
REPOS	DB11.DBX7.1
REF	DB11.DBX7.2
TEACH IN	DB11.DBX7.0

Ziel: MCP - LED	Quelle: Nahtstellen-DB (Parameter BAGNo) Darstellung für BAG 1
INC 1 ... 10 000, INC Var.	DB11.DBX8.0 - 8.5

Tabelle 14-11 Kanalsignale

Ziel: MCP - LED	Quelle: Nahtstellen-DB (Parameter ChanNo)
NC-Start	DB21,DBX35.0
NC-Stop	DB21,DBX35.2 oder DB21,DBX35.3
Einzelsatz	DB21,DBX0.4

Hinweis

LED der Richtungstasten werden durch Betätigen der Richtungstasten angesteuert.

Achsanwahl- und WKS/MKS-LED werden durch Betätigen der jeweiligen Taster angesteuert.

Aufrufbeispiel

```

CALL FC19(      // Maschinensteuertafel-M-Variante Signale an Nahtstelle
  BAGNo :=      B#16#1,          // BAG Nr. 1
  ChanNo :=      B#16#1,          // Kanal Nr. 1
  SpindleIFNo := B#16#4,          // Spindel Interface Number = 4
  FeedHold :=    m22.0,          // Vorschub Halt Signal selbsthaltend
  SpindleHold := db2.dbx151.0);   // Spindel Halt selbsthaltend in
                                   // Meldungs-DB

```

Mit dieser Parametrierung werden die Signale an die 1. BAG, den 1. Kanal und an alle Achsen übertragen. Zusätzlich wird der Spindel-Override in die 4. Achs-/Spindel-Nahtstelle übertragen. Das Vorschub-Halt-Signal wird an Merker 22.0 und das Spindel-Halt-Signal an den Datenbaustein DB2, Datenbit 151.0 übergeben.

Rangierung der Achsanwahlen

Um eine flexible Zuordnung der Achsanwahltasten zur jeweiligen Achse oder Spindel zu ermöglichen, ist **kein Umschreiben oder Neuschreiben** des FC19 erforderlich. Es muss nur die Achsnummer in der Achstabelle DB10.DBB8 und folgende wie gewünscht eingetragen werden. Es muss nur die Achsnummer in der Achstabelle DB10.DBB8 und folgende wie gewünscht eingetragen werden.

Beispiel

Die Spindel ist als 4. Maschinenachse definiert und soll über die Achstaste 9 angewählt werden.

Lösung:

Im DB10 Byte (8 + (9-1)) ist der Wert 4 für die 4. Achse einzutragen.

```

CALL FC19(      // Signale an Nahtstelle
  BAGNo :=      B#16#1,          // BAG Nr. 1
  ChanNo :=      B#16#1,          // Kanal Nr. 1
  SpindleIFNo := B#16#4,          // Spindel Interface Number = 4

  FeedHold :=    m30.0,          // Vorschub Halt Signal selbsthaltend
  SpindleHold := m30.1);         // Spindel Halt selbsthaltend

```

14.17.24 FC21: Transfer - Datenaustausch NC/PLC**14.17.24.1 Funktion**

Der Baustein FC21 dient zum Datenaustausch zwischen PLC und NC. Die Daten werden sofort bei Aufruf des FC21 übertragen, nicht erst mit Beginn des nächsten PLC-Grundprogrammzyklus.

Die Aktivierung des Datentransfers erfolgt durch Aufruf des Bausteins FC21 mit Parameter "Enable" = 1

Funktionen

Der Baustein stellt folgende Funktionen zur Verfügung:

- Signale der Synchronaktionen: PLC → NC-Kanal
- Signale der Synchronaktionen: NC-Kanal → PLC
- Schneller Datenaustausch PLC-NC (Funktion lesen in NC)
- Schnellen Datenaustausch PLC-NC (Funktion schreiben in NC)
- Signale an NC-Kanal aktualisieren
- Signale an Achsen aktualisieren (Datenbyte 2 der Anwender Nahtstelle)
- Signale an Achsen aktualisieren (Datenbyte 4 der Anwender Nahtstelle)

14.17.24.2 Deklaration der Funktion

Deklaration der Funktion

```

VAR_INPUT
    Enable : BOOL ;
    Funct  : BYTE ;
    S7Var  : ANY ;
    IVar1  : INT ;
    IVar2  : INT ;
END_VAR

VAR_OUTPUT
    Error  : BOOL ;
    ErrCode : INT ;
END_VAR

```

14.17.24.3 Erläuterungen der Formalparameter

Erläuterungen der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung	
Enable:	E	BOOL	0 (FALSE), 1 (TRUE)	1:	Daten übertragen
Funct:	E	BYTE	1, 2, 3, ... 7	1:	Synchronaktionen an Kanal
				2:	Synchronaktionen von Kanal
				3:	Daten lesen
				4:	Daten schreiben
				5:	Steuersignale an Kanal
				6:	Steuersignale an Achse
				7:	Steuersignale an Achse

Signal	Art	Typ	Wertebereich	Bedeutung
S7Var:	E	ANY	S7-Datenbereich	Abhängig von "Funct"
IVAR1:	E	INT	---	Abhängig von "Funct"
IVAR2:	E	INT	---	Abhängig von "Funct"
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Fehler liegt vor
ErrCode:	A	INT	---	Abhängig von "Funct"

14.17.24.4 Funktion 1, 2: Signale Synchronaktionen an / von Kanal

Synchronaktionen können von PLC verriegelt bzw. freigegeben werden.

Der Datenbereich liegt in der Anwender-Nahtstelle DB21,DBB300 ...307 (an Kanal) und DB21,DBB308 ...315 (von Kanal). Der Parameter "S7Var" wird bei dieser Funktion nicht ausgewertet, muss aber mit einem Aktualparameter belegt werden (siehe Aufrufbeispiel). Die Daten werden sofort während des FC21-Durchlaufs an/von NC übergeben.

Signal	Art	Typ	Wertebereich	Bedeutung
Enable:	E	BOOL	0 (FALSE), 1 (TRUE)	1: Daten übertragen
Funct:	E	BYTE	1, 2	1: an Kanal 2: von Kanal
S7Var:	E	ANY	S7-Datenbereich	nicht genutzt
IVAR1:	E	INT	1, 2, ... max. Kanal-Nummer	Kanal-Nummer
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1: Fehler liegt vor
ErrCode:	A	INT	1, 10	1: "Funct" ungültig 10: Kanal-Nummer ungültig

Aufrufbeispiel:

```

FUNCTION FC100: VOID
VAR_TEMP
    myAny : ANY ;
END_VAR

BEGIN
NETWORK

//Synchronaktionen mit ID3, ID10 und ID31 deaktivieren im NC-Kanal 1:
SYAK : AUF DB21;
    SET;
    S DBX300. //ID3
    2;
    S DBX301. //ID10
    1;
    S DBX303. //ID31
    6;
    L B#16#1;
    T MB11;

```

```

        SPA    TRAN;

//Synchronaktionen von NCK-Kanal 1:
SYVK : L B#16#2;
      T MB11;
TRAN: CALL FC21 (
          Enable   := M 10.0,      //wenn True, FC 21 aktiv
          Funct    := MB 11,
          S7Var    := #myAny,      //nicht genutzt
          IVAR1    := 1,           //Kanal-Nr
          IVAR2    := 0,
          Error    := M 10.1,
          ErrCode  := MW 12);

END_FUNCTION

```

14.17.24.5 Funktion 3, 4: Schneller Datenaustausch PLC-NC

Allgemeines

Für den schnellen Austausch von Informationen zwischen NC und PLC existiert als Schnittstelle ein eigener interner Datenbereich. Die Schnittstelle umfasst 4096 Bytes. Die Zugriffe (Lesen/Schreiben) von PLC aus erfolgen über FC21. Der interne Aufbau (Struktur) der Schnittstelle wird ausschließlich vom Anwender festgelegt und muss NC- und PLC-seitig identisch vereinbart werden.

Vom NC-Programm kann auf diese Daten mit den Befehlen \$A_DBB[x], \$A_DBW[x], \$A_DBD[x], \$A_DBR[x] zugegriffen werden (Literatur: Listenhandbuch Systemvariablen).

Die konkrete Adresse im Datenfeld wird über einen Byte-Offset (0 bis 4095) im Parameter "IVAR1" angegeben. Dabei muss die Ausrichtung entsprechend dem Datenformat gewählt werden, d. h. ein DWORD beginnt auf einer 4-Byte-Grenze und ein WORD auf einer 2-Byte-Grenze. Bytes können ab einem beliebigen Offset innerhalb des Datenfeldes liegen. Zugriffe auf einzelne Bits werden nicht unterstützt. Sie werden vom FC21 auf Bytezugriff umgesetzt. Die Datentypinformation und Anzahl der Daten wird aus dem über S7Var übergebenen ANY-Parameter entnommen.

Die Datenkonsistenz wird nur für 1- und 2-Byte Zugriffe sowohl von der NC als auch von der PLC ohne zusätzliche programmiertechnische Maßnahmen sichergestellt. Bei 2-Byte Konsistenz trifft dies nur beim Datentyp WORD bzw. INT zu, nicht aber beim Datentyp BYTE.

Bei Verwendung größerer Datentypen oder Übertragung von Feldern, die konsistent übertragen werden müssen, muss das Semaphor-Byte im Parameter "IVAR2" verwendet werden, über das der FC21 die Gültigkeit bzw. Konsistenz eines Blocks erkennen kann. Dieses Handling muss auf NC-Seite, d. h. im NC-Programm, durch schreiben bzw. löschen des Semaphor-Bytes unterstützt werden. Das Semaphor-Byte liegt in dem gleichen Datenbereich wie die Nutzdaten.

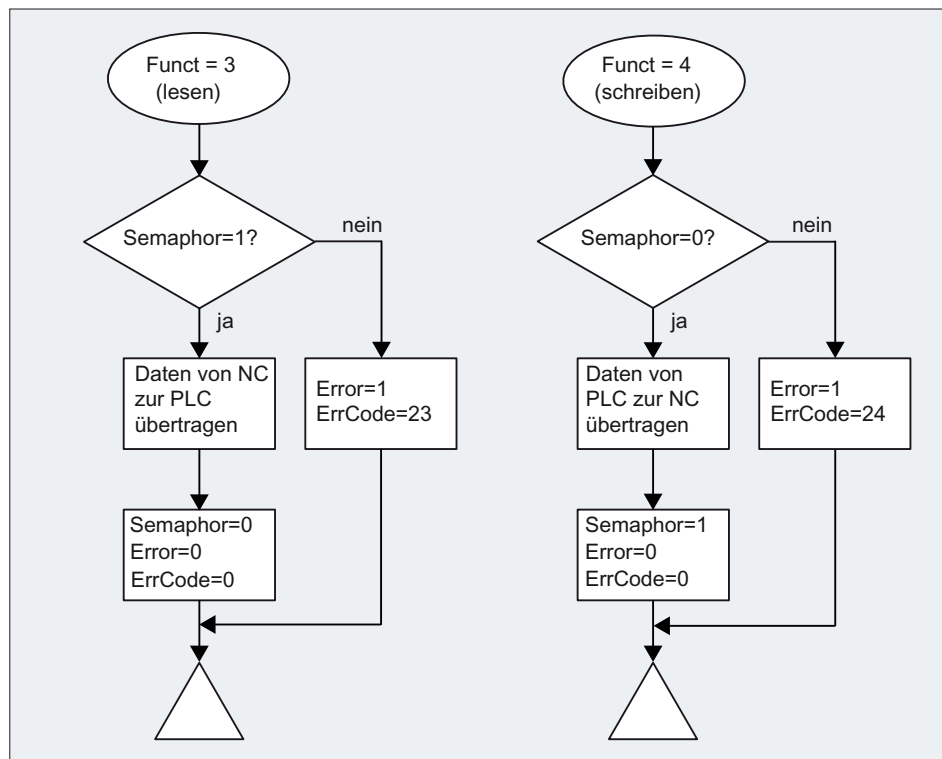
Ein Wert zwischen 0 und 4095 im "IVAR2" bezeichnet das Semaphor-Byte.

Von der PLC wird über den FC21 in dem gleichen Aufruf, bei dem die Nutzdaten übertragen werden, das Semaphorbyte gelesen und beschrieben. Der PLC-Programmierer muss nur die Semaphor-Variablen in der Schnittstelle definieren. Beim Zugriff von der NC über das NC-Programm muss der Semaphor-Mechanismus entsprechend dem unten abgebildeten Flussdiagramm über Einzelanweisungen programmiert werden. Die Sequenz muss für Lesen bzw. Schreiben von Variablen unterschiedlich ausgeführt sein.

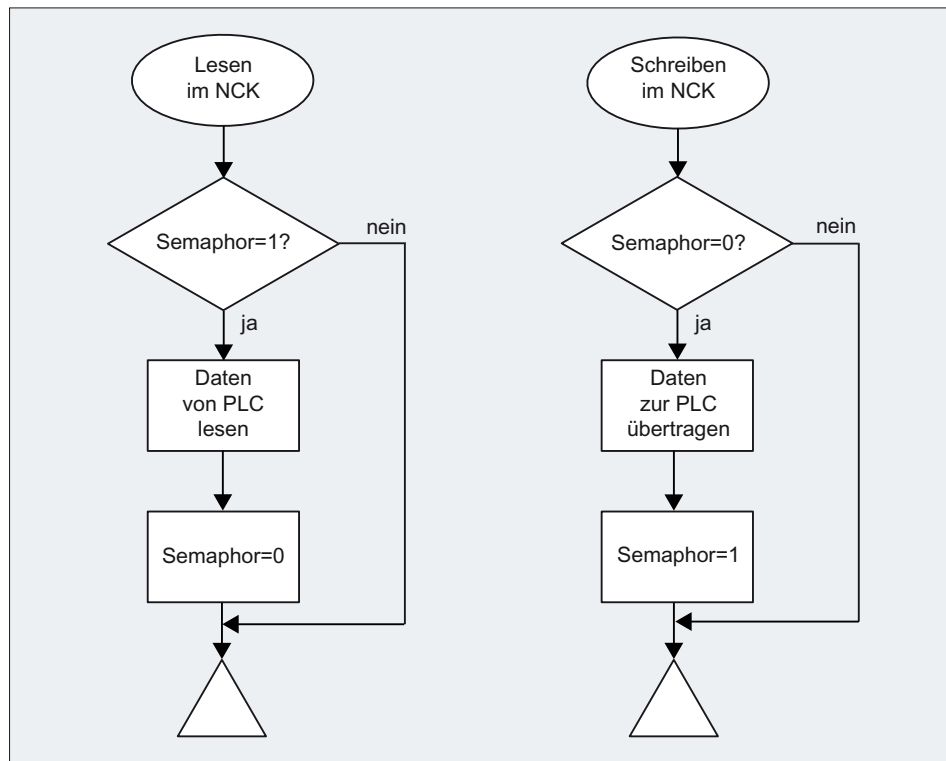
Es werden nur Einzelvariable bzw. Felder direkt durch die Semaphortechnik unterstützt. Eine Übertragung von Strukturen ist in Einzelaufträge aufzuteilen. Hierbei muss für die Datenkonsistenz dieser Struktur durch einen vom Anwender selber zu programmierenden Semaphormechanismus gesorgt werden.

Wird "IVAR2" = -1 gesetzt, erfolgt die Datenübertragung ohne Semaphor.

Datenaustausch mit Semaphor in PLC (Prinzipschaltung des FC21)



Prinzipieller Aufbau in der NC:

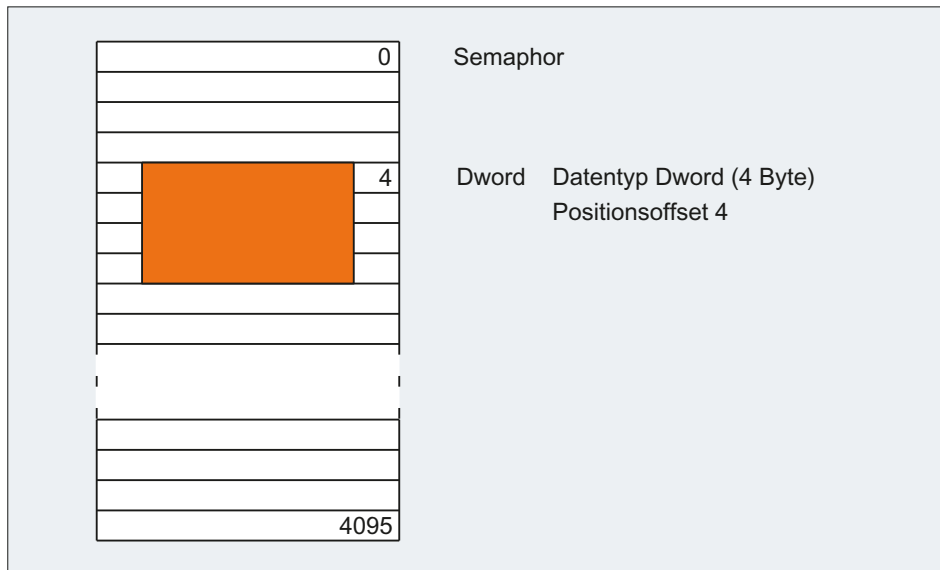


Variablen-Wertebereiche

Signal	Art	Typ	Wertebereich	Bedeutung	
Enable:	E	BOOL	0 (FALSE), 1 (TRUE)	1:	Daten übertragen
Funct:	E	BYTE	3, 4	3:	Daten lesen
				4:	Daten schreiben
S7Var:	E	ANY	S7-Datenbereich, außer Lokaldaten	Quell-/Ziel-Datenbereich	
IVAR1:	E	INT	0 ... 4095	Positionsoffset	
IVAR2:	E	INT	-1 ... 4095	Semaphor-Byte Transfer ohne Semaphor: -1	
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1:	Fehler liegt vor
ErrCode:	A	INT	20, 21, 22, 23, 24, 25	20:	Ausrichtungsfehler
				21:	unerlaubter Positionsoffset
				22:	unerlaubtes Semaphor-Byte
				23:	keine neue Daten zu lesen
				24:	kann keine Daten schreiben
				25:	Lokaldaten bei S7Var paramet-riert

Beispiel 1: Lesen eines DWORD vom Positionsoffset 4 mit Semaphore im Byte 0 und Speichern im Merker-Doppelwort 100

- Datentyp Dword (4 Byte)
- Positionsoffset 4



PLC-Programmierung

```
CALL FC21 (
    Enable      := M 10.0,           //wenn TRUE, FC21 aktiv
    Funct       := B#16#3,           //Daten lesen
    S7Var       := P#M 100.0 DWORD 1,
    IVAR1       := 4,
    IVAR2       := 0,
    Error       := M 10.1,
    ErrCode     := MW12);
UN M10.1;           //Enable so lange 1, bis Wert gelesen
R M10.0;
```

NC-Programmierung mit Synchronaktionen

- Schreiben der Daten zur PLC, Byte 0 dient als Semaphore:
ID=1 WHENEVER \$A_DBB[0] == 0 DO \$A_DBR[4] = \$AA_IM[X] \$A_DBB[0] = 1
- Lesen der Daten von der PLC, Byte1 dient als Semaphore:
ID=2 WHENEVER \$A_DBB[1] == 1 DO \$R1 = \$A_DBR[12] \$A_DBB[1] = 0

Beispiel 2: Lesen eines WORD von Positionsoffset 8 ohne Semaphore und Speichern im Merker-Wort 104

```
CALL FC21 (
    Enable      :=M 10.0,           //wenn TRUE, FC21 aktiv
```

```

Funct      :=B#16#3,           //Daten lesen
S7Var      :=P#M 104.0 WORD 1,
IVAR1      :=8,
IVAR2      :=-1,
Error      :=M 10.1,
ErrCode    :=MW12);

```

14.17.24.6 Funktion 5: Steuersignale an Kanal aktualisieren

Diese Funktion dient einer schnellen Übertragung von wichtigen Steuersignalen, zwischen der zyklischen Datenübertragung. Die Datenbytes 6 und 7 der Anwender-Nahtstelle DB21, ... werden an die NC übertragen. Der Kanal wird im Parameter "IVAR1" angegeben. Hiermit kann z. B. die Vorschubsperr, Einlesesperre außerhalb des PLC Zyklus übertragen werden.

Signal	Art	Typ	Wertebereich	Bedeutung	
Enable:	E	BOOL	0 (FALSE), 1 (TRUE)	1:	Daten übertragen
Funct:	E	BYTE	5	5:	Steuersignale an Kanal
S7Var:	E	ANY	S7-Datenbereich	nicht genutzt	
IVAR1:	E	INT	1. MaxKanal	Kanal - Nummer	
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1:	Fehler liegt vor
ErrCode:	A	INT	1, 10	1:	"Funct" ungültig
				10:	Kanal - Nr. ungültig

14.17.24.7 Funktion 6: Steuersignale an Achsen aktualisieren

Die Funktion 6 dient einer schnellen Übertragung von wichtigen Steuersignalen, zwischen der zyklischen Datenübertragung. Das **Datenbyte 2** der Anwender-Nahtstelle DB31, ... wird an die NC übertragen. Die Übertragung wird für alle aktivierten Achsen vorgenommen. Hiermit kann z. B. die Reglerfreigabe außerhalb des PLC-Zyklus übertragen werden.

Signal	Art	Typ	Wertebereich	Bedeutung	
Enable:	E	BOOL	0 (FALSE), 1 (TRUE)	1:	Daten übertragen
Funct:	E	BYTE	6	6:	Steuersignale an Achsen
S7Var:	E	ANY	S7-Datenbereich	nicht genutzt	
IVAR1:	E	INT	0		
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1:	Fehler liegt vor
ErrCode:	A	INT	1	1:	"Funct" ungültig

14.17.24.8 Funktion 7: Steuersignale an Achsen aktualisieren

Die Funktion 7 dient einer schnellen Übertragung von wichtigen Steuersignalen, zwischen der zyklischen Datenübertragung. Das **Datenbyte 4** der Anwender-Nahtstelle DB31, ... wird an die NC übertragen. Die Übertragung wird für alle aktivierten Achsen vorgenommen. Hiermit kann z. B. der Vorschubhalt außerhalb des PLC-Zyklus übertragen werden.

Signal	Art	Typ	Wertebereich	Bedeutung	
Enable:	E	BOOL	0 (FALSE), 1 (TRUE)		
Funct:	E	BYTE	7	7:	Steuersignale an Achsen
S7Var:	E	ANY	S7-Datenbereich	nicht genutzt	
IVAR1:	E	INT	0		
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1:	Fehler liegt vor
ErrCode:	A	INT	1	1:	"Funct" ungültig

14.17.25 FC22: TM_DIR - Richtungsauswahl für Werkzeugverwaltung**ACHTUNG****Verwendung**

Der Baustein FC22 "TM_DIR" darf nur in Verbindung mit der Werkzeugverwaltung verwendet werden!

Funktion

Der Baustein FC22 "TM_DIR" liefert, bezogen auf die Platznummern, z.B. eines Werkzeugkettenmagazins oder -revolvers (Teilungsachse), anhand der Ziel- und der aktuellen Position, den kürzesten Weg und die Bewegungsrichtung zur Positionierung.

Ausgänge

- Eingang FC22: "Start" = 1 ⇒ Die Ausgänge werden zyklisch aktualisiert.
- Eingang FC22: "Start" = 0 ⇒ Die Ausgänge sind undefiniert.

Sonderpositionierung

Bei Richtungsauswahl mit Sonderpositionierung (Eingang FC22: "Offset" > 0) wird eine neue Zielposition aus der Zielposition, dem Offset für Sonderpositionierung, sowie der Anzahl der Magazinplätze, berechnet:

$$\text{Neue_Zielposition} = (\text{Zielposition} - (\text{Sonderposition} - 1)) \text{ neg. MODULO } \text{Anzahl_der_Magazinplätze}$$

Die neue Zielposition entspricht der Platznummer, auf die das Magazin positioniert werden muss, damit die vom Anwender geforderte Zielposition auf der Platznummer der Sonderposition steht.

Die Richtungsoptimierung ist mit und ohne Sonderpositionierung aktiv.

Aufruf

Pro Magazin ist der Baustein einmal mit entsprechender Parametrierung aufzurufen.

Literatur

- Weitere PI-Dienste zur Werkzeugverwaltung:
 - FB4: PI_SERV - PI-Dienst anfordern (Seite 1005)
 - FC7: TM_REV - Transfer-Baustein für Werkzeugwechsel mit Revolver (Seite 1067)
 - FC8: TM_TRANS - Transfer-Baustein für Werkzeugverwaltung (Seite 1071)
- Funktionshandbuch Werkzeugverwaltung

Deklaration der Funktion

```

FUNCTION FC22: VOID
// NAME :          TM_DIR
VAR_INPUT
    MagNo :          INT;
    ReqPos :          INT;
    ActPos :          INT;
    Offset :          BYTE;
    Start :          BOOL;
END_VAR

VAR_OUTPUT
    Cw :              BOOL;
    Ccw :              BOOL;
    InPos :            BOOL;
    Diff :              INT;
    Error :             BOOL;
END_VAR
BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
MagNo:	E	INT	1, 2, 3, ...	Magazinnummer
ReqPos:	E	INT	1, 2, 3, ...	Zielposition (Magazinplatznummer)
ActPos:	E	INT	1, 2, 3, ...	Aktuelle Position (Magazinplatznummer)
Offset:	E	BYTE	0, 1, 2, ...	Offset für Sonderpositionierung
Start:	E	BOOL	0 (FALSE), 1 (TRUE)	1 = Start der Berechnung
Cw:	A	BOOL	0 (FALSE), 1 (TRUE)	1 = Magazin im Uhrzeigersinn bewegen

Signal	Art	Typ	Wertebereich	Bedeutung
Ccw:	A	BOOL	0 (FALSE), 1 (TRUE)	1 = Magazin gegen den Uhrzeigersinn bewegen
InPos:	A	BOOL	0 (FALSE), 1 (TRUE)	1 = Position erreicht
Diff:	A	INT	0, 1, 2, ...	Betrag des Differenzweges (kürzester Weg)
Error:	A	BOOL	0 (FALSE), 1 (TRUE)	1 = Fehler

Aufrufbeispiel

```

CALL FC22 (                                // Werkzeugverwaltung Richtungsauswahl
                                                // Eingänge
    MagNo :=      2,                        // Magazin Nummer
    ReqPos :=    mw 20,                     // Zielposition
    ActPos :=    mw 22,                     // Aktuelle Position
    Offset :=    b#16#0,                   // Offset für Sonderpositionierung
    Start :=     m 30.4,                    // Start-Anstoß
                                                // Ausgänge
    Cw :=         m 30.0,                   // Magazin im Uhrzeigersinn bewegen
    Ccw :=        m 30.1,                   // Magazin gegen Uhrzeigersinn bewegen
    InPos :=      m 30.2,                   // Magazin in Position
    Diff :=       mw 32,                    // Differenzweg
    Error :=      m 30.3                    // Fehler aufgetreten
);

```

14.17.26 FC24: MCP_IFM2 - Übertragung der MCP-Signale an die Nahtstelle

Funktion

Der Baustein FC24 "MCP_IFM2" (M-Variante z. B. MCP 310) dient zur Übertragung der Daten von der Maschinensteuertafel in die NC/PLC-Nahtstelle:

- Betriebsarten
- Achsanwahlen
- WKS/MKS Umschaltung
- Verfahrtasten
- Overrides bzw. Overridenachbildung
- Schlüsselschalterstellung

Für **Vorschub-Override**, **Achsfahr**- und **INC-Tasten** gelten abhängig von der aktiven Betriebsart bzw. vom angewählten Koordinatensystem folgende Festlegungen:

- **Vorschub-Override:**

- Der Vorschub-Override wird auf die Nahtstelle des angewählten Kanals und auf die Nahtstelle der Achsen transferiert.
- Die Vorschub-Override-Signale werden zusätzlich zum Nahtstellenbyte "Eilgangkorrektur" (DBB 5) an den NC-Kanal übergeben, wenn das HMI-Signal "Vorschub-Override für Eilgang wirksam" gesetzt ist (Ausnahme: Schalterstellung "Null"). Weiterhin wird mit diesem HMI-Signal auch "Eilgangkorrektur wirksam" gesetzt.

- **Maschinenfunktion INC- und Achsfahrtasten:**

- Bei angewähltem MKS werden die Signale auf die Nahtstelle der angewählten Maschinenachsetransferiert.
- Bei angewähltem WKS werden die Signale auf die Geo-Achs-Nahtstelle des parametrisierten Kanals transferiert.
- Bei Umschaltung zwischen MKS und WKS erfolgt generell eine Abwahl der bis dahin angewählten Achse.

Die Ansteuerung der zugehörigen LED's der Maschinensteuertafel wird aus der Rückmeldung entsprechender Anwahlen abgeleitet.

Vorschub- und Spindel-Start/Stopp werden nicht an die Nahtstelle übertragen, sondern als Signal "FeedHold" bzw. "SpindleHold" selbsthaltend ausgehen. Der Anwender kann diese Signale mit weiteren Signalen verknüpfen, die zu Vorschub- oder Spindel-Halt führen sollen (dies kann z. B. über die entsprechenden Eingangssignale des FC10: AL_MSG erfolgen). Zusätzlich werden die zugehörigen LEDs mit angesteuert.

Die **Spindelrichtung** (+, -) wird auch nicht direkt geschaltet, sondern als Ausgangsparameter "SpindleDir" zur Verfügung gestellt. Hierüber kann z. B. der FC18 parametrisiert werden. Zusätzlich wird eine Freigabe der Spindel geschaltet über den Parameter "SpindleHold". Eine Möglichkeit, die Spindel direkt zu bewegen, besteht darin, die Spindel über entsprechende Achsanwahl vorzuwählen und über die (Achs-) Richtungstasten zu verfahren.

Bei Ausfall der Maschinensteuertafel werden die Signale, die von dieser kommen, mit Null vorbesetzt; ebenso die Ausgangssignale "FeedHold" und "SpindleHold".

In einem PLC-Zyklus sind Mehrfachaufrufe des FC24 bzw. auch FC19, FC25, FC26 zulässig. Hierbei steuert der erste Aufruf im Zyklus die LED-Anzeigen an. Weiterhin werden im ersten Aufruf alle Aktionen des parametrisierten Bausteins durchgeführt. Bei den weiteren Aufrufen findet nur noch eine reduzierte Bearbeitung von Kanal und BAG-Schnittstelle statt. Die Geometrieachsen werden nur im ersten Aufruf des Bausteins im Zyklus mit Richtungsvorgaben versorgt.

Die Einzelsatz-An-/Abwahl wird nur durch den ersten Aufruf im Zyklus beeinflusst.

Die zweite Maschinensteuertafel kann bearbeitet werden, wenn der Parameter "BAGNo" um B#16#10 erhöht ist. Bei der Parametrierung ist die BAG Nummer in dem unteren Nibble (untere 4 Bits) enthalten.

"BAGNo" = 0 oder B#16#10 ⇒ keine Bearbeitung der BAG Signale.

"ChanNo" = 0 ⇒ keine Bearbeitung der Kanal-Signale.

Die INC Anwahlen werden in die BAG-Schnittstelle übertragen. Die Aktivierung für diese Vorgabe findet über den DB10.DBX57.0 (INC-Eingänge im BAG-Bereich aktiv) durch diesen Baustein einmalig nach Hochlauf statt.

Weiterhin können 2 Maschinensteuertafeln parallel durch diesen Baustein bearbeitet werden. Hierbei ist der Aufruf des Bausteins für die 2. Maschinensteuertafel im OB1-Zyklus zeitlich hinter den Aufruf für die 1. MCP zu setzen. Eine Unterstützung von 2 MCP ist in den Maschinensteuertafel-Bausteinen bis zu bestimmten Grenzen vorhanden (nicht unterstützt werden vom Standard gegenseitige Verriegelungen der Achsanwahlen bei gleich zugeordneten Achsen bei 2 MCP).

Schlüsselschalterstellung

Ab Softwarestand 4.5 SP2 werden auch im FC24 die Schlüsselschalter-Signale in die Anwendernahtstelle (DBX56.5 bis 7) übertragen. Diese Übertragung findet unabhängig davon statt, ob ein Schlüsselschalter an der MCP montiert ist.

Hinweis

Ergänzende Informationen siehe "FC19: MCP_IFM - Übertragung der MCP-Signale an die Nahtstelle (Seite 1102)".

Flexible Achskonfiguration

Es ist eine Flexibilität bezüglich der Zuordnung Achsanwahlen bzw. Richtungstasten von Maschinen-Achsnummern verfügbar.

Der Einsatz von 2 Maschinensteuertafeln, die gleichzeitig betrieben werden sollen, wird durch die MCP Bausteine insbesondere für den Anwendungsfall 2 Kanäle, 2 BAGs besser unterstützt. Bei den Achs-Tabellen der jeweiligen MCP ist zu beachten, dass die Achsnummern auch in der parametrisierten BAG Nummer des MCP-Bausteins angegeben sind.

Für diese Flexibilität existieren Tabellen für Achsnummern im DB10.

Für die 1. Maschinensteuertafel (MCP) beginnt die Tabelle ab dem Byte 8 (symbolischer Name: MCP1AxisTbl[1..22]) und für die 2. Maschinensteuertafel (MCP) ab dem Byte 32 (symbolischer Name: MCP2AxisTbl[1..22]). Hier sind die Maschinen-Achsnummern byteweise einzutragen. Der Eintrag 0 ist in der Achstabelle zulässig. Eine Prüfung auf eine unzulässige Achsnummer findet nicht statt und kann bei Falscheintrag zum PLC-Stop führen.

Für den **FC24** kann auch eine Begrenzung der **maximal möglichen Achsanwahlen** erfolgen. Diese Obergrenze wird für die 1. Maschinensteuertafel im DB10.DBW30 (symbolischer Name: MCP1MaxAxis) bzw. für die 2. Maschinensteuertafel im DB10.DBW54 (symbolischer Name: MCP2MaxAxis) für die jeweilige MCP eingestellt werden.

Der voreingestellte Wert ist 0, damit wirkt die konfigurierte maximale Achsanzahl. Die Achsnummern und die Begrenzung können auch dynamisch angepasst werden. Danach muss eine erneute Achsanwahl bei FC24 erfolgen. Während des Fahrens von Achsen über die jeweiligen Richtungstasten darf keine Umschaltung der Achsnummern erfolgen. Voreingestellt ist der Kompatibilitätsmodus mit den Achsnummern **1 bis 6** für beide MCP und die Begrenzung auf die konfigurierte Achsanzahl.

Deklaration der Funktion

```

FUNCTION FC24: VOID
// NAME :                      MCP_IFM2

VAR_INPUT
    BAGNo :                    BYTE;
    ChanNo :                    BYTE;
    SpindleIFNo :              BYTE;
END_VAR

VAR_OUTPUT
    FeedHold :                 BOOL;
    SpindleHold :              BOOL;
    SpindleDir :               BOOL;
END_VAR

BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
BAGNo:	E	BYTE	B#16#00 - B#16#0A	1. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
			B#16#10 - B#16#1A	2. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
ChanNo:	E	BYTE	B#16#00 - B#16#0A	Nummer des Kanals in den die Kanalsignale übertragen werden
SpindleIFNo:	E	BYTE	0 - 31 (B#16#1F)	Nummer der Achse/Spindel in welche die Spindelaten übertragen werden (Nummer der zugehörigen Maschinenachse)
FeedHold:	A	BOOL	0 (FALSE), 1 (TRUE)	Vorschub Halt von MCP, selbsthaltend
SpindleHold:	A	BOOL	0 (FALSE), 1 (TRUE)	Spindel Halt von MCP, selbsthaltend
SpindleDir:	A	BOOL	0 (FALSE), 1 (TRUE)	Spindel-Drehrichtung
				0: Drehrichtung + (links)
				1: Drehrichtung - (rechts)

Aufrufbeispiel

```

CALL FC24(                                     // schmale Maschinensteuertafel-M-Variante
                                                // Signale an Nahtstelle
    BAGNo :=      B#16#1,                     // BAG Nr. 1
    ChanNo :=     B#16#1,                     // Kanal Nr. 1
    SpindleIFNo := B#16#4,                     // Spindel Interface Number = 4

```

```

FeedHold := m22.0,           // Vorschub Halt Signal selbsthaltend
SpindleHold := db2.dbx151.0, // Spindel Halt selbsthaltend in Meldungs-
                             DB
SpindleDir := m22.1);        // Rückgabe Spindelrichtung

```

Mit dieser Parametrierung werden die Signale an die 1. BAG, den 1. Kanal und an alle Achsen übertragen. Zusätzlich wird der Spindel-Override in die 4. Achs-/Spindel-Nahtstelle übertragen. Das Vorschub-Halt-Signal wird an Merker 22.0 und das Spindel-Halt-Signal an den Datenbaustein DB2, Datenbit 151.0 übergeben. Mit der Rückmeldung der Spindelrichtung über den Parameter "SpindleDir" kann eine Richtungsvorgabe für einen zusätzlich aufzurufenden FC18 erfolgen.

14.17.27 FC25: MCP_IFT - Übertragung der MCP-/BT-Signale an die Nahtstelle

Funktion

Der Baustein FC25 "MCP_IFT" (T-Variante z. B. MCP 483) dient zur Übertragung der Daten von der Maschinensteuertafel in die NC/PLC-Nahtstelle:

- Betriebsarten
- Richtungstasten von vier Achsen
- WKS/MKS-Umschaltung
- Overrides
- Schlüsselschalter

Für **Vorschub-Override**, **Achsfahr**- und **INC-Tasten** gelten abhängig von der aktiven Betriebsart bzw. vom angewählten Koordinatensystem folgende Festlegungen:

- **Vorschub-Override:**
 - Der Vorschub-Override wird auf die Nahtstelle des angewählten Kanals und auf die Nahtstelle der Achsen transferiert.
 - Die Vorschub-Override-Signale werden zusätzlich zum Nahtstellenbyte "Eilgangkorrektur" (DBB 5) an den NC-Kanal übergeben, wenn das HMI-Signal "Vorschub-Override für Eilgang wirksam" gesetzt ist (Ausnahme: Schalterstellung "Null"). Weiterhin wird mit diesem HMI Signal auch "Eilgangkorrektur wirksam" gesetzt.
- **Maschinenfunktion INC- und Achsfahrtasten:**
 - Bei angewähltem MKS werden die Signale auf die Nahtstelle der angewählten Maschinenachse transferiert.
 - Bei angewähltem WKS werden die Signale auf die Geo-Achs-Nahtstelle des parametrisierten Kanals transferiert.

Die Ansteuerung der zugehörigen LEDs der Maschinensteuertafel wird aus der Rückmeldung entsprechender Anwahlen abgeleitet.

Vorschub- und Spindel-Start/Stopp werden nicht an die Nahtstelle übertragen, sondern als Signal "FeedHold" bzw. "SpindleHold" selbsthaltend ausgegeben. Der Anwender kann diese Signale mit weiteren Signalen verknüpfen, die zu Vorschub- oder Spindel-Halt führen sollen

(dies kann z. B. über die entsprechenden Eingangssignale des FC10: AL_MSG erfolgen). Zusätzlich werden die zugehörigen LEDs mit angesteuert.

Bei Ausfall der Maschinensteuertafel werden die Signale, die von dieser kommen, mit Null vorbesetzt; ebenso die Ausgangssignale "FeedHold" und "SpindleHold".

In einem PLC-Zyklus sind Mehrfachaufrufe des FC25 bzw. auch FC19, FC24, FC26 zulässig. Hierbei steuert der erste Aufruf im Zyklus die LED-Anzeigen an. Weiterhin werden im ersten Aufruf alle Aktionen des parametrisierten Bausteins durchgeführt. Bei den weiteren Aufrufen findet nur noch eine reduzierte Bearbeitung von Kanal und BAG-Schnittstelle statt. Die Geometrieachsen werden nur im ersten Aufruf des Bausteins im Zyklus mit Richtungsvorgaben versorgt.

Die Einzelsatz-An-/Abwahl wird nur durch den ersten Zyklus beeinflusst.

Die zweite Maschinensteuertafel kann bearbeitet werden, wenn der Parameter "BAGNo" um B#16#10 erhöht ist. Bei der Parametrierung ist die BAG Nummer in dem unteren Nibble (untere 4 Bits) enthalten.

"BAGNo" = 0 oder B#16#10 \Rightarrow keine Bearbeitung der BAG Signale.

"ChanNo" = 0 \Rightarrow keine Bearbeitung der Kanal-Signale.

Flexible Achskonfiguration

Es ist eine Flexibilität bezüglich der Zuordnung Achsanwahlen bzw. Richtungstasten von Maschinenachsnummern verfügbar.

Der Einsatz von zwei gleichzeitig betriebenen Maschinensteuertafeln wird durch die MCP Bausteine, insbesondere für den Anwendungsfall "2 Kanäle, 2 BAG" unterstützt. Hierbei ist der Aufruf des Bausteins für die zweite Maschinensteuertafel im OB1-Zyklus zeitlich hinter den Aufruf für die 1. MCP zu setzen. Bei den Achstabellen der jeweiligen MCP ist zu beachten, dass die Achsnummern auch in der parametrisierten BAG Nummer des MCP-Bausteins angegeben sind.

Für diese Flexibilität existieren Tabellen für Achsnummern im DB10.

Für die 1. Maschinensteuertafel (MCP) beginnt die Tabelle ab Byte 8 (symbolischer Name: MCP1AxisTbl[1..22]) und für die 2. Maschinensteuertafel (MCP) ab Byte 32 (symbolischer Name: MCP2AxisTbl[1..22]). Hier sind die Maschinenachsnummern byteweise einzutragen. Der Eintrag 0 ist in der Achstabelle zulässig. Eine Prüfung auf eine unzulässige Achsnummer findet nicht statt und kann bei Falscheintrag zum PLC-Stop führen.

Die Begrenzung der **möglichen Achsanwahlen bei FC25** erfolgt über die 0-Werte in der Achstabelle. Die Achsnummern können auch dynamisch angepasst werden. Während des manuellen Verfahrens von Achsen über Richtungstasten, darf keine Umschaltung der Achsnummern erfolgen. Voreingestellt ist der Kompatibilitätsmodus mit den Achsnummern **1 bis 4** für beide MCP und die Begrenzung auf die konfigurierte Achsanzahl.

Hinweis

Ergänzende Informationen siehe "FC19: MCP_IFM - Übertragung der MCP-Signale an die Nahtstelle (Seite 1102)".

Deklaration der Funktion

```

FUNCTION FC25: VOID
// NAME :                MCP_IPT

VAR_INPUT
    BAGNo :                BYTE;
    ChanNo :                BYTE;
    SpindleIFNo :          BYTE;
END_VAR

VAR_OUTPUT
    FeedHold :              BOOL;
    SpindleHold :           BOOL;
END_VAR

BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
BAGNo:	E	BYTE	B#16#00 - B#16#0A	1. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
			B#16#10 - B#16#1A	2. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
ChanNo:	E	BYTE	B#16#00 - B#16#0A	Nummer des Kanals in welchen die Kanalsignale übertragen werden
SpindleIFNo:	E	BYTE	B#16#00 - B#16#1F	Nummer der Achse/Spindel in welche die Spindeldaten übertragen werden (Nummer der zugehörigen Maschinenachse)
FeedHold:	A	BOOL	0 (FALSE), 1 (TRUE)	Vorschub Halt von MCP, selbsthaltend
SpindleHold:	A	BOOL	0 (FALSE), 1 (TRUE)	Spindel Halt von MCP, selbsthaltend

Aufrufbeispiel

Mit dieser beispielhaften Parametrierung werden die Signale an die 1. BAG, den 1. Kanal und an alle Achsen übertragen. Der Spindel-Override wird in die Nahtstelle der 4. Achs-/Spindel übertragen. Vorschub-Halt wird in den Merker 22.0 und Spindel-Halt an den Datenbaustein DB2, DBX151.0 übertragen.

```

CALL FC25(                                // Maschinensteuertafel-T-Variante
                                                // Signale an Nahtstelle
    BAGNo :=          B#16#1,              // BAG Nr. 1
    ChanNo :=          B#16#1,              // Kanal Nr. 1
    SpindleIFNo :=     B#16#4,              // Spindel Interface Number = 4

```



```

FeedHold :=      m22.0,           // Vorschub Halt Signal selbsthaltend
SpindleHold :=  db2.dbx151.0); // Spindel Halt selbsthaltend in Meldungs-DB

```

14.17.28 FC26: HPU_MCP - Übertragung der HT 8-Signale an die Nahtstelle

Deklaration der Funktion

```

FUNCTION FC26: VOID
//NAME :          HPU_MCP

VAR_INPUT
    BAGNo :        BYTE;
    ChanNo :        BYTE;
END_VAR

BEGIN
END_FUNCTION

```

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
BAGNo:	E	BYTE	B#16#00 - B#16#0A	1. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
			B#16#10 - B#16#1A	2. MCP: Nummer der BAG in welche die Betriebsartensignale übertragen werden
ChanNo:	E	BYTE	B#16#00 - B#16#0A	Nummer des Kanals in den die Kanalsignale übertragen werden

Aufrufbeispiele

Aufruf des FC26 für die erste MCP, die erste BAG und den ersten Kanal der NC.

```

CALL FC26(                                     //Maschinensteuertafel des HT 8
    BAGNo :=      B#16#01,                     //1.MCP, 1.BAG
    ChanNo :=      B#16#01);                   //1.Kanal

```

Aufruf des FC26 für die zweite MCP, die zweite BAG und den dritten Kanal der NC.

```

CALL FC26(                                     //Maschinensteuertafel des HT 8
    BAGNo :=      B#16#12,                     //2.MCP, 2.BAG
    ChanNo :=      B#16#03 );                  //3.Kanal

```

Allgemeine Funktionsbeschreibung

Die Funktion FC26 "HPU_MCP (Maschinensteuertafel-Signale des Bedienhandgerätes HT 8)" überträgt die HT 8-spezifischen Signale der folgenden Funktionen zwischen dem im Funktionsbaustein FB1 parametrisierten HT 8-Ein/Ausgangsdatenbereich (Parameter: MCPxIn und MCPxOut) und der NC/PLC-Nahtstelle:

- Betriebsarten
- Maschinenfunktion INC
- Koordinatensystem WKS oder MKS
- Axiale Verfahrstasten
- Achsanwahl
- Vorschub-Override
- Eilgang-Override
- Schlüsselschalter-Informationen

Hinweis

Betriebsartenumschaltung durch HT 8 und/oder HMI

Die Funktion FC2 "GP_HP - Grundprogramm, zyklischer Teil" überträgt die Signale der Betriebsarten-Umschaltung prinzipiell so, dass eine wahlweise Anwahl vom MCP des HT 8 und des HMI möglich ist. Die Übertragung der HMI-Signale an die NC/PLC-Nahtstelle kann im Funktionsbaustein FB1 mit Parameter "MMCToIF" = FALSE aber auch ausgeschaltet werden.

Aktive Achsen

Über das HT 8 können maximal 6 Achsen gleichzeitig angesprochen werden. Die Auswahl der Achsen ist durch den Anwender/Maschinenhersteller im PLC-Anwenderprogramm zu realisieren.

Flexible Achskonfiguration

Die Funktion FC26 ermöglicht eine flexible Zuordnung der Maschinenachsen zu den Verfahrstasten bzw. zur Achsanwahl. Im DB10 stehen dazu 2 Tabellen zur Verfügung:

- Maschinenachstabelle 1. MCP: DB10.DBB8 bis DBB13 (Tabelle der Maschinenachsnr.)
Symbolischer Name: MCP1AxisTbl[1..22]
- Maschinenachstabelle 2. MCP: DB10.DBB32 bis DBB37 (Tabelle der Maschinenachsnr.)
Symbolischer Name: MCP2AxisTbl[1..22]

In die Tabellen sind die Achsnummern n (mit n = 1, 2, ...) der aktiven Maschinenachsen byteweise einzutragen. Bei nicht verwendeten Tabellenplätzen muss der Wert 0 eingetragen werden.

Die Tabellenlänge kann dem FC26 vorgegeben werden:

- 1. MCP: DB10.DBB30 (Obergrenze der Maschinenachstabelle)
- 2. MCP: DB10.DBB54 (Obergrenze der Maschinenachstabelle)

Ein Wert von z. B. 4 bedeutet, dass vom FC26 nur die ersten 4 Tabelleneinträge bzw. Maschinenachsen berücksichtigt werden. Der Maximalwert für den FC26 ist 6. Bei Wert 0 oder Werten größer 6 wird implizit der Maximalwert genommen.

Hinweis

Folgende Randbedingungen sind zu beachten:

- Eine Überprüfung auf zulässige Maschinenachsennummern findet nicht statt. Ungültige Maschinenachsennummer können zum Stopp der PLC führen.
- Die Maschinenachsennummern können dynamisch geändert werden. Das Beschreiben der Tabelle darf nicht durchgeführt werden, wenn eine Maschinenachse aktuell über eine Verfahrtaste verfahren wird.

Übertragung der Verfahrtastensignale in Abhängigkeit vom aktiven Koordinatensystem

Die Verfahrtastensignale für 6 Achsen liegen im HT 8-Eingangsdatenbereich unter:

- EB n + 2, Bit 0 - Bit 5 (positive Verfahrrichtung)
- EB n + 3, Bit 0 - Bit 5 (negative Verfahrrichtung)

Das Umschalten des Koordinatensystems erfolgt über das Eingangssignal:

- EB n + 0, Bit 0 (MKS/WKS)

Das Eingangssignal wird im FC26 mittels Flankenmerker ausgewertet. Das aktive Koordinatensystem wird in folgendem Ausgangssignal angezeigt:

- AB n + 0, Bit 0 (MKS/WKS) mit 0 = MKS, 1 = WKS

Bei aktivem MKS werden die Verfahrtastensignale der Achsen 1 - 6 in die achsspezifischen Nahtstellen (DB31,DBX4.6 und DBX4.7 (Verfahrtasten +/-)) der in den Maschinenachstabellen (DB10.DBB8 bis DBB13 bzw. DBB32 bis DBB37) angegebenen Achsen übertragen.

Bei aktivem WKS wird davon ausgegangen, dass die Achsen 1 - 3 der Maschinenachstabelle Geometrieachsen sind. Daher werden die Verfahrtastensignale:

- der Achsen **1 - 3** (EB n + 2 / 3, Bit 0 - Bit 2) in die Nahtstelle der Geometrieachsen in DB21,DBB12 + (n * 4), mit n = 0, 1, 2), Bit 6 und Bit 7 (Verfahrtasten +/-) des über Parameter "ChanNo" angegebenen Kanals übertragen.
Die Zuordnung der Verfahrtastensignale der Achsen 1, 2 und 3 auf die Geometrieachsen 1, 2 und 3 des Kanals ist fest und kann nicht verändert werden.
- der Achsen **4 - 6** (EB n + 2 / 3, Bit 3 - Bit 5) in die achsspezifischen Nahtstellen (DB31,DBX4.6 und DBX4.7 (Verfahrtasten +/-)) der in der Maschinenachstabelle eingetragenen Achsen 4 - 6 (DB10.DBB11 bis DBB13 bzw. DBB35 bis DBB37) übertragen.

Kein Verfahren von Maschinenachsen im WKS

Bei aktivem WKS (AB n + 0, Bit 0 = 1) kann das Verfahren von Maschinenachsen verriegelt werden. Dazu sind im PLC-Anwenderprogramm folgende Ausgangssignale zu setzen:

- AB n + 3, Bit 7 = 1 (Bei WKS: keine Maschinenachsen)
Anforderung an den FC26, keine Verfahrtastensignale für Maschinenachsen zu übertragen. Die Verfahrtastensignale für die Achsen 1 - 3 der Maschinenachstabelle werden auf die Geometrieachsen 1 - 3 des angegebenen Kanals übertragen. Die Verfahrtastensignale für die Achsen 4 - 6 der Maschinenachstabelle werden nicht übertragen.
- AB n + 2, Bit 6 (Achsen 7 - n angewählt)
Anforderung an den FC26 keine Verfahrtastensignale zu übertragen, da die Achsen 1 - 6 der Maschinenachstabelle umgeschaltet wurden. Die Achsen 1 - 3 somit keine Geometrieachsen sondern ebenfalls Maschinenachsen sind.

Vorschub-Override

Der Wert des HT 8-Override-Schalters wird als Vorschub-Override in die kanalspezifische Nahtstelle DB21,DBB4 (Vorschub-Override) des programmierten Kanals (Parameter: "ChanNo") und in die achsspezifische Nahtstellen DB31,DBB0 (Vorschub-Override) der in der Tabelle DB10.DBB8 bis DBB13 (Maschinenachsnummer) programmierten Achsen übertragen.

Eilgang-Override

Ist für den programmierten Kanals (Parameter: "ChanNo") das Signal DB21,DBX25.3 = 1 (Vorschub-Override für Eilgang) gesetzt, wird der Wert des HT 8-Override-Schalters als Eilgang-Override in dessen kanalspez. Nahtstelle in DB21,DBB5 (Eilgangkorrektur) übertragen und zusätzlich das Signal DB21,DBX6.6 = 1 (Eilgangkorrektur wirksam) gesetzt.

Maschinenfunktion INC

Die HT 8-Signale der Maschinenfunktion INC werden abhängig vom aktiven Koordinatensystem MKS oder WKS unterschiedlich übertragen:

- Aktives Koordinatensystem: MKS
Die angewählte Maschinenfunktion INC wird für alle 6 Achsen in die achsspez. Nahtstellen in DB31,DBX5.0 bis DBX5.5 (Maschinenfunktion) der in der Tabelle in DB10.DBB8 bis DBB13 (Maschinenachsnummer) programmierten Achsen übertragen.
- Aktives Koordinatensystem: WKS
Für die Achsen 1 bis 3 werden die Signale der Maschinenfunktion INC in die kanalspez. Nahtstelle in DB21,DBX13.0 bis DBX13.5 (Maschinenfunktion) des programmierten Kanals (Parameter: "ChanNo") übertragen.
Für die Achsen 4 bis 6 werden die Signale der Maschinenfunktion INC in die achsspez. Nahtstellen DB31,DBX5.0 bis DBX5.5 (Maschinenfunktion) der in der Tabelle in DB10.DBB11 bis DBB13 (Maschinenachsnummer) programmierten Achsen übertragen.

Die Anwahlsignale der INC-Maschinenfunktionen werden in die BAG-spezifische Nahtstelle DB11.DBB 2 + (n * 20), Bit 0 bis Bit 5 (mit n = 0, 1, 2, ...) übertragen. Der FC26 informiert den NC über die Aktivierung der BAG-Nahtstelle für die INC-Maschinenfunktionen einmalig nach dem Hochlauf über DB10.DBX57.0 (INC-Eingänge im BAG-Bereich aktiv).

Handradanwahl

Die Handradanwahlsignale werden vom HMI ausgewertet und in die entsprechenden NC/PLC-Nahtstellensignale der Maschinen- oder Geometrieachsen übertragen:

- Geometrieachsen: DB21, ... DBB12 + (n * 4), Bit 0 bis Bit 2 (mit n = 0, 1, 2)
- Maschinenachsen: DB31, ... DBX4.0 bis DBX4.2

Voraussetzung: FB1-Parameter: "HWheelMMC" = TRUE

Mehrfachaufruf in einem PLC-Zyklus

In einem PLC-Zyklus sind Mehrfachaufrufe des FC26 zulässig. Beim ersten Aufruf im PLC-Zyklus werden:

- Alle Aktionen des parametrisierten Bausteins durchgeführt
- Die LED-Signale im Ausgangsbereich geschrieben
- Bei angewähltem WKS, die Verfahrtastensignale der Geometrieachsen geschrieben
- Die Signale für Einzelsatz-An- und Abwahl bearbeitet

Bei weiteren Aufrufen des FC26 erfolgt nur noch eine reduzierte Bearbeitung der Kanal- und BAG-Schnittstelle.

Bearbeitung von zwei MCP

Wird die Funktion FC26 im zyklischen Ablauf des PLC-Programms (Organisationsbaustein OB1) zweimal für zwei MCP aufgerufen, muss der Aufruf für die zweite MCP zeitlich nach dem Aufruf für die erste MCP erfolgen.

Hinweis

Kann eine Achse von zwei MCP aus verfahren werden, liegt die Realisierung einer gegenseitigen Verriegelung in der Verantwortung des Anwenders (Maschinenherstellers).

Ausfall der MCP des HT 8

Bei Ausfall der MCP des HT 8 werden alle Eingangssignale auf den Wert 0 gesetzt.

14.17.28.1 Übersicht der NC/PLC-Nahtstellensignale von HT 8

Betriebsarten und Maschinenfunktionen

Quelle: MCP	Ziel: Programmierte BAG (Parameter BAGNo) Darstellung für BAG 1
AUTOMATIC	DB11.DBX0.0
MDA	DB11.DBX0.1
JOG	DB11.DBX0.2

Quelle: MCP	Ziel: Programmierte BAG (Parameter BAGNo) Darstellung für BAG 1
REPOS	DB11.DBX1.1
REF	DB11.DBX1.2
TEACH IN	DB11.DBX1.0
INC 1 ... 10 000, INC Var.	DB11.DBX2.0 - DBX 2.5

Verfahrtasten und Eilgang-Override

Quelle: MCP	Ziel: Geometrie-Achsen des prog. Kanals (Parameter: ChanNo)
Verfahrtaste +	DB21,DBX12.7
Verfahrtaste -	DB21,DBX12.6
Eilgang-Override	DB21,DBX12.5
Verfahrtaste +	DB21,DBX16.7
Verfahrtaste -	DB21,DBX16.6
Eilgang-Override	DB21,DBX16.5
Verfahrtaste +	DB21,DBX20.7
Verfahrtaste -	DB21,DBX20.6
Eilgang-Override	DB21,DBX20.5

Quelle: MCP	Ziel: Prog. Achsen entsprechend der Tabelle in DB10, DBB8 - 13 (1. MCP) bzw. DBB32 - 37 (2. MCP)
Verfahrtaste +	DB31,DBX4.7
Verfahrtaste -	DB31,DBX4.6
Eilgang-Override	DB31,DBX4.5

Override

Quelle: MCP	Ziel: Programmierter Kanal (Parameter: ChanNo)
Vorschub-Override	DB21,DBB4

Quelle: MCP	Ziel: Prog. Achsen entsprechend der Tabelle in DB10, DBB8 - 13 (1. MCP) bzw. DBB32 - 37 (2. MCP)
Vorschub-Override	DB31,DBB0

Kanalsignale

Quelle: MCP	Ziel: Programmierter Kanal (Parameter: ChanNo)
NC-Start	DB21,DBX7.1
NC-Stop	DB21,DBX7.3
RESET	DB21,DBX7.7
Einzelsatz	DB21,DBX0.4

14.17.28.2 Übersicht der NC/PLC-Nahtstellensignale an HT 8

Betriebsarten und Maschinenfunktionen

Ziel: MCP	Quelle: Nahtstellen-DB (Parameter BAGNo) Darstellung für BAG 1
AUTOMATIC	DB11.DBX6.0
MDA	DB11.DBX6.1
JOG	DB11.DBX6.2
REPOS	DB11.DBX7.1
REF	DB11.DBX7.2
TEACH IN	DB11.DBX7.0

14.17.29 FC19, FC24, FC25, FC26 - Beschreibung Quellcode

Aufgabe

Maschinensteuertafel an Anwendernahtstelle (FC19 M-Variante, FC24 schmale Variante, FC25 T-Variante, FC26 HT 8-Variante)

Zugehörige Bausteine

DB7, Anzahl BAG, Kanäle, Achsen
 DB7, Pointer der Maschinensteuertafeln,
 DB8, Ablage für den nächsten Zyklus

Verwendete Ressourcen

Keine.

Allgemeines

Die Bausteine **FC19** (M-Version), **FC24** (schmale Version), **FC25** (T-Version) und **FC26** (HT 8-Version) übertragen die Signale der Maschinensteuertafel von und zur Anwendernahtstelle.

Parameter

- Eingangs-Parameter "BAGNo"
 Mit dem Parameter "BAGNo" wird die BAG ausgewählt, die vom Baustein bearbeitet wird. Über den Parameter "BAGNo" wird auch die Nummer der Maschinensteuertafel selektiert (Bit 4).
- Eingangs-Parameter "ChanNo"
 Mit dem Parameter "ChanNo" wird der Kanal ausgewählt, der bearbeitet wird.

- **Eingangs-Parameter "SpindleIFNo" (Nicht FC26)**
Im Parameter "SpindleIFNo" wird die Nummer der Achse/Spindel (Nummer der zugehörigen Maschinenachse) angegeben, in deren Nahtstelle die Spindelaten übertragen werden. Der Spindel-Override wird auf diese Spindelnahtstelle transferiert. Die Eingangsparameter werden bezüglich Falschparametrierung überprüft.
- **Ausgangs-Parameter "FeedHold" und "SpindleHold" (Nicht FC26)**
Die Ausgangsparameter "FeedHold" und "SpindleHold" werden aus den 4 Tasten Vorschub/Spindel-Sperre, Vorschub/Spindel-Freigabe gebildet. Bei aktivierter Vorschub/Spindel-Sperre liefert der Parameter den Wert "logisch 1"..

Informationen für den nächsten Zyklus werden abhängig von der Maschinensteuertafel-Nummer im DB8, Byte 0 bis 3 bzw. Byte 62 bis 65 abgespeichert. Diese Informationen sind Flankenmarker, Vorschubwert und angewählte Achsnummer. Die Bausteine werden mit Nutzdaten durch die Pointer-Parameter im DB7 "MCP1In" und "MCP1Out" ("MCP2In" und "MCP2Out") versorgt. Die Pointer werden indirekt über einen weiteren Zeiger aus dem VAR-Teil des DB7 adressiert um die Absolutadressierung zu vermeiden. Dieser zusätzliche Zeiger ist im FB1 symbolisch ermittelt worden.

Bausteinbeschreibung

Alle 4 Bausteine sind ähnlich strukturiert aufgebaut und für die einzelnen Teilaufgaben gegliedert:

- **Netzwerk Input**
Im Netzwerk Input werden verschiedene Parameter in lokale Variablen kopiert. Hierbei werden auch die Maschinensteuersignale (Nutzdaten Ein-/Ausgangsbereich) über die verschiedenen Zeiger im DB7 umkopiert. Mit diesen lokalen Variablen wird aus Effizienzgründen im Baustein gearbeitet. Für den Anlauf werden einige Werte initialisiert.
- **Netzwerk Global_IN**
Im Netzwerk Global_IN wird die MKS/WKS Umschaltung mit Flankenauswertung, Achsanwahlen, Richtungstasten und Eilgangüberlagerung für die weitere Bearbeitung im Baustein ermittelt. In diesem Programmteil sollten anwenderspezifische Veränderungen stattfinden, die sich hauptsächlich an den Achsanwahlen orientieren.
- **Netzwerk NC**
Im Netzwerk NC werden nur die Schlüsselschalter-Informationen kopiert.
- **Netzwerk BAG**
Das Netzwerk BAG transferiert die Betriebsarten der Taster als dynamische Signale zum NC. Die INC Rückmeldungen der NC werden temporär für die entsprechenden LEDs abgelegt. Bei BAG-Nummer 0 wird dieses Netzwerk nicht bearbeitet. Einer zu großen Nummer hat eine Meldung 401901 bzw. 402501 zur Folge und schaltet nach Stop um.
- **Netzwerk Channel**
Im Netzwerk Channel werden die Funktionen NC-Start, Stop, Reset und Einzelsatz mit entsprechenden Rückmeldungen aktiviert. Die Richtungstasten der GEO-Achsen werden bei entsprechender Vorwahl versorgt, anderenfalls entsorgt. Bei Kanal-Nummer 0 wird dieses Netzwerk nicht bearbeitet. Eine zu große Nummer hat eine Meldung 401902 bzw. 402502 zur Folge und es wird nach Stopp geschaltet.
- **Netzwerk Spindle**
Das Netzwerk Spindle überträgt den Spindeloverride in die über "SpindleIFNo" parametrisierte Nahtstelle.

- **Netzwerk Axes**
Das Netzwerk Axes überträgt den Vorschub-Override auf die angewählte Achsnahtstelle. Die Richtungstasten werden der angewählten Achse/Spindel zugeordnet. Bei einer vorher angewählten Achse wird die Richtungsinformation auf 0 gesetzt.
- **Netzwerk Global_OUT**
Im Netzwerk Global_OUT werden die Ausgangsparameter aufbereitet und die LED-Signale der INC-Maschinenfunktion gebildet.
- **Netzwerk Output**
Das Netzwerk Output überträgt die Ausgangssignale der Maschinensteuertafel vom Abbild der VAR_TEMP auf die logische Adresse. Weiterhin werden die Daten für den nächsten Zyklus gerettet.

Erweiterung Achsanwahl

Netzwerk Global_IN

Im Netzwerk Global_IN ist für eine Anwahl von mehr als 9 Achsen einzugreifen. Falls hier weitere Tasten und LED der Maschinensteuertafel verwendet werden, sind folgende Maßnahmen zu treffen:

1. Der Befehl UD DW#16#Wert (Kommentar: Clear all axis LED's for display) löscht alle definierten LEDs für Achsanwahlen. Derzeit werden mit der Bitmaske genau die 9 Achsanwahl-LEDs bearbeitet.
2. Der Befehl UW W#16# (Kommentar: "Masking all the axis selection buttons") prüft ob eine neue Richtungsanwahl erfolgte. Hier muss die Bitleiste angepasst werden.
3. Die Sprungleiste (SPL) ist mit neuen Sprungmarken zu erweitern. Die neuen Sprungmarken sind vor der Marke m009 einzufügen in abwärtszählender Reihenfolge. Bei den neuen Sprungmarken sind Anwahlinformationen, wie bei Marke m009, m008 dargestellt, zu erweitern.

Hinweis

Die Bausteine werden als AWL-Quellen bei Bedarf zur Verfügung gestellt. Sie entsprechen aber nicht zwingend dem aktuellen Stand der Bausteine. Die aktuelle Implementierung in C ist in einigen Details schon weiter entwickelt. Es ist sinnvoll die gewünschten zusätzlichen Anforderungen an die Bausteine zu spezifizieren und diese Anforderung über den Vertrieb an die Projektleitung zu geben.

14.17.30 FC1005: AG_SEND - Übergibt Daten an Ethernet-CP

Funktion

Der Funktionsbaustein AG_SEND übergibt an den Ethernet-CP die über eine projektierte Verbindung zu übertragenden Daten. Zusammen mit dem Funktionsbaustein AG_RECV kann so über die integrierte "CP 840D sl" ein Datenaustausch mit einer weiteren Station aufgebaut werden. Diese Station muss dazu in STEP 7, "NetPro" projektiert werden.

Im Grundprogramm ist diese Funktion als Funktionsbaustein FC1005 verfügbar. Dieser entspricht annähernd dem Funktionsbaustein FC5 in der Bibliothek "SIMATIC_NET_CP".

Unterstützt werden die Protokolle TCP, UDP und ISO-on-TCP.

Hinweis

Geringere Anzahl übertragbarer Daten bei SINUMERIK CP und UDP- oder ISO-on-TCP-Protokoll

Bei SINUMERIK CP können mit dem Funktionsbaustein FC1005 bei Verwendung des UDP- oder ISO-on-TCP-Protokolls nur **240** Byte übertragen werden.

Erläuterung der Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
ACT:	E	BOOL	0 (FALSE), 1 (TRUE)	Auftragsanstoß ¹⁾
ID:	E	INT	1, 2, 3 ... 16	Verbindungs-ID
LADDR:	E	WORD	-	Baugruppen-Anfangsadresse ²⁾
SEND:	E	ANY	-	Angabe von Adresse und Länge. Die Adresse des Datenbereichs verweist alternativ auf: <ul style="list-style-type: none"> • Merkerbereich • Datenbausteinbereich
LEN:	E	INT	TCP: 1, 2, ... 8192 UDP: 1, 2, ... 240 ISO-on-TCP: 1, 2, ... 240	Anzahl Bytes, die übertragen werden
DONE:	A	BOOL	0 (FALSE), 1 (TRUE)	0: Auftrag läuft 1: Auftrag ausgeführt
ERROR:	A	BOOL	0 (FALSE), 1 (TRUE)	0: kein Fehler 1: Fehler liegt vor
STATUS:	A	WORD	-	Statusanzeige
1) Der Parameter ACT muss so lange TRUE sein, bis gilt: (DONE == 1) ODER (ERROR == 1)				
2) Bei SINUMERIK 840D sl: Parameter LADDR := W#16#8110				

Dokumentation

Eine ausführliche Bausteinbeschreibung findet sich in:

- SINUMERIK Bedienoberfläche: Online-Hilfe
- SIMATIC Programmierhandbuch: Programmbausteine für SIMATIC NET S7-CPs
Kapitel: "Programmbausteine für Industrial Ethernet" > "Programmbausteine für Offene Kommunikationsdienste (SEND/RECEIVE-Schnittstelle)" > "AG_SEND / AG_LSEND / AG_SSEND"

14.17.31 FC1006: AG_RECV - Empfängt Daten von Ethernet-CP

Funktion

Der Funktionsbaustein AG_RECV übernimmt vom Ethernet-CP die über eine projektierte Verbindung übertragenen Daten. Zusammen mit dem Funktionsbaustein AG_SEND kann so

über die integrierte "CP 840D sl" ein Datenaustausch mit einer weiteren Station aufgebaut werden. Diese Station muss dazu in STEP 7, "NetPro" projektiert werden.

Im Grundprogramm ist diese Funktion als Funktionsbaustein FC1006 verfügbar. Dieser entspricht annähernd dem Funktionsbaustein FC6 in der Bibliothek "SIMATIC_NET_CP".

Unterstützt werden die Protokolle TCP, UDP und ISO-on-TCP.

Hinweis

Geringere Anzahl übertragbarer Daten bei SINUMERIK CP und UDP- oder ISO-on-TCP-Protokoll

Bei SINUMERIK CP können mit dem Funktionsbaustein FC1006 bei Verwendung des UDP- oder ISO-on-TCP-Protokolls nur **240** Byte übertragen werden.

Formalparameter

Signal	Art	Typ	Wertebereich	Bedeutung
ID:	E	INT	1, 2 ...16	Verbindungs-ID
LADDR:	E	WORD	-	Baugruppen-Anfangsadresse ¹⁾
RECV:	E	ANY	-	Angabe von Adresse und Länge. Die Adresse des Datenbereichs verweist alternativ auf: <ul style="list-style-type: none"> • Merkerbereich • Datenbausteinbereich
NDR:	A	BOOL	0 (FALSE), 1 (TRUE)	0: Auftrag läuft 1: Neue Daten übernommen
ERROR:	A	BOOL	0 (FALSE), 1 (TRUE)	0: kein Fehler 1: Fehler liegt vor
STATUS:	A	WORD	-	Statusanzeige
LEN:	A	INT	TCP: 1, 2, ... 8192 UDP: 1, 2, ... 240 ISO-on-TCP: 1, 2, ... 240	Anzahl Bytes, die übertragen werden

1) Bei SINUMERIK 840D sl: Parameter LADDR := **W#16#8110**.

Dokumentation

Eine ausführliche Bausteinbeschreibung findet sich in:

- SINUMERIK Bedienoberfläche: Online-Hilfe
- SIMATIC Programmierhandbuch: Programmbausteine für SIMATIC NET S7-CPs
Kapitel: "Programmbausteine für Industrial Ethernet" > "Programmbausteine für Offene Kommunikationsdienste (SEND/RECEIVE-Schnittstelle)" > "AG_RECV / AG_LRECV / AG_SRECV"

14.18 Signal-/Datenbeschreibungen

14.18.1 Nahtstellensignale NCK/PLC, HMI/PLC, MSTT/PLC

Literatur

Ein vollständiger Überblick aller Nahtstellensignale findet sich in:

Listenhandbuch "NC-Variable und Nahtstellensignale"

Die Nahtstellensignale der NC, die vom PLC-Grundprogramm ausgewertet und aufbereitet zur Anwendernahtstelle übergeben werden, sind in den nachfolgenden Kapiteln aufgeführt.

14.18.2 Dekodierte M-Signale

Die im Teileprogramm, ASUP oder Synchronaktionen programmierten M-Funktionen werden von der NC kanalspezifisch an die PLC übertragen:

- M-Funktionen aus Kanal 1: DB21
- M-Funktionen aus Kanal 2: DB22
- ...

DBX1, ...	Bedeutung
DBX194.0 ... 7	M-Funktionen M0 ... M7
DBX195.0 ... 7	M-Funktionen M8 ... M15
DBX196.0 ... 7	M-Funktionen M16 ... M23
DBX197.0 ... 7	M-Funktionen M24 ... M31
DBX198.0 ... 7	M-Funktionen M32 ... M39
DBX199.0 ... 7	M-Funktionen M40 ... M47
DBX200.0 ... 7	M-Funktionen M48 ... M55
DBX201.0 ... 7	M-Funktionen M56 ... M63
DBX202.0 ... 7	M-Funktionen M64 ... M71
DBX203.0 ... 7	M-Funktionen M72 ... M79
DBX204.0 ... 7	M-Funktionen M80 ... M87
DBX205.0 ... 7	M-Funktionen M88 ... M95
DBX206.0 ... 3	M-Funktionen M96 ... M99

Das Signal für die jeweilige M-Funktion bleibt mindesten für einen PLC-Zyklus in der NC/PLC-Nahtstelle anstehen.

Hinweis

Folgende spindelspezifischen M-Funktionen werden nicht an die NC/PLC-Nahtstelle übertragen: M3, M4, M5 und M70.

Teileprogrammende M02/M30

Die Ausgabe der M-Funktionen M02/M30 besagt nicht sicher, dass das Teileprogramm beendet wurde. Die Ausgabe der M-Funktionen M02/M30 könnte aus einem asynchronen Unterprogramm (ASUP) oder einer Synchronaktion stammen und hat in diesen Fällen nichts mit dem tatsächlichen Teileprogrammende zu tun. Um das Ende eines Teileprogramms im Kanal sicher zu erkennen, muss das folgende Nahtstellensignal ausgewertet werden:

DB21, ... DBX35.7 (Kanalzustand: Reset)

14.18.3 G-Befehle

Die im Teileprogramm, ASUP oder Synchronaktionen programmierten G-Befehle werden von der NC kanalspezifisch an die PLC übertragen:

- G-Befehle aus Kanal 1: DB21
- G-Befehle aus Kanal 2: DB22
- ...

DB21, ...	Bedeutung
DBB208	Aktiver G-Befehl der G-Gruppe 1
DBB209	Aktiver G-Befehl der G-Gruppe 2
DBB210	Aktiver G-Befehl der G-Gruppe 3
...	...
DBB271	Aktiver G-Befehl der G-Gruppe 64

Der jeweilige G-Befehl bleibt mindesten für einen PLC-Zyklus in der NC/PLC-Nahtstelle anstehen.

Adresse im DB21, ...

Die Adresse DBBx einer G-Gruppe im DB21, ... berechnet sich zu:

Byte-Nummer = 208 + "Nummer der G-Gruppen" - 1

Grundstellung nach Power On

Nach Power On wird für alle G-Gruppen der Wert Null, d. h. aktive G-Gruppe undefiniert, übertragen.

Teileprogramm-Ende bzw. -Abbruch

Nach Teileprogramm-Ende bzw. -Abbruch bleibt der zuletzt aktive G-Befehl erhalten.

NC-START

Nach NC-START werden die Werte der im folgenden Maschinendatum angegebenen 8 G-Gruppen, entsprechend der über Maschinendaten festgelegten Grundstellung, sowie der im Teileprogramm programmierten Werte überschrieben:

MD22510 \$MC_GCODE_GROUPS_TO_PLC

Literatur

Eine Übersicht aller G-Befehle findet sich in:

Programmierhandbuch Grundlagen; Kapitel: "Tabellen" > "G-Befehle"

14.18.4 Meldesignale im DB2

Über den DB2 können für einzelne Signale Meldungen auf der Bedientafelfront ausgegeben werden. Die Signale sind in vordefinierte Gruppen unterteilt. Bei Auftreten oder Verschwinden bzw. Quittieren einer Meldung wird die in der Spalte Meldenummer aufgeführte Nummer zum HMI übertragen. Zu jeder Meldenummer kann im HMI ein Text hinterlegt werden.

Literatur:

- Listenhandbuch "NC-Variable und Nahtstellensignale"; Kapitel "Nahtstellensignale - Übersicht" > "PLC-Alarme/Meldungen"
- Inbetriebnahmehandbuch; Kapitel "Alarm- und Meldungstexte"

Hinweis

Die Anzahl der Anwenderbereiche kann über den FB1 parametrierbar werden.

Nach Änderung der Konfiguration (FB1: MsgUser) müssen DB2 / DB3 gelöscht werden.

14.19 Hinweise zur Programmierung in STEP 7

In den nachfolgenden Kapiteln werden Hinweise bezüglich der Vereinfachung der Programmierung komplexer Abläufe und Funktionen in STEP 7 gegeben:

- Daten kopieren (Seite 1139)
- Datentypen ANY und POINTER (Seite 1139)
- Multi-Instanz-DB (Seite 1144)
- Strings (Seite 1145)
- Ermittlung von Offset-Adressen auf Datenbaustein-Strukturen (Seite 1146)
- Aufrufe von FBs (Seite 1147)

Literatur

Grundlegende Hinweise zum Aufbau der Datentypen ANY und POINTER finden sich in:

SIMATIC STEP 7-Handbuch; Kapitel: "Entwerfen von Anwenderprogrammen" > "Register der CPU und Speichern von Daten"

14.19.1 Daten kopieren

Kopiervarianten

Für das schnelle Kopieren von Daten von einem DB in einen anderen empfiehlt sich:

- **bei größeren Datenmengen** der Einsatz der Systemfunktionen SFC BLKMOV bzw. SFC FILL, da hier ein wesentlich schnelleres Kopieren stattfinden kann.
- **bei kleineren Datenmengen** die untenstehende Routine, da die Versorgung der ANY-Parameter an den SFCs zusätzliche Zeit kostet.

Beispiel

```

// DB xx.[AR1] ist Quelle
// DI yy.[AR2] ist Ziel
AUF      DB 100;      //Quell DB
LAR1     P#20.0;      //Anfangsadresse Quelle auf Datenbyte 20
AUF      DI 101;      //Ziel DB
LAR2     P#50.0;      //Anfangsadresse Ziel auf Datenbyte 50
//AR1, AR2, DB, DI vorher geladen

L        4;          //8 Byte transferieren

M001:
L        DBW [AR1,P#0.0]; //Wortweise kopieren
T        DIW [AR2,P#0.0];
+AR1     P#2.0;
+AR2     P#2.0;
TAK;
LOOP     M001;

```

14.19.2 ANY und POINTER

In den folgenden Programmbeispielen sollen Programmiermechanismen dargestellt werden. Hierbei soll der Zugriff auf Ein-/Aus- und DurchgangsvARIABLEN (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT) vom Datentyp POINTER bzw. ANY innerhalb eines FC bzw. FB aufgezeigt werden. Die Zugriffe sind so dargestellt, dass eine teilsymbolische Programmierform verwendet werden kann.

14.19.2.1 Verwendung von POINTER und ANY im FC

Funktion

Der FC99 hat Eingangsparameter, die als POINTER bzw. ANY definiert sind.

In dem Beispiel wird ein Rumpfprogramm gezeigt, um auf die Teilkomponenten des POINTER bzw. ANY zuzugreifen. Hierbei wird der über den POINTER, ANY parametrisierte DB

aufgeschlagen und der Adressoffset als bereichsübergreifender Zeiger im Adressregister AR1 hinterlegt. Somit kann auf Datenelemente der Variablen (in der Regel Strukturen, Arrays) zugegriffen werden, die über den POINTER, ANY adressiert sind.

Dieser Zugriff ist im Beispiel am Ende der jeweiligen Programmsequenz dargestellt. Bei dem Datentyp ANY besteht zusätzlich die Möglichkeit, anhand des Datentyps und der Anzahl von Elementen eine Überprüfung bzw. Verzweigung beim Zugriff der Variablen durchzuführen.

Beispiel

```

FUNCTION FC99: VOID
VAR_INPUT
    Row : BYTE ;
    Convert : BOOL ;           //Zahlenwandlung aktivieren
    Addr : POINTER;           //Zeigt auf Variable
    Addr1 : ANY;
END_VAR
VAR_TEMP
    dbchr : WORD ;
    Anzahl : WORD ;
    typ : BYTE;
END_VAR
BEGIN
NETWORK
TITLE =

                                //POINTER
    L          P##Addr;
    LAR1 ;                      //Pointer holen
    L          W [AR1,P#0.0];   //DB-Nummer holen
    T          #dbchr;
    L          D [AR1,P#2.0];   //Offsetteil des Pointers
    LAR1 ;
    AUF DB [#dbchr];           //DB der Variablen aufschlagen
    L B [AR1,P#40.0];          //hole Byte-Wert über Pointer mit dem
                                //Adressoffset 40
                                //ANY

    L          P##Addr1;
    LAR1 ;                      //ANY holen
    L          B [AR1,P#1.0];   //Typ hole
    T          #typ;
    L          W [AR1,P#2.0];   //Anzahl holen
    T          #Anzahl;
    L          W [AR1,P#4.0];   //DB-Nummer holen
    T          #dbchr;
    L          D [AR1,P#6.0];   //Offsetteil des Pointers
    LAR1 ;

```



```

AUF          DB [#dbchr];          //DB der Variablen aufschlagen
L            B [AR1,P#0.0];        //hole Byte-Wert über ANY

```

14.19.2.2 Verwendung von POINTER und ANY im FB

Funktion

Der FB99 hat Eingangsparameter, die als POINTER bzw. ANY definiert sind.

In dem Beispiel wird ein Rumpfprogramm gezeigt um auf die Teilkomponenten des POINTER bzw. ANY zuzugreifen. Hierbei wird der über den POINTER, ANY parametrisierte DB aufgeschlagen und der Adressoffset als bereichsübergreifender Zeiger im Adressregister AR1 hinterlegt. Somit kann auf Datenelemente der Variable (in der Regel Strukturen, Arrays) zugegriffen werden, die über den POINTER, ANY adressiert sind.

Dieser Zugriff ist im Beispiel am Ende der jeweiligen Programmsequenz dargestellt. Bei dem Datentyp ANY besteht zusätzlich die Möglichkeit anhand des Datentyps und der Anzahl von Elementen eine Überprüfung bzw. Verzweigung beim Zugriff der Variablen durchzuführen.

Beispiel

```

FUNCTIONBLOCK FB99
VAR_INPUT
    Row : BYTE ;
    Convert : BOOL ;                //Zahlenwandlung aktivieren
    Addr : POINTER ;               //Zeigt auf Variable
    Addr1 : ANY;
END_VAR
VAR_TEMP
    dbchr : WORD ;
    Anzahl : WORD ;
    typ : BYTE;
END_VAR
BEGIN
NETWORK
TITLE =

                                //POINTER
L            P##Addr;
LAR1;                          //Pointer holen aus Instanz-DB
L            DIW [AR1,P#0.0];    //DB-Nummer holen
T            #dbchr;
L            DID [AR1,P#2.0];    //Offsetteil des Pointers
LAR1 ;
AUF          DB [#dbchr];        //DB der Variablen aufschlagen
L            B [AR1,P#40.0];     //hole Byte-Wert über Pointer mit
                                dem
                                //Adressoffset 40

```

```

//ANY
L          P##Addr1;
LAR1 ;
//ANY holen aus Instanz-DB
L          DIB [AR1,P#1.0]; //Typ holen
T          #typ;
L          DIW [AR1,P#2.0]; //Anzahl holen
T          #Anzahl;
L          DIW [AR1,P#4.0]; //DB-Nummer holen
T          #dbchr;
L          DID [AR1,P#6.0]; //Offsetteil des Pointers
LAR1 ;
AUF        DB [#dbchr]; //DB der Variablen aufschlagen
L          B [AR1,P#0.0]; //hole Byte-Wert über ANY

```

14.19.2.3 Variable POINTER bzw. ANY für Übergabe an FC oder FB

Variable POINTER bzw. ANY

In STEP 7 ab Version 1 besteht die Möglichkeit, einen POINTER bzw. ANY in der VAR_TEMP zu definieren.

Die Versorgung eines ANY soll im Folgenden an 2 Beispielen dargestellt werden.

Beispiel 1: ANY-Parameter über eine Auswahlliste an einen anderen FB (FC) übergeben

An einem FB (FC) sind **mehrere** ANY-Parameter definiert. Es soll nun über eine Auswahlliste ein bestimmter ANY-Parameter an einen anderen FB (FC) übergeben werden. Dieses funktioniert nur über einen ANY in der VAR_TEMP. Im Parameter "WelcherAny" kann 1 bis 4 parametrisiert sein, um Addr1 bis Addr4 auszuwählen.

Hinweis

Im Baustein wird das Adressregister AR2 verwendet. Dieses Adressregister AR2 wird bei auch bei Multi-Instanz-DB verwendet. Deshalb darf dieser FB **nicht** als Multi-Instanz-DB erklärt sein.

```

FUNCTIONBLOCK FB100
CODE_VERSION1 //ab STEP 7 Version 2, um Multi-Instanz-DB
//zu deaktivieren

VAR_INPUT
WelcherAny : INT;
  Addr1 : ANY; //feste Reihenfolge beachten
  Addr2 : ANY;
  Addr3 : ANY;
  Addr4 : ANY;
END_VAR

```

```

VAR_TEMP
    dbchr : WORD ;
    Anzahl : WORD ;
    typ : BYTE;
    Temp_addr : ANY;
END_VAR
BEGIN
NETWORK
TITLE =
L      WelcherAny;
DEC 1;
L      P#10.0;           //10 Byte pro ANY
*I;
LAR2;
L      P##Addr1;
+AR2;                   //Anfangsadresse der ANYs addieren
L      P##Temp_addr;
LAR1 ;                  //Pointer holen aus VAR_TEMP
L      DID [AR2,P#0.0];  //Zeigerwerte übertragen in VAR_TEM
T      LD [AR1,P#0.0];
L      DID [AR2,P#4.0];
T      LD [AR1,P#4.0];
L      DIW [AR2,P#8.0];
T      LW [AR1,P#8.0];

CALL FB101, DB100
      (ANYPAR := #Temp_addr); //ANYPAR ist der Datentyp ANY

```

Beispiel 2: Vorher zusammengebauten ANY-Parameter an einen anderen FB (FC) übergeben

Es soll **ein** ANY-Parameter, der vorher zusammengebaut wurde, an einen anderen FB (FC) übergeben werden. Dieses funktioniert nur über einen ANY in der VAR_TEMP.

```

FUNCTIONBLOCK FB100
VAR_INPUT
    DBNummer: INT;
    DBOffset : INT;
    Datentyp: INT;
    Anzahl: INT;
END_VAR
VAR_TEMP
    dbchr : WORD ;
    Temp_addr : ANY;
END_VAR
BEGIN
NETWORK

```

```

TITLE =
L      P##Temp_addr;
LAR1 ;                                //Pointer holen aus VAR_TEMP
L      B#16#10;                        //Kennung ANY
T      LB [AR1,P#0.0];
L      Datentyp;
T      LB [AR1,P#1.0];
L      Anzahl;
T      LW [AR1,P#2.0];
L      DBNummer;
T      LW [AR1,P#4.0];
L      DBOffset;
SLD 3;                                //Offset ist ein Bitoffset
T      LD [AR1,P#6.0];
CALL FB101, DB100
      (ANYPAR := #Temp_addr); //ANYPAR ist der Datentyp ANY

```

14.19.3 Multi-Instanz-DB

Funktion

In STEP 7 ab Version 2 können FBs Multi-Instanz-fähig, d. h. mit Multi-Instanz-DBs versehen sein. Multi-Instanz-DBs zeichnen sich dadurch aus, dass ein Datenbaustein für verschiedene Instanzen von FBs verwendet werden kann (siehe Dokumentation STEP 7). Somit kann das Mengengerüst der DBs optimiert werden.

Das Aktivieren von Multi-Instanz-DBs sollte nur dann erfolgen, wenn diese auch genutzt werden, da sich Laufzeit und Codegröße der FBs erhöhen.

Hinweis

Bei komplexeren Programmen in FBs, die Zeiger und Adressregister verwenden, sind bei FBs, die Multi-Instanz-fähig sein sollen, vom Programmierer bestimmte Regeln einzuhalten.

Bei Multi-Instanzen wird die Anfangsadresse der Variablen (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) mit dem DI-Datenbaustein-Register und dem Adressregister AR2 übergeben. Bei Zugriffen innerhalb des Multi-Instanz-fähigen FB regelt der Compiler die Zugriffe auf diese Variablen über das Adressregister AR2 selbstständig. Wenn allerdings komplexere Programmteile in diesem FB auch mit Adressregistern arbeiten müssen (z. B. Daten kopieren), dann ist vor dem Ändern des AR2 der bisherige Inhalt zu retten. Das AR2-Register muss vor einem Zugriff auf eine Instanz-Variable (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) wieder mit dem ursprünglichen Inhalt restauriert sein. Das AR2-Register der Instanz ist am sinnvollsten in eine Lokal-Variable (VAR_TEMP) zu retten.

Der Befehl "Zeiger auf eine Instanzvariable laden" liefert einen Zeigerwert ab dem Beginn der Instanzdaten. Um auf diese Variable über Zeiger zugreifen zu können, ist der Offset, der im AR2 steht, hinzuzurechnen.

Beispiel

```

FUNCTION_BLOCK FB99
VAR_INPUT
    varin: INT;
END_VAR
VAR
    variable1: ARRAY[0..9] OF
        INT;
    variable2: INT;
END_VAR
BEGIN
    L          P##variable1;    //Pointer auf Anfang des ARRAYS
                                //Im Accu steht jetzt der Wert 8500 0010
                                //Im AR2 steht auch ein Bereichs-übergreifender
                                //Zeiger. Wenn Bereichs-übergreifend gearbeitet
                                //werden soll, dann ist bei der Addition dieser
                                //beiden Zeiger ein Bereich auszublenden.
    UD          DW#16#00FF_FFFF, //Ausblenden des Bereichs
    LAR1
                                //Ins AR1 laden
    TAR2;
    +AR1 AR2;                    //AR2 Instanz Offset hinzuaddieren
                                //Jetzt kann über AR1 indirekt auf das ARRAY
                                //von variable1 zugegriffen werden.
    L          DIW [AR1, P#0.0]; //Z. B. Zugriff auf erstes Element
END_FUNCTION_BLOCK

```

14.19.4 Strings

Der Datentyp STRING wird durch bestimmte Dienste des Grundprogramms benötigt. Aus diesem Grund sollen ein paar zusätzliche Informationen zum Aufbau von Strings und dem generellen Umgang bei Parameterzuweisungen beschrieben werden.

Aufbau von Strings

Ein Datum mit dem Datentyp STRING ist generell in einem Datenbaustein abgelegt (definiert). Bei der Definition gibt es 2 Varianten:

1. Einer Variablen wird nur der Datentyp STRING zugewiesen. Hierbei generiert der STEP 7-Compiler eine Länge von 254 Zeichen.
2. Einer Variablen wird der Datentyp STRING mit einer Längenangabe in eckigen Klammern zugewiesen (z. B. [32]). Hierbei generiert der STEP 7-Compiler eine String-Länge entsprechend der Vorgabe.

Für eine Variable vom Datentyp STRING werden immer 2 Byte mehr abgelegt, als die Definition ausweist. Hierbei ist in dem 1. Byte die Anzahl der maximal möglichen Zeichen durch den STEP 7 -Compiler abgelegt. Im 2. Byte ist die Anzahl der genutzten Zeichen enthalten.

Normalerweise wird hier die Nutzlänge des zugewiesenen Strings durch den STEP 7-Compiler abgelegt. Ab dem 3. Byte sind die Zeichen abgelegt (1 Byte pro Zeichen).

Eine Zuweisung von STRING-Parametern an Bausteine des Grundprogramms erfolgt generell über den Datentyp POINTER bzw. ANY. Die Zuweisung soll generell über symbolische Programmierung erfolgen. Hierbei ist der Datenbaustein, in dem sich der zu parametrierende String befindet, in die Symbolliste aufzunehmen. Anschließend ist die Zuweisung an den Grundprogramm-Baustein mit symbolischen Namen des Datenbausteins, anschließenden Punkt und symbolischen Namen der STRING-Variable durchzuführen.

14.19.5 Ermittlung von Offsetadressen auf Datenbaustein-Strukturen

Funktion

Eine weitere häufig vorkommende Aufgabe ist die symbolische Ermittlung einer Offsetadresse innerhalb eines strukturierten DBs. In diesem DB liegt z. B. an beliebiger Stelle ein ARRAY oder eine STRUKTUR. Man möchte, nachdem das Adressregister symbolisch mit der Anfangsadresse geladen wurde, über ein Adressregister auf einzelne Elemente des ARRAY oder der STRUKTUR zugreifen. Ein Weg wie man das Adressregister symbolisch laden kann, führt über einen FC, der als Eingangsparameter einen Pointer hat. Diesem Eingangsparameter des FC wird nun im Programm symbolisch die Adresse des ARRAY oder der STRUKTUR zugewiesen. Der Programmcode im FC ermittelt nun die Offsetadresse aus dem Eingangsparameter und übergibt die Offsetadresse im Adressregister (AR1) an die rufende Funktion. Somit ist auch bei indirekten Zugriffen eine Symbolische Adressierung möglich.

Beispiel

```
FUNCTION FC99: VOID
VAR_INPUT
    Addr : POINTER ;           //Zeigt auf Variable
END_VAR
BEGIN
NETWORK
TITLE =
L      P##Addr;
LAR1 ;                         //Pointer holen von Addr
L      D [AR1,P#2.0];         //Offsetteil des Pointers der Variable
LAR1 ;
END_FUNCTION
```

14.19.6 Aufrufe von FBs

Funktion

Zur Optimierung von Ablaufgeschwindigkeiten ist es sinnvoll, alle Funktionsbausteinanrufe mit vielen statischen Parametern, wie z. B. die vom Grundprogramm zur Verfügung gestellten Bausteine FB2, 3, 4, 5 und 7, schon im Anlauf mit den zugehörigen Instanzparametern aufzurufen. Im Anlauf (OB100) sollte die Vorbesetzung von Parametern erfolgen, die anschließend im zyklischen Teil (OB1) nicht mehr verändert werden. Diese festen Parameterwerte werden in dem zyklischen Aufruf nicht mehr parametrisiert, da sie bereits im Anlauf in den Instanz-DB geschrieben worden sind.

Beispiel: Parametrierung des FB2 mit dem Instanz-DB110

Am folgenden Beispiel soll gezeigt werden, wie eine sinnvolle Aufteilung in OB100-Anteil und OB1-Anteil zu realisieren ist.

Als erstes ist der übliche Aufruf im zyklischen Programm dargestellt:

```
CALL FB2, DB110(
    Req :=      M 100.0,
    NumVar :=    2,                               //2 Variablen lesen
    Addr1 :=    NCVAR.C1_RP_rpa0_0
    Line1 :     W#16#1
    Addr2 :=    NCVAR.C1_RP_rpa0_0
    Line2 .     W#16#2
    Error :=    M1.0,
    NDR :=      M1.1,
    State :=    MW 2,
    RD1 :=      P#M 4.0 REAL 1,
    RD2 :=      P#M 24.0 REAL 1,
```

Ab hier folgt die modifizierte Version des Programmaufrufs.
Hier ist der Aufruf im OB100 dargestellt:

```
CALL FB2, DB110(
    Req :=      FALSE,
    NumVar :=    2,                               //2 Variablen lesen
    Addr1 :=    NCVAR.C1_RP_rpa0_0
    Line1 :     W#16#1
    Addr2 :=    NCVAR.C1_RP_rpa0_0
    Line2 .     W#16#2
    RD1 :=      P#M 4.0 REAL 1,
    RD2 :=      P#M 24.0 REAL 1,
```

Hier ist der noch verbleibende Aufruf im OB1 dargestellt:

```
CALL FB2, DB110 (
    Req :=      M0.0,
    Error :=     M1.0,
    NDR :=      M1.1,
    State :=     MW 2,
```

Hinweis

Durch diese Maßnahme wird eine kürzere Durchlaufzeit im OB1 erreicht, da die statischen Parameterwerte nicht in jedem OB1-Durchlauf in den Instanz-DB kopiert werden.

Ersparnis dieser Variante:

Der zyklische Kopieraufwand von 3 Integerwerten und 4 ANY-Parametern bezogen auf den Instanz-DB, welches aus 3-mal Laden einer Konstante mit einen 3-maligen Transfer in den Instanz-Datenbaustein resultiert. Bei jedem ANY-Transfer werden 4-mal Konstanten mit anschließendem Transfer in den Datenbaustein geladen.

14.20 Datenlisten

14.20.1 Maschinendaten

14.20.1.1 Anzeige-Maschinendaten

Nummer	Bezeichner: \$MM_	Beschreibung
9032	HMI_MONITOR	PLC-Datum für HMI-Bildinfo festlegen

14.20.1.2 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10100	PLC_CYCLIC_TIMEOUT	Zyklische PLC-Überwachungszeit
14504	MAXNUM_USER_DATA_INT	Anzahl der Anwenderdaten (INT)
14506	MAXNUM_USER_DATA_HEX	Anzahl der Anwenderdaten (HEX)
14508	MAXNUM_USER_DATA_FLOAT	Anzahl der Anwenderdaten (FLOAT)
14510	USER_DATA_INT	Anwenderdatum (INT)
14512	USER_DATA_HEX	Anwenderdatum (HEX)
14514	USER_DATA_FLOAT[n]	Anwenderdatum (FLOAT)

Hinweis

Ein Maschinendatum im Integer-/Hexformat wird in der NC als DWORD gehandhabt. Ein Maschinendatum im Fließkomma-Format wird in der NC als FLOAT (8 Byte IEEE) gehandhabt. Sie werden jeweils in der NC/PLC-Nahtstelle abgelegt und können vom PLC-Anwenderprogramm bereits im Hochlauf der PLC aus dem DB20 gelesen werden.

14.20.1.3 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
28150	MM_NUM_VDIVAR_ELEMENTS	Anzahl Elemente für das Schreiben von PLC-Variablen

14.20.2 Signale**14.20.2.1 Signale von Bedientafel**

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Aktiver SINUMERIK-Bedienbereich	DB19.DB21	DB1900.DBB1
Aktuelle Bildnummer	DB19.DBW24	DB1900.DBW4

P4: PLC für SINUMERIK 828D

15.1 Überblick

15.1.1 PLC-Firmware

Die PLC der SINUMERIK 828D ist eine integrierte PLC auf Basis des Befehlssatzes der SIMATIC S7-200.

Die Programmierung der PLC-Anwenderprogramme erfolgt in erster Linie an einem Windows-PC mit dem "PLC Programming Tool". Zusätzlich kann die PLC auch über die Bedienoberfläche der Steuerung diagnostiziert und editiert werden. Hierzu steht ein "Ladder-Add-On-Tool" auf der Steuerung zur Verfügung.

Folgende Besonderheiten sind zu beachten:

- Die Programmierung des PLC-Anwenderprogramms erfolgt durchgehend in Kontaktplan (KOP).
- Es wird eine Untermenge der Programmiersprache S7-200 unterstützt.
- Beim Laden in CPU werden neben dem ablaufrelevanten Code auch die kompletten Projektdaten (einschließlich Symbolik und Kommentaren) in die Steuerung geladen. Damit enthält die Steuerung immer das passende Projekt zum aktuellen laufenden PLC-Anwenderprogramm.
- Beim Laden aus CPU werden die kompletten Projektdaten (einschließlich Symbolik und Kommentaren) in das PLC Programming Tool geladen und sind mit diesem bearbeitbar.
- Der Anwender muss seine Daten und Prozessinformationen typgerecht verwalten. Bei allen Zugriffen auf die Daten muss konsequent der vereinbarte Datentyp verwendet werden.

15.1.2 PLC-Anwendernahtstelle

Die Anwendernahtstelle wird von der PLC-Firmware eingerichtet, welche auch den Austausch aller Signale und Daten zwischen der PLC einerseits und NC und HMI andererseits organisiert.

Die Anwendernahtstelle besteht aus den Teilen:

- Datenschnittstelle mit zyklischem Austausch (siehe "Datenschnittstelle (Seite 1158)")
- Funktionsschnittstelle mit funktions- bzw. auftragsbezogenem Datenaustausch (siehe "Funktionsschnittstelle (Seite 1172)")

Die strukturierten Daten dieser Schnittstellen (remanente und nichtremanente) werden von der Firmware durch Zuordnung auf Datenbausteine dem Anwender zur Verfügung gestellt: "Kommunikationspartner" des PLC-Anwenderprogramms sind die NC (NC, Werkzeugverwaltung, NC-Kanal, Achsen, Spindeln, ...) und das HMI.

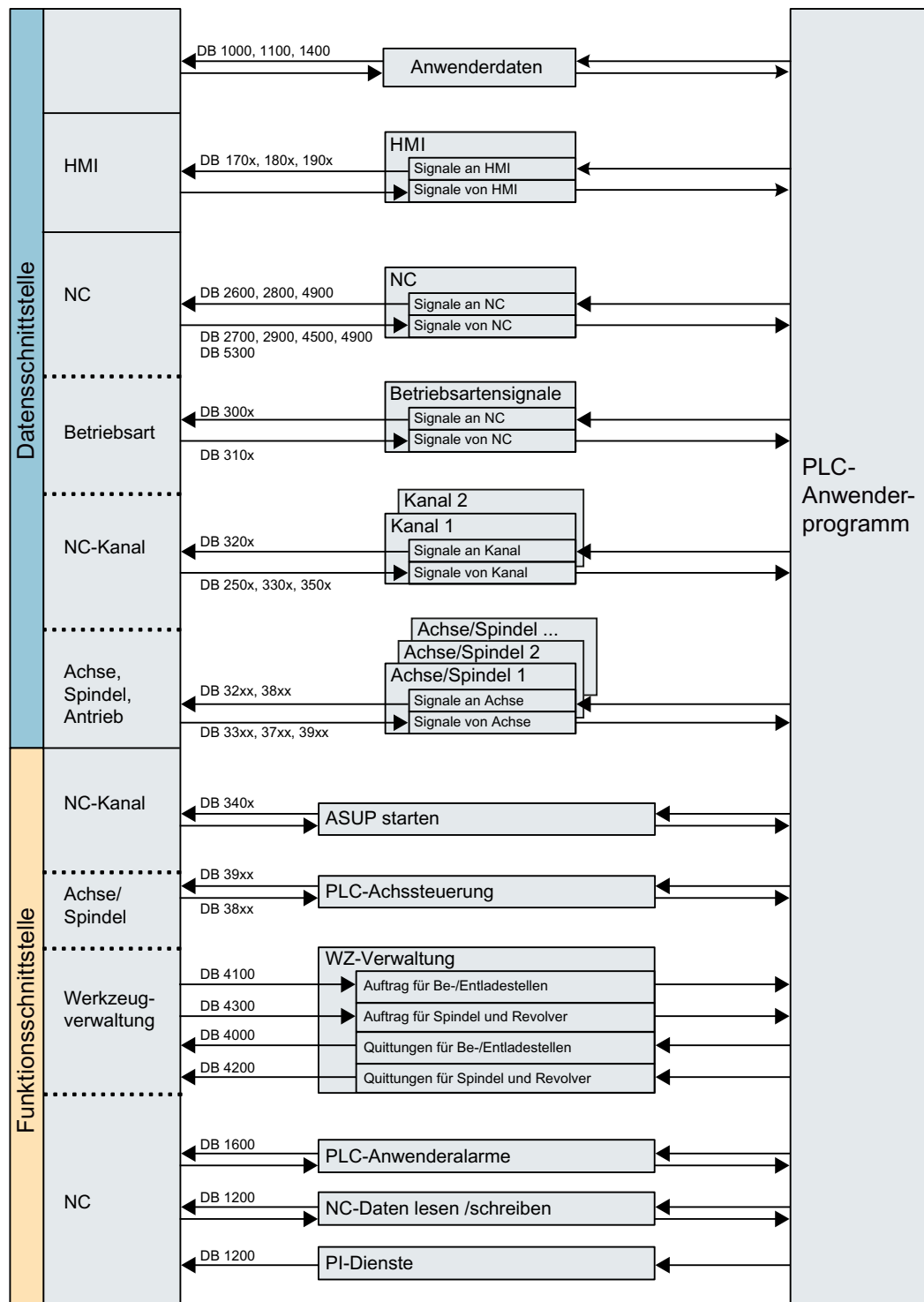


Bild 15-1 Überblick über die Anwendernahstelle der PLC 828D

15.1.2.1 Zyklisch ausgetauschte Daten

Ausgetauscht werden Daten zwischen PLC und NC einerseits sowie zwischen PLC und HMI andererseits.

Daten **an** die PLC werden von der Firmware am Zyklusanfang des Anwenderprogramms bereitgestellt. Dadurch wird gesichert, dass z. B. die Signale vom NC über einen Zyklus konstant bleiben.

Daten **von** der PLC werden von der Firmware am Zyklusende des Anwenderprogramms an NC oder HMI weitergeleitet.

Nahtstelle PLC ↔ NC

Die zyklischen Daten enthalten z. B. Statussignale ("Programm läuft", "Programm unterbrochen") und Steuersignale (Start, Stopp) und Hilfs- und G-Befehle.

Die Daten sind gegliedert in Signale für:

- Betriebsarten
- Kanäle
- Achsen/Spindeln
- Allgemeine NC-Signale

Nahtstelle PLC ↔ HMI

Dies sind Signale für:

- Programmanwahl über Listen
- Messenger Steuerkommando
- Allgemeine Signale von/an HMI
- Signale von/an Wartungsplaner
- Signale von Bedientafel (remanenter Bereich)
- Allgemeine Anwahl-/Statussignale von/an HMI (remanenter Bereich)

15.1.2.2 Alarmer und Meldungen

Die Anwendernahtstelle im DB1600 bietet die Möglichkeit, Fehler- und Betriebsmeldungen auf dem HMI zur Anzeige zu bringen.

Durch die Firmware werden eingetragene Signale ausgewertet und als kommende und gehende Alarmer und Meldungen an den HMI gesendet und dort zur Anzeige gebracht. Die Fehlertexte werden vom HMI verwaltet.

Hinweis

Es werden maximal acht PLC-Alarmer am HMI angezeigt.

15.1.2.3 Remanente Daten

Für die remanenten Daten gibt es die Anwender-Datenbausteine DB9000 - DB9063 und den Datenbereich DB1400.DBW0 - DBW127. Dort kann der Anwender alle Daten ablegen, die über POWER OFF/ON ihre Gültigkeit behalten sollen. Die remanenten Daten werden im nichtflüchtigen Speicher gesichert, jedoch nicht bei der Datensicherung.

15.1.2.4 Nichtremanente Daten

Nichtremanente Daten (z. B. Merker, Zeiten und Zähler) werden bei jedem Steuerungshochlauf gelöscht.

15.1.2.5 PLC-Maschinendaten

Die PLC-Maschinendaten befinden sich im Bereich der NC-Maschinendaten. Diese Daten werden bei POWER ON von der PLC-Firmware in den DB4500 der PLC-Anwendernahtstelle übertragen und können dort vom PLC-Anwenderprogramm ausgewertet werden.

Literatur

Listenhandbuch SINUMERIK 828D

15.1.3 PLC-Eckdaten

Die integrierte PLC verfügt über einen Programmspeicher von 24000 PLC-Anweisungen, welche komplett in einem festen PLC-Zyklus abgearbeitet werden.

Im optional nutzbaren Interruptprogramm INT0 können maximal 500 Anweisungen abgearbeitet werden. Es wird servosynchron ausgeführt und ermöglicht schnellste Reaktionen auf Prozessereignisse. Deshalb werden interruptfähige PLC-Peripheriebaugruppen nicht gebraucht.

Daten		Anzahl	Besonderheiten
Hauptprogramm (MAIN)		1	
Unterprogramm (SBRx)		256	
Interrupts		2	
Zeitgesteuerter Interrupt		1	Servosynchrones Interruptprogramm
Alarmer / Meldungen		1248	Es werden maximal 8 Alarmer angezeigt.
Merker		4096	Nicht remanent.
Zähler		64	Nicht remanent.
Zeiten, davon:		128	Nicht remanent.
	10 ms Inkrementintervall	112	
	100 ms Inkrementintervall	16	
Anwenderdatenbaustein mit je max. 512 Bytes		64	Adressbereich DB9000 bis DB9063
Datenübertragung NC ↔ PLC			Über fest parametrierbare Schnittstelle.

15.1.4 PLC Peripherie, schnelle Onboard-Ein-/Ausgänge

Zu den Eigenschaften der schnellen Onboard-Ein-/Ausgänge und deren Reaktionszeiten, siehe Kapitel "Schnelle Onboard-Ein- und Ausgänge (Seite 1155)".

Erläuterungen zu den Peripheriemodulen, den Maschinensteuertafeln sowie der Zuordnung der Onboard-Ein-/Ausgänge zur PLC siehe:

Literatur:

Gerätehandbuch PPU SINUMERIK 828D

15.1.5 PLC-Toolbox

15.1.5.1 Stern/Dreieck-Umschaltung

Für die Stern/Dreieck-Umschaltung wird in der PLC-Toolbox folgender Baustein mitgeliefert:

- StarDelta

Hinweis

Mit diesem Baustein kann die Stern/Dreieck-Umschaltung auch von Spindelmotoren 1PH8 mit SMI an SINAMICS S120 durchgeführt werden.

15.2 Schnelle Onboard-Ein- und Ausgänge

Für digitale Ein- und Ausgangssignale die von der Steuerung besonders schnell verarbeitet werden sollen, stehen an der PPU-Baugruppe mehrere Schnittstellen zum direkten Anschluss der Signale zur Verfügung:

- Stecker X242: 4 Eingangssignale, 4 Ausgangssignale
- Stecker X252: 4 Eingangssignale, 2 Ausgangssignale

Schnittstelle	E/A-Signale		Adressierung
X242	Eingänge:	DIN1 ... DIN4	I256.0 ... I256.3
	Ausgänge:	DOU1 ... DOU T4	Q256.0 ... Q256.3
X252	Eingänge:	DIN9 ... DIN12	I256.4 ... I256.7
	Ausgänge:	DOU9 ... DOU T10	Q256.4 ... Q256.5

Reaktionszeiten

Abhängig von Lagereglertakt, Ausführungsort und verwendeter digitalen Ein-/Ausgänge ergeben sich folgende Reaktionszeiten:

Lagereglertakt		1,5 ms	3 ms
Verarbeitung über:		Reaktionszeit ¹⁾	
<ul style="list-style-type: none"> Unterprogramm SBRx im zyklischen Betrieb (MAIN, OB1) Digitale Ein-/Ausgänge: Peripheriemodul PP 72/48 		12,5 ms	14 ms
	<ul style="list-style-type: none"> Interruptprogramm INT0 (servosynchron) Digitale Ein-/Ausgänge: Onboard Ein-/Ausgänge der PPU 	3 ms	4,5 ms
¹⁾ Signal an Eingangsklemme → Verarbeitung im PLC-Programm → Signal an Ausgangsklemme			

Lese-/Schreibzugriffe

Das Lesen und Schreiben der Ein/Ausgangssignalen erfolgt direkt an den Ein/Ausgängen der Baugruppe. Für schnellstmögliche Zugriffe (servotaktsynchron) wird die Verwendung der Direktoperationsbefehle im Interruptprogramm **INT0** empfohlen:

Befehl	Symbol
direkter Schließer	- I -
direkter Öffner	- / I -
Bitwert direkt setzen	-(S I)-
Bitwert direkt rücksetzen	-(R I)-
Bitwert direkt zuweisen	-(I)-

Parametrierung

Die Ein/Ausgangssignale können über Maschinendaten exklusiv der NC oder der PLC zugeordnet werden:

- MD10366 \$MN_HW_ASSIGN_DIG_FASTIN[<n>]
- MD10368 \$MN_HW_ASSIGN_DIG_FASTOUT[<n>]

mit <n>: Index zur Adressierung des Ein/Ausgangsbytes (0 = 1. Byte, 1 = 2. Byte, ...)

E/A-Signale		Zuordnung	
		NC	PLC
Eingänge:	DIN1 ... DIN4	MD10366[0] = 00 01 01 01 _H	MD10366[0] = 00 01 00 01 _H
Ausgänge:	DOUT1 ... DOUT4	MD10368[0] = 00 01 01 01 _H	MD10368[0] = 00 01 00 01 _H
Eingänge:	DIN9 ... DIN12	MD10366[1] = 00 01 01 01 _H	MD10366[1] = 00 01 00 01 _H
Ausgänge:	DOUT9 ... DOUT10	MD10368[1] = 00 01 01 01 _H	MD10368[1] = 00 01 00 01 _H

Literatur

Die ausführliche Spezifikation der Schnittstellen findet sich in:

Gerätehandbuch SINUMERK 828D PPU

15.3 Ladder Viewer und Ladder add-on

15.3.1 Überblick

Funktion

Die wesentlichen Funktionen von Ladder Viewer und Ladder add-on sind:

- Ladder Viewer: Anzeige der im PLC-Anwenderprogramm enthaltenen Verknüpfungen von Kontakten und Relais als Kontaktplan (KOP)
- Ladder add-on:
 - Diagnose des PLC-Anwenderprogramms
 - Bearbeiten bzw. Erstellen von Interruptroutinen
 - Rangieren von Daten

Literatur

Eine ausführliche Beschreibung der Funktionalität und Bedienung von Ladder Viewer und Ladder add-on findet sich in den jeweiligen SINUMERIK Bedienhandbüchern:

- Bedienhandbuch Universal
- Bedienhandbuch Drehen
- Bedienhandbuch Fräsen

jeweils im Kapitel "Ladder Viewer und Ladder add-on"

15.3.2 Parametrierung

Darstellung von Adressen

Über das NC-spezifische Maschinendatum wird die Darstellung von Adressen im Ladder Viewer entsprechend der SIMATIC S7 300 Notation eingestellt:

MD51230 \$MN_ENABLE_LADDER_DB_ADDRESSES = <Wert>

<Wert>	Bedeutung
0	Adressdarstellung in SIMATIC S7 200 Notation (z.B. Vxxxxx)
1	Adressdarstellung in SIMATIC S7 300 Notation (z.B. DBxx.DBBxxxx)

Editieren freigeben/sperren

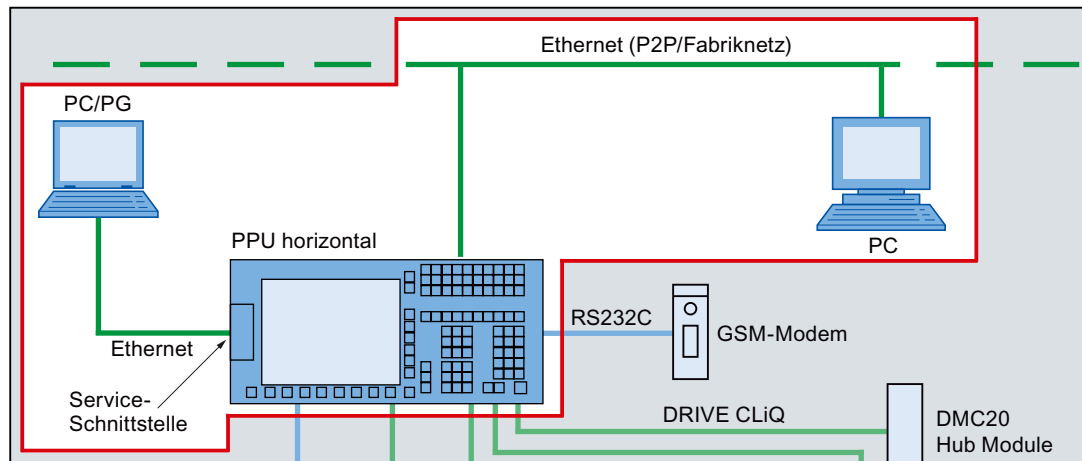
Über das NC-spezifische Maschinendatum wird das Editieren der Interruptprogramme INT_100 und INT_101 freigegeben bzw. gesperrt:

MD51231 \$MN_ENABLE_LADDER_EDITOR = <Wert>

<Wert>	Bedeutung
0	Die Interruptprogramme INT_100 und INT_101 können nicht editiert werden.
1	Interruptprogramme INT_100 und INT_101 können editiert werden.

15.4 PLC Programming Tool

Das "PLC Programming Tool" ist ein Werkzeug, mit dem PLC-Anwenderprogramme komfortabel erstellt werden können. Es ist ein Windows-Programm und muss auf einem Windows-PC installiert werden. Für den Online-Zugang zur Steuerung muss der PC über einen der Industrial Ethernet-Anschlüsse, X130 (Fabriknetz) oder X127 (Serviceschnittstelle), mit der Steuerung verbunden sein.



Mit dem Aufruf des "PLC Programming Tool" wird – ohne Angabe eines bereits existierenden Projekts – ein neues Projekt mit Namen "Projekt1" angelegt. Dieses Projekt kann sofort zum Erstellen des PLC-Anwenderprogramms genutzt, unter einem beliebigen Namen gespeichert und in die Steuerung geladen werden. Bereits existierende Projekte können auf Windows-typische Weise geöffnet werden.

Die Online-Hilfe zum "PLC Programming Tool" steht Windows-typisch über die Funktionstaste "F1" zur Verfügung.

Literatur

- Inbetriebnahmehandbuch Drehen und Fräsen SINUMERIK 828D
Kapitel "Lieferumfang und Voraussetzungen" > "Kommunikation mit der Steuerung" > "So kommunizieren Sie über das Programming Tool mit der Steuerung"
- Online-Hilfe zum PLC Programming Tool

15.5 Datenschnittstelle

Es werden Daten zwischen PLC und NC einerseits sowie zwischen PLC und HMI andererseits zyklisch ausgetauscht. Dies bedeutet insbesondere, dass vom HMI kommende und für die NC bestimmte Daten vom AWP rangiert werden müssen, damit diese wirksam werden.

Daten an die PLC werden von der Firmware am Zyklusanfang des AWP bereitgestellt. Dadurch wird gesichert, dass z. B. die Signale vom NC über einen Zyklus konstant bleiben.

Daten von der PLC werden von der Firmware am Zyklusende des AWP an NC oder HMI weitergeleitet.

Alle Daten dieser Nahtstelle sind im Gerätehandbuch für die SINUMERIK 828D, PPU, aufgeführt.

15.5.1 Nahtstelle PLC-NC

Diese zyklischen Daten enthalten z. B. Statussignale ("Programm läuft", "Programm unterbrochen"), Steuersignale (Start, Stopp) und Hilfs- und G-Befehle.

Die Daten sind gegliedert in Signale für:

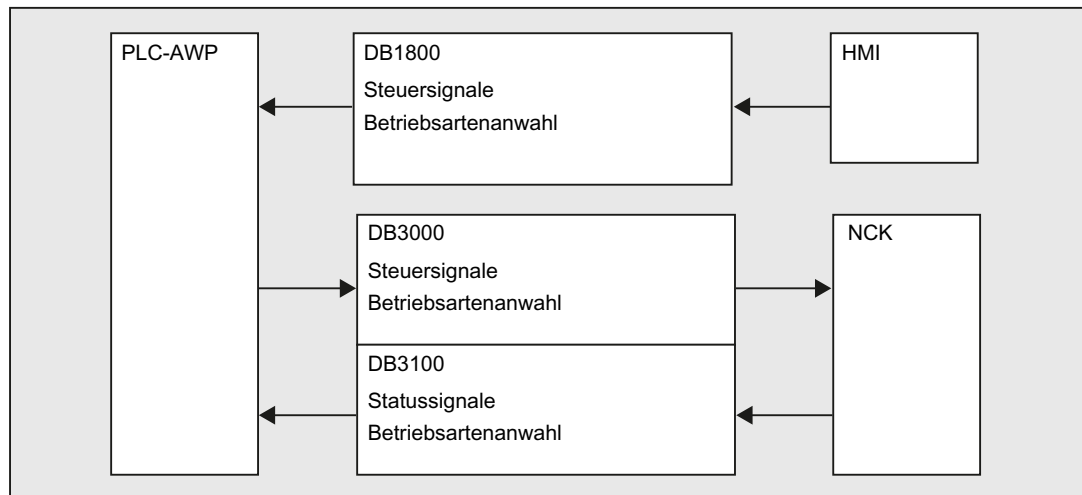
- Betriebsartensignale
- NC-Kanal-Signale
- Achs- und Spindelsignale
- Allgemeine NC-Signale
- Schneller Datenaustausch PLC-NC

15.5.1.1 Betriebsartensignale

DB3000, 3100

Zum NC werden die von der Maschinensteuertafel oder vom HMI vorgegebenen Betriebsartensignale übertragen.

Vom NC werden deren aktuelle Zustände an die PLC gemeldet.

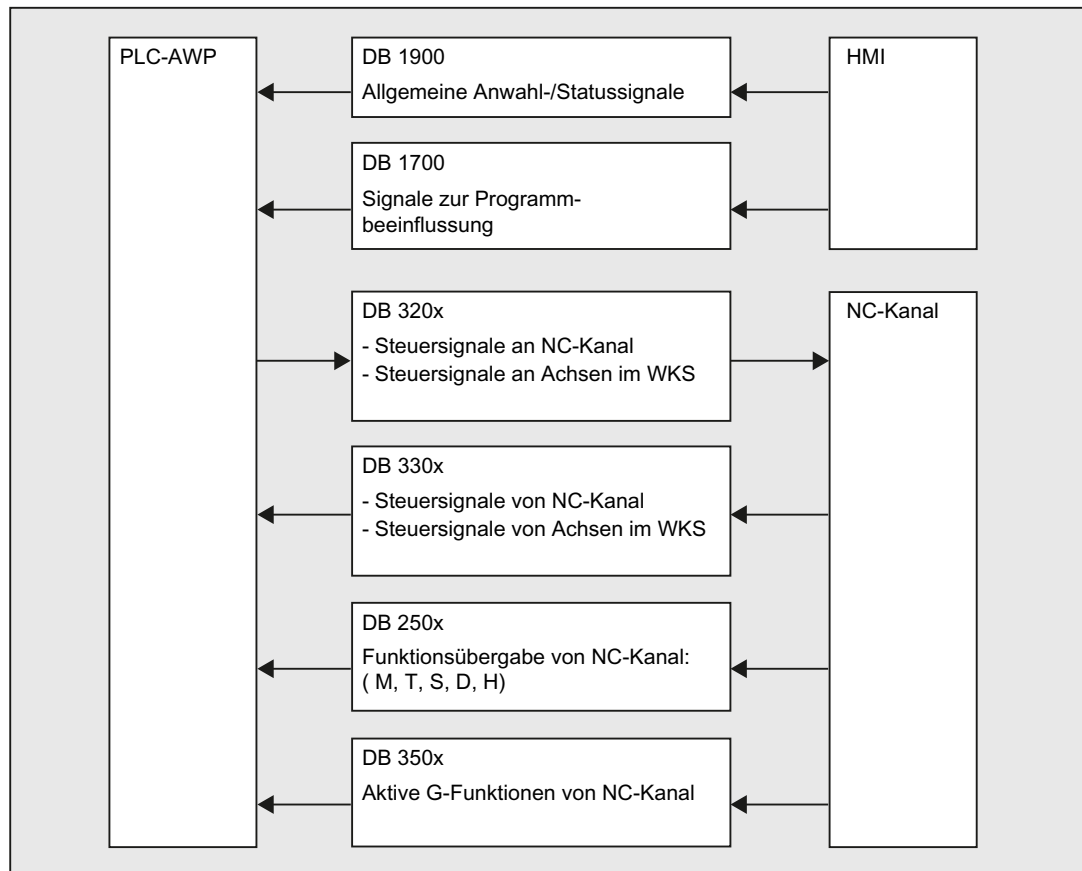


15.5.1.2 NC-Kanal-Signale

DB250x, 320x, 330x, 350x

Die Signale gliedern sich in:

- Steuer-/Status-Signale mit normaler zyklischer Übertragung, siehe "Betriebsartensignale (Seite 1159)".
- Hilfs- und G-Befehle
Diese werden auf zweierlei Weise in die Nahtstellen-DBs eingetragen.
Zunächst werden sie mit den Änderungssignalen eingetragen.
Die M-Signale M0 bis M99 werden zusätzlich dekodiert und die zugehörigen Nahtstellenbits für einen Zyklus gesetzt.
Bei den G-Befehlen werden nur die durch die Maschinendaten ausgewählten Gruppen in den Nahtstellen-DB eingetragen.
Die S-Werte werden zusätzlich zusammen mit den zugehörigen M-Signalen (M03, M04, M05) in die spindelspezifische Nahtstelle eingetragen. Ebenso werden achsspezifische Vorschübe in die entsprechende achsspezifische Nahtstelle eingetragen.



15.5.1.3 Achs- und Spindelsignale

DB370x, 380x, 390x

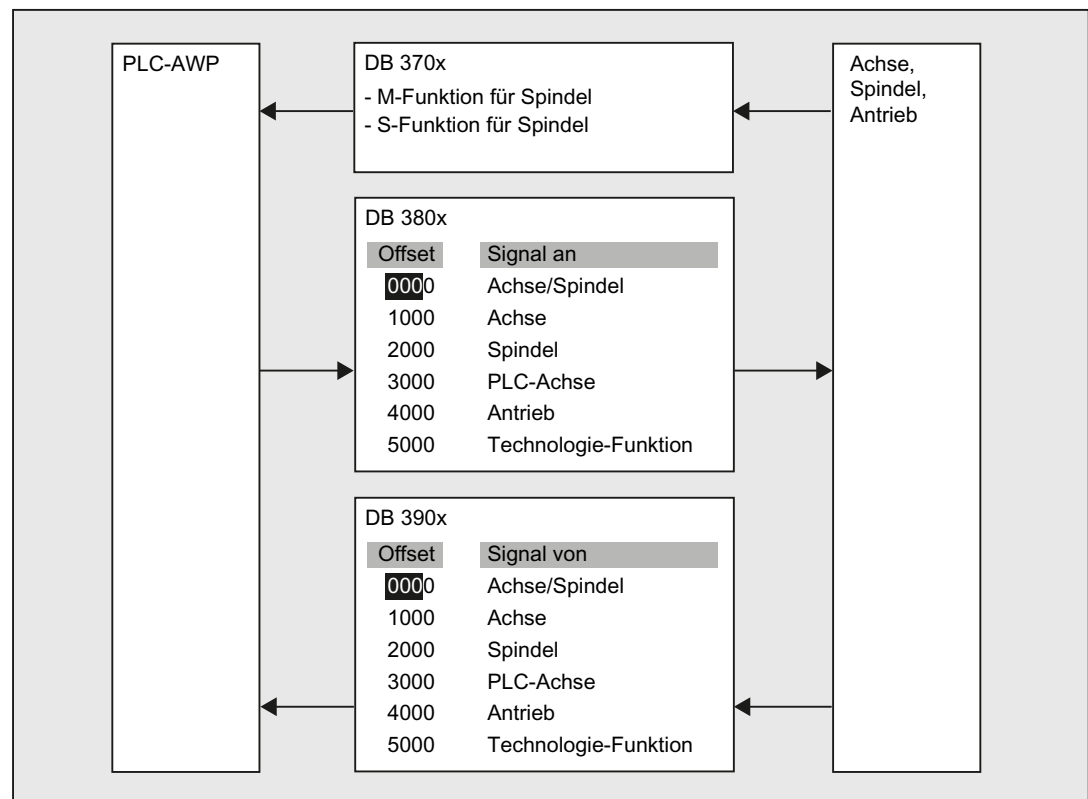
Die achs- und spindelspezifischen Signale sind in folgende Gruppen aufgeteilt:

- Gemeinsame Achs-/Spindelsignale
- Achssignale
- Spindelsignale
- Antriebssignale

Die Signale werden bis auf die im Folgenden beschriebenen Ausnahmen zyklisch übertragen. Zu den Ausnahmen gehören axialer F-Wert, M- und S-Wert.

Ein axialer F-Wert wird über den M-, S-, F-Verteiler dann eingetragen, wenn er im Zuge der NC-Programmbearbeitung an die PLC übertragen wird.

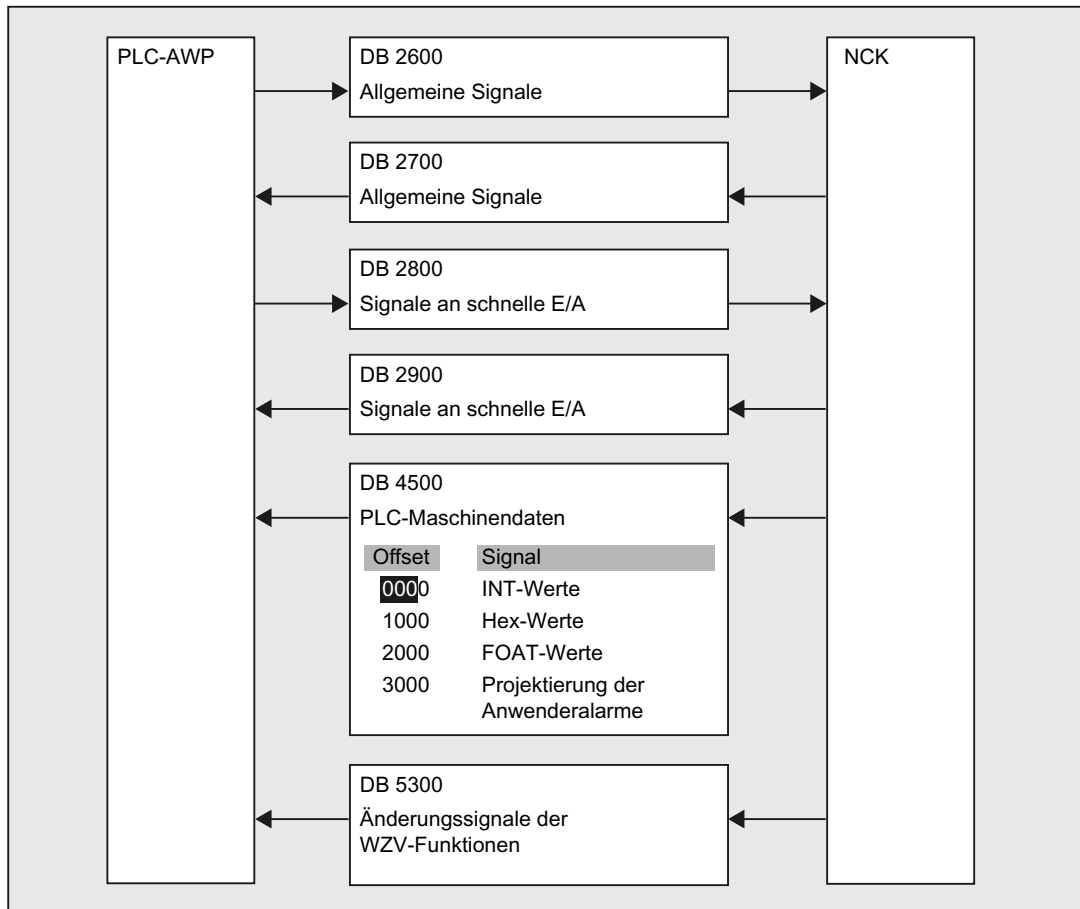
M- und S-Wert werden ebenfalls dann über den M-, S-, F-Verteiler eingetragen, wenn einer oder beide Werte zur Abarbeitung kommen.



15.5.1.4 Allgemeine NC-Signale

DB2600, 2700, 2800, 2900, 4500, 5300

- Sollwerte an die digitalen und analogen Ein-/Ausgänge des NC
- Istwerte von den digitalen und analogen Ein-/Ausgängen des NC
- Schlüsselschalter- und Not-Aus-Signale
- Bereitschafts- und Statussignale des NC



15.5.1.5 Schneller Datenaustausch PLC-NC

DB4900

Für den schnellen Austausch von Informationen zwischen PLC und NC ist der Datenbaustein DB4900 mit einer Größe von 1024 Byte vorgesehen.

Die Belegung dieses Bereichs (Struktur) muss im NC-Teilprogramm und PLC-Anwenderprogramm identisch vereinbart werden.

Vom NC-Teilprogramm kann auf diese Daten mit den Befehlen `$A_DBB[x]`, `$A_DBW[x]`, `$A_DBD[x]` und `$A_DBR[x]`; $0 \leq x \leq 1023$ zugegriffen werden (siehe Listenhandbuch Systemvariablen).

Dabei muss die Ausrichtung der Daten entsprechend ihrem Format gewählt werden, d. h. ein Dword beginnt auf einer 4-Byte-Grenze und ein Word auf einer 2-Byte-Grenze. Bytes können auf einem beliebigen Offset innerhalb des Datenfeldes liegen.

Die Datenkonsistenz wird für Byte-, Word- und Dword-Zugriffe sichergestellt. Bei Übertragung mehrerer Daten ist die Konsistenz anwenderseitig über Semaphore zu sichern, über welche die Gültigkeit bzw. Konsistenz eines Blockes erkannt werden kann.

15.5.2 Nahtstelle PLC-HMI

DB1700, 1800, 1900

Diese Signale sind bereits in den Bildern des Kapitels Nahtstelle PLC-NC (Seite 1159) mit angegeben worden.

Noch einmal sei auf das unter Datenschnittstelle (Seite 1158) Gesagte hingewiesen:

Vom HMI kommende und für die NC bestimmte Daten werden nicht automatisch in den NC-Nahtstellenbereich eingetragen. Vielmehr müssen diese Signale und Daten vom AWP rangiert werden.

Es handelt sich um folgende Signale:

- Programmanwahl über Listen
- Messenger Steuerkommando
- Allgemeine Signale von/an HMI
- Signale von/an Wartungsplaner
- Signale von Bedientafel (remanenter Bereich)
- Allgemeine Anwahl-/Statussignale von/an HMI (remanenter Bereich)

15.5.2.1 Bedienbereichsnummern

Die Nummer des aktiven Bedienbereichs wird angezeigt in: DB1900.DBB1

Bedienbereichsnummern

Bedienbereich	Nummer
Maschine	201
Parameter	205
Programmierung	203
Programm-Manager	202

Bedienbereich	Nummer
Diagnose	204
Inbetriebnahme	206

15.5.2.2 Bildnummern

Die aktuelle Bildnummer wird angezeigt in: DB1900.DBW4

Bildnummernbereiche

Es gibt folgende Bildnummernbereiche:

- JOG, manuelle Maschine (Seite 1164)
- Referenzpunktfahren (Seite 1169)
- MDA (Seite 1169)
- AUTOMATIK (Seite 1169)
- Bedienbereich Parameter (Seite 1170)
- Bedienbereich Programm (Seite 1171)
- Bedienbereich Prog., Programm-Manager (Seite 1172)
- Bedienbereich Diagnose (Seite 1172)

Bildnummern: JOG, manuelle Maschine

Betriebsart JOG

Bild	Nummer
Technologie Drehen	
Zyklenstartmaske für alle Masken die übernommen werden können	81
Technologie Fräsen	
Zyklenstartmaske für alle Masken die übernommen werden können	3
Technologie Drehen / Fräsen	
Grundbild	19
T,S,M	2
NPV setzen	21
Positionieren	4
Planfräsen	18
Abspanen	80
Zyklenstartmaske für alle Anwendermasken	91
Einstellungen Allgemein	1
Einstellungen Mehrkanalfunktion	106
Einstellungen Kollisionsvermeidung	107
Einstellungen Messprotokoll	108
Schwenken	60

Bild	Nummer
Alle G-Befehle	100
Zoom Istwert (MKS/WKS)	101
Gewinde Synchronisieren	102
Rückziehen	103
Handrad	104
Synchronisation Aktionen	105
Technologie Drehen: Werkstück, Nullpunkt	
Nullpunkt Werkstück (Grundmenü)	30
Anwendermaske	31
Anwendermaske	34
Anwendermaske	35
Anwendermaske	36
Anwendermaske	37
Anwendermaske	38
Anwendermaske	40
Messen Kante Z	5
Technologie Drehen: Werkstück, Messen	
Messen Werkzeug (Grundmenü)	50
Manuell X oder Anwendermaske	51
Manuell Y	71
Manuell Z oder Anwendermaske	52
Lupe oder Anwendermaske	53
Anwendermaske	54
Anwendermaske	55
Abgleich Messtaster X oder Anwendermaske	56
Abgleich Messtaster Z oder Anwendermaske	57
Länge Automatisch in Z	58
Länge Automatisch in Y	73
Länge Automatisch in X	59
Technologie Fräsen: Werkstück, Nullpunkt	
Nullpunkt Werkstück (Grundmenü)	30
messen Kante X	5
messen Kante Y	22
messen Kante Z	23
Anwendermaske	7
Kante ausrichten oder Anwendermaske	31
Abstand 2 Kanten oder Anwendermaske	32
Rechtwinklige Ecke	33
Beliebige Ecke oder Anwendermaske	8
Rechtecktasche	34
1 Bohrung oder Anwendermaske	9
2 Bohrungen	35
3 Bohrungen	36

Bild	Nummer
4 Bohrungen	37
Rechteckzapfen	38
1 Kreiszapfen oder Anwendermaske	10
2 Kreiszapfen	39
3 Kreiszapfen	40
4 Kreiszapfen	41
Ebene einrichten	42
Abgleich Taster Länge oder Anwendermaske	11
Abgleich Taster Radius	12
Technologie Fräsen: Werkstück, Messen	
Messen Werkzeug (Grundmenü)	50
- Länge messen Manuell (mit Fräswerkzeug) oder - Länge messen in X Manuell (mit Drehwerkzeug) oder - Anwendermaske	16
Länge messen in Y Manuell (mit Drehwerkzeug)	74
Länge messen in Z Manuell (mit Drehwerkzeug)	24
Durchmesser messen Manuell oder Anwendermaske	17
- Länge messen Automatisch (mit Fräswerkzeug) oder - Länge messen in X Automatisch (mit Drehwerkzeug) oder - Anwendermaske	13
Länge messen in Y Automatisch (mit Drehwerkzeug)	75
Länge messen in Z Automatisch (mit Drehwerkzeug)	25
Durchmesser messen Automatisch oder Anwendermaske	14
Anwendermaske	51
Abgleich Messtaster oder Anwendermaske	15
Abgleich Festpunkt oder Anwendermaske	52
RunMyScreens (nur bei gesetzter JobShopIntegration)	
Anwendermaske für 1. horizontalen Softkey	96
Anwendermaske für 2. horizontalen Softkey	98
Anwendermaske für 3. horizontalen Softkey	99
Anwendermaske für 4. horizontalen Softkey	94
Anwendermaske für 5. horizontalen Softkey	95
Anwendermaske für 6. horizontalen Softkey	92
Anwendermaske für 7. horizontalen Softkey	97
Anwendermaske für 8. horizontalen Softkey	90
Anwendermaske für 9. horizontalen Softkey	83
Anwendermaske für 10. horizontalen Softkey	82
Anwendermaske für 11. horizontalen Softkey	93
Anwendermaske für 12. horizontalen Softkey	84
Anwendermaske für 13. horizontalen Softkey	85
Anwendermaske für 14. horizontalen Softkey	86
Anwendermaske für 15. horizontalen Softkey	87
Anwendermaske für 16. horizontalen Softkey	88

Betriebsart JOG, manuelle Maschine

DB19.DBB24	
Bild	Bildnummer
Technologie Drehen / Fräsen	
Kegel Drehen	61
Winkel Fräsen	62
Anschlag	63
Gerade	1300
Gerade alle Achsen	1330
Gerade X alpha	1340
Gerade Z alpha	1350
Kreis	1360
Bohren	1400
Bohren Mittig	1410
Bohren Gewinde Mittig	1420
Bohren Zentrieren	1433
Bohren Bohren	1434
Bohren Reiben	1435
Bohren Ausdrehen	1436
Bohren Tieflochbohren	1440
Bohren Tieflochbohren 2	1441
Bohren Gewindebohren	1453
Bohrgewindefräsen	1455
Positionen	1473
Positionsreihe	1474
Positionsgitter	1477
Positionsrahmen	1478
Positionskreis	1475
Positionsteilkreis	1479
Hindernis	1476
Drehen	1500
Drehen Abspannen1	1513
Drehen Abspannen2	1514
Drehen Abspannen3	1515
Drehen Einstich1	1523
Drehen Einstich2	1524
Drehen Einstich3	1525
Drehen Freistich Form E	1533
Drehen Freistich Form F	1534
Drehen Freistich Gewinde DIN	1535
Drehen Freistich Gewinde	1536
Drehen Gewinde Längs	1543
Drehen Gewinde Kegel	1544

DB19.DBB24	
Bild	Bildnummer
Drehen Gewinde Plan	1545
Drehen Gewinde Kette	1546
Drehen Abstich	1550
Fräsen	1600
Fräsen Planfräsen	1610
Fräsen Rechtecktasche	1613
Fräsen Kreistasche	1614
Fräsen Rechteckzapfen	1623
Fräsen Kreiszapfen	1624
Fräsen Längsnut	1633
Fräsen Kreisnut	1634
Fräsen offene Nut	1635
Fräsen Mehrkant	1640
Fräsen Gewindefräsen	1454
Fräsen Gravur	1670
Konturdrehen	1200
Konturdrehen Neue Kontur/Letzte Kontur	1210
Konturdrehen Konturabspannen	1220
Konturdrehen Konturstechen	1230
Konturdrehen Konturstechdrehen	1240
Konturfräsen	1100
Konturfräsen Neue Kontur/Letzte Kontur	1110
Konturfräsen Bahnfräsen	1120
Konturfräsen Zentrieren	1130
Konturfräsen Vorbohren	1140
Konturfräsen Konturtasche	1150
Technologie Drehen: Simulation	
Seitenansicht	1740
Stirnansicht	1750
3D-Ansicht	1760
2 Fenster Ansicht	1770
Halbschnitt	1780
Technologie Drehen: Mitzeichnen	
Seitenansicht	1741
Stirnansicht	1751
3D-Ansicht	1761
2 Fensteransicht	1771
Maschinenraum	1791
Halbschnitt	1781
Technologie Fräsen: Simulation	
Draufsicht	1742
3D-Ansicht	1760

DB19.DBB24	
Bild	Bildnummer
Von vorne	1744
Von hinten	1746
Von links	1748
Von rechts	1752
Halbschnitt	1780
Drehansicht	1782
Technologie Fräsen: Mitzeichnen	
Draufsicht	1743
3D-Ansicht	1761
Von vorne	1745
Von hinten	1747
Von links	1749
Von rechts	1753
Maschinenraum	1791
Halbschnitt	1781
Drehansicht	1783

Bildnummern: Referenzpunktfahren

Bild	Nummer
Zoom Istwert MKS/WKS	101

Bildnummern: MDA

Bild	Nummer
MDA	20
Alle G-Befehle	100
Zoom Istwert MKS/WKS	101
Handrad	104
Synchronisation Aktionen	105
Programmbeeinflussung	210
Einstellungen	250

Bildnummern: AUTOMATIK

Bild	Nummer
Automatik	200
Überspeichern	202
Programmbeeinflussung	210
Satzsuchlauf	220
Einstellungen Allgemein	250

Bild	Nummer
Einstellungen Mehrkanalfunktion	106
Einstellungen Kollisionsvermeidung	107
Alle G-Befehle	100
Zoom Istwert MKS/WKS	101
Handrad	104
Synchronisation Aktionen	105
Technologie Drehen: Mitzeichnen	
Seitenansicht	243
Stirnansicht	244
3D-Ansicht	245
2 Fensteransicht	246
Maschinenraum	247
Halbschnitt	253
Technologie Fräsen: Mitzeichnen	
Draufsicht	242
3D-Ansicht	244
Von vorne	248
Von hinten	249
Von links	251
Von rechts	252
Maschinenraum	247
Halbschnitt	253
Drehansicht	254

Bildnummern: Bedienbereich Parameter

Bild	Nummer
Werkzeugliste	600
Werkzeugverschleiß	610
Anwenderwerkzeugliste	620
Magazin	630
Nullpunktverschiebung	
Nullpunktverschiebung Aktiv	642
Nullpunktverschiebung Übersicht	643
Nullpunktverschiebung Basis	644
Nullpunktverschiebung G54 - G509	645
Details von Nullpunktverschiebung Aktiv, Übersicht, Basis oder G54 - G509	647
Anwendervariable	
R-Parameter	650
Globale GUD 1 (SGUD)	660
Globale GUD 2 (MGUD)	661
Globale GUD 3 (UGUD)	662

Bild	Nummer
Globale GUD 4	663
Globale GUD 5	664
Globale GUD 6	665
Globale GUD 7	666
Globale GUD 8	667
Globale GUD 9	668
Kanal GUD 1 (SGUD)	690
Kanal GUD 2 (MGUD)	691
Kanal GUD 3 (UGUD)	692
Kanal GUD 4	693
Kanal GUD 5	694
Kanal GUD 6	695
Kanal GUD 7	696
Kanal GUD 8	697
Kanal GUD 9	698
Lokale LUD	681
Lokale LUD / PUD	684
Settingdaten	
Arbeitsfeldbegrenzung	671
Spindeldaten	670
Spindelfutterdaten	672
CTRL-Energie	
CTRL-Energie Grundmenü	6170
CTRL-Energie Analyse	6171
CTRL-Energie Profile	6172
CTRL-Energie Analyse Grafik	6176
CTRL-Energie Analyse Langzeitmessung	6177
CTRL-Energie Analyse Details	6179
CTRL-Energie Messungen vergleichen	6178

Bildnummern: Bedienbereich Programm

Bild	Nummer
Technologie Drehen: Simulation	
Seitenansicht	413
Stirnansicht	414
3D-Ansicht	415
2 Fenster Ansicht	416
Halbschnitt	423
Technologie Fräsen: Simulation	
Draufsicht	412
3D-Ansicht	414

Bild	Nummer
Von vorne	418
Von hinten	419
Von links	421
Von rechts	422
Halbschnitt	423
Drehansicht	424

Bildnummern: Bedienbereich Prog., Programm-Manager

Bild	Nummer
Verzeichnis NC	300
Lokales Laufwerk	325
USB / Projektirtes Laufwerk1	330
Projektirtes Laufwerk2	340
Projektirtes Laufwerk3	350
Projektirtes Laufwerk4	360
Projektirtes Laufwerk5	383
Projektirtes Laufwerk6	384
Projektirtes Laufwerk7	385
Projektirtes Laufwerk8	386

Bildnummern: Bedienbereich Diagnose

Bild	Nummer
Alarmliste	500
Meldungen	501
Alarmprotokoll	502
NC/PLC-Variable	503

15.6 Funktionsschnittstelle

15.6.1 NC-Variable lesen/schreiben

15.6.1.1 Anwenderschnittstelle

Über die NC/PLC-Nahtstelle "NC-Variable lesen/schreiben" können über das PLC-Anwenderprogramm maximal acht NC-Variablen gleichzeitig gelesen oder geschrieben werden.

Im Rahmen eines Auftrags (lesen/schreiben) sind folgende Schritte auszuführen:

1. Auftragspezifizierung (Seite 1173)
2. Auftragsmanagement: Auftrag starten (Seite 1174)
3. Auftragsmanagement: Warten auf Auftragsende (Seite 1175)
4. Auftragsmanagement: Auftragsabschluss (Seite 1175)
5. Auftragsauswertung (Seite 1177)

Ablaufdiagramm eines Auftrags: siehe "Auftragsmanagement: Ablaufdiagramm (Seite 1176)"

15.6.1.2 Auftragsspezifizierung

Variablenspezifische Auftragsschnittstelle

Jede Variable, die in einem Auftrag bearbeitet werden soll, muss in der **variablenspezifischen Auftragsschnittstelle** über ihre Parameter spezifiziert werden. Die allgemeinen Bezeichner werden später für jede Variable, die von der Schnittstelle bedient werden kann, konkretisiert.

DB120x ¹⁾	NC-Daten lesen/schreiben (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB1000	Variablenindex							
DBB1001	Bereichsnummer							
DBW1002	Zeilenindex NC-Variable x							
DBW1004	Spaltenindex NC-Variable x							
DBW1006	---							
DBD1008	Schreiben: Daten an NC-Variable x (Datentyp der Variablen: 1...4 Byte) ²⁾							
DBD1012	---							
DBD1016	Schreiben: Daten an NC-Variable x (REAL) ³⁾							
DBD1020	Schreiben: Daten an NC-Variable x (DWORD / DINT) ³⁾							
DBW1024	Schreiben: Daten an NC-Variable x (WORD / INT) ³⁾							
DBB1026	Schreiben: Daten an NC-Variable x (BYTE) ³⁾							
DBB1027	---	---	---	---	---	---	---	Schreiben: Daten an NC-Variab- le x ³⁾
1) DB120x, mit x = 0 ... 7 entsprechend Variable 1 ... 8.								
2) Nur für vordefinierte Variable der Anwenderschnittstelle „NC-Variable lesen/schreiben“								
3) Nur für Variablen aus DB9910 NC_DATA								

Hinweis

Kanalspezifische Variablen

Beim Schreiben/Lesen von kanalspezifischen Variablen dürfen in einem Auftrag nur Variablen von genau **einem** Kanal adressiert werden.

Antriebsspezifischen Variablen

Beim Schreiben/Lesen von antriebsspezifischen Variablen dürfen in einem Auftrag nur Variablen von genau **einem** Antriebsobjekt SERVO adressiert werden. Das Antriebsobjekt SERVO muss einer Maschinenachse der NC zugeordnet sein. Der Zeilenindex entspricht der logischen Antriebsnummer.

Fehlerfall

Im Fehlerfall, Schreiben/Lesen von Variablen von unterschiedlichen Antriebsobjekten, oder gleichzeitig von einem Kanal und einem Antriebsobjekt, erfolgt eine Fehlermeldung:

DB1200.DBX3000.1 == 1 (Fehler aufgetreten)

Beispiel: Eine Variable vom Typ "Platztyp" als vierte Variable lesen

```
DB1203.DBB1000: 7
DB1203.DBB1001: -
DB1203.DBW1002: <Platznummer>
DB1203.DBW1004: <Magazinnummer>
DB1203.DBW1006: -
DB1203.DBD1008: -
```

Beispiel: Eine Variable als vierte Variable schreiben

Zum Schreiben eines Datums an die NC, muss der Wert in das Doppelwort DBD1008 eingetragen werden:

```
DB1203.DBB1000: <Variablenindex>
DB1203.DBB1001: <Bereichsnummer>
DB1203.DBW1002: <Spaltenindex>
DB1203.DBW1004: <Zeilenindex>
DB1203.DBW1006: -
DB1203.DBD1008: <Wert>
```

15.6.1.3 Auftragsmanagement: Auftrag starten

Folgende Daten sind vom Anwender in die **globale Auftragsschnittstelle** zu schreiben:

DB120x ¹⁾	NC-Daten lesen/schreiben (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0							Auftrags- typ	Auftrag: Start
DBB1	Anzahl der im Auftrag zu bearbeitenden Variablen							
1) DB120x, mit x = 0 ... 7 entsprechend Variable 1 ... 8.								

Auftragstyp

- Variable **lesen**: DB1200.DBX0.1 = 0
- Variable **schreiben**: DB1200.DBX0.1 = 1

Auftrag starten

Zum Starten des Auftrag über die angegebene Anzahl Variablen ist das Start-Signal zu setzen:

DB1200.DBX0.0 = 1

Hinweis

Ein neuer Auftrag kann nur gestartet werden, wenn der vorhergehende Auftrag abgeschlossen wurde. Siehe Kapitel "Auftragsmanagement: Warten auf Auftragsende (Seite 1175)".

Die Dauer der Ausführung eines Auftrages kann mehrere PLC-Zyklen betragen und je nach Auslastung schwanken. Damit ist diese Funktion nicht zeitlich determiniert.

15.6.1.4 Auftragsmanagement: Warten auf Auftragsende

Das Auftragsende wird von der NC in der **globalen Ergebnisschnittstelle** immer für den **gesamten** Auftrag zurückgemeldet. Die Signale sind vom PLC-Anwender nur lesbar.

DB120x ¹⁾	NC-Daten lesen/schreiben (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB2000							Fehler im Auftrag	Auftrag beendet

1) DB120x, mit x = 0 ... 7 entsprechend Variable 1 ... 8.

Auftragsstatus

- Auftragsende ohne Fehler
DB1200.DBX2000.0 == 1 **UND** DB1200.DBX2000.1 == 0
- Auftragsende mit Fehler
DB1200.DBX2000.0 == 1 **UND** DB1200.DBX2000.1 == 1

Mögliche Fehlerursachen

- Anzahl Variablen (DB1200.DBB1) außerhalb des gültigen Bereichs
- Variablenindex (DB1200.DBB1000) außerhalb des gültigen Bereichs
- Gleichzeitiges Lesen/Schreiben von NC-Daten von unterschiedlichen Antriebsobjekten Servo

15.6.1.5 Auftragsmanagement: Auftragsabschluss**Anforderung**

Um den Auftrag abzuschließen, muss vom PLC-Anwenderprogramm, nach dem Erkennen des Auftragsendes, das Start-Signal des Auftrags zurückgesetzt werden:

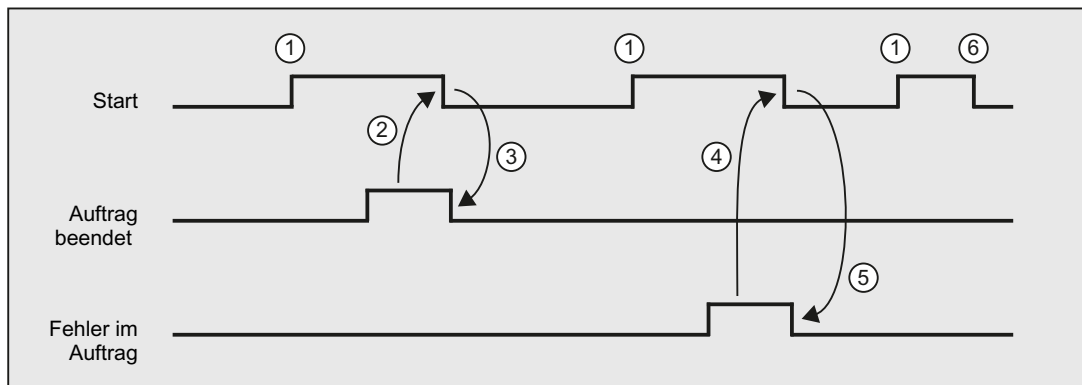
DB1200.DBX0.0 = 0

Rückmeldung

Als Rückmeldung werden von der NC die Status-Signale zurückgesetzt:

- DB1200.DBX2000.0 == 0
- DB1200.DBX2000.1 == 0

Der Auftrag ist damit abgeschlossen.

15.6.1.6 Auftragsmanagement: Ablaufdiagramm

- ① Auftrag starten:
DB1200.DBX0.0 (Start) = 1
- ② Warten auf Auftragsende:
DB1200.DBX2000.0 (Auftrag beendet) == 1 UND
DB1200.DBX2000.1 (Fehler im Auftrag) == 0
⇒ Auftragsanforderung zurücksetzen:
DB1200.DBX0.0 = 0 (Start)
- ③ Durch DB1200.DBX0.0 == 0 (Start) wird der Auftrag vom PLC-Grundprogramm abgeschlossen:
DB1200.DBX2000.0 (Auftrag beendet) = 0
- ④ Warten auf Auftragsende:
DB1200.DBX2000.0 (Auftrag beendet) == 0 UND
DB1200.DBX2000.1 (Fehler im Auftrag) == 1
⇒ Fehlerbehandlung durchführen
⇒ Auftragsanforderung zurücksetzen:
DB1200.DBX0.0 (Start) = 0
- ⑤ Durch DB1200.DBX0.0 == 0 (Start) wird der Auftrag vom PLC-Grundprogramm abgeschlossen:
DB1200.DBX2000.1 (Fehler im Auftrag) = 0
- ⑥ Wird DB1200.DBX0.0 (Start) zurückgesetzt, bevor das Auftragsende vom PLC-Grundprogramm gemeldet wurde, wird der Auftrag ohne weitere Rückmeldung ausgeführt.

15.6.1.7 Auftragsauswertung

Die variablenspezifische Ergebnisschnittstelle muss für jede im Auftrag bearbeitete Variable ausgewertet werden.

DB120x ¹⁾	NC-Dienste (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB3000							Fehler aufgetreten	Variable gültig
DBB3001	Zugriffsergebnis (siehe unten Absatz "Zugriffsergebnis")							
DBW3002	---							
DBD3004	Lesen: Daten von NC-Variable x (Datentyp der Variablen: 1...4 Byte) ²⁾							
DBD3008	---							
DBD3012	---							
DBD3016	Lesen: Daten von NC-Variable x (REAL) ³⁾							
DBD3020	Lesen: Daten von NC-Variable x DWORD / DINT) ³⁾							
DBW3024	Lesen: Daten von NCK-Variable x (WORD / INT) ³⁾							
DBB3026	Lesen: Daten von NCK-Variable x (BYTE) ³⁾							
DBB3027	---	---	---	---	---	---	---	Lesen: Daten von NC-Variable x ³⁾

1) DB120x, mit x = 0 ... 7 entsprechend Variable 1 ... 8.
 2) Nur für vordefinierte Variable der Anwenderschnittstelle "NC-Variable lesen/schreiben"
 3) Nur für Variablen aus DB9910 NC_DATA

Hinweis

Kanalspezifische Variablen

Beim Schreiben/Lesen von kanalspezifischen Variablen dürfen in einem Auftrag nur Variablen von genau **einem** Kanal adressiert werden.

Antriebsspezifischen Variablen

Beim Schreiben/Lesen von antriebsspezifischen Variablen dürfen in einem Auftrag nur Variablen von genau **einem** Antriebsobjekt SERVO adressiert werden. Das Antriebsobjekt SERVO muss einer Maschinenachse der NC zugeordnet sein. Der Zeilenindex entspricht der logischen Antriebsnummer.

Fehlerfall

Im Fehlerfall, Schreiben/Lesen von Variablen von unterschiedlichen Antriebsobjekten, oder gleichzeitig von einem Kanal und einem Antriebsobjekt, erfolgt eine Fehlermeldung:

DB1200.DBX3000.1 == 1 (Fehler aufgetreten)

Zugriffsergebnis

NC-Variablen

DBB3001	
Wert	Bedeutung
0	kein Fehler
3	Objektzugriff nicht erlaubt
5	ungültige Adresse
10	Objekt existiert nicht

Antriebsspezifische Variablen

Tritt beim Lesen/Schreiben einer antriebsspezifischen Variablen ein Fehler auf (DB1200.DBX3000.1 == 1) wird im Zugriffsergebnis eine Fehlernummer angezeigt, die auf den im PROFIdrive-Profil definierten Fehlernummern basiert.

DBB3001	
Wert	Bedeutung
x	<Fehlernummer des PROFIdrive-Profils> + 20 _H bzw. 36 _D

Ermittlung der Bedeutung des Zugriffsergebnisses:

1. Berechnung der Fehlernummer des PROFIdrive-Profils
 <Fehlernummer des PROFIdrive-Profils> = Zugriffsergebnis - 20_H bzw. 36_D.
2. Ermittlung der Bedeutung der Fehlernummer des PROFIdrive-Profils
 Die Fehlernummern des PROFIdrive-Profils sind beschrieben in:
Literatur
 Funktionshandbuch Antriebsfunktionen SINAMICS S120; Kapitel "Kommunikation" >
 "Kommunikation nach PROFIdrive" > "Azyklische Kommunikation" > "Aufbau der Aufträge
 und Antworten" > Absatz "Fehlerwerte in Parameterantworten"

Beispiele: Auftragsstatus

Auftrag ohne Fehler

- DB1200.DBX3000.0 == 1 (Variable gültig) **UND**
- DB1200.DBX3000.1 == 0 (kein Fehler aufgetreten)

Ergebnis:

- DB1200.DBB3001 == 0 (Zugriffsergebnis: "kein Fehler")
- DB1200.DBD3004 == <gelesener Wert>

Auftrag mit Fehler

- DB1200.DBX3000.0 == 0 (Variable nicht gültig) **UND**
- DB1200.DBX3000.1 == 1 (Fehler aufgetreten)

Ergebnis:

- DB1200.DBB3001: mögliche Fehlerursachen siehe oben, Absatz "Zugriffsergebnis"

15.6.1.8 Bedienbare Variablen

Folgende Variablen stehen zu Verfügung:

Variable	Bedeutung
cuttEdgeParam (Seite 1179)	Korrekturwertparameter und Schneidenliste mit D-Nummern zu einem Werkzeug
numCuttEdgeParams (Seite 1180)	Anzahl P-Elemente einer Schneide
linShift (Seite 1180)	Translation einer einstellbaren Nullpunktverschiebung
numMachAxes (Seite 1181)	Nummer der höchsten existierenden Kanalachse
rpa (Seite 1181)	R-Parameter
actLineNumber (Seite 1181)	Zeilennummer des aktuellen NC-Satzes
\$TC_MPPx (Seite 1182)	Magazinplatzdaten
r0078[1] (Seite 1183)	Stromistwert, momentenbildend
r0079[1] (Seite 1183)	Drehmomentsollwert am Ausgang des Drehzahlreglers
r0081 (Seite 1183)	Momentenausnutzung in Prozent
r0082[1] (Seite 1184)	Wirkleistungsistwert
TEMP_COMP_x (Seite 1184)	Temperaturkompensationsdaten

Variable "cuttEdgeParam"

Korrekturwertparameter und Schneidenliste mit D-Nummern zu einem Werkzeug

Die Bedeutung der einzelnen Parameter hängt vom Typ des jeweiligen Werkzeugs ab. Zur Zeit sind 25 Parameter für jede Werkzeugschneide reserviert (jedoch nur ein Teil davon ist mit Werten belegt). Um für zukünftige Erweiterungen flexibel zu sein, sollte jedoch nicht mit einem festen Wert von 25 Parametern sondern mit dem Variablenwert 'numCuttEdgeParams' (Variablenindex 2) gerechnet werden.

Eine detaillierte Beschreibung der Werkzeugparameter ist dem Kapitel W1: Werkzeugkorrektur (Seite 1475) zu entnehmen.

Adresse	Wert / Bedeutung
DB120x.DBB1000	1
DB120x.DBB1001	-
DB120x.DBW1002	(Schneiden-Nr. - 1) * numCuttEdgeParams + ParameterNr (WORD)
DB120x.DBW1004	T-Nummer (1...32000) (WORD)
DB120x.DBD1008	Schreiben: Daten an NC-Variable x (Datentyp: REAL)
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable "numCuttEdgeParams"

Anzahl P-Elemente einer Schneide

Adresse	Wert / Bedeutung
DB120x.DBB1000	2
DB120x.DBB1001	-

Adresse	Wert / Bedeutung
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NCK-Variable x (Datentyp: WORD)

Variable "linShift"

Translation einer einstellbaren Nullpunktverschiebung (kanalspezifische einstellbare Frames)

Die Variable ist nur vorhanden, wenn MD18601 MM_NUM_GLOBAL_USER_FRAMES > 0 ist.

Folgende Frameindizes sind verfügbar:

Index	Bedeutung
0	ACTFRAME = aktuelle resultierende Nullpunktverschiebung
1	IFRAME = aktuelle einstellbare Nullpunktverschiebung
2	PFRAME = aktuelle programmierbare Nullpunktverschiebung
3	EXTFRAME = aktuelle externe Nullpunktverschiebung
4	TOTFRAME = aktuelle Gesamtnullpunktverschiebung = Summe aus ACTFRAME und EXTFRAME
5	ACTBFRAME = aktueller Gesamt-Basisframe
6	SETFRAME = aktueller 1. Systemframe (Istwertsetzen, Ankratzen)
7	EXTSFRAME = aktueller 2. Systemframe (Istwertsetzen, Ankratzen)
8	PARTFRAME = aktueller 3. Systemframe (TCARR und PAROT bei orientierbarem Werkzeugträger)
9	TOOLFRAME = aktueller 4. Systemframe (TOROT und TOFRAME)
10	MEASFRAME = Ergebnisframe für die Werkstück- und Werkzeugvermessung
11	WPFRAME = aktueller 5. Systemframe (Werkstückbezugspunkte)
12	CYCFRAME = aktueller 6. Systemframe (Zyklen)

Adresse	Wert / Bedeutung
DB120x.DBB1000	3
DB120x.DBB1001	-
DB120x.DBW1002	Frameindex * numMachAxes (Seite 1181) + Achsnummer
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable "numMachAxes"

Nummer der höchsten existierenden Kanalachse

Wenn es keine Kanalachslücken gibt, ist der Wert der Variablen auch die Anzahl der vorhandenen Achsen im Kanal.

Adresse	Wert / Bedeutung
DB120x.DBB1000	4
DB120x.DBB1001	-
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: WORD)

Variable "rpa"

R-Parameter

Adresse	Wert / Bedeutung
DB120x.DBB1000	5
DB120x.DBB1001	-
DB120x.DBW1002	R-Nummer + 1
DB120x.DBW1004	-
DB120x.DBD1008	Schreiben: Daten an NC-Variable x (Datentyp: REAL)
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable "actLineNumber"

Zeilennummer des aktuellen NC-Satzes

Nr.	Bedeutung
≥ 1	Zeilennummer des aktuellen NC-Satzes
0	Keine Zeilennummer verfügbar, da Programm nicht gestartet
-1	Keine Zeilennummer verfügbar: Fehler
-2	Keine Zeilennummer verfügbar: Unterdrückung der Satzanzeige mit <code>DISPLOF</code> ist aktiv

Adresse	Wert / Bedeutung
DB120x.DBB1000	6
DB120x.DBB1001	-
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: INT)

Werkzeugverwaltung: Magazinplatzdaten

Platztyp (\$TC_MPP2)

Adresse	Wert / Bedeutung	
DB120x.DBB1000	7	
DB120x.DBB1001	-	
DB120x.DBW1002	Platznummer (1 ... 31999)	
DB120x.DBW1004	Magazinnummer (1 ... 9999)	
DB120x.DBD1008	-	
DB120x.DBW3004	Lesen: Wert der NC-Variable (Datentyp: WORD)	
	Wert	Bedeutung
	> 0	Platztyp für virtuellen Platz
	0	"match all" (Zwischenspeicher)
	9999	undefiniert (kein virtueller Platz)

Platzzustand (\$TC_MPP4)

Adresse	Wert / Bedeutung	
DB120x.DBB1000	8	
DB120x.DBB1001	-	
DB120x.DBW1002	Platznummer (1 ... 31999)	
DB120x.DBW1004	Magazinnummer (1 ... 9999)	
DB120x.DBD1008	-	
DB120x.DBW3004	Lesen: Wert der NC-Variable (Datentyp: WORD)	
	Wert	Bedeutung
	1	gesperrt
	2	frei (<> belegt)
	4	reserviert für Werkzeug im Zwischenspeicher
	8	reserviert für zu beladendes Werkzeug
	16	belegt im linken Halbplatz
	32	belegt im rechten Halbplatz
	64	belegt im oberen Halbplatz
	128	belegt im unteren Halbplatz

T-Nummer des Werkzeuges auf diesem Platz (\$TC_MPP6)

Adresse	Wert / Bedeutung
DB120x.DBB1000	9
DB120x.DBB1001	-
DB120x.DBW1002	Platznummer (1 ... 31999)
DB120x.DBW1004	Magazinnummer (1 ... 9999)
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: T-Nummer des Werkzeuges auf diesem Platz (Datentyp: WORD)

Variable r0078[1]

- Antriebsobjekt: SERVO, SERVO_AC, SERVO_I_AC
- CO: Stromistwert momentenbildend [A_{eff}]
- Index [1]: Geglättet mit p0045

Adresse	Wert / Bedeutung
DB120x.DBB1000	10
DB120x.DBB1001	Nummer des Antriebsmoduls
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable r0079[1]

- Antriebsobjekt: SERVO, SERVO_AC, SERVO_I_AC
- CO: Drehmomentsollwert am Ausgang des Drehzahlreglers (vor der Taktinterpolation) [Nm]
- Index [1]: Geglättet mit p0045

Adresse	Wert / Bedeutung
DB120x.DBB1000	11
DB120x.DBB1001	Nummer des Antriebsmoduls
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable r0081

- Antriebsobjekt: SERVO, SERVO_AC, SERVO_I_AC
- CO: Momentenausnutzung in Prozent

Adresse	Wert / Bedeutung
DB120x.DBB1000	12
DB120x.DBB1001	Nummer des Antriebsmoduls
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable r0082[1]

- Antriebsobjekt: SERVO, SERVO_AC, SERVO_I_AC
- CO: Wirkleistungsistwert [kW]
- Index [1]: Geglättet mit p0045

Adresse	Wert / Bedeutung
DB120x.DBB1000	13
DB120x.DBB1001	Nummer des Antriebsmoduls
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	-
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Temperaturkompensation**Variable "TEMP_COMP_ABS_VALUE" (SD43900)**

Positionsunabhängiger Temperaturkompensationswert

Adresse	Wert / Bedeutung
DB120x.DBB1000	14
DB120x.DBB1001	Nr. der Achse (1, 2, ...)
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	Schreiben: Daten an NC-Variable x (Datentyp: REAL)
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable "TEMP_COMP_SLOPE" (SD43910)

Steigungswinkel für positionsabhängige Temperaturkompensation

Adresse	Wert / Bedeutung
DB120x.DBB1000	15
DB120x.DBB1001	Nr. der Achse (1, 2, ...)
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	Schreiben: Daten an NC-Variable x (Datentyp: REAL)
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable "TEMP_COMP_REF_POSITION" (SD43920)

Bezugsposition für positionsabhängige Temperaturkompensation

Adresse	Wert / Bedeutung
DB120x.DBB1000	16
DB120x.DBB1001	Nr. der Achse (1, 2, ...)

Adresse	Wert / Bedeutung
DB120x.DBW1002	-
DB120x.DBW1004	-
DB120x.DBD1008	Schreiben: Daten an NC-Variable x (Datentyp: REAL)
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

Variable "TOOL_TEMP_COMP" (SD42960[...])

Temperaturkompensation bezogen auf das Werkzeug

Adresse	Wert / Bedeutung
DB120x.DBB1000	17
DB120x.DBB1001	-
DB120x.DBW1002	Index + 1 (1, 2, 3)
DB120x.DBW1004	-
DB120x.DBD1008	Schreiben: Daten an NC-Variable x (Datentyp: REAL)
DB120x.DBW3004	Lesen: Daten von NC-Variable x (Datentyp: REAL)

15.6.1.9 Spezifizierung von ausgewählten NC-Variablen

Die ausgewählten NC-Variablen werden über den Datenbaustein DB9910 (Ausgewählte NC-Variable) spezifiziert. Die Länge des Datenbausteins richtet sich nach der Anzahl der selektierten NC-Variablen in der Variablenliste. Die Variablenliste kann maximal 42 selektierte NC-Variablen enthalten. Der Datenbaustein DB9910 enthält für jede NC-Variable die Daten zur Variablenadressierung und Datentypkonvertierung.

DB9910	Ausgewählte NC-Variable, lesen (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	Variablenindex Variable 1							
DBB1	Syntax-ID							
DBB2	Bereich							
DBB3	Einheit							
DBW4	Spaltenindex							
DBW6	Zeilenindex							
DBB8	Baustein							
DBB9	Zeilenanzahl							
DBB10	Typ							
DBB11	Länge							
....							
DBB492	Variablenindex Variable 42							
DBB493	Syntax-ID							
DBB494	Bereich							
DBB495	Einheit							
DBW496	Spaltenindex							
DBW498	Zeilenindex							
DBB500	Baustein							

DB9910	Ausgewählte NC-Variable, lesen (PLC → NC)
DBB501	Zeilenanzahl
DBB502	Typ
DBB503	Länge

Variablenindex

Der Variablenindex bezieht sich auf den Namen der NC-Variable. Der Variablenindex setzt sich aus einem Anfangswert 100 und dem Offset der NC-Variable in der Liste (0 bis 41) zusammen.

Der Variablenindex wird in der Anwenderschnittstelle DB120x RW_NCDx als DBB1000 A_VarIdx eingetragen.

Der Kommentar enthält folgenden Datensatz mit Leerzeichen als Trenner:

- Bereich
- Baustein
- VariablenName
- VarType
- Spalte
- VarAnzByte

Erweiterte Anwenderschnittstelle

Aufgrund der Datenprüfung in Datenbausteinen kann der Wert einer Variablen nur auf eine Adresse im Datenbaustein geschrieben werden, die den gleichen Typ besitzt, z. B. kann ein REAL-Wert nur auf eine REAL-Adresse geschrieben werden (z. B. mit MOV_R). Die Anwenderschnittstelle "NC-Variable lesen/schreiben" enthält bisher nur eine Adresse vom Typ DWORD für den zu schreibenden Wert (DBD1008). Ein REAL-Wert kann damit nur über eine Hilfsvariable, Merker oder Akkumulator, geschrieben werden. Das Gleiche trifft auch für das Lesen zu (DBD3004). Aus diesem Grund wird die Anwenderschnittstelle "NC-Variable lesen/schreiben" erweitert. Für Schreiben und Lesen wird jeweils eine Adresse für jeden Typ hinzugefügt: REAL, DWORD/DINT, WORD/INT, BYTE und BOOL (DBD1016 ... DBB1027 bzw. DBD3016 ... DBB3027). Diese neuen Adressen werden von der PLC-Firmware nur für die Variablen, die mit dem NC-Variableneditor ausgewählt und beim Übersetzen in den DB9910 NC_DATA eingetragen wurden, bedient (Variablenindex ≥ 100). Die bisher fest in der Anwenderschnittstelle "NC-Variable lesen/schreiben" definierten NC-Variablen verwenden weiterhin die alten Adressen (DBD1008 bzw. DBD3004).

15.6.2 Programminstanz-Dienste (PI-Dienste)

15.6.2.1 Auftragspezifizierung

PI-Dienste werden über die **Auftragsschnittstelle** (DB1200 ab Offset 4000) spezifiziert.

DB1200			PI-Dienst [r/w]					
			Nahtstelle PLC → NC					
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB 4000	-	-	-	-	-	-	-	Start ¹⁾
DBB 4001	PI-Index ²⁾							
DBB 4002	-							
DBB 4003	-							
DBW 4004	PI-Parameter 1 ³⁾							
DBW 4006	PI-Parameter 2							
DBW 4008	PI-Parameter 3							
DBW 4010	PI-Parameter 4							
DBW 4012	PI-Parameter 5							
DBW 4014	PI-Parameter 6							
DBW 4016	PI-Parameter 7							
DBW 4018	PI-Parameter 8							
DBW 4020	PI-Parameter 9							
DBW 4022	PI-Parameter 10							
1) DB1200.DBX4000.1, Start: DBX4000.1 = 1 ⇒ Start des PI-Dienstes; DBX4000.1 = 0 ⇒ PI-Dienst beendet								
2) DB1200.DBB4001, PI-Index: Spezifiziert den konkreten PI-Dienst								
3) DB1200.DBW4004 ..., PI-Parameter: PI-spezifische Parameter								

Übersicht der PI-Dienste:

- PI-Dienst ASUP (Seite 1188)
- PI-Dienst LOGOUT (Seite 1190)
- PI-Dienst DATA_SAVE (Seite 1190)
- PI-Dienst TMMVTL (Seite 1191)

15.6.2.2 Auftragsrückmeldung

Die PLC gibt Rückmeldungen über Erfolg oder Nichterfolg des gestarteten PI-Dienstes in der **Ergebnisschnittstelle** (DB1200 ab Offset 5000).

Das Auftragsende wird durch folgende Signalen gemeldet:

- DB1200.DBX5000.0
- DB1200.DBX5000.1

Die Signale werden von der PLC geschrieben und sind vom Anwender nur lesbar.

Ein Auftrag ist abgeschlossen, sobald der Anwender das Signal "Start" (DB1200.DBX4000.1) rücksetzt. Daraufhin werden die Statussignale DB1200.DBX5000.0 und .1 auf Null gesetzt.

DB1200			PI-Dienst [r]					
			Nahtstelle NCK → PLC					
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB 5000	-	-	-	-	-	-	Fehler im Auftrag ²⁾	Auftrag beendet ¹⁾
DBB 5001	-							
DBB 5002	-							
1) DB1200.DBX5000.0, Auftragsstatus: DBX5000.0 == 1 ⇒ Auftrag beendet								
2) DB1200.DBX5000.1, Fehlerstatus: DBX5000.1 == 0 ⇒ keine Fehler; DBX5000.1 == 1 ⇒ Fehler								

Mögliche Fehlerursachen

- Der Index des parametrisierten PI-Dienstes (DB1200.DBB4001) ist außerhalb des gültigen Bereichs
- Parameterfehler

15.6.2.3 PI-Dienst ASUP

Interrupts

Hinweis

Zusammenhang Kanäle und Interrupts

Bei Steuerungen mit maximal **einem** Kanal stehen **zwei** Interrupts und bei Steuerungen mit maximal **zwei** Kanälen stehen **vier** Interrupts zur Verfügung.

Mit dem PI-Dienst "ASUP" kann von der PLC aus jeder Interruptnummer ein Interruptprogramm (ASUP) zugeordnet werden. Das Interruptprogramm wird dann auf der NC ausgeführt, wenn der zugehörige Interrupt ausgelöst wird. Die Interruptprogramme müssen dazu in der NC im Verzeichnis für den Maschinenhersteller (_N_CMA_DIR) mit folgenden Programmnamen vorliegen:

Interrupt	Programmname
1	PLCASUP1_SPF
2	PLCASUP2_SPF
3	PLCASUP3_SPF
4	PLCASUP4_SPF

Sind die Interruptprogramme nicht vorhanden, müssen sie neu erstellt werden. Anschließend muss auf der NC ein NC-Reset (Power On) ausgelöst werden.

Der PI-Dienst "ASUP" muss nach dem Hochlauf der Steuerung pro Interruptzuordnung nur einmal ausgeführt werden. Die Zuordnung von Interrupt zu Interruptprogramm bleibt bis zum nächsten Hochlauf erhalten.

Auftragsspezifizierung

PI-Dienst: ASUP		
Adresse	Beschreibung	Gültige Werte
DB1200.DBW4001	PI-Index ¹⁾	1, 2, 13, 14
DB1200.DBW4004	Parameter 1: LIFTFAST ²⁾	0 (FALSE), 1 (TRUE)
DB1200.DBW4006	Parameter 2: BLSYNC ³⁾	0 (FALSE), 1 (TRUE)
DB1200.DBW4008	Parameter 3: Kanalindex ⁴⁾	0, 1
DB1200.DBW4010	Parameter 4: Interrupt-Priorität ⁵⁾	0, 1, 2, 3, 4
1) <ul style="list-style-type: none"> PI-Index = 1: Interrupt 1 \Rightarrow <code>_N_CMA_DIR / PLCASUP1_SPF</code>, defaultmäßige Interrupt-Priorität: 1 PI-Index = 2: Interrupt 2 \Rightarrow <code>_N_CMA_DIR / PLCASUP2_SPF</code>, defaultmäßige Interrupt-Priorität: 2 PI-Index = 13: Interrupt 3 \Rightarrow <code>_N_CMA_DIR / PLCASUP3_SPF</code>, defaultmäßige Interrupt-Priorität: 3 PI-Index = 14: Interrupt 4 \Rightarrow <code>_N_CMA_DIR / PLCASUP4_SPF</code>, defaultmäßige Interrupt-Priorität: 4 		
2) LIFTFAST bewirkt, dass nach dem Auslösen des Interrupts zuerst ein Schnellabheben erfolgt. Erst danach wird die Interrupt-Routine ausgeführt. Literatur Funktionshandbuch Sonderfunktionen; Kapitel "R3: Erweitertes Stillsetzen und Rückziehen" > "Steuerungsgeführtes ESR" > "Rückziehen"		
3) BLSYNC bewirkt, dass nach dem Auslösen des Interrupts zuerst gewartet wird, bis der aktuelle Satz abgearbeitet ist. Erst danach wird die Interrupt-Routine ausgeführt.		
4) 0 \rightarrow Kanal 1, 1 \rightarrow Kanal 2		
5) Bei Verwendung der defaultmäßigen Interrupt-Prioritäten ist der Parameter auf den Wert 0 zu setzen.		

Maschinendaten

- Niedrigste Interrupt-Priorität
Im folgenden Maschinendatum wird für die NC die niedrigste Interrupt-Priorität festgelegt, dessen zugehöriger Interrupt bearbeitet wird. Interrupts mit einer niedrigeren Priorität als der im Maschinendatum angegebenen, werden in der Steuerung nicht bearbeitet:
`MD11604 $MN_ASUP_START_PRIO_LEVEL`

Randbedingungen**Kanalzustand**

Der PI-Dienst "ASUP" darf nur ausgeführt werden, wenn sich der Kanal, in dem er angefordert wird, im Zustand "Reset" befindet.

ProgEvent "Hochlauf"

Wenn für den ereignisgesteuerten Programmaufruf (ProgEvent) als auslösendes Ereignis "Hochlauf" projektiert ist (`MD20108 $MC_PROG_EVENT_MASK`), darf der PI-Dienst "ASUP" erst nach Beendigung des ProgEvent-Programms (`PROG_EVENT_SPF` oder `MD11620 $MN_PROG_EVENT_NAME = <Anwender_Prog_event_SPF>`) gestartet werden.

Siehe auch

Auftragsspezifizierung (Seite 1187)

Auftragsrückmeldung (Seite 1187)

15.6.2.4 PI-Dienst LOGOUT**Funktion**

Das zuletzt an die NC übergebene Kennwort wird zurückgesetzt.

Auftragsspezifizierung

PI-Dienst: LOGOUT		
Adresse	Beschreibung	Gültige Werte
DB1200.DBW4001	PI-Index	3 (Kennwort rücksetzen)

15.6.2.5 PI-Dienst DATA_SAVE**Funktion**

Datenrettung des aktuellen Zustands der NC auf die System CompactFlash Card durchführen.

Hinweis**Hochlauf**

Im nächsten Hochlauf der Steuerung kann im "Startup menu" über "Reload saved user data" der gespeicherte Zustand in die NC geladen werden.

Auftragsspezifizierung

PI-Dienst: DATA_SAVE		
Adresse	Beschreibung	Gültige Werte
DB1200.DBW4001	PI-Index	4

Literatur

Inbetriebnahmehandbuch SINUMERIK 828D; Kapitel "Inbetriebnahme CNC" > "Lieferumfang und Voraussetzungen" > "Hochlauf der Steuerung"

15.6.2.6 PI-Dienst TMMVTL

Funktion

Mit dem PI-Dienst TMMVTL ist es möglich, von der PLC aus einen Auftrag zum Umsetzen eines Werkzeugs anzufordern. Aufgrund des PI-Dienstes führt die Werkzeugverwaltung für das im PI-Dienst angegebene Werkzeug (Werkzeugnummer oder Quell-Platznummer / Quell-Magazinnummer) eine Leerplatzsuche im Ziel-Magazin durch. Anschließend erhält die PLC über den DB41xx.DBB0 einen Auftrag zum Umsetzen des Werkzeugs.

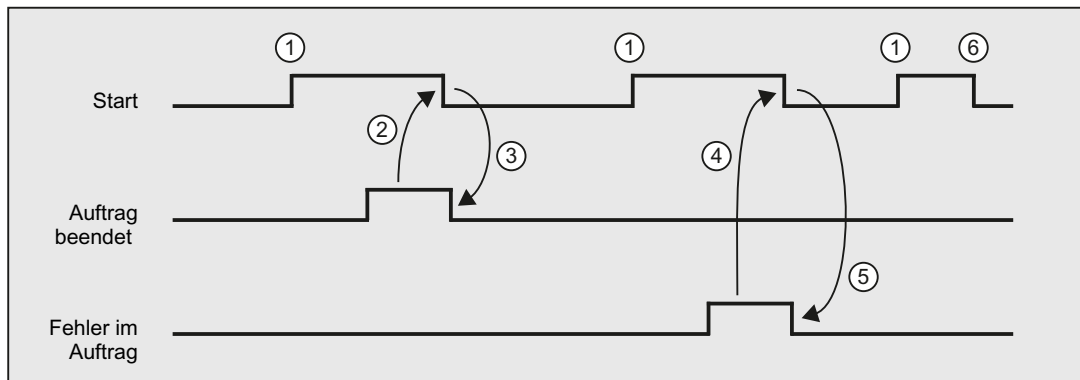
Auftragsspezifizierung

Adresse	Beschreibung	Gültige Werte
DB1200.DBW4001	PI Index	5
DB1200.DBW4004	Parameter 1: Werkzeugnummer ¹⁾	-1, 1 ... 31999
DB1200.DBW4006	Parameter 2: Quell-Platznummer ¹⁾	-1, 1 ... 31999
DB1200.DBW4008	Parameter 3: Quell-Magazinnummer ¹⁾	-1, 1 ... 9999
DB1200.DBW4010	Parameter 4: Ziel-Platznummer ²⁾	-1, 1 ... 31999
DB1200.DBW4012	Parameter 5: Ziel-Magazinnummer ³⁾	-1, 1 ... 9999
<p>1) Das Werkzeug kann wahlweise angegeben werden über:</p> <ul style="list-style-type: none"> • Werkzeugnummer (T-Nummer) • Quell-Platz- und Quell-Magazinnummer <p>Für die nicht benutzten Parameter der jeweils anderen Variante ist der Wert -1 einzutragen.</p> <p>2) Mit Ziel-Platznummer = -1 wird entsprechend der eingestellten Suchstrategie im gesamten Magazin ein Leerplatz für das Werkzeug gesucht. Ist ein Zielplatz angegeben wird geprüft ob der Platz mit der angegebenen Ziel-Platznummer für das Werkzeug frei und geeignet ist.</p> <p>3) Bei Ziel-Magazinnummer = -1 wird für Werkzeuge auf einem Zwischenspeicher entsprechend der sich aus \$TC_MDP2 ergebenden Zuordnung gesucht.</p>		

Anwendungsbeispiele

- Bei Verwendung von Zwischenspeichern zum Rücktransport des Werkzeugs (zum Beispiel Toolboy und/oder Shifter) kann im Zuge des asynchronen Rücktransportes eine explizite Leerplatzsuche im Magazin notwendig sein. In diesem Fall muss sich die PLC nicht den Ursprungsplatz merken, der PI-Dienst TMMVTL sucht nach einem geeigneten Platz.
- Aus einem Hintergrundmagazin soll ein Werkzeug in das Vordergrundmagazin geholt werden.

15.6.2.7 PI-Dienste: Taktdiagramm



- ① Anwender setzt Signal "Start", Auftragsausführung beginnt.
- ② Nachdem die PLC-Firmware "Auftrag beendet" signalisiert, setzt der Anwender das Signal "Start" zurück.
- ③ Durch Rücksetzen des Signals "Start" setzt die PLC-Firmware das Signal "Auftrag beendet" zurück.
- ④ Nachdem die PLC-Firmware "Fehler im Auftrag" signalisiert, setzt der Anwender das Signal "Start" zurück.
- ⑤ Durch Rücksetzen des Signals "Start" setzt die PLC-Firmware das Signal "Fehler im Auftrag" zurück.
- ⑥ Wird das Signal "Start" vom Anwender versehentlich zurückgesetzt, ehe eines der Signale "Auftrag beendet" oder "Fehler im Auftrag" kommt, so werden die Ergebnissignale für diesen Auftrag nicht aktualisiert. Der Auftrag wird aber dennoch ausgeführt.

15.6.3 PLC-Anwenderalarme

15.6.3.1 Anwenderschnittstelle

Hinweis

Obwohl im Folgenden die Bezeichnung Anwender-"*Alarme*" Verwendung findet, wird erst beim Eintragen des jeweiligen **Löschkriteriums** (siehe "Anwenderalarme projektieren (Seite 1195)") festgelegt, ob es sich um eine **Meldung** oder um einen **Alarm** handelt.

Die Anwendernachrichten im DB1600 bietet die Möglichkeit, Fehler- und Betriebsmeldungen auf dem HMI zur Anzeige zu bringen.

Dies beinhaltet folgende Leistungen:

- Aktivierung der Anwenderalarme 700000 bis 700247 und der erweiterten Anwenderalarme 701000 bis 701999.

Hinweis

Für die Nutzung der erweiterten PLC-Anwenderalarme 701000 bis 701999 gelten folgende Voraussetzungen:

- Der Kompatibilitätsmodus muss ausgeschaltet sein.
- Der Datenbaustein DB9913 muss im PLC-Projekt enthalten sein (d. h. DB9913 wurde im PLC Programming Tool unter "Bibliotheken" > "Spezielle Datenbausteine" ausgewählt und mittels Kopieren & Einfügen / Doppelklicken in das PLC-Projekt übernommen).

- Die Anwenderalarme 700000 bis 700247 und 701000 bis 701247 können mit einem zusätzlichen numerischen Parameter versehen werden.
- Deaktivierung und Quittierung der Anwenderalarme.
- Auswertung der durch die Anwenderalarme initiierten Systemreaktionen.

Durch die Firmware werden eingetragene Signale ausgewertet und als kommende und gehende Alarme und Meldungen an den HMI gesendet und dort zur Anzeige gebracht. Die Fehlertexte werden vom HMI verwaltet.

15.6.3.2 Aktivierungsschnittstelle der Anwenderalarme

Jeder Anwenderalarm wird über sein zugeordnetes Aktivierungsbit aktiviert. Diese Bits werden in der **Aktivierungsschnittstelle** gesetzt.

Ein neuer Anwenderalarm wird mit einer 0/1 Flanke des jeweiligen Bits aktiviert.

Aktivierungsschnittstelle für die Alarme 700000 bis 700247

DB1600	Aktivierung Alarm [r/w] (PLC → HMI)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Aktivierung Alarm-Nr.							
DBB0	700007	700006	700005	700004	700003	700002	700001	700000
	Aktivierung Alarm-Nr.							
DBB1	700015	700014	700013	700012	700011	700010	700009	700008
	Aktivierung Alarm-Nr.							
DBB2	700023	700022	700021	700020	700019	700018	700017	700016
	Aktivierung Alarm-Nr.							
DBB3	700031	700030	700029	700028	700027	700026	700025	700024
	Aktivierung Alarm-Nr.							
DBB4	700039	700038	700037	700036	700035	700034	700033	700032
	Aktivierung Alarm-Nr.							
DBB5	700047	700046	700045	700044	700043	700042	700041	700040
...							

DB1600	Aktivierung Alarm [r/w] (PLC → HMI)							
	Aktivierung Alarm-Nr.							
DBB30	700247	700246	700245	700244	700243	700242	700241	700240

Aktivierungsschnittstelle für die Alarmer 701000 bis 701999

DB1600	Aktivierung Alarm [r/w] (PLC → HMI)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Aktivierung Alarm-Nr.							
DBB4000	701007	701006	701005	701004	701003	701002	701001	701000
	Aktivierung Alarm-Nr.							
DBB4001	701015	701014	701013	701012	701011	701010	701009	701008
	Aktivierung Alarm-Nr.							
DBB4002	701023	701022	701021	701020	701019	701018	701017	701016
	Aktivierung Alarm-Nr.							
DBB4003	701031	701030	701029	701028	701027	701026	701025	701024
	Aktivierung Alarm-Nr.							
DBB4004	701039	701038	701037	701036	701035	701034	701033	701032
	Aktivierung Alarm-Nr.							
DBB4005	701047	701046	701045	701044	701043	701042	701041	701040
...	...							
	Aktivierung Alarm-Nr.							
DBB4124	701999	701998	701997	701996	701005	701994	701993	701992

15.6.3.3 Variablenschnittstelle der Anwenderalarme

Den Anwenderalarmen 700000 bis 700247 und 701000 bis 701247 kann eine Variable als Parameter mitgegeben werden. In der **Variablenschnittstelle** ist dafür je ein Doppelwort reserviert. Gültige Offsets müssen demzufolge durch 4 teilbar sein.

Variablenschnittstelle für die Alarmer 700000 bis 700247

DB1600	Variable für Alarm [r32/w32] (PLC → HMI)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD1000	Variable für Alarm 700000							
DBD1004	Variable für Alarm 700001							
DBD1008	Variable für Alarm 700002							
	...							
DBD1980	Variable für Alarm 700245							
DBD1984	Variable für Alarm 700246							
DBD1988	Variable für Alarm 700247							

Variablenschnittstelle für die Alarmer 701000 bis 701247

DB1600	Variable für Alarm [r32/w32] (PLC → HMI)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD5000	Variable für Alarm 701000							
DBD5004	Variable für Alarm 701001							
DBD5008	Variable für Alarm 701002							
	...							
DBD5980	Variable für Alarm 701245							
DBD5984	Variable für Alarm 701246							
DBD5988	Variable für Alarm 701247							

15.6.3.4 Anwenderalarme projektieren

Für jeden Alarm können folgende Attribute festgelegt werden:

- Alarmreaktion: wie die Steuerung beim Auftreten des Alarms reagiert.
- Löschkriterium: was muss unternommen werden, um den Alarm wieder zu löschen oder zu quittieren. Das Löschkriterium bestimmt gleichzeitig Typ und Priorität des Alarms.
- Kanalzuordnung: welchem Kanal wird der Alarm zugeordnet.

Anwenderalarme 700000 bis 700247

Die Anwenderalarme 700000 bis 700247 werden über Maschinendaten projiziert.

Alarmreaktion und Löschkriterium

Die Festlegung der Alarmreaktionen und Löschkriterien erfolgt bitcodiert im Maschinendatum:

MD14516 \$MN_USER_DATA_PLC_ALARM [x] = <Alarmreaktion und Löschkriterium>

mit x = Anwenderalarmnummer - 700000; Wertebereich: $0 \leq x \leq 247$

<Alarmreaktion und Löschkriterium>	
Bit	Bedeutung
Alarmreaktionen	
0	NC-Startsperre
1	Einlesesperre
2	Vorschubsperr aller Achsen
3	Not-Halt
4	PLC-Stop
5	Alarmprotokoll
Löschkriterien	
6	Abbruch mit DB1600 DBX3000.0
7	Power On
Bit x = 1: Aktivierung der Funktionalität	
Bit x = 0: Abwahl der Funktionalität	

Kanalzuordnung

Die Kanalzuordnung erfolgt bitcodiert im Maschinendatum:

MD14518 \$MN_USER_DATA_PLC_ALARM_ASSIGN [x] = <Kanalzuordnung>

mit x = Anwenderalarmnummer - 700000; Wertebereich: $0 \leq x \leq 247$

<Kanalzuordnung>	
Bit	Bedeutung
0	Geltungsbereich NC-Kanal 1
1	Geltungsbereich NC-Kanal 2
Bit x = 1: Kanal angewählt	
Bit x = 0: Kanal abgewählt	

Anwenderalarme 701000 bis 701999

Die Anwenderalarme 701000 bis 701999 werden im Datenbaustein DB9913 (ALARM_INI) projektiert. Für jeden Alarm werden zwei Byte Projektierungsdaten benötigt. Ein Byte für die Auswahl der Alarmreaktionen und Löschkriterien und ein Byte für die Kanalzuordnung. Diese zwei Bytes werden in einem Wort zusammengefasst.

DB9913	Projektion Anwenderalarme 701000 bis 701999 [r]
DBW0	Alarm 701000
DBW2	Alarm 701001
DBW4	Alarm 701002
	...
DBW1998	Alarm 701999

Bit	Bedeutung
Alarmreaktion	
0	NC-Startsperre
1	Einlesesperre
2	Vorschubsperr aller Achsen
3	Not-Halt
4	PLC-Stop
5	Alarmprotokoll
Löschkriterium	
6	Abbruch mit DB1600 DBX3000.0
7	Power On
Kanalzuordnung	
8	PLC-Anwenderalarm für NC-Kanal 1
9	PLC-Anwenderalarm für NC-Kanal 2
10	reserviert
11	reserviert
12	reserviert
13	reserviert

Bit	Bedeutung
14	reserviert
15	reserviert

Löschkriterium und Priorität

Das Löschkriterium und damit implizit auch der Typ und die Priorität eines Anwenderalarms wird eingestellt über die Bits 6 und 7:

Bit 7	Bit 6	Löschkriterium:	Typ	Priorität
0	0	Rücksetzen des Aktivierungsbits	Meldung	niedrig
0	1	Quittieren in DB1600.DBX3000.0 (siehe "Quittierungsschnittstelle der Anwenderalarme (Seite 1198)")	Alarm	mittel
1	0	Power On	Alarm	hoch
1	1	reserviert (intern gewertet als: Bit 7 = 1, Bit 6 = 0)	-	-

Anzeigemeldung

Wird für einen der oben aufgeführten Anwenderalarme keine Alarmreaktion aktiviert (Maschinendatenbit 0 bis 4 = 0), wird damit festgelegt, dass es sich um eine sogenannte "Anzeigemeldung" ohne Auswirkung auf das System handelt. Dies bedeutet insbesondere, dass auch das Löschkriterium (Maschinendatenbit 6 und 7) des entsprechenden Maschinendatums nicht ausgewertet wird.

15.6.3.5 Aktive Alarmreaktionen und Löschkriterien auslesen

Die momentan **aktiven Alarmreaktionen** (also die IST-Reaktionen), die **aktiven Löschkriterien** und die **aktive Kanalzuordnung** können aus der Schnittstelle global ausgelesen werden:

DB1600	Aktive Alarmreaktion [r]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB2000	Power On	Abbruch mit DB1600 DBX3000.0		PLC-Stop	Not-Halt	Kanal 1		
						Vorschub-sperre aller Achsen	Einlese-sperre	NC-Start-sperre
DBB2001						Kanal 2		
						Vorschub-sperre aller Achsen	Einlese-sperre	NC-Start-sperre

Ein Bit wird gesetzt, wenn mindestens für einen aktiven Alarm die entsprechende Reaktion oder das entsprechende Löschkriterium projiziert ist. Es wird gelöscht, wenn für keinen der anstehenden Alarme mehr diese Reaktion/Löschkriterium projiziert ist.

15.6.3.6 Quittierungsschnittstelle der Anwenderalarme

Voraussetzung für das Quittieren eines Anwenderalarms ist das Rücksetzen des entsprechenden Aktivierungsbits.

- Meldungen mit Löschkriterium {0,0} verschwinden dann selbständig aus der Anzeige.
- Alarme mit Löschkriterium {0,1} werden durch das Quittungsbit *Quit* gelöscht.
- Alarme mit Löschkriterium {1,0} bleiben vom Setzen des Quittungsbits unbeeinflusst, sie können nur durch Power On gelöscht werden.

DB1600	Alarmquittung [r/w]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB3000								Quit
DBB3001								
DBB3002								
DBB3003								

15.6.3.7 Schnittstelle zum HMI

Die PLC kann dem HMI acht Meldungen oder Alarme zur Anzeige übergeben, die in der Reihenfolge ihres Auftretens angezeigt werden.

Beim Auftreten weiterer Meldungen/Alarme bleiben die ersten sieben im HMI erhalten, und die jüngste Meldung bzw. der jüngste Alarm wird nach folgenden Regeln von einem gerade aufgetretenen verdrängt:

- Systemmeldung/-alarm verdrängt Anwendermeldung/-alarm.
- Meldungen/Alarme höherer Priorität verdrängen solche niederer Priorität.

Die ersten sieben Meldungen/Alarme verbleiben deshalb in der Anzeige, weil es sehr wahrscheinlich ist, dass diese die Ursache des Problems benennen und die Folgenden lediglich sekundärer Art sind. Werden jedoch eine oder mehrere Meldungen/Alarme quitiert und damit gelöscht, rückt eine entsprechende Anzahl neu hinzugekommener Alarme/Meldungen im HMI nach.

15.6.4 PLC-Achssteuerung

15.6.4.1 Allgemeine Informationen

Von der PLC können Achsen/Spindeln über Datenbausteine der Anwendernahtstelle gesteuert werden, dabei wird die Achse/Spindel über die DB-Nummer spezifiziert:

- **DB380x** Nahtstelle PLC → NC (an Achse/Spindel)
- **DB390x** Nahtstelle NC → PLC (von Achse/Spindel)

mit Achsindex x: $0 \leq x \leq \text{max. Achsindex}$; Achsindex = Achsnummer - 1

Folgende Funktionen werden unterstützt:

- Positionierachsen
- Spindel positionieren
- Spindel drehen
- Spindel pendeln
- Teilungsachsen

Literatur

- Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2) und Teilungsachsen (T1)
- Funktionshandbuch Grundfunktionen; Spindeln (S1)

Voraussetzung

Die zu steuernde Achse muss der PLC zugeordnet sein (PLC-Achse). Über die Anwendernahtstelle "*Achstausch*" (DB3800.DBB8/DB3900.DBB8) kann eine Achse zwischen NC und PLC getauscht werden.

Funktionsstart

Jede Funktion wird durch die positive Flanke des entsprechenden Signals "Start" angestoßen. Dieses Signal muss solange auf logisch "1" bleiben, bis die Funktion positiv oder negativ quittiert wurde (z. B. durch *Position erreicht* = "1" oder *Fehler* = "1"). Das Signal "Positionierachse aktiv" = "1" zeigt an, dass die Funktion aktiv ist und die Ausgangssignale gültig sind.

Abbruch

Das **Abbrechen** der Funktion ist über Rücksetzen des Startsignals **nicht** möglich, sondern nur über andere Nahtstellensignale (etwa über das achsspezifische Signal *Restweg löschen/ Spindel-Reset*, DB380x DBX2.2).

Die Achsnahtstelle liefert Statussignale der Achse zurück, die gegebenenfalls auszuwerten sind (z. B. *Genauhalt*, *Fahrbefehl*, → DB390x).

Wenn die Achse/Spindel bei Aufruf der PLC-Achssteuerung durch Verfahren der Achse durch die NC belegt ist (Fahrbefehle stehen an), dann wird die Funktion erst nach Beenden dieser Verfahrensbewegung gestartet. In dieser Situation wird kein Fehlercode ausgegeben.

Achssperr

Bei gesetzter **Achssperr** bewegt sich eine über PLC-Achssteuerung gesteuerte Achse nicht. Es wird nur ein simulierter Istwert erzeugt. (Verhalten wie bei NC-Programmierung).

15.6.4.2 Anwenderschnittstelle: Vorbereiten der NC-Achse als PLC-Achse

Anforderungssignale an Achse/Spindel (Auszug)

Zunächst muss die Achse/Spindel von der PLC angefordert werden:

DB380x	Signale an Achse/Spindel (PLC → NC) [r/w]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB8	PLC-Achse/ Spindel anfordern			Aktiviersignal bei Veränderung dieses Bytes			NC-Achse/Spindel Kanal zuordnen	
							B	A

Eine Änderung eines Anforderungssignals (DB380x.DBX8.7 oder 8.0) muss der NC über eine 0→1-Flanke des Aktiviersignals (DB380x.DBX8.4) mitgeteilt werden. Nach einem PLC-Zyklus muss das Aktiviersignal wieder zurückgesetzt werden.

Statussignale von Achse/Spindel (Auszug)

DB390x	Signale von Achse/Spindel (NC → PLC) [r]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB8	PLC-Achse/ Spindel	Neutrale Achse/ Spindel	Achsaustausch möglich	Neuer Typ von PLC gefordert			Aktuelle Zuordnung der NC-Achse/Spindel in Kanal	
							B	A

Hinweis

Simulation

Zur Aktivierung der Nahtstellensignale muss bei Simulation für jede benötigte Achse das Maschinendatum MD30350 \$MA_SIMU_AX_VDI_OUTPUT gesetzt werden.

PLC-Achse anfordern und abgeben

t	DB380x.DB0008	DB390x.DB0008	Bemerkung
	0 0 0 0 0 0 0 0	0 x 1 0 0 0 0 0 1	PLC erhält Signal "Achstausch möglich", "NC-Achse" und evtl. "neutrale Achse/Spindel" (x).
	1 0 0 1 0 0 0 0	1 0 0 0 0 0 0 1	PLC fordert Achse an.
	1 0 0 0 0 0 0 0	1 0 1 0 0 0 0 1	PLC setzt Aktiviersignal zurück.
..... → Achse ist PLC zugeordnet, kann verwendet werden und wird danach an NC abgegeben:			
	0 0 0 1 0 0 0 1	1 0 0 0 0 0 0 1	PLC gibt Achse an NC zurück.
	0 0 0 0 0 0 0 1	0 0 1 0 0 0 0 1	PLC setzt Aktiviersignal zurück; Status "NC-Achse" und "Achstausch möglich".
..... → Achse ist PLC zugeordnet, kann verwendet werden und wird danach freigegeben:			
	0 0 0 1 0 0 0 0	0 1 0 0 0 0 0 1	PLC gibt Achse frei.
	0 0 0 0 0 0 0 0	0 1 1 0 0 0 0 1	PLC setzt Aktiviersignal zurück; Status "neutrale Achse/Spindel", "NC-Achse" und "Achstausch möglich".

15.6.4.3 Anwenderschnittstelle: Funktionalität

Die beiden Tabellen geben einen Überblick über die verfügbaren Nahtstellensignale. Die genaue Beschreibung der Signale und die Erläuterung, welche Signale für die einzelnen Funktionen relevant sind, wird im Folgenden erläutert.

Signale an PLC-Achse

DB380x	Signale an PLC-Achse (PLC → NC) [r/w]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB3000	Start Positionierachse	Start Spindelpositionieren	Start Spindeldrehen	Start Spindelpendeln	-	-	-	-
DBB3001	-	-	Stop Spindeldrehen	Stop Spindelpendeln	-	-	-	-
DBB3002	Automatische Getriebeanwahl	konstante Schnittgeschwindigkeit	Drehrichtung wie M4	-	Handradüberlagerung ein	Verfahrmaß Inch (nicht metrisch)	Wegbedingung kürzester Weg (DC)	Wegbedingung inkrementell (IC)
DBB3003	Teilungsposition	-	-	-	-	-	Wegbedingung abs. pos. Richtung (ACP)	Wegbedingung abs. neg. Richtung (ACN)

DB380x	Signale an PLC-Achse (PLC → NC) [r/w]
DBD3004	Position (REAL, bei Teilungsachse: DINT)
DBD3008	Vorschubgeschwindigkeit (REAL), wenn < 0, wird Wert aus Maschinendatum POS_AX_VELO genommen

Die Bits der Wegbedingungen und der Drehrichtungsvorgabe geben den jeweiligen Positionier- oder Verfahrensmodus vor, es darf nur eines der Bits gesetzt werden:

Bedeutung	Zu setzende Wegbedingung
Positionieren absolut	Kein Mode-Bit gesetzt
Positionieren inkrementell	DBB3002.0 = 1
Positionieren kürzester Weg	DBB3002.1 = 1
Positionieren absolut, positive Anfahrriichtung	DBB3003.1 = 1
Positionieren absolut, negative Anfahrriichtung	DBB3003.0 = 1
Drehrichtung wie M4	DBB3002.5 = 1

Die übrigen Bits dienen dem Spezifizieren und Starten der jeweiligen Funktion, diese Funktionsbits sowie Position und Geschwindigkeit werden bei den einzelnen Funktionen näher erläutert.

Signale von PLC-Achse

DB390x	Signale von PLC-Achse (NC → PLC) [r]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB3000	Positionier- achse aktiv	Position er- reicht		-	-	-	Fehler wäh- rend Ver- fahren	Achse nicht startbar
DBB3001	-	-	-	-	-	-	-	-
DBB3002	-	-	-	-	-	-		-
DBB3003	Fehlernummer							

Für die Verwendung der nachfolgend aufgeführten Funktionen müssen folgende Voraussetzungen erfüllt sein:

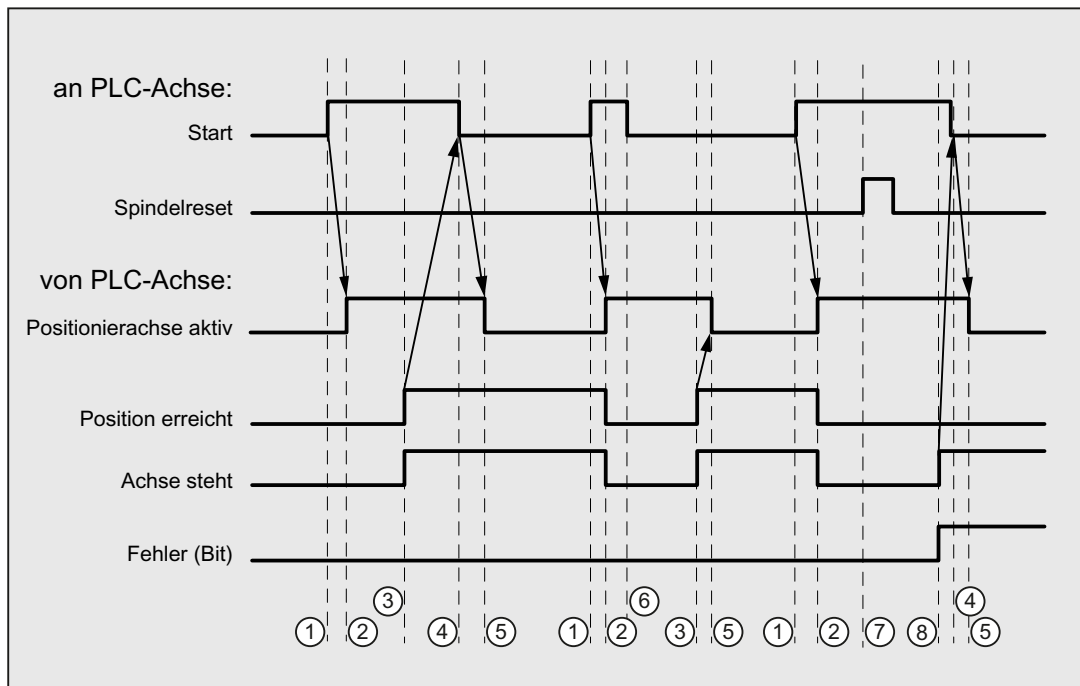
- Die Achse oder Spindel ist ordnungsgemäß der PLC zugeordnet.
- Regler- und Impulsfreigabe sind gesetzt.
- Nach Setzen aller Steuersignale, ist nur eines der Startsignale in DB380x.DBB3000 gesetzt.

15.6.4.4 Spindel positionieren

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	Es darf gleichzeitig immer nur eines der vier Signale gesetzt sein. Alle Signale 0: absolutes Positionieren
DBX3002.1	Verfahren auf kürzestem Weg	
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBD3004	Sollposition/Sollweg	Sollweg, wenn DBX3002.0 == 1

DB380x	Steuersignale PLC → NC	Bemerkung
DBD3008	Vorschubgeschwindigkeit	0: Verfahren wird mit dem Wert aus MD35300 \$MA_SPIND_POSCTRL_VELO
DBX3000.6	Start	Hinweis Das Rücksetzen des Signals führt nicht zum Stopp!
DBX2.2	Restweg löschen, Spindelreset	Abbruchsignal, beendet die Funktion

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	Auch dann 1, wenn Override = 0 oder Sollposition erreicht wenn Start = 1
DBX3000.6	Position erreicht	1: Sollposition mit "Genauhalt Fein" erreicht
DBX3000.0	Spindel nicht startbar	
DBX3000.1	Fehler während Verfahren	1: Fehler während des Verfahrens, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	
DBX1.4	Achse/Spindel steht	1: wenn $n < n_{\min}$

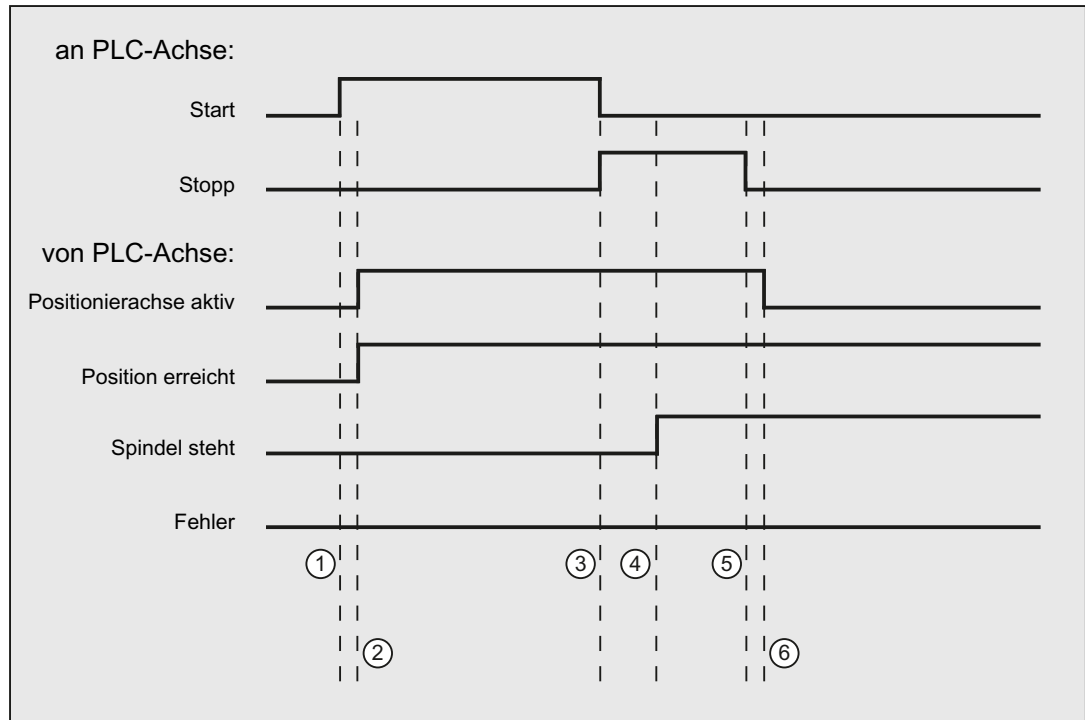


- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
- ② Meldung *Positionierachse aktiv* zeigt an, dass die Funktion aktiv ist und die Ausgangssignale gültig sind, *Position erreicht* und *Achse steht* werden gegebenenfalls zurückgenommen. Bei Wegvorgabe = 0 werden die Signale nicht zurückgenommen.
- ③ Das Erreichen der Position wird gemeldet (*Position erreicht*), *Spindel steht* wird gesetzt.
- ④ Der Anwender nimmt daraufhin *Start* zurück.
- ⑤ Daraufhin wird das Signal *Positionierachse aktiv* zurückgesetzt.
- ⑥ Der Anwender nimmt das Signal *Start* mit Erhalt des Signals *Positionierachse aktiv* sofort zurück.
- ⑦ Das Positionieren wird über das Setzen von *Spindel Reset* abgebrochen. Dieses Signal muss mindestens einen PLC-Takt anstehen.
- ⑧ Die Spindel kommt zum Stillstand (*Spindel steht*), das Signal *Fehler* wird gesetzt. (In diesem Fall wird die Fehlernummer 115 ausgegeben.)

15.6.4.5 Spindel drehen

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	-
DBX3002.1	Verfahren auf kürzestem Weg	-
DBX3002.5	Drehrichtung wie M4	1: Drehrichtung durch M4 vorgegeben 0: Drehrichtung durch M3 vorgegeben
DBX3003.0	absolut, negative Richtung	-
DBX3003.1	absolut, positive Richtung	-
DBD3008	Vorschubgeschwindigkeit	Spindeldrehzahl
DBX3000.5	Start Spindel drehen	-
DBX3001.5	Stopp Spindel drehen	-

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	1: bei Start oder Stopp == 1
DBX3000.6	Position erreicht	1: Funktion wurde ohne Fehler gestartet
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-
DBX1.4	Achse/Spindel steht	-

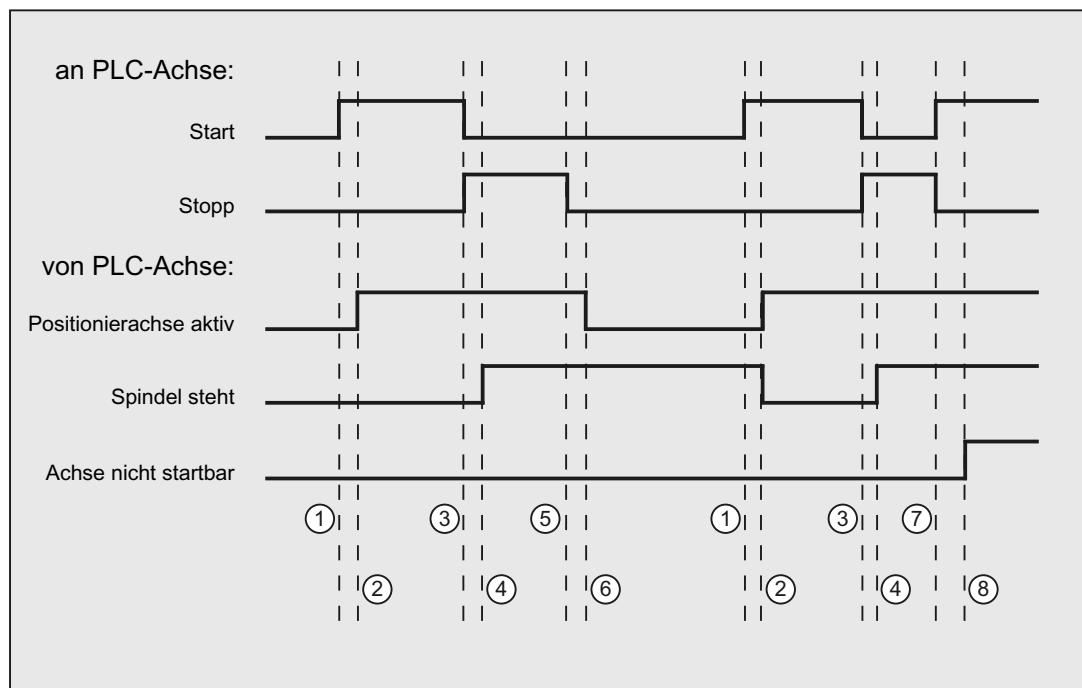


- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
- ② Meldungen *Positionierachse aktiv* und *Position erreicht* werden zurückgemeldet, *Position erreicht* ist dabei irrelevant.
- ③ Das Drehen der Spindel wird vom Anwender dadurch gestoppt, dass *Start* zurückgesetzt und *Stopp* gesetzt wird.
- ④ Die Spindel hält an und das Signal *Spindel steht* wird gesetzt.
- ⑤ Daraufhin setzt der Anwender *Stopp* zurück.
- ⑥ Das Rücksetzen von *Stopp* bewirkt das Rücksetzen von *Positionierachse aktiv*.

15.6.4.6 Spindel pendeln

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	inkrementell	Es darf keines der Bits gesetzt werden.
DBX3002.1	kürzester Weg	
DBX3002.5	Drehrichtung wie M4	
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBD3004	Sollgetriebestufe	MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE = 0 0 - 5: Pendeln MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE = 1 0: Pendeln 1: Pendeln mit Getriebestufenwechsel M41 2: Pendeln mit Getriebestufenwechsel M42 3: Pendeln mit Getriebestufenwechsel M43 4: Pendeln mit Getriebestufenwechsel M44 5: Pendeln mit Getriebestufenwechsel M45
DBD3008	Vorschubgeschwindigkeit	Beim Pendeln ohne Bedeutung! Die Pendeldrehzahl stammt aus dem Maschinendatum MD35400, \$MA_SPIND_OSCILL_DES_VELO.
DBX3000.5	Start Spindel pendeln	Start darf nicht unmittelbar auf Stopp folgen. Erst muss Stopp rückgesetzt werden (beide 0).
DBX3001.5	Stopp Spindel pendeln	

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	1: bei Start oder Stopp == 1
DBX3000.6	Position erreicht	1: nach Start
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBX3000.0	Achse nicht startbar	1: Fehler beim Start, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	
DBX1.4	Achse/Spindel steht	



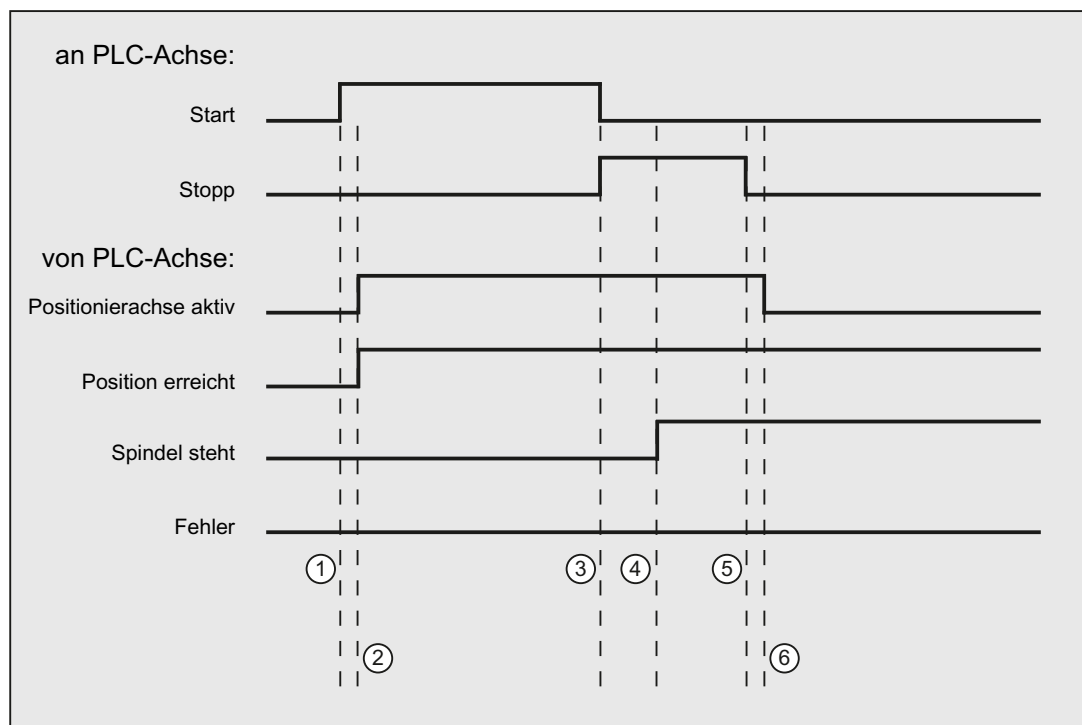
- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
Hinweis: Dies ist nur bei rückgesetztem Signal *Positionierachse aktiv* möglich!
- ② Meldungen *Positionierachse aktiv* und *Position erreicht* werden zurückgemeldet, *Position erreicht* ist dabei irrelevant und darum nicht eingezeichnet.
- ③ Das Pendeln der Spindel wird vom Anwender dadurch gestoppt, dass *Start* zurückgesetzt und *Stopp* gesetzt wird.
- ④ Die Spindel hält an und das Signal *Spindel steht* wird gesetzt.
- ⑤ Daraufhin setzt der Anwender *Stopp* zurück.
- ⑥ Das Rücksetzen von *Stopp* bewirkt das Rücksetzen von *Positionierachse aktiv*.
- ⑦ Im AWP wird *Stopp* zurückgesetzt und unrichtigerweise im gleichen PLC-Zyklus *Start* auch wieder gesetzt. Dadurch wird *Positionierachse aktiv* nicht rückgesetzt, sondern...
- ⑧ ...es wird das Signal *Achse nicht startbar* gesetzt (Fehlernummer 106).

15.6.4.7 Teilungsachse

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	inkrementelles Verfahren	Es darf gleichzeitig immer nur eines der Signale gesetzt sein. Alle Signale 0: absolutes Positionieren
DBX3002.1	Verfahren auf kürzestem Weg	
DBX3002.5	Drehrichtung wie M4	
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBX3003.7	Teilungsposition	Teilungsachse EIN
DBD3004	Sollposition/Sollweg	Sollweg, wenn DBX3002.0 == 1
DBD3008	Vorschubgeschwindigkeit	0: Verfahren wird mit dem Wert aus MD32060 \$MA_POS_AX_VELO

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3000.7	Start Positionierachse	Hinweis Das Rücksetzen des Signals führt nicht zum Stopp!
DBX2.2	Restweg löschen, Spindelreset	Abbruchsignal, beendet die Funktion

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	Auch dann 1, wenn Override = 0 oder Sollposition erreicht.
DBX3000.6	Position erreicht	1: Sollposition mit "Genauhalt Fein" erreicht.
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-
DBX1.4	Achse/Spindel steht	-

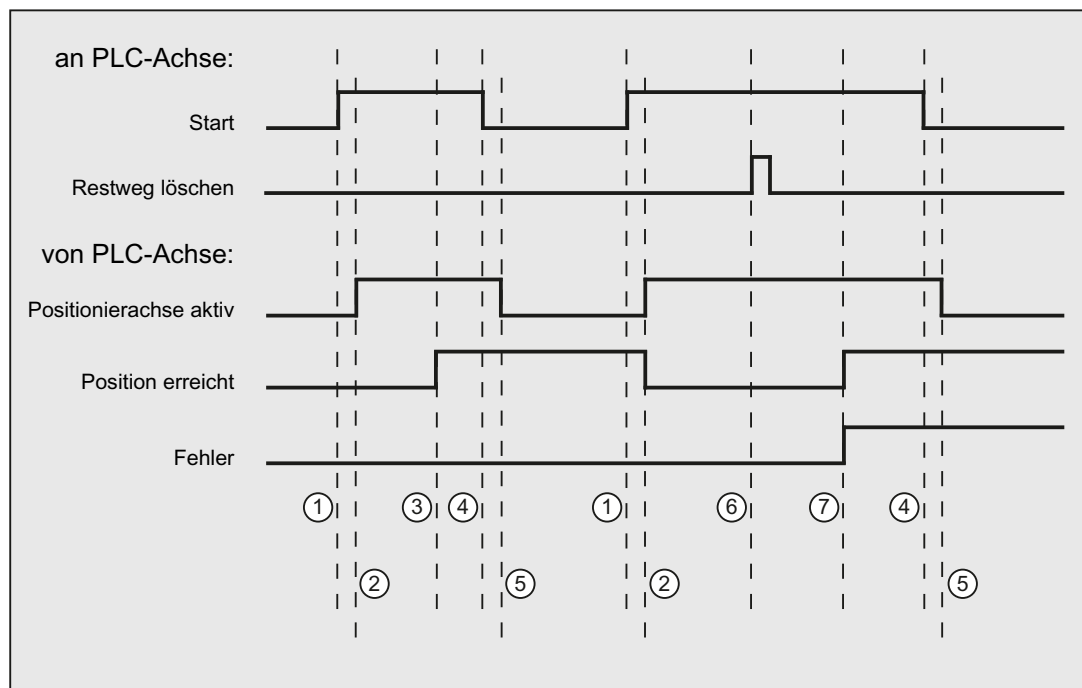


- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
Hinweis: Dies ist nur bei rückgesetztem Signal *Positionierachse aktiv* möglich!
- ② Meldungen *Positionierachse aktiv* und *Position erreicht* werden zurückgemeldet, *Position erreicht* ist dabei irrelevant.
- ③ Das Pendeln der Spindel wird vom Anwender dadurch gestoppt, dass *Start* zurückgesetzt und *Stopp* gesetzt wird.
- ④ Die Spindel hält an und das Signal *Spindel steht* wird gesetzt.
- ⑤ Daraufhin setzt der Anwender *Stopp* zurück.
- ⑥ Das Rücksetzen von *Stopp* bewirkt das Rücksetzen von *Positionierachse aktiv*.

15.6.4.8 Positionierachse metrisch

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	Es darf gleichzeitig immer nur eines der Signale gesetzt sein. Alle Signale 0: absolutes Positionieren
DBX3002.1	Verfahren auf kürzestem Weg	
DBX3002.5	Drehrichtung wie M4	
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBD3002.2	Verfahrmaß Inch	0: Verfahrmaß metrisch
DBX3002.3	Handradüberlagerung	0: Überlagerung AUS
DBD3004	Sollposition/Sollweg	Sollweg, wenn DBX3002.0 == 1
DBD3008	Vorschubgeschwindigkeit	0: Verfahren wird mit dem Wert aus MD32060 \$MA_POS_AX_VELO
DBX3000.7	Start Positionierachse	Hinweis Das Rücksetzen des Signals führt nicht zum Stopp!
DBX2.2	Restweg löschen, Spindelreset	Abbruchsignal, beendet die Funktion

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	Auch dann 1, wenn Override = 0 oder Sollposition erreicht.
DBX3000.6	Position erreicht	1: Achse hat Sollposition erreicht.
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-



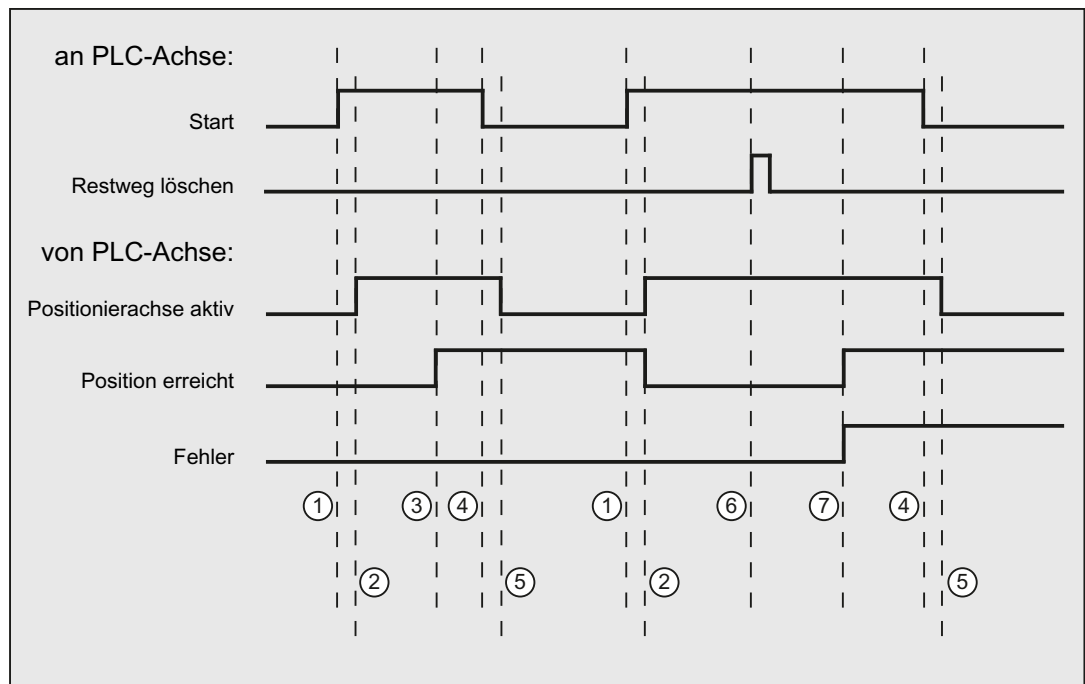
- ① Erster Funktionsanstoß durch positive Flanke von *Start*.
- ② *Positionierachse aktiv* = 1 zeigt an, dass die Funktion aktiv ist und die Ausgangssignale gültig sind, *Position erreicht* und *Achse steht* werden gegebenenfalls zurückgenommen.
- ③ Positive Quittung *Position erreicht* = 1 und *Positionierachse aktiv* = 1
- ④ Rücksetzen vom Funktionsanstoß nach Erhalt der Quittung
- ⑤ Signalwechsel durch Funktion
- ⑥ Positionierung wird durch Restweg löschen abgebrochen, Signaldauer mind. 1 PLC-Zyklus.
- ⑦ Die Signale *Position erreicht* und *Fehler* werden gesetzt, die *Fehlernummer* (in diesem Fall 30) kann gelesen werden.

15.6.4.9 Positionierachse Inch

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	Es darf gleichzeitig immer nur eines der Signale gesetzt sein. Alle Signale 0: absolutes Positionieren
DBX3002.1	Verfahren auf kürzestem Weg	
DBX3002.5	Drehrichtung wie M4	
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBX3002.2	Verfahrmaß Inch	0: Verfahrmaß metrisch
DBX3002.3	Handradüberlagerung	0: Überlagerung AUS
DBD3004	Sollposition/Sollweg	Sollweg, wenn DBX3002.0 == 1
DBD3008	Vorschubgeschwindigkeit	0: Verfahren wird mit dem Wert aus MD32060 \$MA_POS_AX_VELO

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3000.7	Start Positionierachse	Hinweis Das Rücksetzen des Signals führt nicht zum Stopp!
DBX2.2	Restweg löschen, Spindelreset	Abbruchsignal, beendet die Funktion

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	Auch dann 1, wenn Override = 0 oder Sollposition erreicht.
DBX3000.6	Position erreicht	1: Achse hat Sollposition erreicht.
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-

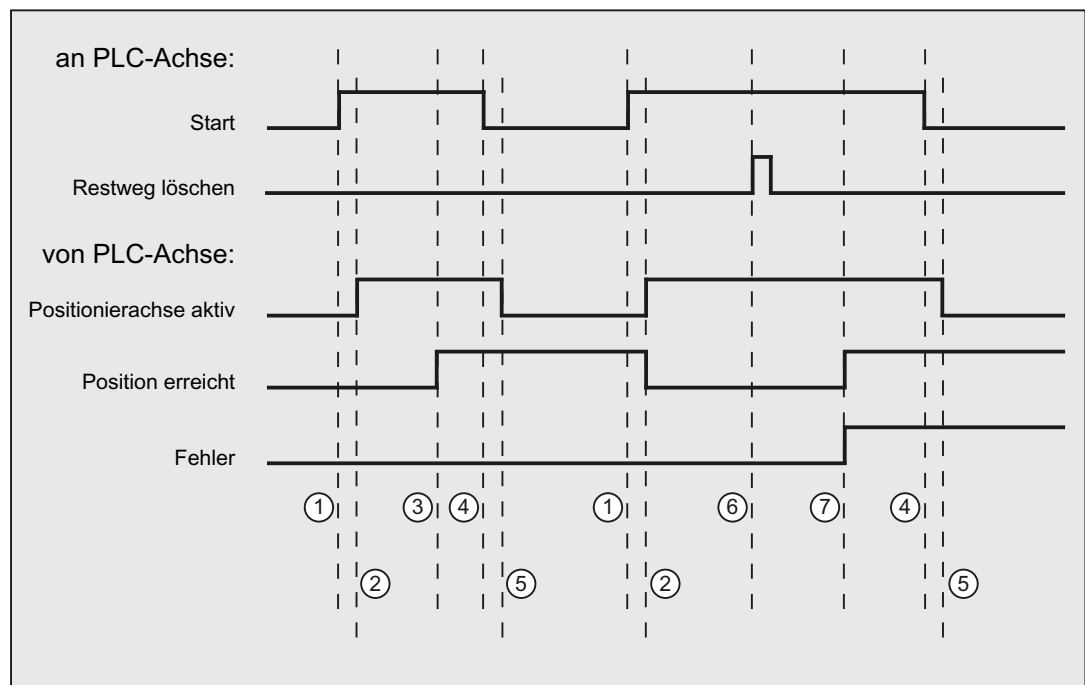


- ① Erster Funktionsanstoß durch positive Flanke von *Start*.
- ② *Positionierachse aktiv* = 1 zeigt an, dass die Funktion aktiv ist und die Ausgangssignale gültig sind, *Position erreicht* und *Achse steht* werden gegebenenfalls zurückgenommen.
- ③ Positive Quittung *Position erreicht* = 1 und *Positionierachse aktiv* = 1
- ④ Rücksetzen vom Funktionsanstoß nach Erhalt der Quittung
- ⑤ Signalwechsel durch Funktion
- ⑥ Positionierung wird durch Restweg löschen abgebrochen, Signaldauer mind. 1 PLC-Zyklus.
- ⑦ Die Signale *Position erreicht* und *Fehler* werden gesetzt, die *Fehlernummer* (in diesem Fall 30) kann gelesen werden.

15.6.4.10 Positionierachse metrisch mit Handradüberlagerung

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	Es darf gleichzeitig immer nur eines der Signale gesetzt sein. Alle Signale 0: absolutes Positionieren
DBX3002.1	Verfahren auf kürzestem Weg	
DBX3002.5	Drehrichtung wie M4	
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBX3002.2	Verfahrmaß Inch	0: Verfahrmaß metrisch
DBX3002.3	Handradüberlagerung	0: Überlagerung AUS
DBD3004	Sollposition/Sollweg	Sollweg, wenn DBX3002.0 == 1
DBD3008	Vorschubgeschwindigkeit	0: Verfahren wird mit dem Wert aus MD32060 \$MA_POS_AX_VELO
DBX3000.7	Start Positionierachse	Hinweis Das Rücksetzen des Signals führt nicht zum Stopp!
DBX2.2	Restweg löschen, Spindelreset	Abbruchsignal, beendet die Funktion

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	Auch dann 1, wenn Override = 0 oder Sollposition erreicht.
DBX3000.6	Position erreicht	1: Achse hat Sollposition erreicht.
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-



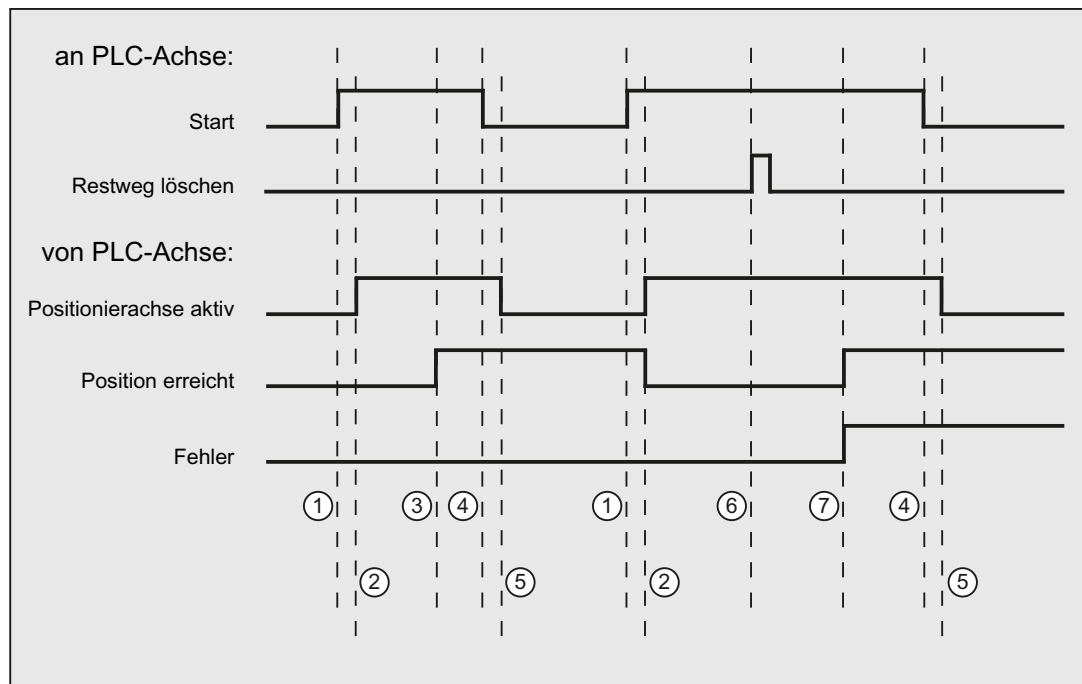
- ① Erster Funktionsanstoß durch positive Flanke von *Start*.
- ② *Positionierachse aktiv* = 1 zeigt an, dass die Funktion aktiv ist und die Ausgangssignale gültig sind, *Position erreicht* und *Achse steht* werden gegebenenfalls zurückgenommen.
- ③ Positive Quittung *Position erreicht* = 1 und *Positionierachse aktiv* = 1
- ④ Rücksetzen vom Funktionsanstoß nach Erhalt der Quittung
- ⑤ Signalwechsel durch Funktion
- ⑥ Positionierung wird durch Restweg löschen abgebrochen, Signaldauer mind. 1 PLC-Zyklus.
- ⑦ Die Signale *Position erreicht* und *Fehler* werden gesetzt, die *Fehlernummer* (in diesem Fall 30) kann gelesen werden.

15.6.4.11 Positionierachse Inch mit Handradüberlagerung

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	Es darf gleichzeitig immer nur eines der Signale gesetzt sein.
DBX3002.1	Verfahren auf kürzestem Weg	
DBX3002.5	Drehrichtung wie M4	Alle Signale 0: absolutes Positionieren
DBX3003.0	absolut, negative Richtung	
DBX3003.1	absolut, positive Richtung	
DBX3002.2	Verfahrmaß Inch	0: Verfahrmaß metrisch
DBX3002.3	Handradüberlagerung	0: Überlagerung AUS
DBD3004	Sollposition/Sollweg	Sollweg, wenn DBX3002.0 == 1
DBD3008	Vorschubgeschwindigkeit	0: Verfahren wird mit dem Wert aus MD32060 \$MA_POS_AX_VELO

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3000.7	Start Positionierachse	Hinweis Das Rücksetzen des Signals führt nicht zum Stopp!
DBX2.2	Restweg löschen, Spindelreset	Abbruchsignal, beendet die Funktion

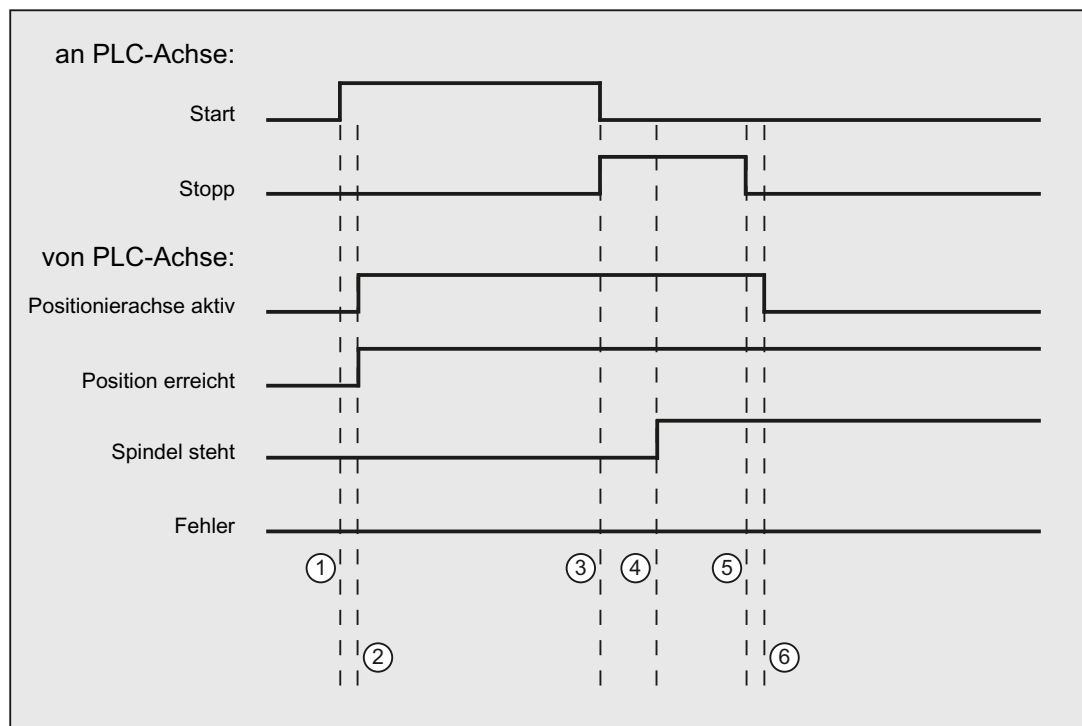
DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	Auch dann 1, wenn Override = 0 oder Sollposition erreicht.
DBX3000.6	Position erreicht	1: Achse hat Sollposition erreicht.
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-



15.6.4.12 Spindel drehen mit automatischer Getriebestufenanwahl

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	-
DBX3002.1	Verfahren auf kürzestem Weg	-
DBX3002.5	Drehrichtung wie M4	1: Drehrichtung durch M4 vorgegeben 0: Drehrichtung durch M3 vorgegeben
DBX3003.0	absolut, negative Richtung	-
DBX3003.1	absolut, positive Richtung	-
DBX3002.7	automatische Getriebestufenwahl	1: automatische Getriebestufenanwahl EIN
DBD3008	Vorschubgeschwindigkeit	Spindeldrehzahl
DBX3000.5	Start Spindel drehen	-
DBX3001.5	Stopp Spindel drehen	-

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	1: bei Start oder Stopp == 1,
DBX3000.6	Position erreicht	1: Solldrehzahl wird ausgegeben
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-
DBX1.4	Achse/Spindel steht	-

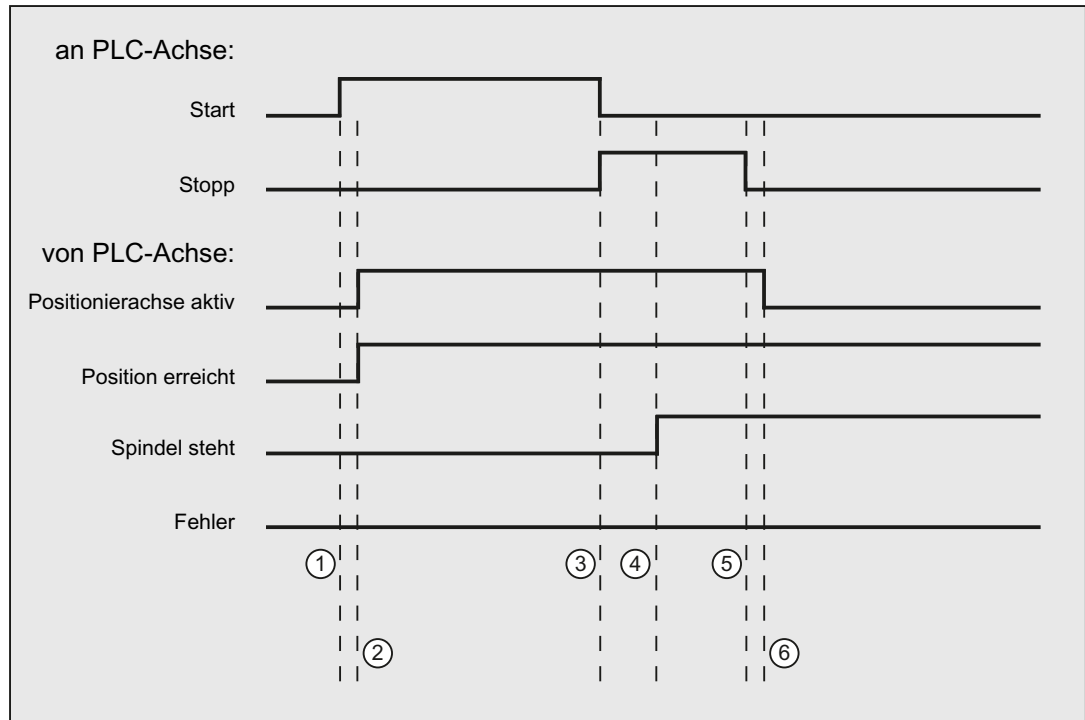


- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
- ② Meldungen *Positionierachse aktiv* und *Position erreicht* werden zurückgemeldet, *Position erreicht* ist dabei irrelevant.
- ③ Das Drehen der Spindel wird vom Anwender dadurch gestoppt, dass *Start* zurückgesetzt und *Stopp* gesetzt wird.
- ④ Die Spindel hält an und das Signal *Spindel steht* wird gesetzt.
- ⑤ Daraufhin setzt der Anwender *Stopp* zurück.
- ⑥ Das Rücksetzen von *Stopp* bewirkt das Rücksetzen von *Positionierachse aktiv*.

15.6.4.13 Spindel drehen mit konstanter Schnittgeschwindigkeit [m/min]

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	-
DBX3002.1	Verfahren auf kürzestem Weg	-
DBX3002.5	Drehrichtung wie M4	1: Drehrichtung durch M4 vorgegeben 0: Drehrichtung durch M3 vorgegeben
DBX3003.0	absolut, negative Richtung	-
DBX3003.1	absolut, positive Richtung	-
DBX3002.2	Verfahrmaß Inch	0 Verfahrmaß metrisch
DBX3002.6	konst. Schnittgeschwindigkeit	1 konstante Schnittgeschwindigkeit EIN
DBD3008	Vorschubgeschwindigkeit	Spindeldrehzahl
DBX3000.5	Start Spindel drehen	-
DBX3001.5	Stopp Spindel drehen	-

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	1: bei Start oder Stopp == 1,
DBX3000.6	Position erreicht	1: Solldrehzahl wird ausgegeben
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-
DBX1.4	Achse/Spindel steht	-



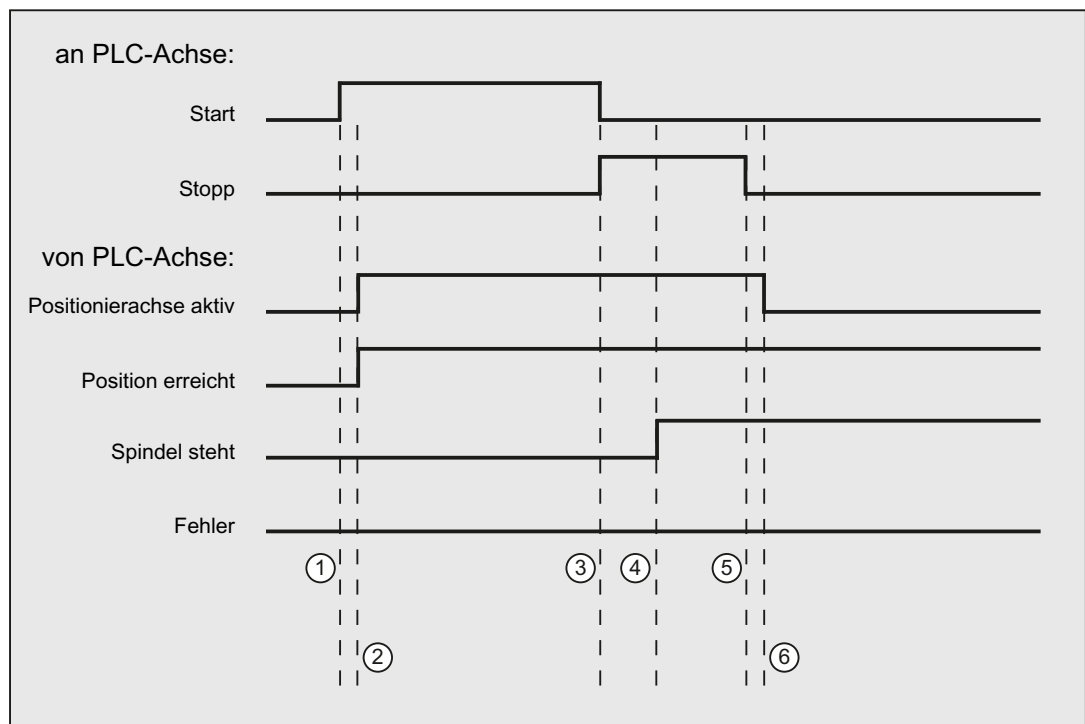
- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
- ② Meldungen *Positionierachse aktiv* und *Position erreicht* werden zurückgemeldet, *Position erreicht* ist dabei irrelevant.
- ③ Das Drehen der Spindel wird vom Anwender dadurch gestoppt, dass *Start* zurückgesetzt und *Stopp* gesetzt wird.
- ④ Die Spindel hält an und das Signal *Spindel steht* wird gesetzt.
- ⑤ Daraufhin setzt der Anwender *Stopp* zurück.
- ⑥ Das Rücksetzen von *Stopp* bewirkt das Rücksetzen von *Positionierachse aktiv*.

15.6.4.14 Spindel drehen mit konstanter Schnittgeschwindigkeit [feet/min]

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.0	Inkrementelles Verfahren	-
DBX3002.1	Verfahren auf kürzestem Weg	-

DB380x	Steuersignale PLC → NC	Bemerkung
DBX3002.5	Drehrichtung wie M4	1: Drehrichtung durch M4 vorgegeben 0: Drehrichtung durch M3 vorgegeben
DBX3003.0	absolut, negative Richtung	-
DBX3003.1	absolut, positive Richtung	-
DBX3002.2	Verfahrmaß Inch	1: Verfahrmaß inch
DBX3002.6	konst. Schnittgeschwindigkeit	1: konstante Schnittgeschwindigkeit EIN
DBD3008	Vorschubgeschwindigkeit	Spindeldrehzahl
DBX3000.5	Start Spindel drehen	-
DBX3001.5	Stopp Spindel drehen	-

DB390x	Statussignale NC → PLC	Bemerkung
DBX3000.7	Positionierachse aktiv	1: bei Start oder Stopp == 1
DBX3000.6	Position erreicht	1: Solldrehzahl wird ausgegeben
DBX3000.1	Fehler	1: Fehler beim Verfahren, Fehlernummer in DBB3003 auswerten!
DBB3003	Fehlernummer	-
DBX1.4	Achse/Spindel steht	-



- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
- ② Meldungen *Positionierachse aktiv* und *Position erreicht* werden zurückgemeldet, *Position erreicht* ist dabei irrelevant.
- ③ Das Drehen der Spindel wird vom Anwender dadurch gestoppt, dass *Start* zurückgesetzt und *Stopp* gesetzt wird.
- ④ Die Spindel hält an und das Signal *Spindel steht* wird gesetzt.
- ⑤ Daraufhin setzt der Anwender *Stopp* zurück.
- ⑥ Das Rücksetzen von *Stopp* bewirkt das Rücksetzen von *Positionierachse aktiv*.

15.6.4.15 Fehlermeldungen

Konnte eine Funktion nicht ausgeführt werden, werden je nach Fehler folgende Signale gesetzt:

- DB390x.DBX3000.0 == 1 (Achse nicht startbar)
- DB390x.DBX3000.1 == 1 (Fehler während Verfahren)

Die genaue Fehlerursache wird angezeigt über:

- DB390x.DBB3003 (Fehlernummer)

Fehlernummer		Bedeutung
Dezimal	Hex	
1	01	Es wurden mehrere Funktionen der Achse/Spindel gleichzeitig aktiviert
20	14	Es wurde eine Funktion gestartet, ohne dass die Position erreicht wurde
30	1E	Die Achse/Spindel wurde vor Ende der Bewegung an die NC abgegeben
40	28	Die Achse ist durch NC Programm programmiert, NC interner Fehler

Fehlernummer		Bedeutung
Dezimal	Hex	
50	32	Fest zugeordnete PLC Achse: verfährt (JOG) oder referenziert
60	3C	Fest zugeordnete PLC Achse: Kanalzustand lässt keinen Start zu
100	64	Falsche Position bei Achse/Spindel programmiert (Alarmnummer ¹⁾ 16830)
101	65	Programmierte Drehzahl ist zu hoch
102	66	Falscher Wertebereich konstante Schnittgeschwindigkeit (Alarmnummer ¹⁾ 14840)
104	68	Folgespindel: unerlaubte Programmierung (Alarmnummer ¹⁾ 22030)
105	69	Kein Messsystem vorhanden (Alarmnummer ¹⁾ 16770)
106	6A	Positioniervorgang der Achse noch aktiv (Alarmnummer ¹⁾ 22052)
107	6B	Referenzmarke nicht gefunden (Alarmnummer ¹⁾ 22051)
108	6C	Kein Übergang von der Drehzahlregelung in die Lageregelung (Alarmnummer ¹⁾ 22050)
109	6D	Referenzmarke nicht gefunden (Alarmnummer ¹⁾ 22051)
110	6E	Geschwindigkeit/Drehzahl ist negativ
111	6F	Soll-Drehzahl ist Null
112	70	Ungültige Getriebestufe
115	73	Programmierte Position wurde nicht erreicht
117	75	in der NC ist G96/G961 nicht aktiv
118	76	in der NC ist G96/G961 noch aktiv
120	78	Achse ist keine Teilungsachse (Alarmnummer ¹⁾ 20072)
121	79	Teilungsposition- Fehler (Alarmnummer ¹⁾ 17510)
125	7D	DC (kürzester Weg) nicht möglich (Alarmnummer ¹⁾ 16800)
126	7E	Absolutwert Minus nicht möglich (Alarmnummer ¹⁾ 16820)
127	7F	Absolutwert Plus nicht möglich (Alarmnummer ¹⁾ 16810)
128	80	Keine Planachse für Durchmesserprogrammierung vorhanden (Alarmnummer ¹⁾ 16510)
130	82	Software-Endschalter Plus (Alarmnummer ¹⁾ 20070)
131	83	Software-Endschalter Minus (Alarmnummer ¹⁾ 20070)
132	84	Arbeitsfeldbegrenzung Plus (Alarmnummer ¹⁾ 20071)
133	85	Arbeitsfeldbegrenzung Minus (Alarmnummer ¹⁾ 20071)
134	85	Frame ist unzulässig für Teilungsachse
135	87	Teilungsachse mit "Hirth-Verzahnung" ist aktiv (Alarmnummer ¹⁾ 17501)
136	88	Teilungsachse mit "Hirth-Verzahnung" aktiv und Achse nicht referenziert (Alarmnummer ¹⁾ 17503)
137	89	Spindelbetrieb für transformierte Spindel/Achse nicht möglich (Alarmnummer ¹⁾ 22290)
138	8A	für die Achse wird die entsprechende wirksame Koordinatensystem-spezifische Arbeitsfeldbegrenzung Plus verletzt (Alarmnummer ¹⁾ 20082)
139	8B	für die Achse wird die entsprechende wirksame Koordinatensystem-spezifische Arbeitsfeldbegrenzung Minus verletzt (Alarmnummer ¹⁾ 20082)
200	C8	System-Alarmnummer ¹⁾ 450007

1) Die ausführliche Alarmbeschreibung findet sich in: Diagnosehandbuch Alarme; SINUMERIK 828D, SINAMICS S120

15.6.5 ASUP starten

15.6.5.1 Auftragsstart

DB340x	ASUP: Auftrag [r/w] (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	-	-	-	-	-	-	-	Start INT1
DBB1	-	-	-	-	-	-	-	Start INT2
DBB2	-	-	-	-	-	-	-	Start INT3
DBB3	-	-	-	-	-	-	-	Start INT4

Interrupts

"Start INT1":

- DBX0.0 = 1: Anforderung zum Starten des Interruptprogramms (ASUP), das INT1 zugeordnet ist.
- DBX0.0 = 0: Rücksetzen der ASUP-Anforderung nachdem die Quittierung in der Ergebnisschnittstelle DB340x.DBB1000.0 - 3 erfolgt ist.

"Start INT2":

- DBX1.0 = 1: Anforderung zum Starten des Interruptprogramms (ASUP), das INT2 zugeordnet ist.
- DBX1.0 = 0: Rücksetzen der ASUP-Anforderung nachdem die Quittierung in der Ergebnisschnittstelle DB340x.DBB1001.0 - 3 erfolgt ist.

"Start INT3":

- DBX2.0 = 1: Anforderung zum Starten des Interruptprogramms (ASUP), das INT3 zugeordnet ist.
- DBX2.0 = 0: Rücksetzen der ASUP-Anforderung nachdem die Quittierung in der Ergebnisschnittstelle DB340x.DBB1002.0 - 3 erfolgt ist.

"Start INT4":

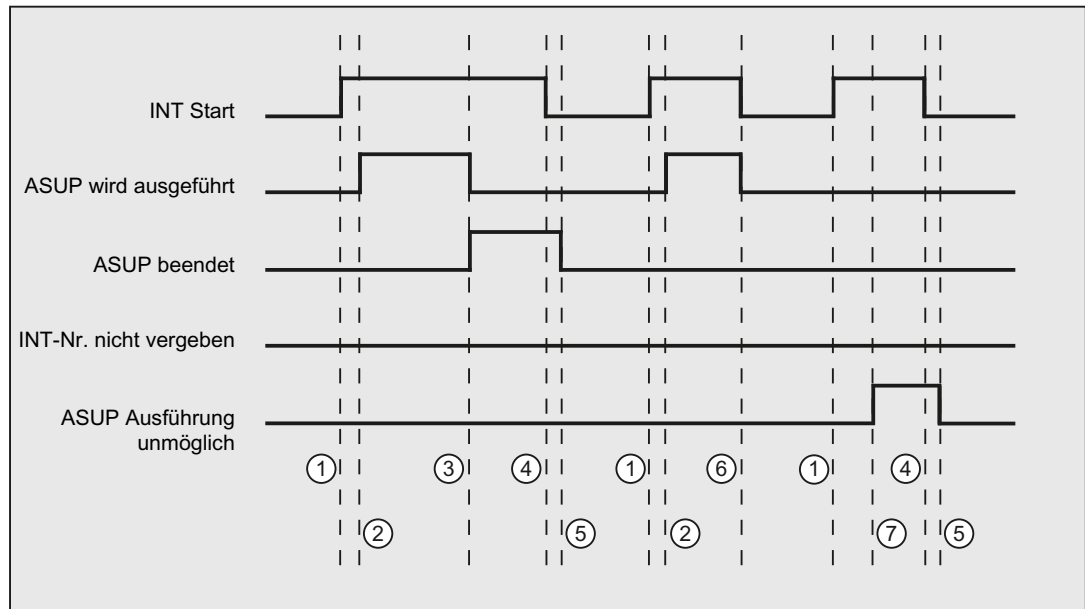
- DBX3.0 = 1: Anforderung zum Starten des Interruptprogramms (ASUP), das INT4 zugeordnet ist.
- DBX3.0 = 0: Rücksetzen der ASUP-Anforderung nachdem die Quittierung in der Ergebnisschnittstelle DB340x.DBB1003.0 - 3 erfolgt ist.

15.6.5.2 Auftragsergebnis

DB340x			ASUP: Ergebnis [r] (NC → PLC)					
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB1000	Signale zu Start INT1 (DB340x.DBX0.0)							
					ASUP-Ausführung nicht möglich ¹⁾	Interrupt-Nr nicht vergeben ²⁾	ASUP wird ausgeführt	ASUP beendet ³⁾
DBB1001	Signale zu Start INT2 (DB340x.DBX1.0)							
					ASUP-Ausführung nicht möglich ¹⁾	Interrupt-Nr nicht vergeben ²⁾	ASUP wird ausgeführt	ASUP beendet ³⁾
DBB1002	Signale zu Start INT3 (DB340x.DBX2.0)							
					ASUP-Ausführung nicht möglich ¹⁾	Interrupt-Nr nicht vergeben ²⁾	ASUP wird ausgeführt	ASUP beendet ³⁾
DBB1003	Signale zu Start INT4 (DB340x.DBX3.0)							
					ASUP-Ausführung nicht möglich ¹⁾	Interrupt-Nr nicht vergeben ²⁾	ASUP wird ausgeführt	ASUP beendet ³⁾
¹⁾ Negative Quittung: z. B. bei Not-Halt oder Kanal-Reset-Anforderung. ²⁾ Negative Quittung: Die nummernmäßige Zuordnung ist noch nicht erfolgt. Abhilfe: PI-Dienst "ASUP" ausführen. ³⁾ Positive Quittung: ASUP erfolgreich beendet. Start-Signal zurücksetzen: DB340x.DBXn.0 = 0, mit n = 0, 1, 2, 3								

15.6.5.3 Signalfluss

Signalfluss



- ① Funktionsanstoß vom Anwender durch positive Flanke von *Start*.
- ② *ASUP wird ausgeführt* wird zurückgemeldet.
- ③ Die Quittung *ASUP beendet* zeigt die erfolgreiche Ausführung an, *ASUP wird ausgeführt* wird zurückgenommen.
- ④ Das Signal zum Funktionsanstoß wird nach Erhalt der Quittung vom Anwender zurückgesetzt.
- ⑤ Signalwechsel durch Firmware.
- ⑥ Nicht zulässig! Wird Funktionsanstoß vor Erhalt der Quittung rückgesetzt, werden die Ausgangssignale nicht aktualisiert – ohne Einfluss auf Ablauf der angestoßenen Funktion.
- ⑦ *ASUP-Ausführung nicht möglich*: negative Quittung, Fehler aufgetreten.

Bild 15-2 Beispiel: Signalfluss

15.6.6 Kanalanwahl am HMI

Funktion

Der am HMI angezeigte Kanal, z.B. im Maschinengrundbild, kann vom PLC-Anwenderprogramm aus über die HMI/PLC-Nahtstelle angewählt werden.

Voraussetzung

In der NC ist mehr als ein Kanal parametrierbar.

Auftrags- und Quittierungsschnittstelle

DB1900			Kanalwahl [r/w] (PLC ↔ HMI)					
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB5011	Fehlerkennung (siehe Absatz unten) (HMI → PLC)							
...	...							
DBB5021	Status (HMI → PLC)	Anforde- rung (PLC ↔ HMI)	Funktionsnummer (PLC → HMI)					
DBB5022	Kanalnummer (siehe Absatz unten) (PLC → HMI)							

Kanalnummer

- Kanalnummer: 1, 2, ... max. Anzahl Kanäle
- Nächster Kanal: FF_H

Fehlerkennung

- 0: Kein Fehler
- 1: Funktionsnummer ungültig (DBX32.0 - .5)
- 2: Parameter ungültig (DBB33 - DBB35)
- 3: Das Schreiben der HMI-internen Variable ist fehlgeschlagen
- 10: Kanal nicht vorhanden (DBB33)

Funktionsablauf

PLC → HMI

Vom PLC-Anwenderprogramm ist folgender Ablauf einzuhalten:

1. Prüfen, ob die Schnittstelle für einen neuen Auftrag frei ist:
 - DB1900.DBX**5021**.6 == 0 (Funktionsanforderung)
 - DB1900.DBX**5021**.7 == 0 (Status)
2. Ist die Schnittstelle frei, sind die Auftragsdaten einzutragen und die Funktionsanforderung zu setzen:
 - DB1900.DBB**5022** = <Kanalnummer>
 - DB1900.DBX**5021**.0 - .5 = 1 (Funktionsnummer: Kanalwahl)
 - DB1900.DBX**5021**.6 = 1 (Funktionsanforderung)

HMI → PLC

Vom HMI erfolgen bei **fehlerfreier** Parametrierung folgende Reaktionen:

1. Nachdem HMI die Funktionsanforderung zur Kanalanwahl erkennt hat, wird der Status auf "Funktion wird ausgeführt" gesetzt und die Funktionsanforderung zurückgesetzt:
 - DB1900.DBX**5021**.7 = 1 (Status)
 - DB1900.DBX**5021**.6 = 0 (Funktionsanforderung)
2. Nachdem die Kanalanwahl durchgeführt wurde, wird der Status wieder zurückgesetzt und als Fehlerkennung der Wert 0 gesetzt:
 - DB1900.DBX**5021**.7 = 0 (Status)
 - DB1900.DBX**5011** = 0 (Fehlerkennung)

Vom HMI erfolgen bei **fehlerhafter** Parametrierung folgende Reaktionen:

- Die Funktionsanforderung wird zurückgesetzt und die entsprechende Fehlerkennung gesetzt:
 - DB1900.DBX**5021**.6 = 0 (Funktionsanforderung)
 - DB1900.DBX**5021**.7 = 0 (Status)
 - DB1900.DBX**5011** = <Fehlerkennung>

15.7 CNC-Sperrfunktion (Option)

15.7.1 Funktion

Hinweis

Die "CNC-Sperrfunktion" ist eine lizenzpflichtige Option (Artikelnummer: 6FC5800-0AP76-0YB0).

Für die Verwendung der CNC-Sperrfunktion muss die entsprechenden Lizenz bei SIEMENS erworben werden. Die Verwendung der Sperrfunktion über eine Trial Lizenz ist **nicht** möglich.

ACHTUNG
Lizenzzertifikat
Der Einrichter der CNC-Sperrfunktion (Maschinenhersteller bzw. Händler) muss das Lizenzzertifikat für diese Option (CoL) behalten!
Mit Hilfe dieses Zertifikats erfolgt im Fall einer vergessenen PIN die Legitimierung gegenüber SIEMENS. Der Besitzer des Zertifikats (CoL) kann die Sperre der Maschine aufheben lassen.

Mit der "CNC-Sperrfunktion" kann der Maschinenhersteller mit Hilfe einer verschlüsselten Datei, die er mit der SINUMERIK Integrate-Applikation Access MyMachine (AMM) erzeugt hat, ein Sperrdatum in der Steuerung aktivieren. Darüber kann die Nutzung der Maschine auf die

Zeit bis zum Erreichen des Sperrdatums begrenzt werden. Bei Überschreiten des Sperrdatums wird die Funktion NC-Start der Steuerung gesperrt.

Mit einer weiteren verschlüsselten Datei kann die CNC-Sperrfunktion verlängert oder deaktiviert werden. Diese Datei schickt der Maschinenhersteller dem Endkunden, wenn dieser seinen vereinbarten Verpflichtungen nachgekommen ist.

15.7.2 Voraussetzungen

Für die Nutzung der "CNC-Sperrfunktion" müssen folgende Voraussetzungen erfüllt sein:

- Die Option "CNC-Sperrfunktion" muss gesetzt sein.
- Es muss ein PLC-Projekt vom CPU-Typ "828D Step 2 x.yy" verwendet werden. Der Kompatibilitätsmodus muss ausgeschaltet sein.
- Die SINUMERIK Integrate-Applikation Access MyMachine/ P2P (PC) muss installiert sein.

15.7.3 Einschränkungen

Die CNC-Sperrfunktion unterstützt das Geschäftsmodell mit zeitlich limitierter Nutzung. Der Schutz dient dazu, eine nicht autorisierte Nutzung über den eingestellten Zeitrahmen hinaus zu unterbinden. Durch den direkten Zugang zur CNC-Steuerung eröffnen sich aber ggf. Möglichkeiten zur Umgehung der Funktion. Die CNC-Sperrfunktion bietet keinen absoluten Schutz gegen Manipulation. Die nicht autorisierte Nutzung der Maschine wird durch die Verriegelung des Automatikbetriebs der CNC-Steuerung unterbunden. Da dabei ein laufendes Automatikprogramm nicht unterbrochen wird, kann dies zu einer Verlängerung der Laufzeit über das Sperrdatum hinaus führen. Alle anderweitigen Funktionen der SINUMERIK-Steuerung stehen weiterhin zur Verfügung.

Damit die CNC-Sperrfunktion ihre Wirkung entfalten kann, ist die Unterstützung durch den Maschinenhersteller nötig. Bei der Anwendung der CNC-Sperrfunktion sind daher folgender Hinweis und folgende Randbedingungen unbedingt zu beachten!

Hinweis

Die CNC-Sperrfunktion baut auf eine Bindung des PLC-Projekts an die entsprechende SINUMERIK-Steuerung auf. Die SINUMERIK-Steuerung besteht dabei aus einer Kombination von Panel Processing Unit (PPU) und CF-Card mit Systemsoftware.

Randbedingungen

- Die CNC-Sperrfunktion kann durch Manipulationsversuche bzw. Inkonsistenzen zum Stillstand der Maschine führen.
- Durch den Einsatz der CNC-Sperrfunktion können zusätzliche Service-Einsätze des Maschinenherstellers bzw. Händlers beim Kunden nötig werden.
- Das PLC-Projekt sollte niemals ohne hinterlegte OEM-PIN zum Kunden gegeben werden. Die Nutzung eines "freien" PLC-Projekts ermöglicht die Umgehung der CNC-Sperrfunktion!
- Eine erneute Implementierung des PLC-Projekts der entsprechenden Werkzeugmaschine ermöglicht eine Umgehung der CNC-Sperrfunktion!

- Der Schutz für die Programmorganisationseinheiten in der PLC der SINUMERIK ist unbedingt zu nutzen. Die Aktivierung ist im PLC Programming Tool möglich. Die Codierung des PLC-Projekts muss geheim gehalten werden.
- Für einen besseren Schutz muss für jede einzelne Maschine eine eigenständige OEM-PIN vergeben werden.
- Vor dem ersten Einrichten der Sperrfunktion (siehe Kapitel "Ersteinrichtung der CNC-Sperrfunktion (Seite 1227)") muss der Einrichter das Datum und die Uhrzeit in der SINUMERIK-Steuerung korrekt stellen. Steht das Datum in der Vergangenheit, verlängert sich die Laufzeit der Maschine entsprechend um die Differenz zum realen Datum.
- Die CNC-Sperrfunktion baut auf die Echtzeituhr der SINUMERIK 828D auf. Aufgrund des wartungsfreien Aufbaus der SINUMERIK 828D kann die Uhrzeit verloren gehen. Die CNC-Sperrfunktion führt eine Plausibilitätsprüfung der Uhrzeit durch. Diese Prüfung kann durch Spannungsverlust der Echtzeituhr beeinträchtigt werden. In diesem Fall wird die Zeit ohne Spannungsversorgung nicht mitgezählt.
- Der Einrichter der CNC-Sperrfunktion behält das zugehörige Lizenzzertifikat (CoL). Es weist ihn gegenüber SIEMENS als berechtigten Einrichter der Funktion und damit implizit als berechtigten Besitzer der Maschine aus.
- Aufgrund einer Softwarefehlfunktion kann es zur unbeabsichtigten Sperrung der Steuerung kommen.

15.7.4 Schutz vor Manipulation

Die CNC-Sperrfunktion dient dazu, im Rahmen des heute zu erwartenden Gebrauchs und Missbrauchs die Nutzung der CNC-Steuerung nur innerhalb des eingestellten Zeitrahmens zu erlauben. Trotz der vorhandenen Schutzmechanismen gegen unerlaubte Manipulation der CNC-Sperrfunktion verbleibt ein Restrisiko, das die Schutzmechanismen umgangen werden können. Die CNC-Sperrfunktion überprüft zyklisch die installierte Kombination aus PPU, CF-Card und PLC-Projekt. Beim Tausch aller drei Bestandteile entfällt die Sperrfunktion. Zum Schutz vor Manipulation müssen daher zwingend die Randbedingungen aus dem Kapitel "Einschränkungen (Seite 1226)" eingehalten werden.

Hinweis

Die CNC-Sperrfunktion nutzt kryptografische Verfahren. Die eingesetzten Verfahren entsprechen zum Zeitpunkt der ersten Vermarktung dem Stand der Technik. Mit fortschreitender Zeit erhöht sich die Wahrscheinlichkeit, dass diese Verfahren umgangen werden können.

15.7.5 Ersteinrichtung der CNC-Sperrfunktion

Die Ersteinrichtung der CNC-Sperrfunktion verkoppelt die Steuerungshardware, die Panel Processing Unit (PPU), zusammen mit der Systemsoftware auf der CF-Card und dem zur Maschine gehörenden PLC-Projekt. Gleichzeitig wird ein erstes Sperrdatum gesetzt.

Zur Einrichtung der Sperrfunktion ist die Erzeugung einer zur Hardware passenden verschlüsselten Lockset-Datei (lockset.clc) erforderlich. Die Datei wird mit Hilfe der SINUMERIK Integrate-Applikation Access MyMachine (AMM) erzeugt.

Erzeugen der Lockset-Datei

Zur Erstellung der Lockset-Datei werden folgende Daten benötigt:

- Seriennummer der CF-Card
- Seriennummer der Steuerung (Panel Processing Unit, PPU)
- OEM-PIN
- Sperrdatum

Hinweis

Die **Seriennummern** der CF-Card und der Steuerung (PPU) finden sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Hardware" > "NCU/PLC" > "ncu1":

- **CF-Card:** Bereich "CF-Karte" > "SerialNo."
 - **PPU:** Bereich "SINUMERIK 828D PPU" > "SerialNo."
-

Hinweis

Die **OEM-PIN** dient zur Erhöhung der Sicherheit vor Manipulation der CNC-Sperrfunktion.

Die OEM-PIN wird vom System beim Aktivieren der CNC-Sperrfunktion im PLC-Anwenderprogramm hinterlegt. Die OEM-PIN kann nicht vom Anwender im PLC-Anwenderprogramm eingesehen oder geändert bzw. gelöscht werden.

Die Daten werden über Access MyMachine im Dialogfenster "CNC-Sperrfunktion" eingegeben (Aufruf über das Hauptmenü: "Extras" > "CNC-Sperrfunktion"). Die Schaltfläche "**Aktivieren**" muss angewählt sein:

Nach Eingabe aller erforderlichen Daten wird mit der Schaltfläche "Lockset-Datei erstellen ..." die Lockset-Datei zum Aktivieren erzeugt.

Übertragen der Lockset-Datei

Die Lockset-Datei muss anschließend über eine Ethernet-Verbindung direkt oder alternativ über ein Speichermedium, z. B. USB-FlashDrive, auf die Steuerung übertragen werden.

Die Datei liegt unter: /System-CF-Card/User/sinumerik/data/license.

Einlesen der Lockset-Datei

Vor dem Einlesen der Lockset-Datei in der Steuerung muss vom Maschinenhersteller die Uhrzeit der Steuerung korrekt eingestellt werden, da die Uhrzeit zum Zeitpunkt des Aktivierens der CNC-Sperrfunktion als Startwert für die Überwachung gespeichert wird. Danach kann das Einlesen über die Bedienoberfläche gestartet werden: Bedienbereich "Inbetriebnahme" > "Lizenzen" > "License Key einlesen".

Zum Einlesen wird keine Zugriffstufe benötigt.

Tritt beim Einlesen der Lockset-Datei kein Fehler auf, ist die CNC-Sperrfunktion in der Steuerung aktiv.

Hinweis

Treten beim Einlesen der Lockset-Datei Fehler auf, wird ein fehlerspezifischer Alarm angezeigt. Der Zustand der CNC-Sperrfunktion bleibt unverändert.

Hinweis

Es wird empfohlen, dass vom Maschinenhersteller nach Inbetriebnahme der Maschine und Aktivierung der CNC-Sperrfunktion ein vollständiges **Inbetriebnahmearchiv** über alle Steuerungskomponenten erstellt wird. Dadurch wird die Datenkonsistenz für die CNC-Sperrfunktion sichergestellt. Bei Bedarf kann mit diesem Inbetriebnahmearchiv die Steuerung wieder in Betrieb genommen werden, ohne dass dabei ein Service-Einsatz zum erneuten Aktivieren der CNC-Sperrfunktion notwendig wird.

Kontrolle des Sperrdatums

Die Aktivierung der CNC-Sperrfunktion ist dadurch ersichtlich, dass an der Bedienoberfläche das in der Lockset-Datei eingetragene Sperrdatum angezeigt wird:

Bedienoberfläche SINUMERIK Operate: Bedienbereich "Inbetriebnahme" > "Lizenzen"

Einstellen der Vorwarnzeit

Die Vorwarnzeit ist der Zeitraum vor Erreichen des Sperrdatums, ab dem einmal täglich der Alarm 8063 angezeigt wird. Der Alarm weist darauf hin, dass das Sperrdatum demnächst erreicht und NC-Start für die Steuerung gesperrt wird. Die Vorwarnzeit wird eingestellt über das Maschinendatum:

MD17300 \$MN_CNC_LOCK_WARNING_TIME

Wichtige Hinweise

Für ein korrektes und sicheres Funktionieren der CNC-Sperrfunktion sind folgende Hinweise zu beachten!

Hinweis

Vor dem Aktivieren der CNC-Sperrfunktion muss an der Steuerung die **Uhrzeit** korrekt eingestellt werden.

Hinweis

Das PLC-Anwenderprogramm sollte konsequent mit dem **POE-Passwortschutz** versehen werden. Damit wird verhindert, dass der Anwender das maschinenspezifische Know-how kopieren und in einem eigenen PLC-Anwenderprogramm verwenden kann und dieses PLC-Anwenderprogramm gegen das PLC-Anwenderprogramm des Maschinenherstellers, welches den PLC-Schlüssel der CNC-Sperrfunktion enthält, austauschen kann.

Hinweis

Das **Inbetriebnahmearchiv** für den Endanwender darf erst **nach dem Aktivieren** der CNC-Sperrfunktion ausgelesen werden.

Hinweis

Nach dem Deaktivieren der CNC-Sperrfunktion (siehe "Deaktivierung der CNC-Sperrfunktion (Seite 1233)") ist unbedingt ein **neues Inbetriebnahmearchiv** auszulesen und gegen das ursprüngliche Archiv auszutauschen. Nur so wird sichergestellt, dass es später nicht zu einem versehentlichen erneuten Aktivieren der CNC-Sperrfunktion kommt. Dies hätte einen Alarm und die Sperre von NC-Start zur Folge. Um dies wieder rückgängig zu machen, muss die Lockset-Datei zum Deaktivieren der CNC-Sperrfunktion erneut eingelesen werden.

Hinweis

Die Verantwortung für ein korrektes und sicheres Funktionieren der CNC-Sperrfunktion liegt ausschließlich beim Maschinenhersteller.

15.7.6 Verlängerung der CNC-Sperrfunktion

Zur Verlängerung der CNC-Sperrfunktion muss der Maschinenhersteller mit Access MyMachine (AMM) eine neue Lockset-Datei (lockset.clc) **mit neuem Sperrdatum** für die CNC-Sperrfunktion erzeugen.

Erzeugen der Lockset-Datei

Zur Erstellung der Lockset-Datei werden folgende Daten benötigt:

- Seriennummer der CF-Card
- Seriennummer der Steuerung (Panel Processing Unit, PPU)
- OEM-PIN
- **Neues Sperrdatum**

Hinweis

Die Seriennummern der CF-Card und der Steuerung (Panel Processing Unit, PPU) sowie die OEM-PIN müssen mit den Werten übereinstimmen, die bei der Ersteinrichtung für das Aktivieren der CNC-Sperrfunktion verwendet wurden.

Hinweis

Die **Seriennummern** der CF-Card und der Steuerung (PPU) finden sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Hardware" > "NCU/PLC" > "ncu1":

- **CF-Card:** Bereich "CF-Karte" > "SerialNo."
 - **PPU:** Bereich "SINUMERIK 828D PPU" > "SerialNo."
-

Die Daten werden über Access MyMachine im Dialogfenster "CNC-Sperrfunktion" eingegeben (Aufruf über das Hauptmenü: "Extras" > "CNC-Sperrfunktion"). Die Schaltfläche "**Verlängern**" muss ausgewählt sein:

Nach Eingabe aller erforderlichen Daten wird mit der Schaltfläche "Lockset-Datei erstellen ..." die Lockset-Datei zum Verlängern erzeugt.

Übertragen der Lockset-Datei

Die neue Lockset-Datei muss auf die Steuerung übertragen werden. Die Übertragung kann dabei vom Maschinenhersteller über eine Ethernet-Verbindung direkt auf die Steuerung erfolgen.

Die Datei liegt unter: /System-CF-Card/User/sinumerik/data/license.

Oder der Maschinenhersteller sendet die neue Lockset-Datei an den Endkunden und dieser überträgt die Datei auf die Steuerung.

Einlesen der Lockset-Datei

Das Einlesen der Lockset-Datei wird über die Bedienoberfläche gestartet: Bedienbereich "Inbetriebnahme" > "Lizenzen" > "License Key einlesen".

Zum Einlesen wird keine Zugriffstufe benötigt.

Tritt beim Einlesen der Lockset-Datei kein Fehler auf, ist die CNC-Sperrfunktion mit neuem Sperrdatum in der Steuerung aktiv.

Hinweis

Treten beim Einlesen der Lockset-Datei Fehler auf, wird ein fehlerspezifischer Alarm angezeigt. Der Zustand der CNC-Sperrfunktion bleibt unverändert.

Kontrolle des geänderten Sperrdatums

Dass die Änderung des Sperrdatums in der Steuerung erfolgreich durchgeführt wurde, kann an der Bedienoberfläche kontrolliert werden:

Bedienoberfläche SINUMERIK Operate: Bedienbereich "Inbetriebnahme" > "Lizenzen"

15.7.7 Deaktivierung der CNC-Sperrfunktion

Zur Deaktivierung der CNC-Sperrfunktion muss der Maschinenhersteller mit Access MyMachine (AMM) eine Lockset-Datei (lockset.clc) **ohne Sperrdatum** erzeugen.

Erzeugen der Lockset-Datei

Zur Erstellung der Lockset-Datei werden folgende Daten benötigt:

- Seriennummer der CF-Card
- Seriennummer der Steuerung (Panel Processing Unit, PPU)
- OEM-PIN

Hinweis

Die Seriennummern der CF-Card und der Steuerung (Panel Processing Unit, PPU) sowie die OEM-PIN müssen mit denen Werten übereinstimmen, die bei der Ersteinrichtung für das Aktivieren der CNC-Sperrfunktion verwendet wurden.

Hinweis

Die **Seriennummern** der CF-Card und der Steuerung (PPU) finden sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Hardware" > "NCU/PLC" > "ncu1":

- **CF-Card:** Bereich "CF-Karte" > "SerialNo."
 - **PPU:** Bereich "SINUMERIK 828D PPU" > "SerialNo."
-

Die Daten werden über Access MyMachine im Dialogfenster "CNC-Sperrfunktion" eingegeben (Aufruf über das Hauptmenü: "Extras" > "CNC-Sperrfunktion"). Die Schaltfläche "**Deaktivieren**" muss angewählt sein:

Nach Eingabe aller erforderlichen Daten wird mit der Schaltfläche "Lockset-Datei erstellen ..." die Lockset-Datei zum Deaktivieren erzeugt.

Übertragen der Lockset-Datei

Die neue Lockset-Datei muss auf die Steuerung übertragen werden. Die Übertragung kann dabei vom Maschinenhersteller über eine Ethernet-Verbindung direkt auf die Steuerung erfolgen.

Die Datei liegt unter: /System-CF-Card/User/sinumerik/data/license.

Oder der Maschinenhersteller sendet die neue Lockset-Datei an den Endkunden und dieser überträgt die Datei auf die Steuerung.

Einlesen der Lockset-Datei

Das Einlesen der Lockset-Datei wird über die Bedienoberfläche gestartet: Bedienbereich "Inbetriebnahme" > "Lizenzen" > "License Key einlesen".

Zum Einlesen wird keine Zugriffstufe benötigt.

Tritt beim Einlesen der Lockset-Datei kein Fehler auf, ist die CNC-Sperrfunktion deaktiviert.

Hinweis

Treten beim Einlesen der Lockset-Datei Fehler auf, wird ein fehlerspezifischer Alarm angezeigt. Der Zustand der CNC-Sperrfunktion bleibt unverändert.

Hinweis

Es wird empfohlen, dass vom Endkunden **nach dem Deaktivieren** des Sperrdatums ein vollständiges **Inbetriebnahmearchiv** über alle Steuerungskomponenten erstellt wird. Bei Bedarf kann mit diesem Inbetriebnahmearchiv die Steuerung wieder in Betrieb genommen werden, ohne dass dabei die CNC-Sperrfunktion wieder aktiviert wird.

Kontrolle

Dass die Deaktivierung des Sperrdatums in der Steuerung erfolgreich durchgeführt wurde, ist dadurch ersichtlich, dass an der Bedienoberfläche kein Sperrdatum mehr angezeigt wird:

Bedienoberfläche SINUMERIK Operate: Bedienbereich "Inbetriebnahme" > "Lizenzen"

15.7.8 Austausch einer defekten Steuerungshardware (PPU)

Beim Austausch einer defekten Steuerungshardware (PPU) muss die System-CF-Card bei der Maschine verbleiben, damit sie in die neue Steuerungshardware eingesetzt werden kann.

Nach dem erstmaligen Hochlauf der Steuerung mit der neuen Hardware ist ein vorhandenes, mit der vorhergehenden Steuerungshardware erstelltes Inbetriebnahmearchiv einzulesen. Nach dem nächsten Hochlauf der Steuerung wird der Alarm 8062: "CNC-Sperrfunktion: Der Ablauf der Funktion wurde gestört: Ursache 2" (Die Hardware wurde getauscht) angezeigt und NC-Start gesperrt. Ursache des Alarms ist die neue Seriennummer der neuen Steuerungshardware.

Neue Lockset-Datei anfordern

Um die Steuerung wieder entsperren zu können, muss vom Endkunden eine neue, zur aktuellen Steuerung passende Lockset-Datei (lockset.clc) vom Einrichter der CNC-Sperrfunktion (Maschinenhersteller bzw. Händler) angefordert werden.

Erzeugen der Lockset-Datei

Zur Erstellung der Lockset-Datei werden folgende Daten benötigt:

- Seriennummer der bei der Maschinen verbliebenen CF-Card
- Seriennummer der **neuen** Steuerung (PPU)
- die ursprünglich vergebene OEM-PIN
- letztes Sperrdatum oder neues Sperrdatum

Hinweis

Die ursprünglich vergebene **OEM-PIN** muss für den Austausch einer defekten Steuerungshardware (PPU) beim Einrichter der CNC-Sperrfunktion bekannt sein!

Hinweis

Die **Seriennummern** der CF-Card und der Steuerung (PPU) finden sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Hardware" > "NCU/PLC" > "ncu1":

- **CF-Card:** Bereich "CF-Karte" > "SerialNo."
- **PPU:** Bereich "SINUMERIK 828D PPU" > "SerialNo."

Die Daten werden über Access MyMachine im Dialogfenster "CNC-Sperrfunktion" eingegeben (Aufruf über das Hauptmenü: "Extras" > "CNC-Sperrfunktion"). Die Schaltfläche **"Aktivieren"** muss angewählt sein:

Nach Eingabe aller erforderlichen Daten wird mit der Schaltfläche "Lockset-Datei erstellen ..." die Lockset-Datei zum Entsperren erzeugt.

Übertragen der Lockset-Datei

Die neuen Lockset-Datei muss auf die Steuerung übertragen werden. Die Übertragung kann dabei vom Maschinenhersteller über eine Ethernet-Verbindung direkt auf die Steuerung erfolgen.

Die Datei liegt unter: /System-CF-Card/User/sinumerik/data/license.

Oder der Maschinenhersteller sendet die neue Lockset-Datei an den Endkunden und dieser überträgt die Datei auf die Steuerung.

Einlesen der Lockset-Datei

Hinweis

Vor dem Einlesen der Lockset-Datei muss die **Uhrzeit** an der neuen Steuerungshardware korrekt eingestellt werden, da die Uhrzeit zum Zeitpunkt des Aktivierens der CNC-Sperrfunktion als Startwert für die Überwachung gespeichert wird. Der Einrichter der CNC-Sperrfunktion kann den Vorgang dadurch absichern, indem der Tausch und das Stellen der Uhrzeit durch sein Servicepersonal vorgenommen wird.

Das Einlesen der Lockset-Datei wird über die Bedienoberfläche gestartet: Bedienbereich "Inbetriebnahme" > "Lizenzen" > "License Key einlesen".

Zum Einlesen wird keine Zugriffsstufe benötigt.

Tritt beim Einlesen der Lockset-Datei kein Fehler auf, ist die CNC-Sperrfunktion in der neuen Steuerung aktiv.

15.7.9 Austausch einer defekten CF-Card

Beim Austausch einer defekten System-CF-Card muss die defekte CF-Card an SIEMENS gesandt werden. Im Gegenzug wird dann von SIEMENS eine neue CF-Card geliefert.

Nach dem erstmaligen Hochlauf der Steuerung mit der neuen CF-Card ist ein vorhandenes, mit der vorhergehenden Steuerungshardware erstelltes Inbetriebnahmearchiv einzulesen. Nach dem nächsten Hochlauf der Steuerung wird der Alarm 8062: "CNC-Sperrfunktion: Der Ablauf der Funktion wurde gestört: Ursache 1" (Es hat ein Tausch der CF-Karte stattgefunden) angezeigt und NC-Start gesperrt. Ursache des Alarms ist die neue Seriennummer der neuen CF-Card.

Neue Lockset-Datei anfordern

Um die Steuerung wieder entsperren zu können, muss vom Endkunden eine neue, zur aktuellen Steuerung passende Lockset-Datei (lockset.clc) vom Einrichter der CNC-Sperrfunktion (Maschinenhersteller bzw. Händler) angefordert werden.

Erzeugen der Lockset-Datei

Zur Erstellung der Lockset-Datei werden folgende Daten benötigt:

- Seriennummer der **neuen** CF-Card
- Seriennummer der Steuerung (PPU)
- die ursprünglich vergebene OEM-PIN
- das Sperrdatum entsprechend dem Status der CNC-Sperrfunktion (letztes Sperrdatum, neues Sperrdatum, kein Sperrdatum)

Hinweis

Die ursprünglich vergebene **OEM-PIN** muss für den Austausch einer defekten CF-Card beim Einrichter der CNC-Sperrfunktion bekannt sein!

Hinweis

Die **Seriennummern** der CF-Card und der Steuerung (PPU) finden sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Hardware" > "NCU/PLC" > "ncu1":

- **CF-Card:** Bereich "CF-Karte" > "SerialNo."
- **PPU:** Bereich "SINUMERIK 828D PPU" > "SerialNo."

Die Daten werden über Access MyMachine im Dialogfenster "CNC-Sperrfunktion" eingegeben (Aufruf über das Hauptmenü: "Extras" > "CNC-Sperrfunktion"). Die Schaltfläche "**Verlängern**" oder "**Deaktivieren**" muss angewählt sein:

Bild 15-3 CNC-Sperrfunktion aktualisieren

Nach Eingabe aller erforderlichen Daten wird mit der Schaltfläche "Lockset-Datei erstellen ..." die Lockset-Datei zum Entsperren erzeugt.

Übertragen der Lockset-Datei

Die neue Lockset-Datei muss auf die Steuerung übertragen werden. Die Übertragung kann dabei vom Maschinenhersteller über eine Ethernet-Verbindung direkt auf die Steuerung erfolgen.

Die Datei liegt unter: /System-CF-Card/User/sinumerik/data/license.

Oder der Maschinenhersteller sendet die neue Lockset-Datei an den Endkunden und dieser überträgt die Datei auf die Steuerung.

Einlesen der Lockset-Datei

Hinweis

Vor dem Einlesen der Lockset-Datei muss die **Uhrzeit** an der neuen Steuerungshardware korrekt eingestellt werden, da die Uhrzeit zum Zeitpunkt des Aktivierens der CNC-Sperrfunktion als Startwert für die Überwachung gespeichert wird. Der Einrichter der CNC-Sperrfunktion kann den Vorgang dadurch absichern, indem der Tausch und das Stellen der Uhrzeit durch sein Servicepersonal vorgenommen wird.

Das Einlesen der Lockset-Datei wird über die Bedienoberfläche gestartet: Bedienbereich "Inbetriebnahme" > "Lizenzen" > "License Key einlesen".

Zum Einlesen wird keine Zugriffsstufe benötigt.

Tritt beim Einlesen der Lockset-Datei kein Fehler auf, ist die CNC-Sperrfunktion in der neuen Steuerung aktiv.

15.7.10 OEM-PIN vergessen

Der Einrichter der CNC-Sperrfunktion (Maschinenhersteller bzw. Händler) hat die bei der Ersteinrichtung von ihm vergebene OEM-PIN vergessen und kann daher für die entsprechende Steuerung keine gültige Lockset-Datei mehr erzeugen.

Entsperren der Maschine

Damit der Einrichter der CNC-Sperrfunktion diese wieder bedienen kann, muss er sich an die SIEMENS Hotline wenden und dabei Folgendes bereithalten:

- Lizenzzertifikat (CoL) für die Option "CNC-Sperrfunktion"
- Seriennummer der CF-Card
- Seriennummer der Steuerung (PPU)
- Softwarestand der CNC-Software

Hinweis

Die **Seriennummern** der CF-Card und der Steuerung (PPU) finden sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Hardware" > "NCU/PLC" > "ncu1":

- CF-Card: Bereich "CF-Karte" > "SerialNo."
- PPU: Bereich "SINUMERIK 828D PPU" > "SerialNo."

Der **Softwarestand der CNC-Software** findet sich auf der Bedienoberfläche SINUMERIK Operate im Bedienbereich "Diagnose" > "Version" > "Ist-Version".

Das Lizenzzertifikat (CoL) muss im Original zu SIEMENS gesendet werden. Das Entsperren durch SIEMENS ist ein kostenpflichtiger Vorgang.

Von der Hotline erhält der Einrichter der CNC-Sperrfunktion eine Lockset-Datei zum Entsperren der Maschine. Das Entsperren wirkt allerdings nur auf die Hardware der Steuerung

(PPU). Das PLC-Projekt wird nicht entsperrt. Aus diesem Grund muss auch das originale, zur Maschine passende PLC-Projekt vorliegen.

Zum Entsperren der Maschine ist der Vororteinsatz eines Servicemitarbeiters des Einrichters der CNC-Sperrfunktion beim Endanwender (Betreiber) der Maschine erforderlich.

Weitere Informationen zum Ablauf erhalten sie von der SIEMENS Hotline.

15.7.11 Sonstige Informationen

Projektdatei

Hinweis

Für die Dokumentation der Zugehörigkeit von Seriennummern und OEM-PIN ist der Maschinenhersteller selbst verantwortlich.

Access MyMachine bietet die Möglichkeit, eine unverschlüsselte Projektdatei ("ucls" für "User-CNC-Lock-Set-Datei") anzulegen, die folgende Daten enthält:

- Seriennummer der CF-Card
- Seriennummer der Steuerung (PPU)
- OEM-PIN
- Erstellungsdatum
- Sperrdatum

Aufgerufen wird diese Funktion über die Schaltfläche "Daten speichern ..." im Dialogfenster "CNC-Sperrfunktion".

Mit der Schaltfläche "Daten laden ..." können die in der Projektdatei hinterlegten unverschlüsselten Daten wieder eingelesen werden.

Fehlerhafte Einstellungen von Datum oder Uhrzeit

Werden bei aktivierter CNC-Sperrfunktion fehlerhafte Einstellungen von Datum oder Uhrzeit festgestellt, werden folgende Aktionen ausgelöst:

- Ausgabe von Alarm 8065: "CNC-Sperrfunktion: Bitte Datum/Uhrzeit korrekt einstellen!"
- Sperre von NC-Start.

Um die Sperre aufzuheben, müssen die fehlerhaften Einstellungen vor dem Ausschalten der Steuerung korrigiert werden.

ACHTUNG

Dauerhafte Sperre

Die fehlerhaften Einstellungen von Datum oder Uhrzeit müssen unbedingt vor dem Ausschalten der Steuerung korrigiert werden. Ansonsten droht die dauerhafte Sperre durch Alarm 8064: "CNC-Sperrfunktion: Das Sperrdatum ist erreicht, kein NC-Start möglich!"

Abhilfe: Datum/Uhrzeit **vor** dem Ausschalten der Steuerung korrekt einstellen.

Wird bei aktivierter CNC-Sperrfunktion das Datum versehentlich in die Zukunft gestellt, wird der folgende Alarm ausgegeben:

Alarm 8066 "CNC-Sperrfunktion: Durch Änderung des Datums wurde die verbleibende Laufzeit verringert!"

Solange die Steuerung nicht ausgeschaltet wird, kann das Datum noch korrigiert werden.

ACHTUNG

Verkürzte Nutzungsdauer

Nach Ausschalten der Steuerung wird ein in die Zukunft gestelltes Datum als wirkliches Datum angesehen und kann nicht mehr zurückgestellt werden. Dadurch verkürzt sich die Nutzungsdauer bis zum Sperrdatum.

Abhilfe: Datum **vor** dem Ausschalten der Steuerung korrekt einstellen.

Weitere Informationen

Siehe:

- Online-Hilfe zu SINUMERIK Integrate Access MyMachine/ P2P (PC)
- Online-Hilfe zum PLC Programming Tool

R1: Referenzieren

16.1 Kurzbeschreibung

Funktion

Beim Referenzieren einer Maschinenachse wird das Koordinatensystem der Maschinenachse mit dem Koordinatensystem der Maschine synchronisiert. Dazu wird die Maschinenachse auf den Maschinennullpunkt gefahren und dann die Istposition der Maschinenachse auf den Wert Null gesetzt.

Kann der Maschinennullpunkt konstruktionsbedingt nicht direkt angefahren werden, bedient man sich zum Synchronisieren der Maschinenachse eines im Verfahrbereich der Maschinenachse festgelegten Referenzpunktes. Dessen Position in Bezug zum Maschinennullpunkt muss bekannt sein. Beim Referenzieren wird die Istposition der Maschinenachse dann auf diesen Wert gesetzt

Messsysteme und Referenzierarten

Beim Referenzieren können Maschinenachsen mit folgenden Messsystemen und Referenzierarten synchronisiert werden:

- Messsysteme
 - Inkrementelles rotatorisches Messsystem mit mindestens einer Nullmarke
 - Inkrementelles lineares Messsystem (Längenmesssystem)
 - Rotatorisches Messsystem mit abstandscodierten Referenzmarken (Fa. Heidenhain)
 - Lineares Messsystem mit abstandscodierten Referenzmarken (Fa. Heidenhain)
 - Absolutes rotatorisches Messsystem
 - Absolutes lineares Messsystem (Längenmesssystem)
- Referenzierarten
 - Referenzieren bei inkrementellen Messsystemen mit berührungslosem Näherungsschalter und Ein- und Zwei-Flankenerkennung
 - Referenzieren bei inkrementellen Messsystemen mit Ersatz des Referenznockens durch berührungslosen Näherungsschalter
 - Referenzieren bei inkrementellen Messsystemen mit berührungslosem Näherungsschalter mit projektierter Anfahrgeschwindigkeit bei Spindelapplikationen
 - Referenzieren bei Messsystemen mit abstandscodierten Referenzmarken durch Überfahren von 2 bzw. 4 Nullmarken
 - Referenzieren von passiven Messsystemen mittels Messsystemabgleich
 - Referenzieren im Nachführbetrieb
 - Referenzieren mit Nockenschalter am Antrieb

Start

Das Referenzieren einer Maschinenachse kann manuell oder über das Teileprogramm gestartet werden:

- Manuell: Betriebsart JOG und MDA, Maschinenfunktion REF
- Teileprogramm: Befehl G74

16.2 Achsspezifisches Referenzieren

Beim achsspezifischen Referenzieren durch Referenzpunktfahren, muss das Verfahren einzeln für jede zu referenzierende Maschinenachse gestartet werden.

Anwahl der Betriebsart und Maschinenfunktion

Vor dem Referenzpunktfahren der Maschinenachsen ist zuerst die entsprechende BAG in die Betriebsart JOG oder MDA zu versetzen:

DB11, ... DBX0.2 (Betriebsart JOG)

DB11, ... DBX0.1 (Betriebsart MDA)

Anschließend ist die Maschinenfunktion REF (Referenzpunktfahren) anzuwählen:

DB11, ... DBX1.2 (Maschinenfunktion REF)

Start des Referenzpunktfahrens

Beim achsspezifischen Referenzpunktfahren muss der Start für jede Maschinenachse erfolgen.

Der Start des Referenzpunktfahrens erfolgt über die achsspezifischen Verfahrtasten:

DB31, ... DBX4.6 (Verfahrtaste minus)

DB31, ... DBX4.7 (Verfahrtaste plus)

Richtungsfreigabe

Zur Vermeidung von Fehlbedienungen muss die Richtungsfreigabe parametrierung werden. Über die Richtungsfreigabe wird festgelegt, durch welche Verfahrtaste das Referenzpunktfahren gestartet wird:

MD34010 \$MA_REFP_CAM_DIR_IS_MINUS = <Wert>

<Wert>	Bedeutung
0	Referenzpunktfahren in Plusrichtung
1	Referenzpunktfahren in Minusrichtung

Tippbetrieb

Ob das Referenzpunktfahren nach dem einmaligen Betätigen der Richtungstaste vollständig ausgeführt wird oder der Maschinenbediener aus Sicherheitsgründen die Richtungstaste kontinuierlich gedrückt halten muss (Tippbetrieb), ist über folgendes Maschinendatum zu parametrieren:

MD11300 \$MN_JOG_INC_MODE_LEVELTRIGGRD (INC und REF im Tippbetrieb)

Lässt der Maschinenbediener die Richtungstaste im Tippbetrieb los, wird die Maschinenachse bis zum Stillstand abgebremst. Das Referenzpunktfahren wird nicht abgebrochen. Mit dem nächsten Drücken der Richtungstaste wird das Referenzpunktfahren fortgesetzt.

Referenzierstatus

Mit dem Start des Referenzpunktfahrens wird der Referenzierstatus der Maschinenachse zurückgesetzt:

DB31, ... DBX60.4 (Referenziert / Synchronisiert 1)

DB31, ... DBX60.5 (Referenziert / Synchronisiert 2)

DB21, ... DBX36.2 (alle referenzpunktpflichtigen Achsen sind referenziert)

Abstandscodierte Messsysteme

Bei abstandscodierten Messsystemen kann das Referenzpunktfahren mit einer beliebigen Verfahrtaste gestartet werden.

Reihenfolge

Die Einhaltung einer bestimmten Reihenfolge, in der Maschinenachsen zu referenzieren sind, ist vom Maschinenbediener oder vom Maschinenhersteller per PLC-Anwenderprogramm sicherzustellen:

- Maschinenbediener
Die Maschinenachsen sind vom Maschinenbediener in der erforderlichen Reihenfolge zu starten.
- Maschinenhersteller
Das PLC-Anwenderprogramm des Maschinenherstellers ermöglicht das Starten nur in der entsprechenden Reihenfolge.

Gleichzeitiges Referenzpunktfahren mehrerer Maschinenachsen

Es können steuerungsabhängig mehrere Maschinenachsen gleichzeitig referenziert werden:

Abschluss des Referenzpunktfahrens

Ist das Referenzpunktfahren einer Maschinenachse erfolgreich abgeschlossen, erfolgt die entsprechende Rückmeldung durch Setzen des Referenzierstatus:

DB31, ... DBX60.4 (Referenziert / Synchronisiert 1)

DB31, ... DBX60.5 (Referenziert / Synchronisiert 2)

Abbruch des Referenzpunktfahrens

Beim achsspezifischen Referenzpunktfahren wird die Maschinenachse in dem Kanal verfahren, der als Master-Kanal der Maschinenachse parametrierung wurde:

MD30550 \$MA_AXCONF_ASSIGN_MASTER_CHAN

Zum Abbruch des Referenzpunktfahrens muss daher entweder BAG-Reset oder Kanal-Reset für den Master-Kanal der Maschinenachse ausgelöst werden:

DB11, ... DBX0.7 (BAG-Reset)

DB21, ... DBX7.7 (Kanal-Reset)

Alle Maschinenachsen, die zum Zeitpunkt des Abbruchs das Referenzpunktfahren noch nicht erfolgreich abgeschlossen haben, bleiben im Status "Nicht referenziert":

DB31, ... DBX60.4 (Referenziert / Synchronisiert 1)

DB31, ... DBX60.5 (Referenziert / Synchronisiert 2)

16.3 Kanalspezifisches Referenzieren

Beim kanalspezifischen Referenzieren durch Referenzpunktfahren werden nach dem Anstoß des Referenzpunktfahrens alle Maschinenachsen des Kanals in der parametrisierten Reihenfolge referenziert.

Anwahl der Betriebsart und Maschinenfunktion

Vor dem Referenzpunktfahren der Maschinenachsen muss die zugehörige BAG in die Betriebsart JOG oder MDA geschaltet werden:

DB11, ... DBX0.2 (Betriebsart JOG)

DB11, ... DBX0.1 (Betriebsart MDA)

Anschließend ist die Maschinenfunktion REF (Referenzpunktfahren) anzuwählen:

DB11, ... DBX1.2 (Maschinenfunktion REF)

Parametrieren der Achsreihenfolge

Die Reihenfolge, in der die Maschinenachsen des Kanals referenziert werden, wird über folgendes Maschinendatum parametrierung:

MD34110 \$MA_REFP_CYCLE_NR = <Nummer>

<Nummer>	Bedeutung
-1	Für NC-START im Kanal muss die Maschinenachse nicht referenziert sein.
0	Die Maschinenachse nimmt nicht am kanalspezifischen Referenzpunktfahren teil.
1 - 15	Nummer der Reihenfolge beim kanalspezifischen Referenzpunktfahren.

Die Maschinenachsen werden in aufsteigender Reihenfolge der Nummern referenziert.

Maschinenachsen mit gleicher Nummer werden gleichzeitig referenziert.

Gleichzeitiges Referenzpunktfahren mehrerer Maschinenachsen

Es können steuerungsabhängig mehrere Maschinenachsen gleichzeitig referenziert werden:

Start des Referenzpunktfahrens

Der Start des kanalspezifischen Referenzpunktfahrens erfolgt über:

DB21, ... DBX1.0 (Referenzieren aktivieren)

Den Status des kanalspezifischen Referenzpunktfahrens meldet der Kanal über:

DB21, ... DBX33.0 (Referenzieren aktiv)

Referenzierstatus

Mit dem Start des Referenzpunktfahrens wird der Referenzierstatus der Maschinenachse zurückgesetzt:

DB31, ... DBX60.4 (Referenziert / Synchronisiert 1)

DB31, ... DBX60.5 (Referenziert / Synchronisiert 2)

Abschluss des Referenzpunktfahrens

Ist das kanalspezifische Referenzpunktfahren für alle beteiligten Maschinenachsen erfolgreich abgeschlossen, erfolgt die entsprechende Rückmeldung über:

DB21, ... DBX36.2 (alle referenzpunktpflichtigen Achsen sind referenziert)

Abbruch des Referenzpunktfahrens

Beim kanalspezifischen Referenzpunktfahren wird die Maschinenachse in dem Kanal verfahren, dem sie als Kanalachse aktuell zugeordnet ist.

Zum Abbruch des Referenzpunktfahrens muss entweder BAG-Reset oder Kanal-Reset für den entsprechenden Kanal ausgelöst werden:

DB11, ... DBX0.7 (BAG-Reset)

DB21, ... DBX7.7 (Kanal-Reset)

Alle Maschinenachsen, für die zum Zeitpunkt des Abbruchs das Referenzpunktfahren noch nicht erfolgreich abgeschlossen wurde, bleiben im Status "Nicht referenziert":

DB31, ... DBX60.4 (Referenziert / Synchronisiert 1)

DB31, ... DBX60.5 (Referenziert / Synchronisiert 2)

16.4 Referenzpunktfahren aus dem Teileprogramm (G74)

Funktion

Mit dem Befehl G74 können Maschinenachsen aus einem Teileprogramm oder Synchronaktion heraus erstmalig oder erneut referenziert werden.

Ein erneutes Referenzieren ist z. B. erforderlich nach:

- Istwertverschiebung PRESETON
Literatur: Programmieranleitung Arbeitsvorbereitung, Kapitel "Koordinatentransformationen (Frames)" > "Preset-Verschiebung mit PRESETON"
- Parken einer Maschinenachse (Seite 122)
- Überschreitung der Gebergrenzfrequenz des Lagemesssystems

Programmierung

Syntax

```
G74 <Maschinenachse> {<Maschinenachse>}
```

Bedeutung

G74:	Befehl zum Referenzieren von Maschinenachsen	
	Alleine im Satz:	ja
	Wirksamkeit:	satzweise
<Maschinenachse>:	Es ist der Name einer Maschinenachse (MD10000 \$MN_AXCONF_MACHAX_NAME_TAB) anzugeben. Die Maschinenachse muss Kanalachse des Kanals sein, in dem das Teileprogramm oder die Synchronaktion abgearbeitet wird.	

Reset-Verhalten

Durch BAG- oder Kanal-Reset wird das Referenzpunktfahren für alle programmierten Maschinenachsen abgebrochen:

- DB11, ... DBX0.7 (BAG-Reset)
- DB21, ... DBX7.7 (Kanal-Reset)

Alle Maschinenachsen, für die zum Reset-Zeitpunkt das Referenzpunktfahren noch nicht erfolgreich abgeschlossen wurde, bleiben im Status "Nicht referenziert":

- DB31, ... DBX60.4 (Referenziert / Synchronisiert 1) == 0
- DB31, ... DBX60.5 (Referenziert / Synchronisiert 2) == 0

16.5 Referenzieren bei inkrementellen Messsystemen

16.5.1 Hardware-Signale

Abhängig vom Maschinenaufbau und den Eigenschaften des verwendeten inkrementellen Messsystems, müssen verschiedene Hardware-Signale angeschlossen werden.

Referenznocken

- Anschluss
Das Referenznockensignal kann an einem digitalen Eingang einer externen PLC-Peripheriebaugruppe oder einem schnellen Eingang der NCU-Schnittstelle X142 angeschlossen werden.
- NC/PLC-Nahtstellensignal
Das Referenznockensignal muss vom PLC-Anwenderprogramm in die axiale NC/PLC-Nahtstelle übertragen werden:
DB31, ... DBX12.7 (Verzögerung Referenzpunktfahren)

Nullmarkenauswahl

Werden während des Referenzpunktfahres der Achse oder Spindel mehrere Nullmarken des Messsystems erkannt (z.B. Messgetriebe zwischen Motor und Geber), muss mit einem zusätzlichen berührungslosen Näherungsschalter-Signal die spezifische Nullmarke ausgewählt werden.

- Anschluss
Der berührungslose Näherungsschalter muss an einem schnellen digitalen Eingang der NCU-Schnittstellen X122 oder X132 angeschlossen werden.
- Aktivierung
Damit das berührungslose Näherungsschalter-Signal ausgewertet wird, muss für die Achse / Spindel im Antriebsparameter p0493 der digitale Eingang ausgewählt werden, an dem der berührungslose Näherungsschalter angeschlossen ist.

Nullmarkenersatz

Liefert das verwendete Messsystem kein Nullmarkensignal, kann über ein berührungsloses Näherungsschalter-Signal ein Nullmarkenersatz erzeugt werden.

- Anschluss
Der berührungslose Näherungsschalter muss an einem schnellen digitalen Eingang der NCU-Schnittstellen X122 oder X132 angeschlossen werden.
- Aktivierung
Damit das berührungslose Näherungsschalter-Signal ausgewertet wird, muss für die Achse / Spindel im Antriebsparameter p0494 oder p0495 der digitale Eingang ausgewählt werden, an dem der berührungslose Näherungsschalter angeschlossen ist.

Übersicht

Signal	Anschluss: dig. Eingang über	Einstellen
Referenznocken	ext. PLC-Peripheriebaugruppe oder NCU: X142	PLC-Anwenderprogramm: DB31, ... DBX12.7
Nullmarkenauswahl	NCU: X122 oder X132	Antriebsparameter: p0493
Externe Nullmarke bzw. Nullmarkenersatz	NCU: X122 oder X132	Antriebsparameter: p0494 oder p0495

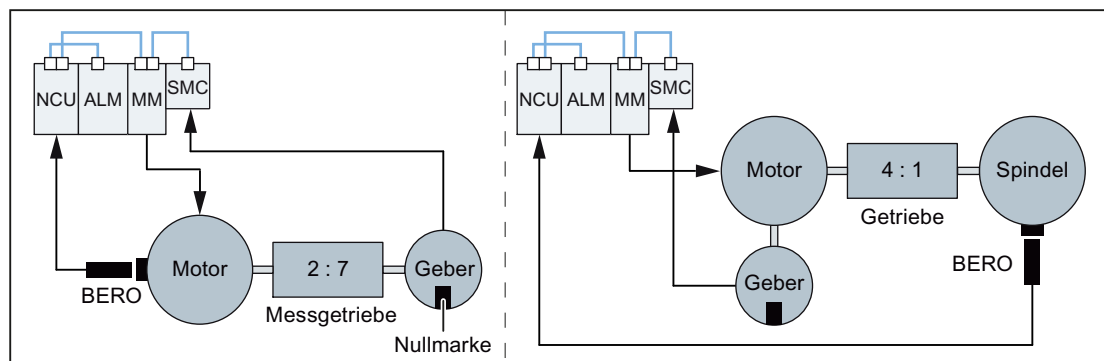
Literatur

- NCU-Schnittstellen: SINUMERIK 840D sl Gerätehandbuch NCU7x0.3 PN, Kapitel "Anschliessen" > "Digitale Ein-/Ausgänge"
- Antriebsparameter: SINAMICS S120/S150 Listenhandbuch

16.5.2 Nullmarkenauswahl

Funktion

Das Referenzieren von inkrementellen Messsystemen basiert auf der eindeutigen Lage der Gebernulldmarke bezogen auf den gesamten Verfahrbereich der Maschinenachse. Werden aufgrund von maschinenspezifischen Gegebenheiten mehrere Gebernulldmarken im Verfahrbereich der Maschinenachse erkannt (Beispiele siehe folgendes Bild), muss zur eindeutigen Bestimmung des Referenzpunktes ein berührungsloser Näherungsschalter an der Maschine angebracht werden. Die Lage des Referenzpunktes ergibt sich dann aus der Kombination des Signals des berührungslosen Näherungsschalters und der Gebernulldmarke.



BE- Berührungsloser Näherungsschalter
RO

Bild 16-1 Messgetriebe zwischen Motor und Geber bzw. Untersetzungsgetriebe zwischen Motor und Spindel

Parametrierung

NC: Referenziermodus

Als Referenziermodus ist "Referenzieren von inkrementellen, rotatorischen oder linearen Messsystemen: Nullimpuls auf der Geberspur" zu parametrieren:

MD34200 \$MA_ENC_REFP_MODE[<Achse>] = 1

Antrieb: Nullmarkenauswahl

Im Parameter p0493 ist der digitale Eingang der NCU-Schnittstelle einzustellen, an dem der berührungslose Näherungsschalter angeschlossen ist.

Hinweis

Nullmarkenauswahl

Die Verarbeitung des Signals des berührungslosen Näherungsschalters erfolgt ausschließlich im Antrieb. Anschluss und Parametrierung siehe Kapitel "Hardware-Signale (Seite 1249)".

16.5.3 Zeitlicher Ablauf

Der zeitliche Ablauf des Referenzpunktfahrens bei inkrementellen Messsystemen unterteilt sich in drei Phasen:

- Phase 1: "Phase 1: Fahren auf den Referenznocken (Seite 1253)"
- Phase 2: "Phase 2: Synchronisation mit der Nullmarke (Seite 1255)"
- Phase 3: "Phase 3: Fahren auf den Referenzpunkt (Seite 1260)"

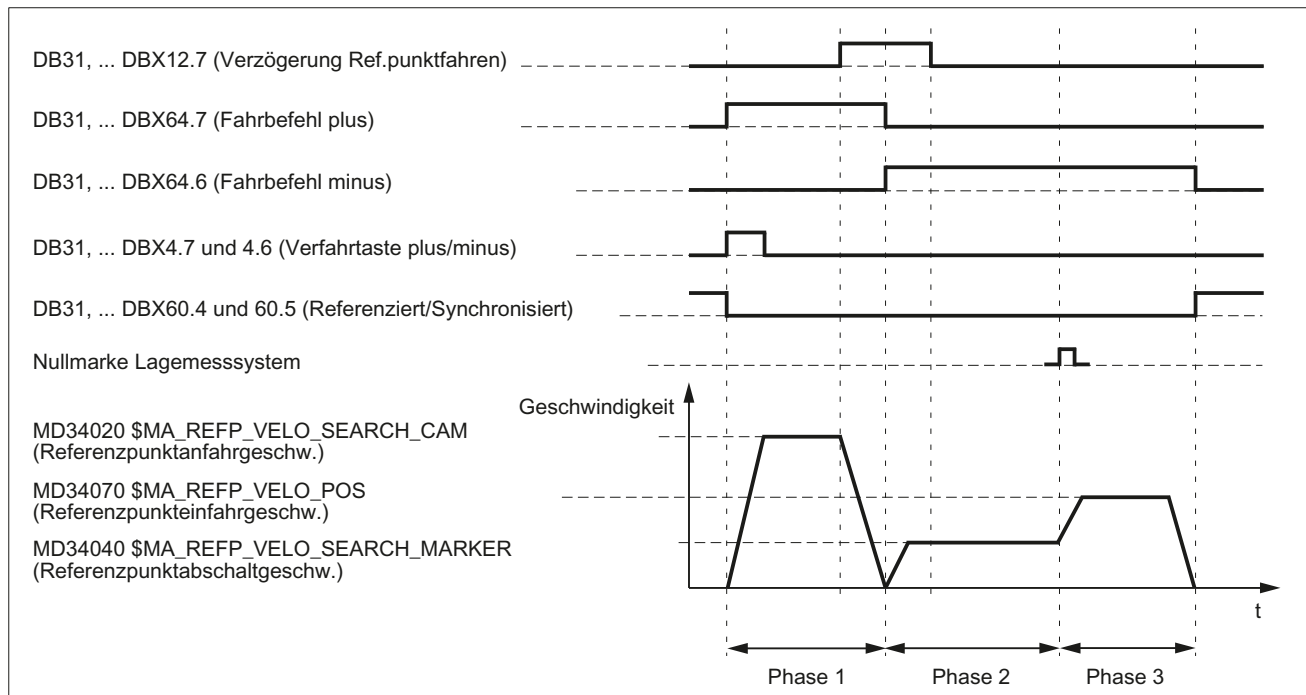


Bild 16-2 Zeitlicher Ablauf beim Referenzieren mit inkrementellen Messsystemen (Beispiel)

16.5.4 Phase 1: Fahren auf den Referenznocken

Phase 1: Grafische Darstellung

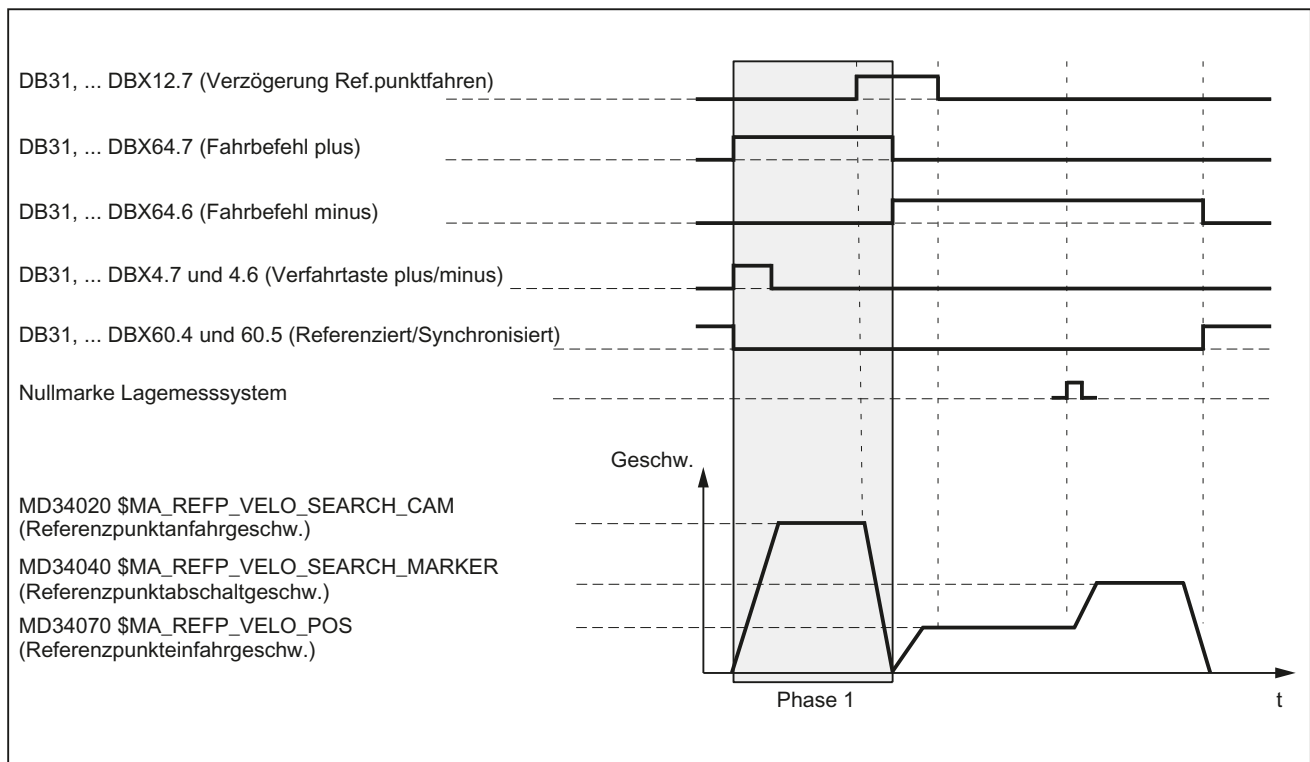


Bild 16-3 Phase 1: Fahren auf den Referenznocken

Phase 1: Start

Zum Starten des Referenzpunktfahrens siehe Kapitel "Achsspezifisches Referenzieren (Seite 1244)" und "Achsspezifisches Referenzieren (Seite 1244)".

Phase 1: Ablauf

Für Phase 1 sind, je nach Lage der Maschinenachse bezüglich des Referenznockens, drei Fälle zu unterscheiden:

1. Die Maschinenachse steht vor dem Referenznocken
2. Die Maschinenachse steht auf dem Referenznocken
3. Die Maschinenachse hat keinen Referenznocken

Fall 1: Die Maschinenachse steht vor dem Referenznocken

Nach dem Start des Referenzpunktfahrens wird die Maschinenachse in die parametrierte Richtung und auf die parametrierte Referenzpunktanfahrgeschwindigkeit beschleunigt:

- MD34010 \$MA_REFP_CAM_DIR_IS_MINUS (Referenzpunkt anfahren in Minusrichtung)
- MD34020 \$MA_REFP_VELO_SEARCH_CAM (Referenzpunktanfahrgeschwindigkeit)

Das Erreichen des Referenznockens muss durch Abfrage eines digitalen EIngangs im PLC-Anwenderprogramm erkannt und der NC über folgendes Nahtstellensignal mitgeteilt werden:

DB31, ... DBX12.7 = 1 (Verzögerung Referenzpunktfahren)

Mit Erkennen des NC/PLC-Nahtstellensignals wird die Maschinenachse bis zum Stillstand abgebremst. Dabei wird mindestens noch die Wegstrecke s_{\min} verfahren. Dadurch wird sichergestellt, dass die Maschinenachse den Referenznocken in Phase 2 mit der parametrierten Referenzpunktabschaltgeschwindigkeit verlässt.

$$s_{\min} = \frac{(\text{MD34040 } \$MA_REFP_VELO_SEARCH_MARKER)^2}{2 * \text{MD32300 } \$MA_MAX_AX_ACCEL}$$

Phase 1 ist damit beendet. Das Referenzpunktfahren wird mit Phase 2 fortgesetzt.

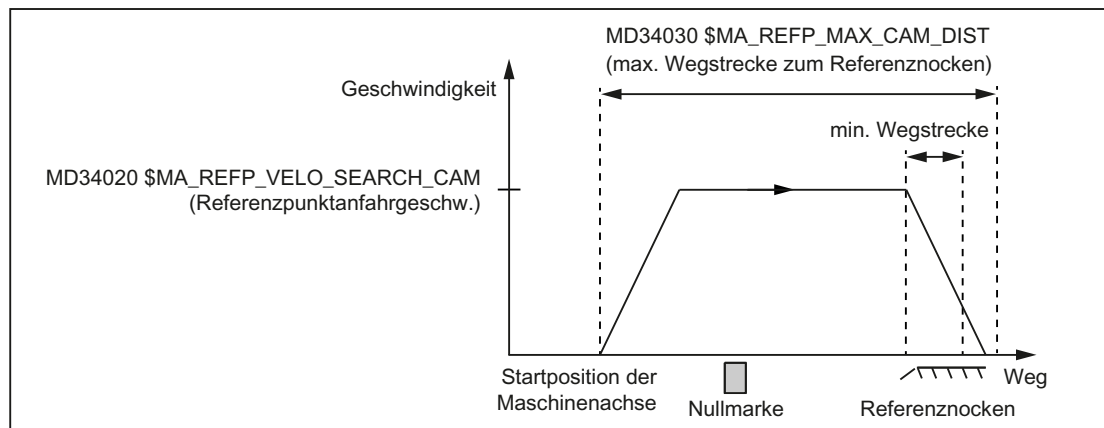


Bild 16-4 Minimale Wegstrecke für Verzögerung

Fall 2: Die Maschinenachse steht auf dem Referenznocken

Die Maschinenachse bleibt auf ihrer Ausgangsposition stehen.

Die Phase 1 ist damit beendet. Das Referenzpunktfahren wird mit Phase 2 fortgesetzt.

Fall 3: Die Maschinenachse hat keinen Referenznocken

Maschinenachsen ohne Referenznocken bleiben auf ihrer Ausgangsposition stehen.

Dazu gehören z. B.:

- Maschinenachsen, die über ihren gesamten Verfahrbereich nur eine Nullmarke haben
- Rundachsen, die nur eine Nullmarke pro Umdrehung haben

Für Maschinenachsen ohne Referenznocken ist im folgenden Maschinendatum Null einzutragen:

MD34000 \$MA_REFP_CAM_IS_ACTIVE = 0 (Achse mit Referenznocken)

Die Phase 1 ist damit beendet. Das Referenzpunktfahren wird mit Phase 2 fortgesetzt.

Phase 1: Eigenschaften

- Vorschub-Override ist wirksam.
- Vorschubhalt (kanal- und achsspezifisch) ist wirksam.
- NC-Stop und NC-Start sind wirksam.
- Die Maschinenachse wird angehalten, wenn der Referenznocken nicht innerhalb der parametrisierten maximalen Wegstrecke erreicht wird:
MD34030 \$MA_REFP_MAX_CAM_DIST (max. Wegstrecke zum Referenznocken)

Siehe auch

Kanalspezifisches Referenzieren (Seite 1246)

16.5.5 Phase 2: Synchronisation mit der Nullmarke

Phase 2: Grafische Darstellung

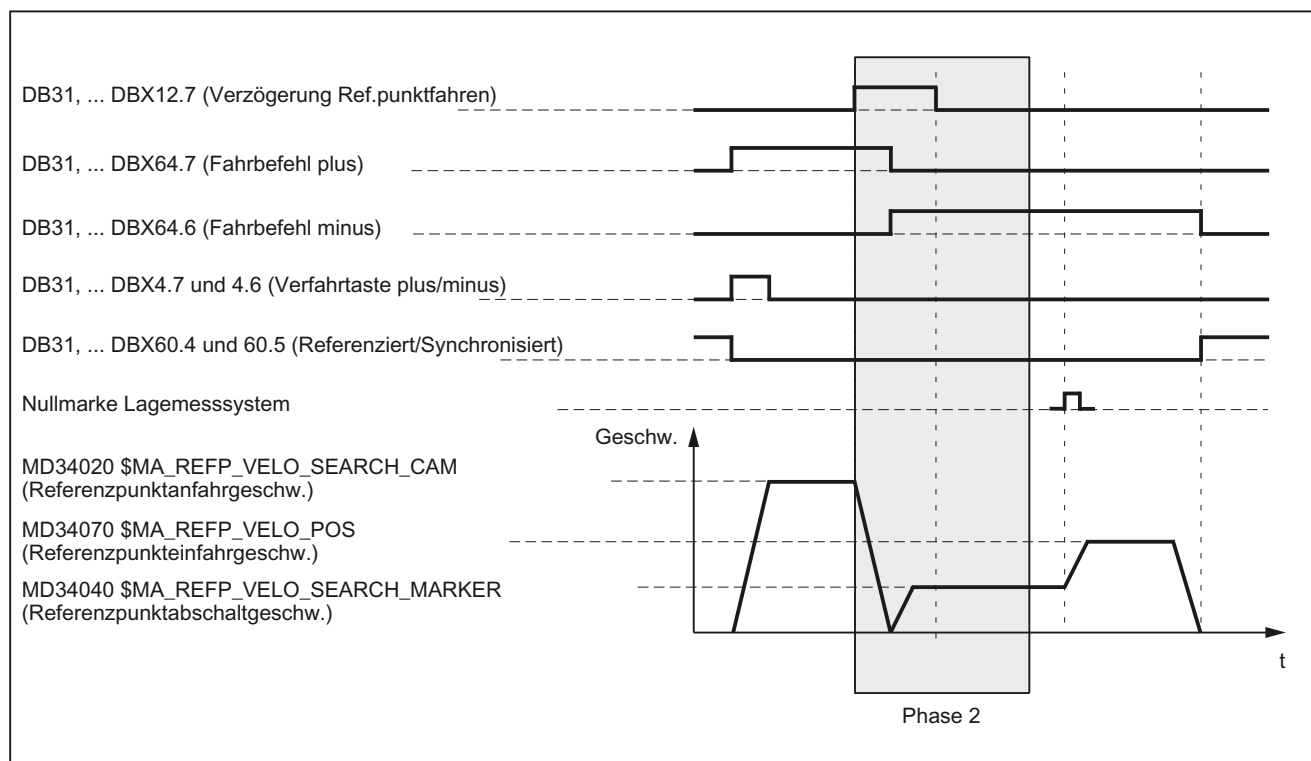


Bild 16-5 Phase 2: Synchronisation mit der Nullmarke

Phase 2: Start

Phase 2 wird automatisch gestartet, sobald Phase 1 ohne Alarm beendet wurde.

Ausgangssituation:

Die Maschinenachse steht auf dem Referenznocken.

Nullmarkensuchrichtung:

Die Richtung der Nullmarkensuche ergibt sich aus den Einstellungen in den Maschinendaten:

- MD34010 \$MA_REFP_CAM_DIR_IS_MINUS (Referenzpunkt anfahren in Minusrichtung)
- MD34050 \$MA_REFP_SEARCH_MARKER_REVERSE (Richtungsumkehr auf Referenznocken)

Phase 2: Ablauf

Die Synchronisation in Phase 2 kann über die fallende oder steigende Referenznockenflanke erfolgen. Die Parametrierung erfolgt über:

MD34050 \$MA_REFP_SEARCH_MARKER_REVERSE[<Achse>] = <Wert>

Wert	Bedeutung
0	Synchronisation mit fallender Referenznockenflanke
1	Synchronisation mit steigender Referenznockenflanke

Hinweis

Hat die Istgeschwindigkeit der Maschinenachse beim Anfahren des Referenznockens noch nicht die Sollgeschwindigkeit der Phase 2 innerhalb der parametrisierten Toleranzgrenzen erreicht, wird noch einmal mit Phase 1 gestartet. Das ist z. B. der Fall, wenn die Maschinenachse beim Starten des Referenzpunktfahrens auf dem Referenznocken steht.

MD35150 \$MA_SPIND_DES_VELO_TOL (Spindeldrehzahltoleranz)

Fall 1: Synchronisation mit fallender Referenznockenflanke

Bei der Synchronisation mit fallender Referenznockenflanke beschleunigt die Maschinenachse auf die parametrisierte Referenzpunktabschaltgeschwindigkeit entgegen der parametrisierten Referenzpunktanfahrriichtung (Verfahrriichtung der Phase 1).

Nach dem Verlassen des Referenznockens wird auf die nächste Gebernulmarke gewartet:
DB31, ... DBX12.7 == 0

Mit dem Erkennen der Gebernulmarke ist die Phase 2 beendet. Die Maschinenachse fährt mit konstanter Geschwindigkeit weiter und das Referenzpunktfahren wird mit Phase 3 fortgesetzt.

- MD34040 \$MA_REFP_VELO_SEARCH_MARKER
(Referenzpunktabschaltgeschwindigkeit)

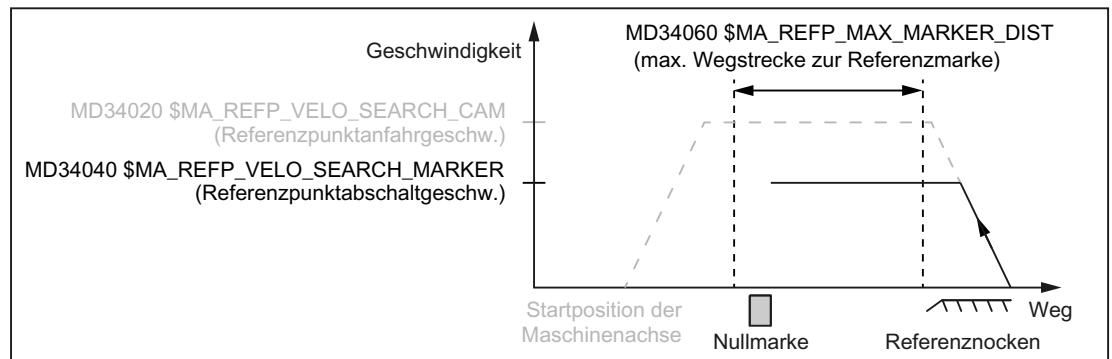


Bild 16-6 Synchronisation mit fallender Referenznockenflanke

Fall 2: Synchronisation mit steigender Referenznockenflanke

Bei der Synchronisation mit steigender Referenznockenflanke beschleunigt die Maschinenachse auf die parametrisierte Referenzpunktanfahrsgeschwindigkeit entgegen der parametrisierten Referenzpunktanfahrriichtung (Verfahrriichtung der Phase 1):

- MD34020 \$MA_REFP_VELO_SEARCH_CAM (Referenzpunktanfahrsgeschwindigkeit)
- MD34010 \$MA_REFP_CAM_DIR_IS_MINUS (Referenzpunkt anfahren in Minusrichtung).

Nach dem Verlassen des Referenznockens wird die Maschinenachse auf Stillstand abgebremst:

DB31, ... DBX12.7 == 0

Anschließend verfährt die Maschinenachse mit der parametrisierten Referenzpunktabschaltgeschwindigkeit zurück auf den Referenznocken:

MD34040 \$MA_REFP_VELO_SEARCH_MARKER (Referenzpunktabschaltgeschwindigkeit)

Mit Erreichen des Referenznockens (DB31, ... DBX12.7 = 1) wird auf die nächste Gebernulmarke gewartet.

Mit dem Erkennen der Gebernulmarke ist die Phase 2 beendet. Die Maschinenachse fährt mit konstanter Geschwindigkeit weiter und das Referenzpunktfahren wird mit Phase 3 fortgesetzt.

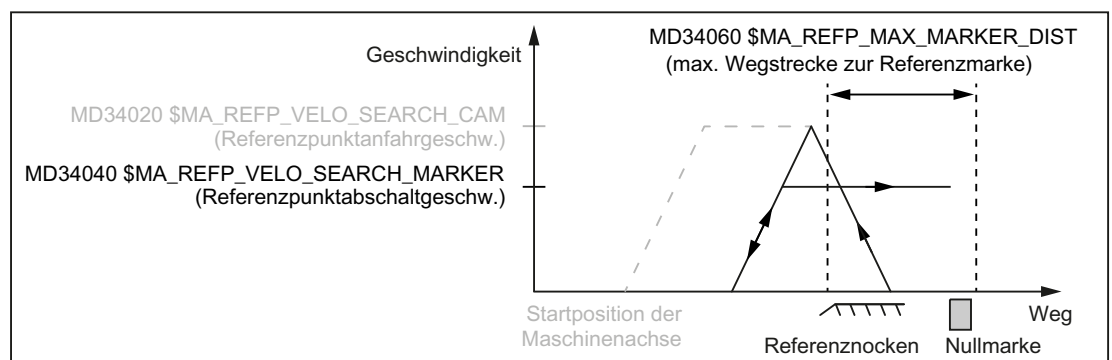


Bild 16-7 Synchronisation mit steigender Referenznockenflanke

Elektronische Referenznockenverschiebung

Die elektronische Referenznockenverschiebung dient der Kompensation temperaturbedingter Längenausdehnungen des Referenznockens, so dass sicher immer auf dieselbe Gebernulldmarke synchronisiert wird:

MD34092 \$MA_REFP_CAM_SHIFT (Elektronische Referenznockenverschiebung für inkrementelle Messsysteme mit äquidistanten Nullmarken)

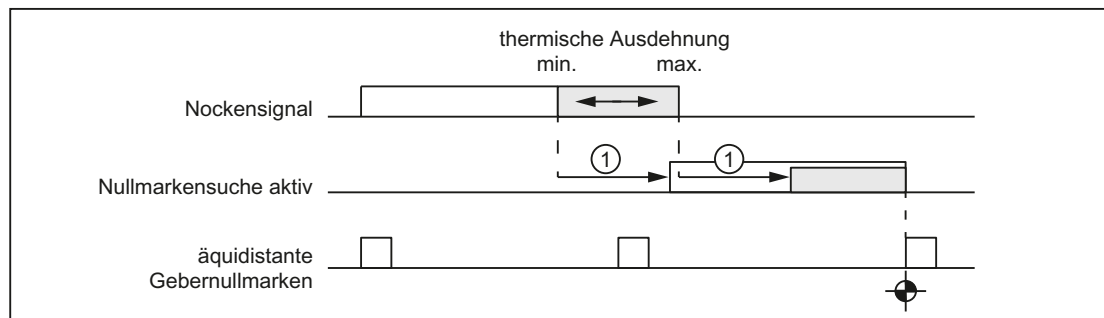
Durch die elektronische Referenznockenverschiebung wird nach dem Erkennen der Referenznockenflanke nicht sofort auf die nächste Gebernulldmarke synchronisiert, sondern erst nach dem zusätzlichen Zurücklegen des parametrisierten Verschiebungsweges.

Aufgrund der Ermittlung des, seit dem Erkennen der Referenznockenflanke zurückgelegte Verfahrensweges im Interpolatortakt, ergibt sich der wirksame Verschiebungsweg s_{shift} zu:

$$s_{\text{shift_min}} = \text{MD34092 \$MA_REFP_CAM_SHIFT}$$

$$s_{\text{shift_max}} = \text{MD34092 \$MA_REFP_CAM_SHIFT} + \text{MD34040 \$MA_REFP_VELO_SEARCH_MARKER} * \text{Interpolatortakt}$$

Die elektronische Referenznockenverschiebung wirkt immer in Suchrichtung der Nullmarke.



① Referenznockenverschiebung

Bild 16-8 Elektronische Referenznockenverschiebung

Voraussetzung

Die elektronische Referenznockenverschiebung wirkt nur bei Maschinenachsen mit Referenznocken:

MD34000 \$MA_REFP_CAM_IS_ACTIVE == 1

Referenznockenjustage

Geber mit äquidistanten Nullmarken

Bei Gebern, die in äquidistanten Abständen Nullmarken liefern, ist auf exakte Justage des Referenznockens zu achten, damit beim Referenzpunktfahren immer die korrekte Nullmarke erkannt wird.

Zeitverhalten

Folgende Faktoren beeinflussen das zeitliche Verhalten vom Erreichen des Referenznockens an der Maschine bis zum Erkennen des vom PLC-Anwenderprogramm an die NC übertragenen Referenznockensignals:

- Schaltgenauigkeit des Referenznockenschalters
- Zeitverzögerung des Referenznockenschalters (Öffner)
- Zeitverzögerung am PLC-Eingang
- PLC-Zykluszeit
- Zykluszeit zur Aktualisierung der NC/PLC-Nahtstelle
- Interpolatortakt
- Lageregeltakt

Einstellhinweise

- Referenznocken
In der Praxis hat sich die Justierung der Flanke des Referenznockens in der Mitte zwischen zwei Nullmarken bewährt.
- Elektronische Referenznockenverschiebung

WARNUNG

Kollisionsgefahr

Bei fehlerhafter oder ungenauer Justage des Referenznockens kann eine falsche Nullmarke ausgewertet werden. Die Steuerung ermittelt dadurch einen falschen Maschinennullpunkt. Demzufolge fährt die Maschinenachse auf falsche Positionen. Software-Endschalter, Schutzbereiche und Arbeitsfeldbegrenzungen wirken auf falsche Positionen und können so die Maschine nicht schützen. Die Wegdifferenz entspricht +/- dem von der Maschinenachse zurückgelegten Weg zwischen 2 Nullmarken.

Als Anhaltspunkt zur Parametrierung der elektronischen Referenznockenverschiebung dient das nur lesbare Maschinendatum:

MD34093 \$MA_REFP_CAM_MARKER_DIST (Abstand Referenznocken/Referenzmarke)

Der angezeigte Wert entspricht der Distanz zwischen dem Verlassen des Referenznockens und dem Auftreten der Referenzmarke. Bei zu kleinen Werten besteht die Gefahr, dass die Ermittlung des Referenzpunktes aufgrund von Temperatureinflüssen oder einer schwankenden Laufzeit des Nockensignals nicht deterministisch ist.

Phase 2: Eigenschaften

- Vorschub-Override ist **nicht** wirksam.
Intern wird mit Vorschub-Override = 100% verfahren.
Bei einer vorgegebenen Vorschub-Override von 0% erfolgt Abbruch.
- Vorschubhalt (kanal- und achsspezifisch) ist wirksam.
- NC-Stop und NC-Start sind **nicht** wirksam.
- Erreicht die Maschinenachse in Phase 2 nicht innerhalb der parametrierten Wegstrecke die Referenzmarke (Gebernulmarke), wird die Maschinenachse angehalten:
MD34060 \$MA_REFP_MAX_MARKER_DIST (max. Wegstrecke zur Referenzmarke)

16.5.6 Phase 3: Fahren auf den Referenzpunkt

Phase 3: Grafische Darstellung

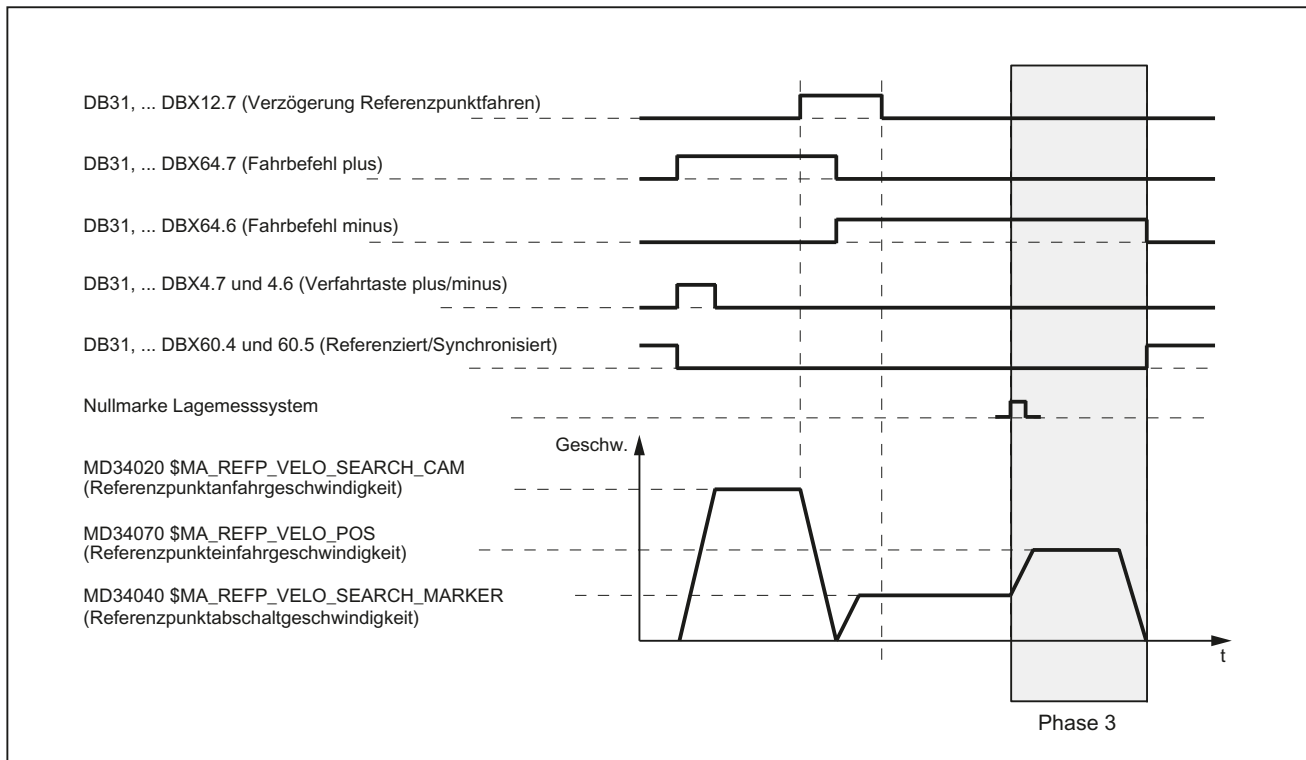


Bild 16-9 Phase 3: Fahren auf den Referenzpunkt

Phase 3: Start

Die Maschinenachse fährt am Ende der Phase 2 mit Referenzpunktabschaltgeschwindigkeit. Sobald Phase 2 erfolgreich ohne Alarm beendet wurde, wird daher ohne Anzuhalten in Phase 3 gewechselt.

Ausgangssituation

Die Gebernulmarke wurde erkannt.

Phase 3: Ablauf

Die Maschinenachse verfährt mit der parametrisierten Referenzpunkteinfahrtgeschwindigkeit: MD34070 \$MA_REFP_VELO_POS (Referenzpunkteinfahrtgeschwindigkeit) von der in Phase 2 erkannten Gebernulmarke bis zum Referenzpunkt.

Der dabei zu verfahrenende Weg s_{ref} ergibt sich aus der Summe von Referenzpunktabstand und Referenzpunktverschiebung:

MD34080 \$MA_REFP_MOVE_DIST (Referenzpunktabstand)

MD34090 \$MA_REFP_MOVE_DIST_CORR (Referenzpunktverschiebung)

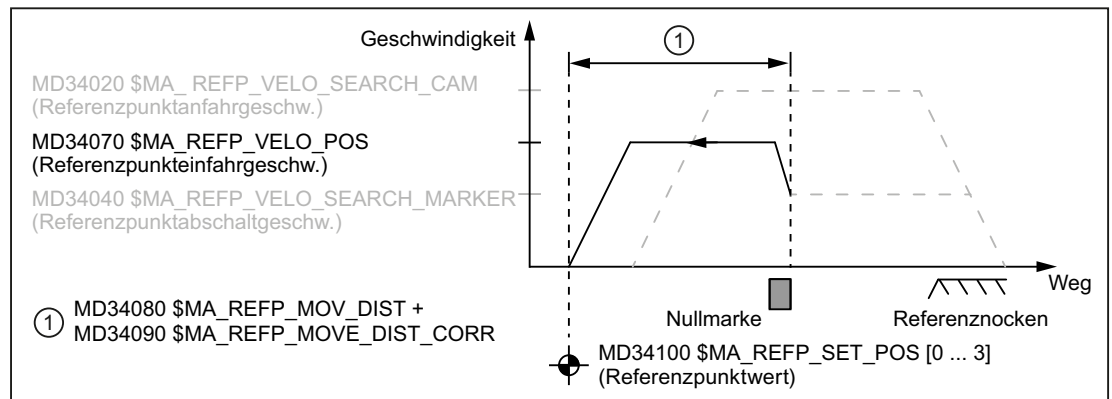


Bild 16-10 Referenzpunktposition

Mit Erreichen des Referenzpunktes, wird die Maschinenachse angehalten und das Istwertsystem auf den über die NC/PLC-Nahtstelle vorgegeben Referenzpunktwert n synchronisiert:

MD34100 \$MA_REFP_SET_POS[< n >] (Referenzpunktwert)

Die Auswahl des Referenzpunktwertes erfolgt über die NC/PLC-Nahtstelle:

DB31, ... DBX2.4 ... 7 (Referenzpunktwert 1 ... 4)

Das Istwertsystem wird auf den Referenzpunktwert synchronisiert, der zum Zeitpunkt des Erreichens des Referenznockens in Phase 1 (DB31, ... DBX12.7 == 1) ausgewählt war.

Die Maschinenachse ist jetzt referenziert. Als Rückmeldung an das PLC-Anwenderprogramm wird, abhängig vom aktiven Messsystem, das Nahtstellensignal gesetzt:

DB31, ... DBX60.4 / .5 (Referenziert/Synchronisiert 1 / 2) = 1

Eigenschaften der Phase 3

- Vorschub-Override ist wirksam.
- Vorschubhalt (kanal- und achsspezifisch) ist wirksam.
- NC-Stop und NC-Start sind wirksam.

Besonderheiten der Phase 3

In folgenden Fällen hält die Maschinenachse nach dem Erkennen der Nullmarke zuerst an, um dann auf den Referenzpunkt zurückzufahren:

- Die Summe aus Referenzpunktabstand und Referenzpunktverschiebung ist kleiner dem, aufgrund der Referenzpunkteinfahrtsgeschwindigkeit, benötigten Bremsweg:
 $MD34080 + MD34090 < \text{"Benötigter Bremsweg aufgrund MD34070"}$
- Der Referenzpunkt liegt, entgegen der aktuellen Fahrtrichtung, "hinter" dem Referenznocken.

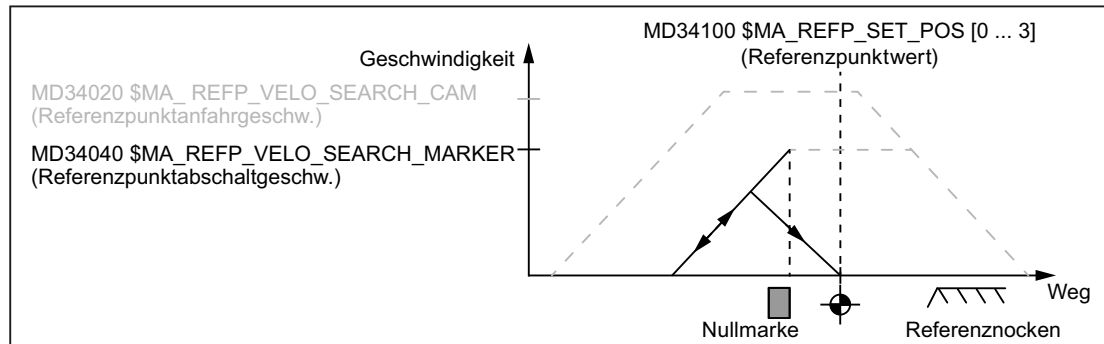


Bild 16-11 Referenzpunktabstand plus Referenzpunktverschiebung kleiner Bremsweg

16.6 Referenzieren bei abstandscodierten Referenzmarken

16.6.1 Allgemeiner Überblick

Abstandscodierte Referenzmarken

Messsysteme mit abstandscodierten Referenzmarken bestehen aus zwei parallel verlaufenden Maßstabsspuren:

- Inkrementales Strichgitter
- Referenzmarkenspur

Der Abstand zwischen jeweils zwei aufeinander folgenden Referenzmarken ist definiert. Somit lässt sich nach dem Überfahren zweier aufeinander folgender Referenzmarken die absolute Position der Maschinenachse bestimmen. Bei einem Referenzmarken-Abstand von z. B. ca. 10 mm wird zum Referenzieren der Maschinenachse nur ein Fahrweg von ca. 20 mm benötigt.

Das Referenzieren kann von jeder beliebigen Achsposition aus in positiver wie negativer Richtung erfolgen (Ausnahme: Verfahrbereichsenden).

16.6.2 Grundlegende Parametrierung

Lineare Messsysteme

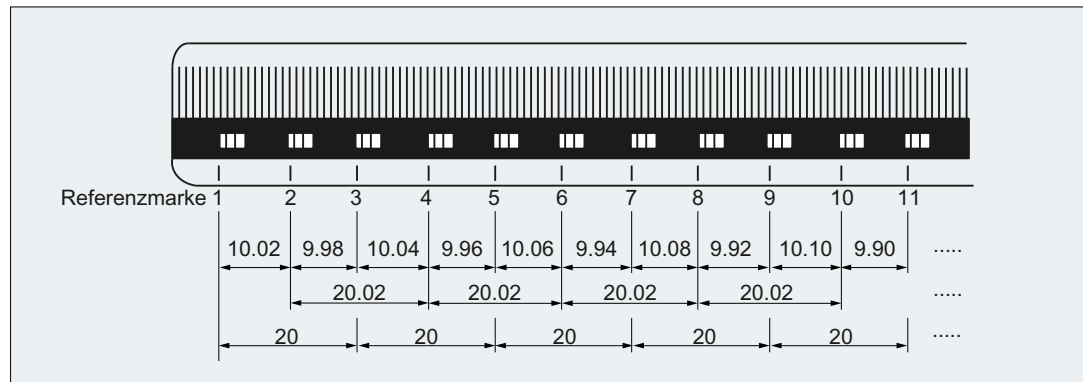


Bild 16-12 Glasmaßstab mit abstandscodierten Referenzmarken, Teilungsperiode: 20 mm

Zur Parametrierung von linearen Messsystemen sind folgende Daten einzustellen:

- Die absolute Verschiebung zwischen dem Maschinennullpunkt und der Position der ersten Referenzmarke des Längenmesssystems:
MD34090 \$MA_REFP_MOVE_DIST_CORR (Referenzpunkt-/Absolutverschiebung)
Siehe dazu unten: Ermittlung der Absolutverschiebung
- Die Orientierung des Längenmesssystems (gleich- oder gegensinnig) gegenüber dem Maschinenkoordinatensystem:
MD34320 \$MA_ENC_INVERS (Längenmesssystem gegensinnig zum Maschinensystem)

Rotatorisches Messsystem

Für rotatorische Messsysteme gilt sinngemäß das Gleiche wie für lineare Messsysteme (s. o.).

Ermittlung der Absolutverschiebung

Zur Ermittlung der Absolutverschiebung zwischen dem Maschinennullpunkt und der Position der ersten Referenzmarke einer Maschinenachse wird folgendes Vorgehen empfohlen:

1. Für die Absolutverschiebung den Wert Null eingeben:
MD34090 \$MA_REFP_MOVE_DIST_CORR = 0
2. Referenzpunktfahren durchführen.
Hinweis: Das Referenzpunktfahren sollte an einer Stelle in der Maschine durchgeführt werden, an der z. B. mittels eines Laserinterferometers die genaue Position der Maschinenachse bezüglich des Maschinennullpunkts einfach ermittelt werden kann.
3. Von der Bedienoberfläche z.B. SINUMERIK Integrate, die angezeigte Istposition der Maschinenachse im Maschinenkoordinatensystem MKS ablesen.

4. Die aktuelle Position der Maschinenachse bezogen auf den Maschinennullpunkt messen.
5. Die Absolutverschiebung berechnen und in das MD34090 eintragen.
Die Absolutverschiebung berechnet sich, abhängig von der Orientierung des Messsystems gegenüber dem Maschinensystemkoordinatensystem, zu:

Orientierung des Messsystems	Absolutverschiebung
gleichsinnig	(Gemessene Position) + (Angezeigte Istposition)
gegensinnig	(Gemessene Position) - (Angezeigte Istposition)

WARNUNG

Referenzpunktabweichung

Nach der Ermittlung der Absolutverschiebung und dem Eintrag im MD34090 muss das Referenzpunktfahren für die Maschinenachse erneut durchgeführt werden.

Referenziermode

Für das Referenzieren mit abstandscodierten Referenzmarken stehen zwei Modi zur Verfügung:

- Auswertung von **zwei** aufeinander folgenden Referenzmarken:
MD34200 \$MA_ENC_REFP_MODE = 3
Vorteil:
 - Kurzer Verfahrenweg
- Auswertung von **vier** aufeinander folgenden Referenzmarken:
MD34200 \$MA_ENC_REFP_MODE = 8
Vorteil:
 - Plausibilitätsprüfung durch die NC möglich
 - Erhöhung der Sicherheit des Referenzierergebnisses

16.6.3 Zeitlicher Ablauf

Zeitlicher Ablauf

Der zeitliche Ablauf beim Referenzieren mit abstandscodierten Referenzmarken unterteilt sich in 2 Phasen:

- Phase 1: Überfahren der Referenzmarken mit Synchronisation
- Phase 2: Fahren auf einen festen Zielpunkt

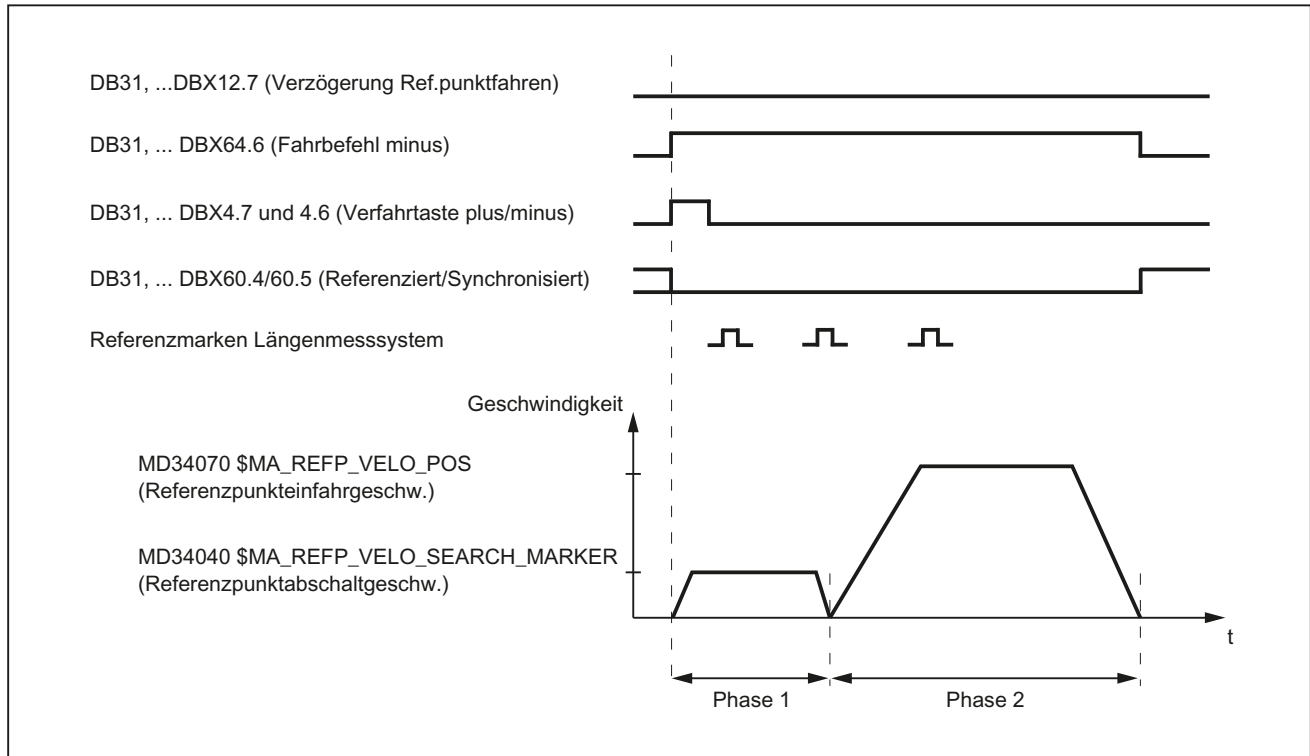


Bild 16-13 Abstandscodierte Referenzmarken

16.6.4 Phase 1: Überfahren der Referenzmarken mit Synchronisation

Phase 1: Start

Zum Starten des Referenzpunktfahrens siehe Kapitel "Achsspezifisches Referenzieren (Seite 1244)" und "Kanalspezifisches Referenzieren (Seite 1246)".

Referenznocken

Bei Messsystemen mit abstandscodierten Referenzmarken ist für das eigentliche Referenzieren kein Referenznocken erforderlich. Aus funktionalen Gründen wird aber für das kanalspezifische Referenzpunktfahren und beim Referenzpunktfahren aus dem Teileprogramm (G74) vor einem Verfahrbereichsende der Maschinenachse ein Referenznocken benötigt.

Phase 1: Ablauf

Ablauf ohne Berührung eines Referenznockens

Nach dem Start des Referenzpunktfahrens beschleunigt die Maschinenachse auf die parametrierte Referenzpunktabschaltgeschwindigkeit:

MD34040 \$MA_REFP_VELO_SEARCH_MARKER (Referenzpunktabschaltgeschwindigkeit)

Nach dem Überfahren der parametrierten Anzahl von Referenzmarken wird die Maschinenachse wieder angehalten und das Istwertsystem der Maschinenachse auf die von der NC ermittelte Absolutposition synchronisiert.

Ablauf bei Start auf dem Referenznocken

Steht die Maschinenachse beim Start des Referenzpunktfahrens auf dem Referenznocken, beschleunigt sie auf die parametrierte Referenzpunktabschaltgeschwindigkeit entgegen der parametrierten Referenzpunktanfahrriichtung:

MD34040 \$MA_REFP_VELO_SEARCH_MARKER (Referenzpunktabschaltgeschwindigkeit)

MD34010 \$MA_CAM_DIR_IS_MINUS (Referenzpunktanfahren in Minusrichtung)

Somit ist sichergestellt, dass die Maschinenachse nicht vor dem Überfahren der parametrierten Anzahl von Referenzmarken an die Verfahrbereichsgrenze stößt.

Nach dem Überfahren der parametrierten Anzahl von Referenzmarken wird die Maschinenachse wieder angehalten und das Istwertsystem der Maschinenachse auf die von der NC ermittelte Absolutposition synchronisiert.

Ablauf bei Berührung des Referenznockens während des Referenzierens

Nach dem Start des Referenzpunktfahrens beschleunigt die Maschinenachse auf die parametrierte Referenzpunktabschaltgeschwindigkeit:

MD34040 \$MA_REFP_VELO_SEARCH_MARKER (Referenzpunktabschaltgeschwindigkeit)

Vor dem Überfahren der parametrierten Anzahl von Referenzmarken berührt die Maschinenachse den Referenznocken. Daraufhin wird sie reversiert und die Referenzmarkensuche in entgegengesetzter Richtung erneut gestartet.

Nach dem Überfahren der parametrierten Anzahl von Referenzmarken wird die Maschinenachse wieder angehalten und das Istwertsystem der Maschinenachse auf die von der NC ermittelte Absolutposition synchronisiert.

Plausibilitätsprüfung des Referenzmarkenabstands

Ein Fehler liegt vor, wenn die NC beim Referenzpunktfahren für zwei aufeinander folgende Referenzmarken einen größeren Abstand als dem Doppelten des parametrierten Referenzmarkenabstandes ermittelt:

MD34300 \$MA_ENC_REFP_MARKER_DIST (Referenzmarkenabstand)

Die Maschinenachse wird daraufhin mit der halben parametrierten Referenzpunktabtastgeschwindigkeit (MD34040) in entgegengesetzter Richtung verfahren und die Referenzmarkensuche erneut gestartet.

Wird erneut ein fehlerhafter Referenzmarkenabstand ermittelt, wird die Maschinenachse angehalten und das Referenzpunktfahren abgebrochen (Alarm 20003 "Fehler im Messsystem").

Abbruchkriterium

Wird nicht innerhalb der parametrierten Wegstrecke die parametrierte Anzahl von Referenzmarken erkannt, wird die Maschinenachse angehalten und das Referenzpunktfahren abgebrochen:

MD34060 \$MA_REFP_MAX_MARKER_DIST (max. Wegstrecke zur Referenzmarke)

Eigenschaften der Phase 1

Nach erfolgreichem Abschluss der Phase 1 ist das Istwertsystem der Maschinenachse synchronisiert.

16.6.5 Phase 2: Fahren auf Zielpunkt

Phase 2: Start

Phase 2 wird automatisch gestartet, sobald Phase 1 ohne Alarm beendet wurde.

Ausgangssituation:

- Die Maschinenachse steht unmittelbar nach der letzten der parametrierten Anzahl von Referenzmarken.
- Das Istwertsystem der Maschinenachse ist synchronisiert.

Phase 2: Ablauf

In Phase 2 wird die Maschinenachse zum Abschluss des Referenzpunktfahrens auf eine definierte Zielposition (Referenzpunkt) gefahren. Dieses Verhalten kann zur Abkürzung des Referenzpunktfahrens auch unterdrückt werden:

MD34330 \$MA_STOP_AT_ABS_MARKER = <Wert>

Wert	Bedeutung
0	Fahren auf Zielposition
1	Kein Fahren auf Zielposition

Fahren auf Zielposition (Normalfall)

Die Maschinenachse beschleunigt auf die parametrierte Referenzpunkteinfahrtgeschwindigkeit und fährt auf den parametrierten Zielpunkt (Referenzpunkt):

MD34070 \$MA_REFP_VELO_POS (Referenzpunkteinfahrtgeschwindigkeit)

MD34100 \$MA_REFP_SET_POS (Referenzpunktwert)

Die Maschinenachse ist referenziert. Als Kennung setzt die NC, abhängig vom aktiven Messsystem, das entsprechende Nahtstellensignal:

DB31, ... DBX60.4 / 60.5 (Referenziert/Synchronisiert 1/2) = 1

Kein Fahren auf Zielposition

Die Maschinenachse ist jetzt referenziert. Als Kennung setzt die NC, abhängig vom aktiven Messsystem, das entsprechende Nahtstellensignal:

DB31, ... DBX60.4 / 60.5 (Referenziert/Synchronisiert 1/2) = 1

Eigenschaften der Phase 2

Abhängig davon, ob für die Maschinenachse ein Referenznocken parametriert ist, weist die Phase 2 unterschiedliche Eigenschaften auf.

Maschinenachse ohne Referenznocken

MD34000 \$MA_REFP_CAM_IS_ACTIVE (Achse mit Referenznocken) = 0

Eigenschaften:

- Vorschub-Override ist wirksam.
- Vorschubhalt (kanalspezifisch und achsspezifisch) ist wirksam.
- NC-Stop und NC-Start sind wirksam.

Maschinenachse mit Referenznocken

MD34000 \$MA_REFP_CAM_IS_ACTIVE (Achse mit Referenznocken) = 1

Eigenschaften:

- Vorschub-Override ist **nicht** wirksam.
Intern wird mit Vorschub-Override = 100% verfahren.
Bei einer vorgegebenen Vorschub-Override von 0% erfolgt Abbruch.
- Vorschubhalt (kanalspezifisch und achsspezifisch) ist wirksam.

- NC-Stop und NC-Start sind **nicht** wirksam.
- Wird nach dem Verlassen des Referenznockens nicht innerhalb der parametrierten Wegstrecke die parametrierte Anzahl von Referenzmarken erkannt, wird die Maschinenachse angehalten:
MD34060 \$MA_REFP_MAX_MARKER_DIST (max. Wegstrecke zur Referenzmarke)

Besonderheiten bei rotatorischen Messsystemen

Bei rotatorischen, abstandscodierten Messsystemen ist die absolute Position nur innerhalb einer Umdrehung eindeutig bestimmbar. Abhängig vom mechanischen Anbau des Gebers stimmt der Hardware-seitige Überlauf der Absolutposition nicht immer mit dem Verfahrbereich der Rundachse überein.

Besonderheiten bei Modulo-Rundachsen

Bei Modulo-Rundachsen wird die Referenzpunktposition auf den parametrierten Modulo-Bereich abgebildet:

MD30330 \$MA_MODULO_RANGE (Größe des Modulo-Bereichs)

MD30340 \$MA_MODULO_RANGE_START (Startposition des Modulo-Bereichs)

Hinweis

Die Abbildung der Referenzpunktposition auf den parametrierten (fiktiven) Modulo-Bereich erfolgt auch bei Achsfunktion "Bestimmung der Referenzpunktposition rotatorischer, abstandscodierter Geber innerhalb des projizierten Modulo-Bereichs":

MD30455 \$MA_MISC_FUNCTION_MASK (Achsfunktionen), BIT1 = 1

16.7 Referenzieren über Istwertabgleich

16.7.1 Istwertabgleich auf das referenzierende Messsystem

Funktion

Bei Istwertabgleich auf das referenzierende Messsystem wird nach dem erfolgreichen Referenzieren eines Messsystems einer Maschinenachse die ermittelte absolute Istposition direkt auf alle anderen Messsysteme der Maschinenachse übertragen und diese als referenziert gekennzeichnet:

DB31, ... DBB60.4 / 60.5 (Referenziert/Synchronisiert 1/2) = 1

Vorteil

Die Maschinenachse kann beim Umschalten vom explizit referenzierten Messsystem auf das per Istwertabgleich referenzierte Messsystem kontinuierlich in Regelung bleiben (Reglerfreigabe aktiv), da durch die abgegliche Istposition kein Istwertsprung stattfindet.

Hinweis

Zur Erhöhung der Positioniergenauigkeit durch Ermittlung der Messsystem-spezifischen Geberfeininformation wird empfohlen, das per Istwertabgleich referenzierte Messsystem nach dem Umschalten explizit neu zu referenzieren.

Aktivierung

Die Aktivierung des Istwertabgleichs auf das referenzierende Messsystem erfolgt maschinenachsspezifisch über:

MD34102 \$MA_REFP_SYNC_ENCS = 1

16.7.2 Istwertabgleich bei Messsystemen mit abstandscodierten Referenzmarken

Funktion

Zur Erhöhung der Positioniergenauigkeit durch Ermittlung der Messsystem-spezifischen Geberfeininformation wird empfohlen, das per Istwertabgleich referenzierte Messsystem nach einer Messsystemumschaltung explizit neu zu referenzieren.

Wird für das passive Messsystem ein Geber mit abstandscodierten Referenzmarken verwendet, kann unter folgenden Voraussetzungen das Referenzieren vermieden werden:

1. Aktives Messsystem: indirektes Messsystem (Motormesssystem) mit z. B. Absolutwertgeber
2. Passives Messsystem: direktes Messsystem mit abstandscodierten Referenzmarken
3. Verfahrbewegung der Maschinenachse mit dem referenzierten indirekten Messsystem vor der Messsystemumschaltung so, dass die zum Referenzieren erforderliche Anzahl von Referenzmarken überfahren wird. Dabei wird das passive direkte Messsystem automatisch referenziert.

Parametrierung

Neben den spezifischen Maschinendaten zum Referenzieren der einzelnen Messsysteme sind folgende Maschinendaten einzustellen:

- Freigabe des Istwertabgleichs:
MD34102 \$MA_REFP_SYNC_ENCS = 1
- Direktes Messsystem mit abstandscodierten Referenzmarken:
 - MD34200 \$MA_ENC_REFP_MODE[*Messsystem*] = 3
Abstandscodierte Referenzmarken
 - MD30242 \$MA_ENC_IS_INDEPENDENT[*Messsystem*] = 2
Beim Istwertabgleich wird das passive direkte Messsystem auf die Istposition des aktiven indirekten Messsystems abgeglichen, jedoch nicht als referenziert gekennzeichnet. Nach dem Überfahren der parametrierten Anzahl von Referenzmarken wird das passive direkte Messsystem automatisch referenziert. Das Referenzieren erfolgt in jeder Betriebsart.

Ablauf

1. Ausgangssituation: Beide Messsysteme sind nicht referenziert:
DB31, ... DBX60.4 = 0 (Referenziert / Synchronisiert 1)
DB31, ... DBX60.5 = 0 (Referenziert / Synchronisiert 2)
2. Referenzieren des indirekten Messsystems entsprechend Messsystemtyp:
DB31, ... DBX60.4 = 1 (Referenziert / Synchronisiert 1)
DB31, ... DBX60.5 = 0 (Referenziert / Synchronisiert 2)
3. Verfahren der Maschinenachse über die parametrierte Anzahl von Referenzmarken.
Dadurch automatisches Referenzieren des direkten Messsystems:
DB31, ... DBX60.4 = 1 (Referenziert / Synchronisiert 1)
DB31, ... DBX60.5 = 1 (Referenziert / Synchronisiert 2)

16.8 Referenzieren im Nachführbetrieb

Funktion

Inkrementelle Messsysteme und Messsysteme mit abstandscodierten Referenzmarken können auch während Nachführbetrieb der Maschinenachse referenziert werden. Voraussetzung dafür ist die korrekte Parametrierung des Referenzpunktfahrens entsprechend dem verwendeten Messsystem (siehe Kapitel "Referenzieren bei inkrementellen Messsystemen (Seite 1249)" und "Referenzieren bei abstandscodierten Referenzmarken (Seite 1262)").

Beim Referenzieren im Nachführbetrieb wird die Maschinenachse nicht von der NC, sondern über eine externe Verfahrbewegung über die Gebernulmarke bzw. die parametrierte Anzahl von abstandscodierten Referenzmarken bewegt. Mit dem Erkennen der Gebernulmarke bzw.

der parametrierten Anzahl von abstandscodierten Referenzmarken ist das Messsystem referenziert.

Hinweis

Reproduzierbarkeit des Referenzierungsergebnisses

Beim NC-geführten Referenzpunktfahren ist die Reproduzierbarkeit des Referenzierungsergebnisses durch die Einhaltung der parametrierten Verfahrensgeschwindigkeiten während des Referenzvorgangs gewährleistet. Beim Referenzieren im Nachführbetrieb liegt die Verantwortung beim Maschinenhersteller / Anwender, durch geeignete Maßnahmen die Reproduzierbarkeit der Referenzierungsergebnisse zu erzielen.

Eindeutigkeit der Nullmarke

Das Referenzieren eines inkrementellen Messsystems basiert auf der eindeutigen Lage der Gebernulldmarke bezogen auf den gesamten Verfahrbereich der Maschinenachse.

Da beim Referenzieren im Nachführbetrieb das Referenznocken-Signal von der NC nicht ausgewertet wird, ist beim Referenzieren im Nachführbetrieb ohne weitere Maßnahmen eine eindeutige Ermittlung des Referenzpunktes nur gegeben bei:

- nur einer Gebernulldmarke im Verfahrbereich der Maschinenachse.
- linearen Messsystemen mit abstandscodierten Referenzmarken.
- Modulo-Rundachsen (Absolutposition innerhalb einer Umdrehung).

Nullmarkenauswahl bei mehreren Nullmarkensignalen

Werden aufgrund von maschinenspezifischen Gegebenheiten, z. B. Untersetzungsgetriebe zwischen Geber und Last, mehrere Gebernulldmarken im Verfahrbereich der Maschinenachse erkannt, muss zur eindeutigen Bestimmung des Referenzpunktes ein berührungsloser Näherungsschalter an der Maschine angebracht und über einen digitalen Eingang der NCU-Schnittstelle angeschlossen werden.

Hinweis

Berührungsloser Näherungsschalter-Signal: Nullmarkenauswahl

Die Verarbeitung des berührungslosen Näherungsschalter-Signals erfolgt ausschließlich im Antrieb. Anschluss und Parametrierung siehe Kapitel "Hardware-Signale (Seite 1249)".

Freigabe

Die Freigabe der Funktion "Referenzieren im Nachführbetrieb" erfolgt über:

MD34104 \$MA_REFP_PERMITTED_IN_FOLLOWUP = TRUE

Start des Referenzvorgangs

Befindet sich die Maschinenachse bei Start des Referenzpunktfahrens im Nachführbetrieb (DB31, ... DBX61.3 == TRUE) wird das Messsystem im Nachführbetrieb referenziert.

Befindet sich die Maschinenachse bei Start des Referenzpunktfahrens nicht im Nachführbetrieb, wird das "normale" von der NC gesteuerte Referenzpunktfahren ausgeführt

Das Referenzieren im Nachführen kann in folgenden Betriebsarten gestartet werden:

- JOG-REF: Verfahrtasten
- AUTOMATIK: Teileprogrammanweisung G74

Ablauf des Referenziervorgangs (Betriebsart JOG-REF)

1. Nachführbetrieb der Maschinenachse aktivieren:
DB31, ... DBX1.4 (Nachführbetrieb) = 1
DB31, ... DBX2.1 (Reglerfreigabe) = 0
2. Warten auf Aktivierung des Nachführbetriebs:
DB31, ... DBX61.3 (Nachführen aktiv) == 1
3. Wechsel in die Betriebsart JOG, Maschinenfunktion REF.
4. Externes Verfahren der Maschinenachse über die Gebernulmarke bzw. parametrisierte Anzahl von abstandscodierten Referenzmarken. Der Referenziervorgang wird NC-intern gestartet, sobald die Maschinenachse bewegt wird. Als Rückmeldung wird folgendes NC/PLC-Nahtstellensignal zurückgesetzt:
DB31, ... DBX61.4 (Achse/Spindel steht) == 0
5. Nach erfolgreichem Erkennen der Gebernulmarke bzw. parametrisierten Anzahl von abstandscodierten Referenzmarken ist das Messsystem referenziert. Als Rückmeldung wird folgendes NC/PLC-Nahtstellensignal gesetzt:
DB31, ... DBX60.4/60.5 (Referenziert/Synchronisiert 1/2) == 1

Abbruch des Referenziervorgangs

Ein aktiver Referenziervorgang wird abgebrochen durch:

- Abwahl des Nachführbetriebs
- NC-Reset

Verhalten bei bereits referenzierten Messsystemen

Ein bereits referenziertes Messsystem kann nur in der Betriebsart AUTOMATIK über die Teileprogrammanweisung G74 erneut referenziert werden.

Ablauf des Referenziervorgangs (Betriebsart AUTOMATIK)

1. Wechsel in die Betriebsart AUTOMATIK.
2. Start des Teileprogramms.
3. Nachführbetrieb der Maschinenachse aktivieren:
DB31, ... DBX1.4 (Nachführbetrieb) = 1
DB31, ... DBX2.1 (Reglerfreigabe) = 0
4. Warten auf Aktivierung des Nachführbetriebs:
DB31, ... DBX61.3 (Nachführen aktiv) == 1
5. Der Referenziervorgang wird NC-intern gestartet, sobald der Befehl G74 abgearbeitet wird.

6. Externes Verfahren der Maschinenachse über die Gebernulldmarke bzw. parametrisierte Anzahl von abstandscodierten Referenzmarken.
7. Nach erfolgreichem Erkennen der Gebernulldmarke bzw. parametrisierten Anzahl von abstandscodierten Referenzmarken ist das Messsystem referenziert. Als Rückmeldung wird folgendes NC/PLC-Nahtstellensignal gesetzt:
DB31, ... DBX60.4 / 60.5 (Referenziert/Synchronisiert 1/2) == 1
8. Der Satzwechsel erfolgt nach dem erfolgreichen Abschluss des Referenziervorgangs.

Abbruch des Referenziervorgangs

Ein aktiver Referenziervorgang wird abgebrochen durch:

- Abwahl des Nachführbetriebs
- NC-Reset

Verhalten bei bereits referenzierten Messsystemen

Ein bereits referenziertes Messsystem kann erneut referenziert werden.

16.9 Referenzieren bei Absolutwertgebern

16.9.1 Informationen zur Justage

Maschinenachsen mit Absolutwertgebern

Der Vorteil von Maschinenachsen mit Absolutwertgebern ist, dass nach einem einmaligen Justage-Vorgang das bei inkrementellen Messsystemen notwendige Referenzpunktfahren (z. B. Hochlauf der Steuerung, Abwahl von "Parken" der Maschinenachse, etc.) entfallen kann und das Istwertsystem der Maschinenachse sofort auf die ermittelte Absolutposition synchronisiert wird.

Justage

Bei der Justage eines Absolutwertgebers wird der Istwert des Gebers einmalig mit dem Maschinennullpunkt abgeglichen und anschließend gültig gesetzt.

Der aktuelle Justage-Status eines Absolutwertgebers wird in dem folgenden achsspezifischen Maschinendatum der Maschinenachse angezeigt, an die er angeschlossen ist:

MD34210 \$MA_ENC_REFP_STATE (Status Absolutwertgeber)

Wert	Bedeutung
0	Geber nicht justiert
1	Geberjustage freigegeben
2	Geber ist justiert

Justage-Arten

Folgende Justage-Arten werden unterstützt:

- Justage durch Eingabe der Referenzpunktverschiebung
- Justage durch Eingabe der Referenzpunktwertes
- Automatische Justage mittels Messtaster
- Justage mittels berührungslosen Näherungsschalters

Neujustage

Eine erneute Justage des Absolutwertgebers ist erforderlich nach:

- Getriebeumschaltung zwischen Last und Absolutwertgeber
- Ab-/Anbau des Absolutwertgebers
- Ab-/Anbau des Motors mit Absolutwertgeber
- Datenverlust im statischen NC-Speicher

- Batteriespannungsausfall
- Istwertsetzen (PRESETON)

WARNUNG

Datensicherung

Beim Sichern von Maschinendaten einer Maschine A wird der Geberstatus (MD34210) der Maschinenachsen mitgesichert.

Beim Laden dieses Datensatzes in eine Maschine B des gleichen Typs, z. B. im Rahmen einer Serieninbetriebnahme oder nach einem Servicefall, werden die betroffenen Maschinenachsen von der NC automatisch als justiert / referenziert betrachtet. Es liegt in der besonderen Verantwortung des Maschinenherstellers / Anwenders, in solchen Fällen eine Neujustage vorzunehmen.

Siehe dazu auch die Erläuterungen zum Maschinendatum:

MD30250 \$MA_ACT_POS_ABS (Absolute Geberposition zum Ausschaltzeitpunkt)

Hinweis

Dass eine Neujustage des Absolutwertgebers erforderlich ist, wird von der Steuerung nur bei folgenden Ereignissen erkannt:

- Getriebeumschaltung mit Wechsel des Übersetzungsverhältnisses
- Ansprechen der Nullmarkenüberwachung
- Neue Geber-Seriennummer nach einem Tausch des Absolutwertgebers

Die Steuerung setzt daraufhin den Status des Absolutwertgebers auf den Wert "0":

MD34210 \$MA_ENC_REFP_STATE = 0 (Geber nicht justiert)

Es wird folgender Alarm angezeigt:

Alarm 25022 "Achse <Achsname> Geber <Nummer> Warnung 0"

Bei Ansprechen der Nullmarkenüberwachung wird zusätzlich folgender Alarm angezeigt:

Alarm 25020 "Achse <Achsname> Nullmarkenueberwachung aktiver Geber"

In allen anderen Fällen (z. B. PRESETON) liegt es in der alleinigen Verantwortung des Anwenders die Dejustage des Absolutwertgebers durch **manuelles** Setzen des Status auf den Wert "0" anzuzeigen und eine Neujustage durchzuführen.

16.9.2 Justage durch Eingabe der Referenzpunktverschiebung

Funktion

Bei der Justage durch Eingabe der Referenzpunktverschiebung wird die Differenz zwischen der auf der Bedienoberfläche angezeigten und der tatsächlichen Istposition in der Maschine ermittelt und der NC als Referenzpunktverschiebung bekannt gemacht.

Vorgehensweise

1. Position der Maschinenachse bezogen auf den Maschinennullpunkt ermitteln, z. B. durch:
 - Messen der Position (z. B. Laserinterferometer)
 - Fahren der Maschinenachse auf eine bekannte Position (z. B. Festanschlag)
2. Auf der Bedienoberfläche angezeigte Istposition der Maschinenachse ablesen.
3. Referenzpunktverschiebung berechnen (Differenz der unter Punkt 1 und 2 ermittelten Istpositionen) und im Maschinendatum eintragen:
MD34090 \$MA_REFP_MOVE_DIST_CORR (Referenzpunktverschiebung)
4. Absolutwertgeber als justiert kennzeichnen:
MD34210 \$MA_ENC_REFP_STATE = 2

Hinweis

Die Geberjustage wird erst mit der nächsten Geberaktivierung (z. B. Hochlauf der Steuerung) wirksam.

5. POWER ON - Reset auslösen.
6. Auf der Bedienoberfläche angezeigte Position der Maschinenachse kontrollieren.

Hinweis**Losekompensation**

Wenn für ein Messsystem mit Absolutwertgeber eine Losekompensation parametrier ist, muss auf Folgendes geachtet werden:

Beim Verfahren der Maschinenachse auf die Justage-Maschinenposition darf keine Lose entstehen!

Referenzpunktverschiebung dauerhaft aktivieren

Erst nach POWER ON - Reset wird die eingegebene Referenzpunktverschiebung (MD34090) dauerhaft wirksam. Wird die Maschinenachse nach der Justage des Absolutwertgebers ohne zwischenzeitlichen POWER ON - Reset verfahren, kann die im Maschinendatum eingegebene Referenzpunktverschiebung z. B. im Rahmen interner Überlaufkorrekturen überschrieben werden.

Überprüfen der Istposition

Es wird nach der Justage des Absolutwertgebers empfohlen, nach dem nächsten Hochlauf der Steuerung (POWER ON) die Istposition der Maschinenachse zu verifizieren.

16.9.3 Justage durch Eingabe des Referenzpunktwertes

Funktion

Bei der Justage durch Eingabe des Referenzpunktwertes wird die absolute Position der Maschinenachse bezogen auf den Maschinennullpunkt ermittelt, z. B. durch:

- Messen der Position (z. B. Laserinterferometer)
- Fahren der Maschinenachse auf eine bekannte Position (z. B. Festanschlag)

Der so ermittelte Positionswert wird der NC als Referenzpunktwert bekannt gemacht. Die NC berechnet dann aus der Differenz von Geberabsolutwert und Referenzpunktwert die entsprechende Referenzpunktverschiebung.

Vorgehensweise

1. Überprüfen, ob der Referenziermodus auf "Übernahme des Referenzpunktwertes" eingestellt ist:
MD34200 \$MA_ENC_REFP_MODE == 0
Falls nicht, in das Maschinendatum den Wert 0 eintragen und Power On-Reset auslösen.
2. Die Maschinenachse in der Betriebsart JOG auf die zu messende (z. B. Laserinterferometer) bzw. bekannte Position (z. B. Festanschlag) fahren.

Hinweis

Die Maschinenachse kann über die Verfahrtasten nur in die zum Referenzpunktfahren freigegebene Richtung verfahren werden:

MD34010 \$MA_REFP_CAM_DIR_IS_MINUS (Referenzpunkt anfahren in Minusrichtung)

Um Verfälschungen der Position aufgrund von Lose im Antriebsstrang zu vermeiden, muss das Anfahren der bekannten Position mit geringer Geschwindigkeit erfolgen.

3. Die auf den Maschinennullpunkt bezogene Position der Maschinenachse als Referenzpunktwert in das Maschinendatum eintragen:
MD34100 \$MA_REFP_SET_POS = *Position*
4. Die Geberjustage freigeben:
MD34210 \$MA_ENC_REFP_STATE = 1
5. Zur Übernahme der eingegebenen Maschinendatenwerte Power On-Reset auslösen.
6. Wechsel in die Betriebsart JOG-REF.

7. Betätigen der bereits in Schritt 2 verwendeten, zum Referenzpunktfahren freigegebenen Verfahrtaste.

Beim Betätigen der Verfahrtaste bewegt sich die Maschinenachse nicht!

Die NC berechnet die Referenzpunktverschiebung aus der vom Absolutwertgeber gemeldeten und dem eingegebenen Referenzpunktwert. Das Ergebnis wird eingetragen in das Maschinendatum:

MD34090 \$MA_REFP_MOVE_DIST_CORR (Referenzpunktverschiebung)

Der Status des Absolutgebers wird auf "Geber ist justiert" gesetzt:

MD34210 \$MA_ENC_REFP_STATE = 2

Das Istwertsystem der Maschinenachse wird synchronisiert.

Die Maschinenachse ist jetzt referenziert. Als Kennung setzt die NC, abhängig vom aktiven Messsystem, das entsprechende Nahtstellensignal:

DB31, ... DBB60.4 / 60.5 (Referenziert/Synchronisiert 1/2) = 1

8. POWER ON - Reset auslösen.

Hinweis

Referenzpunktverschiebung dauerhaft aktivieren

Erst nach POWER ON - Reset wird die eingegebene Referenzpunktverschiebung (MD34090) dauerhaft wirksam.

Wird die Maschinenachse nach der Justage des Absolutwertgebers ohne zwischenzeitlichen POWER ON - Reset verfahren, kann die im Maschinendatum eingegebene Referenzpunktverschiebung z. B. im Rahmen interner Überlaufkorrekturen überschrieben werden.

Überprüfen der Istposition

Es wird nach der Justage des Absolutwertgebers empfohlen, nach dem nächsten Hochlauf der Steuerung (POWER ON) die Istposition der Maschinenachse zu verifizieren.

16.9.4 Automatische Justage mittels Messtaster

Funktion

Bei der automatischen Justage mittels Messtaster wird mit der Maschinenachse aus einem Teileprogramm heraus eine bekannte Position in der Maschine angefahren. Der Positionswert ist in der NC als Referenzpunkt看wert hinterlegt. Sobald der Messtaster schaltet, ist die Position erreicht und die NC berechnet aus der Differenz von Geberwert und Referenzpunkt看wert die entsprechende Referenzpunktverschiebung.

Hinweis

Teileprogramm zur automatischen Justage

Das Teileprogramm zur automatischen Justage mittels Messtaster muss vom Maschinenhersteller / Anwender im Hinblick auf die jeweiligen Maschinenspezifika selbst erstellt werden.

Kollisionsfreiheit

Da für die zu referenzierenden Maschinenachsen keine Istwert-bezogenen Überwachungen aktiv sind, ist durch den Maschinenbediener während des Verfahrens der Maschinenachsen besonders auf die Kollisionsfreiheit in der Maschine zu achten!

Teileprogramm

Das Teileprogramm zur automatischen Justage von Absolutwertgebern mittels Messtaster muss für jede zu referenzierende Maschinenachse die nachfolgend aufgeführten Punkte in dieser Reihenfolge realisieren:

1. Anfahren der Justageposition der Maschinenachse, die über das Ansprechen des Messtasters erkannt wird.
Das Anfahren sollte mehrfach aus der gleichen Richtung mit stufenweise reduzierter Geschwindigkeit erfolgen, um eine möglichst hohe Genauigkeit des Messwertes zu erhalten. Der Messwert wird in der Systemvariablen \$AA_IM abgelegt.
2. Berechnen und Schreiben der Referenzpunktverschiebung:

```
MD34090 $MA_REFP_MOVE_DIST_CORR = MD34100 $MA_REFP_SET_POS - $AA_IM
```

3. Setzen des Absolutgeberstatus auf "Geber ist justiert":
MD34210 \$MA_ENC_REFP_STATE = 2

Ablauf

Bei der automatischen Justage mittels Messtaster ist folgendermaßen vorzugehen:

1. Teileprogrammstart auch bei nicht-referenzierten Maschinenachsen freigeben:
MD20700 \$MC_REFP_NC_START_LOCK = 0
2. Für alle relevanten Maschinenachsen die Maschinenachseposition bezogen auf den Maschinennullpunkt bei Schalten des Messtasters als Referenzpunkt看wert eingeben:
MD34100 \$MA_REFP_SET_POS = Referenzpunkt看wert

3. Zur Übernahme der eingegebenen Maschinendatenwerte NCK-Reset auslösen.
4. Teileprogramm starten.
5. Nach Ablauf des Teileprogramms Teileprogrammstart bei nicht-referenzierten Maschinenachsen wieder sperren:
MD20700 \$MC_REFP_NC_START_LOCK = 1
6. POWER ON - Reset auslösen, damit die vom Teileprogramm geschriebene Referenzpunktverschiebung dauerhaft wirksam wird:
MD34090 \$MA_REFP_MOVE_DIST_CORR (Referenzpunktverschiebung)

Hinweis**Referenzpunktverschiebung dauerhaft aktivieren**

Erst nach POWER ON - Reset wird die eingegebene Referenzpunktverschiebung (MD34090) dauerhaft wirksam.

Wird die Maschinenachse nach der Justage des Absolutwertgebers ohne zwischenzeitlichen POWER ON - Reset verfahren, kann die im Maschinendatum eingegebene Referenzpunktverschiebung z. B. im Rahmen interner Überlaufkorrekturen überschrieben werden.

Überprüfen der Istposition

Es wird nach der Justage des Absolutwertgebers empfohlen, nach dem nächsten Hochlauf der Steuerung (POWER ON) die Istposition der Maschinenachse zu verifizieren.

16.9.5 Justage mittels BERO

Funktion

Bei der Justage mittels berührungslosen Näherungsschalters wird, wie bei inkrementellen Messsystemen, ein Referenzpunktfahren auf eine definierte Maschinenposition durchgeführt. Der berührungslose Näherungsschalter ersetzt dabei die bei Absolutwertgebern nicht vorhandene Gebernullmarke. Nach erfolgreichem Abschluss des Referenzpunktfahrens berechnet die NC automatisch die Referenzpunktverschiebung aus der Differenz von Geberabsolutwert und parametrimtem Referenzpunktwert.

Parametrierung**NC: Referenziermodus**

Der Referenziermodus ist auf "Referenzieren von inkrementellen, rotatorischen oder linearen Messsystemen: Nullimpuls auf der Geberspur" einzustellen:

MD34200 \$MA_ENC_REFP_MODE[<Achse>] = 1

NC: Referenzpunktwert

Die Parametrierung des Referenzpunktwertes erfolgt über:

MD34100 \$MA_REFP_SET_POS[<Achse>] = <Referenzpunktwert>

Antrieb: Nullmarkenersatz

Im Paramter p0494 oder p0495 ist der digitale Eingang der NCU-Schnittstelle einzustellen, an dem der berührungslose Näherungsschalter angeschlossen ist.

Durchführung

Das Referenzpunktfahren kann in der Betriebsart JOG-REF manuell oder in den Betriebsarten AUTOMATIK oder MDA über ein Teileprogramm (G74) gestartet werden.

Nach erfolgreichem Abschluss des Referenzpunktfahrens ist der Absolutwertgeber justiert und das Istwertsystem der Maschinenachse synchronisiert.

Als Rückmeldung für das PLC-Anwenderprogramm setzt die NC, abhängig vom aktiven Messsystem, für die Maschinenachse das NC/PLC-Nahtstellensignal:

DB31, ... DBB60.4 / 60.5 (Referenziert/Synchronisiert 1 / 2) = 1

Hinweis

Wird nach der Justage des Absolutwertgebers der berührungslose Näherungsschalter entfernt, muss der Referenziermodus auf "Referenzieren mit Absolutwertgeber" umparametriert werden:

MD34200 \$MA_ENC_REFP_MODE[<Achse>] = 0

16.9.6 Referenzpunktfahren bei Absolutwertgebern**Parametrierung****Verfahrbewegungsfreigabe**

Wird für eine Maschinenachse mit justiertem Absolutwertgeber als aktivem Messsystem Referenzpunktfahren ausgelöst (manuell in der Betriebsart JOG-REF oder automatisch per Teileprogrammanweisung G74), wird die Maschinenachse abhängig von der parametrierten Verfahrbewegungsfreigabe verfahren:

MD34330 \$MA_REFP_STOP_AT_ABS_MARKER = <Wert>

Wert	Bedeutung
0	Verfahrbewegung ist freigegeben . Nach dem Auslösen des Referenzpunktfahrens fährt die Maschinenachse auf die Referenzpunktposition. Mit Erreichen der Referenzpunktposition ist das Referenzpunktfahren abgeschlossen.
1	Verfahrbewegung ist nicht freigegeben. Nach dem Auslösen des Referenzpunktfahrens fährt die Maschinenachse nicht und das Referenzpunktfahren ist sofort abgeschlossen.

16.9.7 Referenzpunktfahren bei rotatorischen Absolutwertgebern mit Ersatznullmarke

Funktion

Damit das Referenzpunktfahren mittels Nullmarke (siehe Kapitel "Referenzieren bei inkrementellen Messsystemen (Seite 1249)") auch bei Absolutwertgebern verwenden kann, wird die bei Absolutwertgebern die hardware-mäßig nicht vorhandene Nullmarke simuliert. Dazu wird von der Steuerung einmal pro Geberumdrehung, immer an der gleichen Position innerhalb der Umdrehung, das Signal für die Ersatznullmarke erzeugt.

Unterschiede zum Referenzieren mit inkrementellen Gebern

Ein Absolutwertgeber mit Ersatznullmarke ist nicht als vollständiger Ersatz für einen inkrementellen Geber zu sehen. Alle Eigenschaften des Absolutwertgebers werden beibehalten. In der folgenden Tabelle sind die unterschiedlichen Eigenschaften von inkrementellen und absoluten Gebern aufgelistet:

Tabelle 16-1 Eigenschaften von inkrementellen und absoluten Gebern

Eigenschaft	Inkrementeller Geber	Absolutgeber
Gebertyp	MD30240 \$MA_ENC_TYPE =	
	1	4
Interne Geberposition	MD30250 \$MA_ACT_POS_ABS =	
	Wert wird nur bei MD34210 ≥ 1 aktualisiert	Wert wird nur bei MD30270 = 0 aktualisiert
Verfahrenbereichserweiterung	MD30270 \$MA_ENC_ABS_BUFFERING =	
	Keine Auswirkung	= 0 (Standard): aktiv
Referenzpunktverschiebung	MD34090 \$MA_REFP_MOVE_DIST_CORR =	
	Werteingabe zulässig	Wert wird ausschließlich durch Steuerung aktualisiert
Unterstützte Referenzierarten	MD34200 \$MA_ENC_REFP_MODE =	
	1, 3, 4, 8	0, 1
Justagestatus	MD34210 \$MA_ENC_REFP_STATE = 0, 1, 2	
	Automatische Dejustage des Gebers beim Ausschalten in der Bewegung.	Automatische Dejustage des Gebers beim Parametersatzwechsel mit Positionssprung bzw. bei Seriennummeränderung.
Modulo-Bereich Absolutposition	MD34220 \$MA_ENC_ABS_TURNS_MODULO =	
	0	1 - 4096
Geberseriennummer	MD34230 \$MA_ENC_SERIAL_NUMBER =	
	0	Wert muss bei jedem Geberwechsel von der PLC aus aktualisiert werden, sonst Verlust der Justage plus Alarm.
Übernahme von Serien-IBN Files	Uneingeschränkt möglich.	Wegen Gebereigenschaften (MD30250, MD30270, MD34090, MD34210, MD34220, MD34230) nur eingeschränkt möglich.
Einschaltzeit	0 Sekunden	mehrere Sekunden
Nullmarke	1 pro Geberumdrehung	keine
Nullmarkenüberwachung	Hardware	Software

Eigenschaft	Inkrementeller Geber	Absolutgeber
Position nach POWER ON ohne Istwert-Pufferung	0.0	Letzte Position innerhalb von MD34220.
	MD34210 = 0	MD30270 = 1
Position nach POWER ON mit Istwert-Pufferung	Letzte Stillstandsposition vor dem Ausschalten.	Letzte Position inklusive kleiner Bewegungen während POWER OFF.
	MD34210 = 1	MD30270 = 0
Referenziert nach POWER ON	abhängig vom Justagestatus	

Voraussetzung

Die Funktion ist nur mit rotatorischen Absolutgebern einsetzbar:

- MD31000 \$MA_ENC_IS_LINEAR == 0
- MD30240 \$MA_ENC_TYPE == 4

Parametrierung

Referenzpunktfahren mit Ersatznullmarke:

MD34200 \$MA_ENC_REFP_MODE = 1

Randbedingungen

- Eine Referenzpunktverschiebung (MD34090 \$MA_REFP_MOVE_DIST_CORR) darf nicht parametrierung werden:
Dieses MD beschreibt in Verbindung mit Absolutwertgebern die Verschiebung zwischen Maschinen- und Absolutwertgebernnullpunkt und hat damit eine andere Bedeutung.
- Die lastseitige Nullmarkensuchgeschwindigkeit (MD34040 \$MA_REFP_VELO_SEARCH_MARKER) sollte die Grenzfrequenz der Absolutspur des Gebers (MD36302 \$MA_ENC_FREQ_LIMIT_LOW) nicht überschreiten. Bei zu großer Geschwindigkeit kann keine Absolutinformation mehr gelesen und damit keine Ersatznullmarke generiert werden.
- Wird keine Nullmarke innerhalb des parametrisierten Weges (MD34060 \$MA_REFP_MAX_MARKER_DIST) gefunden, wird ein Alarm angezeigt:
- Soll der Absolutwertgeber außer der letzten Position auch den Referenziert-Status über POWER OFF beibehalten, muss folgendes MD gesetzt werden:
MD34210 \$MA_ENC_REFP_STATE = 2

Datensicherung und Serieninbetriebnahme

Einige Eigenschaften eines Absolutwertgebers schränken die Übertragbarkeit eines Inbetriebnahmearchivs auf andere Maschinen ein. Folgenden Maschinendaten müssen nach dem Einspielen eines Inbetriebnahmearchivs in die Steuerung kontrolliert und eventuell korrigiert werden:

- MD30250 \$MA_ACT_POS_ABS (Interne Geberposition)
- MD30270 \$MA_ENC_ABS_BUFFERING (Verfahrbereichserweiterung)
- MD34090 \$MA_REFP_MOVE_DIST_CORR (Absolutverschiebung)

- MD34210 \$MA_ENC_REFP_STATE (Justagestatus)
- MD34220 \$MA_ENC_ABS_TURNS_MODULO (Modulo-Bereich)
- MD34230 \$MA_ENC_SERIAL_NUMBER (Geber-Seriennummer)

16.9.8 Einschalten des Messsystems

In folgenden Situationen wird das Messsystem einer Maschinenachse eingeschaltet:

- Hochlauf der Steuerung (POWER ON)
- Einschalten des Messsystems per Nahtstellensignal (Abwahl von "Parken"):
DB31, ... DBB1.5 / 1.6 (Lagemesssystem 1/2)
DB31, ... DBB2.1 (Reglerfreigabe)
- Unterschreiten der parametrisierten Gebergrenzfrequenz (Spindeln):
MD36300 \$MA_ENC_FREQ_LIMIT

Beim Einschalten des Messsystems synchronisiert die NC das Istwertsystem der Maschinenachse auf den aktuellen Absolutwert. Verfahrbewegungen sind bei Achsen während der Synchronisation gesperrt, bei Spindeln nicht.

Parametrieren der Gebergrenzfrequenz (Spindeln)

Der Absolutwertgeber der Fa. Heidenhain EQN 1325 besitzt eine Inkremental- und eine Absolutspur.

Wird eine Spindel mit Drehzahlen oberhalb der Gebergrenzfrequenz der Inkrementalspur betrieben, muss als Gebergrenzfrequenz die wesentlich kleinere Grenzfrequenz der Absolutspur parametrieren werden:

MD36300 \$MA_ENC_FREQ_LIMIT

Ansonsten würde beim Einschalten des Messsystems durch Unterschreiten der parametrisierten Gebergrenzfrequenz eine falsche Absolutposition gelesen werden. Dies hätte einen Positionsversatz im Istwertsystem der Maschinenachse zur Folge.

Bestimmen der Gebergrenzfrequenz

Die zu parametrierende Gebergrenzfrequenz ergibt sich aus der kleineren der beiden folgenden Grenzdrehzahlen:

- Geber
Die Grenzdrehzahl bzw. die Gebergrenzfrequenz ist dem Datenblatt des Gebers zu entnehmen (z. B. Grenzdrehzahl = 2000 [1/min])
- NC
Bedingt durch das NC-interne Auswerteverfahren ergibt sich die Grenzdrehzahl, bis zu der eine fehlerfreie Ermittlung des Absolutwertes durch die NC möglich ist, zu 4 Geber-Umdrehungen pro Interpolatortakt.
Bei einem Interpolatortakt von z. B. 12 ms: Grenzdrehzahl = 4 / 12 ms = 20000 1/min
Die der Grenzdrehzahl entsprechende Grenzfrequenz berechnet sich zu:

$$MD36300 = \frac{4 * MD31020}{MD10050 * MD10070}$$

MD31020 \$MA_ENC_RESOL (Geberstriche pro Umdrehung)
MD10050 \$MN_SYSCLOCK_CYCLE_TIME (Systemgrundtakt)
MD10070 \$MN_IPO_SYSCLOCK_TIME_RATIO (Faktor für Interpolatortakt)

Hinweis

Die für Spindeln relevante Lageregeleinschalt-drehzahl ist entsprechend der Gebergrenzfrequenz des Absolutwertgebers der Spindel einzustellen:

MD35300 \$MA_SPIND_POSCTRL_VELO (Lageregeleinschalt-drehzahl)

MD36300 \$MA_ENC_FREQ_LIMIT (Gebergrenzfrequenz)

16.9.9 Nicht unterstützte Referenziervarianten

Folgende Referenziervarianten werden im Zusammenhang mit Absolutwertgebern nicht unterstützt:

- Referenzieren / Justieren mittels Gebernullmarke
- Abstandscodierte Referenzmarken
- berührungsloser Näherungsschalter mit Zweiflankenauswertung

16.10 Automatisches Wiederherstellen des Maschinenbezugs

Ohne definierten Maschinenbezug können beim Verfahren von Maschinenachsen keine positionsabhängigen Funktionen wie Transformationen oder Toolframe, ausgeführt werden. In verschiedenen Maschinsituationen müssen diese Funktionen sofort mit der Geberaktivierung, z.B. nach dem Einschalten der Steuerung oder nach Beenden von "Parken einer Maschinenachse (Seite 122)", zum Verfahren von Achsen zur Verfügung stehen. Dabei sollen bzw. können die Maschinenachsen aber nicht für ein erneutes Referenzieren verfahren werden.

Absolutwertgeber

Bei Messsystemen mit justierten Absolutwertgebern wird der Maschinenbezug ohne zusätzliche Maßnahmen sofort mit dem Lesen des Geberwertes wiederhergestellt.

Inkrementalgeber

Bei inkrementellen Messsystemen kann das Wiederherstellen des Maschinenbezugs ohne Verfahren der Achsen durch "Automatisches Referenzieren" oder "Restaurieren der Istposition" wiederhergestellt werden.

Randbedingungen

WARNUNG

Fehlsynchronisation des Lagemesssystems durch Versatz der Maschinenachsen-Istposition

Während der Zeit, in der das Messsystem der Maschinenachse abgeschaltet ist, darf diese mechanisch **nicht** bewegt werden. Andernfalls kommt es zu einem Versatz zwischen der letzten gepufferten Istposition und der tatsächlichen Istposition der Maschinenachse. Dies würde zu einer Fehlsynchronisation des Messsystems mit Gefahr für Mensch und Maschine führen.

Die Unveränderlichkeit der Istposition ist vom Maschinenhersteller durch entsprechende Maßnahmen an der Maschine, z. B. Haltebremsen, zu unterstützen und vom Anwender sicherzustellen. Die Verantwortung dafür liegt ausschließlich beim Maschinenhersteller/Anwender.

Kann das Bewegen der Achse im abgeschalteten Zustand maschinenbaulich nicht verhindert werden, ist entweder ein Absolutgeber zu verwenden oder die Achse muss nach dem Einschalten per Referenzpunktfahren neu referenziert werden.

Hinweis

Sensor Module SMExx

Automatisches Referenzieren bzw. Restaurieren der Istposition nach dem Wiedereinschalten der Steuerung auf die letzte gepufferte Position, ist nur in Verbindung mit Sensor Modules SMExx (Externally Mounted) möglich. Bei Verwendung von Sensor Modules SMCxx (Cabinet) oder SMIxx (Integrated) kann die Istposition nach dem Wiedereinschalten der Steuerung (Power On) **nicht** wiederhergestellt werden. Das Messsystem der Maschinenachse muss neu referenziert werden.

16.10.1 Automatisches Referenzieren

Funktion

Beim automatischen Referenzieren wird nach dem Einschalten der Steuerung die Istposition des aktiven Messsystem der Maschinenachse auf die letzte gepufferte Position und als Geberstatus "referenziert" gesetzt. Dadurch ist das Starten von Programmen in den Betriebsarten AUTOMATIK und MDA direkt nach dem Hochlauf der Steuerung möglich.

Voraussetzungen

- Das nach dem Einschalten der Steuerung aktive Messsystemen, muss vor dem Ausschalten bereits erstmalig referenziert worden sein.
- Zum Zeitpunkt das Ausschaltens der Steuerung muss sich die Maschinenachse im Stillstand mit "Genauhalt fein" (DB31, ... DBX60.7 == 1) befinden.

Hinweis

Beindet sich die Maschinenachse zum Zeitpunkt das Ausschaltens **nicht** im Stillstand mit "Genauhalt fein", wird nach dem Einschalten die Istposition mit dem Wert "0" initialisiert. Als Geberstatus wird "nicht referenziert" angezeigt.

Parametrierung

Die Freigabe des Automatischen Referenzierens erfolgt durch Einstellen des Geberstatus auf "Automatisches Referenzieren ist freigegeben, aber der Geber ist noch nicht referenziert":

MD34210 \$MA_ENC_REFP_STATE[<Geber>] = 1

Nach dem Referenzieren des Messsystem wird im Geberstatus angezeigt, dass das Automatische Referenzieren bei der nächsten Geberaktivierung ausgeführt wird:

MD34210 \$MA_ENC_REFP_STATE[<Geber>] == 2

NC/PLC-Nahtstellensignale

Nach dem Automatischen Referenzieren wird für das aktive Messsystem der Geberstatus "referenziert" angezeigt:

DB31, ... DBX60.4 / .5 == 1 (Referenziert/Synchronisiert 1 / 2)

Randbedingungen

Geberaktivierung bei MD34210 \$MA_ENC_REFP_STATE[<Geber>] == 1

Ein Geberstatus zum Zeitpunkt der Geberaktivierung gleich "1" bedeutet, dass "Automatische Referenzieren" freigegeben ist. Das Messsystem wurde aber entweder noch nicht referenziert

oder es wurde nicht bei Stillstand der Maschinenachse im Zustand "Genauhalt fein" abgeschaltet. Für die Maschinenachse bzw. das aktive Messsystem wird folgendes gesetzt:

- Istposition = 0
- Aktives Messsystem, Geberstatus = "Nicht referenziert":
DB31, ... DBX60.4 / .5 = **0** (Referenziert/Synchronisiert 1 / 2)

Literatur

Funktionsbeschreibung Grundfunktionen, Kapitel "R1 Referenzieren" > "Referenzieren bei inkrementellen Messsystemen (Seite 1249)"

16.10.2 Restaurieren der Istposition

Funktion

Beim Restaurieren der Istposition auf die letzte gepufferte Position, wird der Geberstatus des aktiven Messsystems auf "**restauriert**" gesetzt. Die Achse kann nur manuell verfahren werden.

Betriebsart AUTOMATIK

Zur Freigabe von NC-Start zum automatischen Abarbeiten von Programmen in der Betriebsart AUTOMATIK, muss das Messsystem der Maschinenachse neu referenziert werden.

Betriebsart MDA und Überspeichern

In der Betriebsart MDA und für das Überspeichern, kann die Bearbeitung auch ohne Referenzieren der Achsen, mit restaurierten Positionen erfolgen. Dazu muss kanalspezifisch NC-Start mit restaurierten Positionen explizit freigegeben werden:

MD20700 \$MC_REFP_NC_START_LOCK = 2

Voraussetzung

Das nach dem Einschalten der Steuerung aktive Messsystem muss vor dem Ausschalten bereits erstmalig referenziert worden sein.

Parametrierung

Freigabe: Restaurieren der Istposition

Die Freigabe für das Restaurieren der Istposition erfolgt durch Einstellen des Geberstatus auf "Die letzte vor dem Ausschalten gepufferte Achsposition wird restauriert, kein automatisches Referenzieren":

MD34210 \$MA_ENC_REFP_STATE[<Geber>] = 3

Freigabe: NC-START für Betriebsarten "MDA" und "Überspeichern"

Die Freigabe von NC-START zum Abarbeiten von Teileprogrammen oder Teileprogrammsätzen in den Betriebsarten "MDA" und "Überspeichern", mit Status "Position restauriert" erfolgt über:

MD34110 \$MA_REFP_CYCLE_NR \neq -1 (Achsenreihenfolge beim kanalspez. Referenzieren)

MD20700 \$MC_REFP_NC_START_LOCK = 2 (NC-START-Sperre ohne Referenzpunkt)

NC/PLC-Nahtstellensignale

Die restaurierte Istposition wird als qualitativ nicht gleichwertig mit einer Istposition nach Referenzpunktfahren angesehen. Daher wird für das Messsystem der Maschinenachse der Status "Position restauriert" und nicht "Referenziert/Synchronisiert" angezeigt.

Istposition restauriert:

- DB31, ... DBX60.4 / .5 = 0 (Referenziert/Synchronisiert 1 / 2)
- DB31, ... DBX71.4 / .5 = 1 (**Position restauriert**, Geber 1 / 2)

Messsystem referenziert:

- DB31, ... DBX60.4 / .5 = 0 \rightarrow 1 (**Referenziert/Synchronisiert** 1 / 2)
- DB31, ... DBX71.4 / .5 = 1 \rightarrow 0 (Position restauriert, Geber 1 / 2)

Hinweis

Im Zustand "Position restauriert" ist die Überwachung der Verfahrbereichsgrenzen (Software-Endschalter, Arbeitsfeldebegrenzung, etc.) bereits aktiv.

Randbedingungen

Spindeln

Bei Überschreitung der Gebergrenzfrequenz wird eine Spindeln auf den Zustände "Nicht Referenziert/Synchronisiert" zurückgesetzt:

- DB31, ... DBX60.4 / .5 = 1 \rightarrow 0 (Referenziert/Synchronisiert 1 / 2)
- DB31, ... DBX71.4 / .5 = 1 \rightarrow 0 (Position restauriert, Geber 1 / 2)

Literatur

Funktionsbeschreibung Grundfunktionen, Kapitel "R1 Referenzieren" > "Referenzieren bei inkrementellen Messsystemen (Seite 1249)"

16.11 Randbedingungen

16.11.1 Große Verfahrbereiche

Linearachsen mit einem Verfahrbereich > 4096 Geberumdrehungen, rotatorischem Absolutwertgeber EQN 1325 und einem parametrisierten Absolutwertgeber-Bereich von MD34220 \$MA_ENC_ABS_TURNS_MODULO = 4096:

Der maximale mögliche Verfahrbereich entspricht dem von inkrementellen Gebern.

Endlos drehende Rundachsen mit Absolutwertgeber:

- Es sind beliebig ganzzahlige Übersetzungsverhältnisse zulässig.
- Es wird empfohlen, endlos drehende Rundachsen mit Absolutwertgeber als Modulo-Rundachsen (Verfahrbereich 0...360 Grad) zu parametrisieren:
MD30310 \$MA_ROT_IS_MODULO = 1
Ansonsten kann es dazu kommen, dass die Rundachse nach dem Einschalten des Messsystems einen sehr großen Verfahrweg benötigt, um auf absolut Null zu fahren.

Maschinenachsen mit Absolutwertgeber:

Damit die Steuerung nach dem Wiedereinschalten des Messsystems die aktuelle Istposition korrekt ermittelt kann, darf bei ausgeschaltetem Messsystem die Maschinenachse nur um weniger als den halben Absolutwertgeber-Bereich bewegt werden:

MD34220 \$MA_ENC_ABS_TURNS_MODULO

Hinweise zur Eindeutigkeit von Geberpositionen

Hinweis

Lineare Absolutwertgeber

Der Absolutwert von Linearmaßstäben, z. B. Fa. Heidenhain LC181, ist im Rahmen der lieferbaren Maßstabslängen immer eindeutig.

Rotatorische Absolutwertgeber

Der Absolutwert von rotatorischen Absolutwertgebern ist nur im Bereich der spezifischen maximalen Geberumdrehungen eindeutig.

Der rotatorische Absolutwertgeber der Fa. Heidenhain EQN 1325 liefert z. B. einen eindeutigen Absolutwert im Bereich von 0 bis 4096 Geberumdrehungen.

Je nach Anbauverhältnissen ergeben sich daraus:

- Rundachse mit Geber an der Last: 4096 Last-Umdrehungen
- Rundachse mit Geber am Motor: 4096 Motor-Umdrehungen
- Linearachse mit Geber am Motor: 4096 Motor-Umdrehungen

Beispiel:

Ein rotatorischer Absolutwertgeber EQN 1325 ist am Motor einer Linearachse angebracht. Bei einer effektiven Spindelsteigung von 10 mm ergibt sich daraus ein eindeutiger Absolutwert im Verfahrbereich von -20,48 m bis +20,48 m.

16.12 Datenlisten

16.12.1 Maschinendaten

16.12.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
11300	JOG_INC_MODE_LEVELTRIGGRD	INC/REF im Tippbetrieb/Dauerbetrieb

16.12.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20700	REFP_NC_START_LOCK	NC-Startsperre ohne Referenzpunkt

16.12.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30200	NUM_ENCS	Anzahl der Geber
30240	ENC_TYP	Istwert Gebertyp
30242	ENC_IS_INDEPENDENT	Geber ist unabhängig
30250	ACT_POS_ABS	Absolute Geberposition zum Ausschaltzeitpunkt
30270	ENC_ABS_BUFFERING	Absolutgeber: Verfahrbereichserweiterung
30300	IS_ROT_AX	Rundachse / Spindel
30310	ROT_IS_MODULO	Modulo-Wandlung für Rundachse / Spindel
30330	MODULO_RANGE	Größe des Modulo-Bereichs
30340	MODULO_RANGE_START	Startposition Modulo-Bereich
30355	MISC_FUNCTION_MASK	Achsfunktionen
31122	BERO_DELAY_TIME_PLUS	BERO-Verzögerungszeit in Plus-Richtung
31123	BERO_DELAY_TIME_MINUS	BERO-Verzögerungszeit in Minus-Richtung
34000	REFP_CAM_IS_ACTIVE	Achse mit Referenznocken
34010	REFP_CAM_DIR_IS_MINUS	Referenzpunktanfahren in Minusrichtung
34020	REFP_VELO_SEARCH_CAM	Referenzpunktanfahrsgeschwindigkeit
34030	REFP_MAX_CAM_DIST	max. Wegstrecke zum Referenznocken
34040	REFP_VELO_SEARCH_MARKER	Referenzpunktabschaltgeschwindigkeit
34050	REFP_SEARCH_MARKER_REVERSE	Richtungsumkehr auf Referenznocken
34060	REFP_MAX_MARKER_DIST	max. Wegstrecke zur Referenzmarke; max. Wegstrecke zu 2 Referenzmarken bei abstands- codierten Maßstäben
34070	REFP_VELO_POS	Referenzpunkteinfahrtgeschwindigkeit

Nummer	Bezeichner: \$MA_	Beschreibung
34080	REFP_MOVE_DIST	Referenzpunktabstand/Zielpunkt bei abstandscodiertem System
34090	REFP_MOVE_DIST_CORR	Referenzpunkt-/Absolutverschiebung abstandscodiert
34092	REFP_CAM_SHIFT	Elektronische Referenznockenverschiebung für inkrementelle Messsysteme mit äquidistanten Nullmarken.
34093	REFP_CAM_MARKER_DIST	Abstand Referenznocken/Referenzmarke
34100	REFP_SET_POS	Referenzpunktwert
34102	REFP_SYNC_ENCS	Istwertabgleich auf das referenzierende Messsystem
34104	REFP_PERMITTED_IN_FOLLOWUP	Freigabe Referenzieren im Nachführbetrieb
34110	REFP_CYCLE_NR	Achsreihenfolge beim kanalspez. Referenzieren
34120	REFP_BERO_LOW_ACTIVE	Polaritätswechsel des BERO Nockens
34200	ENC_REFP_MODE	Referenzier-Modus
34210	ENC_REFP_STATE	Status Absolutwertgeber
34220	ENC_ABS_TURNS_MODULO	Bereich Absolutwertgeber bei rotatorischen Gebern
34230	ENC_SERIAL_NUMBER	Geber-Seriennummer
34232	EVERY_ENC_SERIAL_NUMBER	Reichweite der Geber-Seriennummer
34300	ENC_REFP_MARKER_DIST	Grundabstand der Referenzmarken bei abstandscodierten Gebern
34310	ENC_MARKER_INC	Differenzabstand zweier Referenzmarken bei abstandscodierten Maßstäben
34320	ENC_INVERS	Längenmesssystem gegensinnig zum Maschinensystem
34330	REFP_STOP_AT_ABS_MARKER	Abstandscodiertes Längenmesssystem ohne Zielpunkt
35150	SPIND_DES_VELO_TOL	Spindeldrehzahltoleranz
36302	ENC_FREQ_LIMIT_LOW	Gebergrenzfrequenz Neusynchronisation
36310	ENC_ZERO_MONITORING	Nullmarkenüberwachung

16.12.2 Signale

16.12.2.1 Signale an BAG

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
BAG-Reset	DB11.DBX0.7	DB3000.DBX0.7
Maschinenfunktion REF	DB11.DBX1.2	DB3000.DBX1.2

16.12.2.2 Signale von BAG

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Aktive Maschinenfunktion REF	DB11.DBX5.2	DB3100.DBX1.2

16.12.2.3 Signale an Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Referenzieren aktivieren	DB21,DBX1.0	DB320x.DBX1.0
OEM Kanal-Signal: (HMI → PLC) REF	DB21,DBX28.7	-

16.12.2.4 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Referenzieren aktiv	DB21,DBX33.0	DB330x.DBX1.0
Reset	DB21,DBX35.7	DB330x.DBX3.7
alle referenzpunktpflichtigen Achsen sind referenziert	DB21,DBX36.2	DB330x.DBX4.2

16.12.2.5 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Nachführbetrieb (anfordern)	DB31,DBX1.4	DB380x.DBX1.4
Lagemesssystem 1 / Lagemesssystem 2	DB31,DBX1.5-6	DB380x.DBX1.5-6
Referenzpunktwert 1 bis 4	DB31,DBX2.4-7	DB380x.DBX2.4-7
Verfahrtasten minus / plus	DB31,DBX4.6-7	DB380x.DBX4.6-7
Verzögerung Referenzpunktfahren	DB31,DBX12.7	DB380x.DBX1000.7

16.12.2.6 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Referenziert, Synchronisiert 1 / 2	DB31,DBX60.4-5	DB390x.DBX0.4-5
Nachführen aktiv	DB31,DBX61.3	DB390x.DBX1.3
Fahrbefehl minus / plus	DB31,DBX64.6-7	DB390x.DBX4.6-7
Restauriert 1 / 2	DB31,DBX71.4-5	DB390x.DBX11.4-5

S1: Spindeln

17.1 Kurzbeschreibung

Spindeln dienen primär dazu, ein Werkzeug oder ein Werkstück in eine Drehbewegung zu versetzen, um dadurch die Voraussetzungen für eine spanabhebende Bearbeitung zu schaffen.

Dazu muss die Spindel, je nach Typ der Maschine, folgende Funktionen unterstützen:

- Vorgabe einer Spindeldrehrichtung (M3, M4)
- Vorgabe einer Spindeldrehzahl (S, SVC)
- Spindel Halt ohne Orientierung (M5)
- Spindel Halt mit Orientierung / Spindelpositionierung (SPOS, M19 und SPOSA)
- Getriebestufenumschaltung (M40 bis M45)
- Spindel-Achsfunktionalität (Spindel wird zur Rundachse und umgekehrt)
- Gewindeschneiden (G33, G34, G35)
- Gewindebohren ohne Ausgleichsfutter (G331, G332)
- Gewindebohren mit Ausgleichsfutter (G63)
- Umdrehungsvorschub (G95)
- konstante Schnittgeschwindigkeit (G96, G961, G97, G971)
- programmierbare Spindeldrehzahlbegrenzungen (G25, G26, LIMS)
- Lagemessgeber an der Spindel oder am Spindelmotor montierbar
- Spindelüberwachungen auf min. und max. Drehzahl sowie max. Gebergrenzfrequenz und Zielpunktüberwachung der Spindel
- Ein-, Ausschalten der Lageregelung (SPCON, SPCOF, M70)
- Programmierung von Spindelfunktionen:
 - aus den Teileprogramm
 - über Synchronaktionen
 - über PLC mit FC18 oder über spezielle Spindelschnittstelle für einfache Spindelansteuerungen

17.2 Betriebsarten

17.2.1 Übersicht

Eine Spindel ist eine endlos drehende Rundachse, die in verschiedenen Betriebsarten in Lageregelung oder Drehzahlregelung betrieben werden kann.

Betriebsarten

- Steuerbetrieb (Drehzahlregelung)
- Pendelbetrieb (Drehzahlregelung)
- Positionierbetrieb (Lageregelung)
- Synchronbetrieb

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Synchronspindel (S3)

- Gewindebohren ohne Ausgleichsfutter (G331 / G332)

Literatur:

Programmierhandbuch Grundlagen; Kapitel: "Wegbefehle" > "Gewindebohren" > "Gewindebohren ohne Ausgleichsfutter (G331 / G332)"

- Achsbetrieb (Lageregelung)

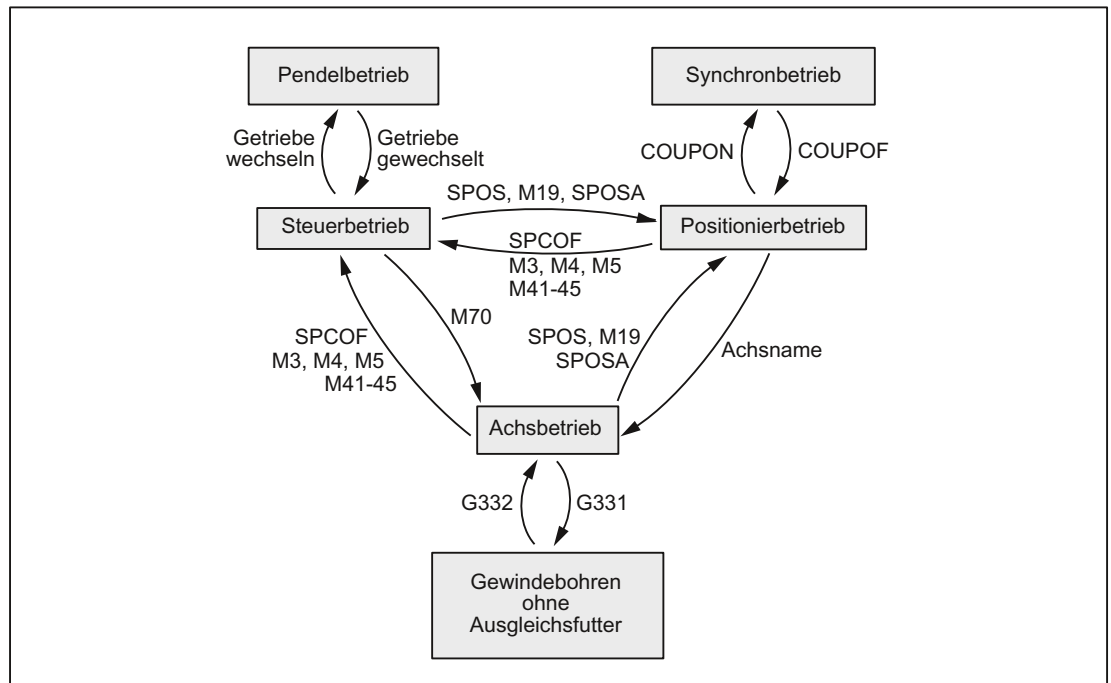
Hinweis

Umschaltung Achs / Spindelbetrieb

Eine Spindel kann zwischen Achs- und Spindelbetrieb umgeschaltet werden, wenn für den Achs- und Spindelbetrieb ein gemeinsamer Motor verwendet wird.

17.2.2 Betriebsartenwechsel

Zwischen den Spindelbetriebsarten und dem Achsbetrieb kann wie folgt gewechselt werden:



- Steuerbetrieb → Pendelbetrieb**
 Die Spindel wechselt in den Pendelbetrieb, wenn durch die automatische Getriebestufenauswahl (M40) in Verbindung mit einem neuen S-Wert oder durch M41 bis M45 eine neue Getriebestufe vorgegeben wurde. Die Spindel wechselt nur dann in den Pendelbetrieb, wenn die neue Getriebestufe ungleich der aktuellen Istgetriebestufe ist.
- Pendelbetrieb → Steuerbetrieb**
 Ist die neue Getriebestufe eingelegt, wird das Nahtstellensignal: DB31, ... DBX84.6 (Pendelbetrieb) rückgesetzt und mit dem Nahtstellensignal: DB31, ... DBX16.3 (Getriebe ist umgeschaltet) in den Steuerbetrieb gewechselt. Die letzte programmierte Spindeldrehzahl (S-Wert) ist wieder wirksam.
- Steuerbetrieb → Positionierbetrieb**
 Soll die Spindel aus der Drehung (M3 oder M4) mit Orientierung angehalten oder aus dem Stillstand (M5) neu orientiert werden, wird mit SPOS, M19 oder SPOSA in den Positionierbetrieb gewechselt.
- Positionierbetrieb → Steuerbetrieb**
 Soll die Orientierung der Spindel beendet werden, wird mit M3, M4 oder M5 in den Steuerbetrieb gewechselt. Die letzte programmierte Spindeldrehzahl (S-Wert) ist wieder wirksam.
- Positionierbetrieb → Pendelbetrieb**
 Soll die Orientierung der Spindel beendet werden, kann mit M41 bis M45 in den Pendelbetrieb gewechselt werden. Ist der Getriebestufenwechsel beendet, wird die letzte programmierte Spindeldrehzahl (S-Wert) und M5 (Steuerbetrieb) wieder wirksam.

- Positionierbetrieb → Achsbetrieb
Wurde eine Spindel mit Orientierung angehalten, wird durch die Programmierung des zugeordneten Achsnamens in den Achsbetrieb gewechselt. Die Getriebestufe bleibt erhalten.
- Steuerbetrieb → Achsbetrieb
Soll aus dem Steuerbetrieb in den Achsbetrieb gewechselt werden, so kann dies auch durch Programmieren von M70 erfolgen. Dabei wird eine drehende Spindel wie bei M5 abgebremst, die Lageregelung eingeschaltet und der Parametersatz Null angewählt.
- Achsbetrieb → Steuerbetrieb
Soll der Achsbetrieb beendet werden, kann mit M3, M4 oder M5 in den Steuerbetrieb gewechselt werden. Die letzte programmierte Spindeldrehzahl (S-Wert) ist wieder wirksam.
- Achsbetrieb → Pendelbetrieb
Soll der Achsbetrieb beendet werden, kann mit M41 bis M45 in den Pendelbetrieb gewechselt werden (nur wenn die programmierte Getriebestufe ungleich der Istgetriebestufe ist). Ist der Getriebestufenwechsel beendet, wird die letzte programmierte Spindeldrehzahl (S-Wert) und M5 (Steuerbetrieb) wieder wirksam.

17.2.3 Steuerbetrieb

Wann Steuerbetrieb?

Bei folgenden Funktionen befindet sich die Spindel im Steuerbetrieb:

- Konstante Spindeldrehzahl:
 - S... M3/M4/M5 und G93, G94, G95, G97, G971
 - S... M3/M4/M5 und G33, G34, G35
 - S... M3/M4/M5 und G63
- Konstante Schnittgeschwindigkeit:
 - G96/G961 S... M3/M4/M5

Die Spindel muss nicht synchronisiert sein.

Voraussetzungen

Ein Spindellageistwertgeber ist zwingend erforderlich für M3/M4/M5 in Verbindung mit:

- Umdrehungsvorschub (G95)
- konstante Schnittgeschwindigkeit (G96, G961, G97, G971)
- Gewindeschneiden (G33, G34, G35)
- Gewindebohren ohne Ausgleichsfutter (G331, G332)
- Lageregelung einschalten (SPCON, M70)

Ein Spindellageistwertgeber ist **nicht** erforderlich für M3/M4/M5 in Verbindung mit:

- zeitreziproker Vorschubverschlüsselung (G93)
- Vorschubgeschwindigkeit in mm/min bzw. inch/min (G94)
- Gewindebohren mit Ausgleichsfutter (G63)

Drehzahlsteuerbetrieb

Der Drehzahlsteuerbetrieb ist besonders dann geeignet, wenn eine konstante Spindeldrehzahl verlangt wird, die Position der Spindel aber ohne Bedeutung ist (z. B. konstante Fräserdrehzahl zur Erzeugung eines gleichmäßigen Bildes der Werkstückoberfläche).

- Der Drehzahlsteuerbetrieb wird im Teileprogramm mit M3, M4, M5 oder mit SPCOF eingeschaltet.
- Das folgende NC/PLC-Nahtstellensignal ist gesetzt:
DB31, ... DBX84.7 (Steuerbetrieb)
- Das NC/PLC-Nahtstellensignal:
DB31, ... DBX61.5 (Lageregler aktiv)
ist rückgesetzt, wenn ohne Lageregelung gearbeitet wird.
- Die Beschleunigung im Drehzahlsteuerbetrieb wird getriebestufenabhängig festgelegt im Maschinendatum:
MD35200 \$MA_GEAR_STEP_SPEEDCTRL_ACCEL
Der Wert sollte den physikalischen Gegebenheiten möglichst entsprechen.

Lageregelbetrieb

Der Lageregelbetrieb ist besonders dann geeignet, wenn die Position der Spindel über einen längeren Zeitraum genau bekannt sein muss oder wenn Synchronspindelsollwertkopplung eingeschaltet werden soll.

- Der Lageregelbetrieb wird im Teileprogramm eingeschaltet mit:
SPCON(<Spindelnummer>)
- Das folgende NC/PLC-Nahtstellensignal ist gesetzt:
DB31, ... DBX61.5 (Lageregler aktiv)
- Die Beschleunigung im Lageregelbetrieb wird getriebestufenabhängig festgelegt im Maschinendatum:
MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Eigener Spindel-Reset

Das Verhalten der Spindel nach Reset bzw. Programmende (M2, M30) wird eingestellt mit dem Maschinendatum:

MD35040 \$MA_SPIND_ACTIVE_AFTER_RESET (eigener Spindel-Reset)

Wert	Bedeutung
0	Die Spindel wird mit Reset bzw. Programmende sofort mit der gültigen Beschleunigung auf Stillstand abgebremst. Die letzte programmierte Spindeldrehzahl und Spindeldrehrichtung werden gelöscht.
1	Bei Reset bzw. Programmende bleiben die letzte programmierte Spindeldrehzahl (S-Wert) und die letzte programmierte Spindeldrehrichtung (M3, M4, M5) erhalten. Die Spindel wird nicht abgebremst.

Ist vor Reset bzw. Programmende die konstante Schnittgeschwindigkeit (G96, G961) aktiv, wird die aktuelle Spindeldrehzahl (bezogen auf 100% Spindel-Override) intern als letzte programmierte Spindeldrehzahl übernommen.

Die Spindel kann nur stillgesetzt werden mit dem NC/PLC-Nahtstellensignal:

DB31, ... DBX2.2 (Restweg löschen / Spindel-Reset)

Bei Alarmen, die für die Spindel Schnellstopp erzeugen, wird die Drehrichtung gelöscht. Die letzte programmierte Spindeldrehzahl (S-Wert) bleibt erhalten. Nach Beseitigung der Alarmursache muss die Spindel neu gestartet werden.

Spindel-Istdrehzahlanzeige und Spindelverhalten bei G96, G961

DB31, ... DBX61.4 (Achse/Spindel steht)

Die Drehzahl, bei der die Spindel als "stehend" gilt, wird eingestellt mit dem Maschinendatum:

MD36060 \$MA_STANDSTILL_VELO_TOL

Der Wert sollte so bemessen werden, dass im Stillstand das folgende NC/PLC-Nahtstellensignal sicher ansteht:

DB31,... DBX61.4 (Achse/Spindel steht)

Wenn DB31,... DBX61.4 (Achse/Spindel steht) gemeldet wird und für die Spindel keine Lageregelung aktiv ist, dann wird an der Bedienoberfläche die Istdrehzahl mit Null angezeigt und mit der Systemvariablen \$AA_S[<n>] wird Null gelesen.

Spindelverhalten bei konstanter Schnittgeschwindigkeit G96, G961

- Zu Beginn der Bearbeitung (Übergang von G0 zu Gx) und nach NC-Stop, G60 (Genauhalt modal) und G9 (Genauhalt satzweise) wird für den Bahnstart abgewartet, bis die Istdrehzahl den Toleranzbereich der Solldrehzahl erreicht hat:
DB31, ... DBX83.5 (nIst = nSoll)
- Die NC/PLC-Nahtstellensignale:
DB31, ... DBX83.5 (nIst = nSoll)
und
DB31, ... DBX83.1 (Solldrehzahl begrenzt)
sind auch bei starken Drehzahländerungsvorgaben (Plan-Achse fährt gegen Position 0) definiert gesetzt.

- Beim Unterschreiten der Minimaldrehzahl oder beim Erkennen des NC/PLC-Nahtstellensignals: DB31, ... DBX61.4 (Achse/Spindel steht) wird das NC/PLC-Nahtstellensignal: DB31, ... DBX83.5 (nIst = nSoll) zurückgesetzt (z. B. für Maschinennotstrategie).
- Eine begonnene Bahnbearbeitung (G64, Überschleifen) wird nicht unterbrochen.

Zusätzlich wird das Spindelverhalten von dem folgenden Maschinendatum beeinflusst:

MD35500 \$MA_SPIND_ON_SPEED_AT_IPO_START (Vorschubfreigabe bei Spindel im Sollbereich)

Spindelverhalten bei Getriebestufenwechsel-Ende

- Mit dem NC/PLC-Nahtstellensignal: DB31, ... DBX16.3 (Getriebe ist umgeschaltet) wird der NC mitgeteilt, dass die neue Getriebestufe (NC/PLC-Nahtstellensignale DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)) gültig ist, und der Pendelbetrieb beendet wird. Dabei ist es unwesentlich, ob das NC/PLC-Nahtstellensignal: DB31, ... DBX18.5 (Pendeldrehzahl) noch gesetzt ist. Die Istgetriebestufe sollte der Sollgetriebestufe entsprechen. Zur Anwahl des Parametersatzes ist die gemeldete Istgetriebestufe relevant.
- Nach der Quittierung des Getriebestufenwechsel (GSW) durch die PLC (DB31, ... DBX16.3) befindet sich die Spindel im Drehzahlsteuerbetrieb (DB31, ... DBX84.7 = 1). War vor dem GSW eine Drehrichtung (M3, M4, M5 oder FC18: "Start Spindel drehen") oder eine Spindeldrehzahl (S-Wert) programmiert, dann werden nach dem GSW die letzte Drehrichtung und Drehzahl wieder aktiv.

17.2.4 Pendelbetrieb

Für die Spindel wird während des Getriebestufenwechsels der Pendelbetrieb aktiviert.

Die Wirkungsweise ist ausführlich beschrieben unter dem Thema "Getriebestufenwechsel mit Pendelbetrieb (Seite 1356)".

17.2.5 Positionierbetrieb

17.2.5.1 Allgemeine Funktionalität

Wann Positionierbetrieb?

Beim Positionierbetrieb wird die Spindel an der vorgegebenen Position angehalten. Dabei wird die Lageregelung eingeschaltet und bleibt bis zur Abwahl aktiv.

Bei folgenden Funktionen befindet sich die Spindel im Positionierbetrieb:

- SPOS[<n>]=...
- SPOS[<n>]=ACP(...)
- SPOS[<n>]=ACN(...)
- SPOS[<n>]=AC(...)
- SPOS[<n>]=IC(...)
- SPOS[<n>]=DC(...)
- SPOSA[<n>]=ACP(...)
- SPOSA[<n>]=ACN(...)
- SPOSA[<n>]=AC(...)
- SPOSA[<n>]=IC(...)
- SPOSA[<n>]=DC(...) gleich mit SPOSA[<n>]=...
- M19 bzw. M[<n>]=19

Die Adresserweiterung [<n>] mit <n> = Spindelnummer kann für die Hauptspindel entfallen.

SPOS[<n>]=AC(...)

Spindelpositionierung auf eine absolute Position (0 bis 359,999 Grad). Die Positionierrichtung wird entweder durch die momentane Spindeldrehrichtung (Spindel dreht) oder durch den Restweg bestimmt.

SPOS[<n>]=IC(...)

Spindelpositionierung auf eine inkrementelle Position (+/- 999999.99 Grad) bezüglich der letzten programmierten Position. Die Positionierrichtung wird durch das Vorzeichen des zu verfahrenen Wegs festgelegt.

SPOS[<n>]=DC(...)

Spindelpositionierung auf kürzestem Weg auf eine absolute Position (0 bis 359,999 Grad). Die Positionierrichtung wird entweder durch die momentane Spindeldrehrichtung (Spindel dreht) oder automatisch durch die Steuerung bestimmt (Spindel steht).

SPOS[<n>]=...

Gleicher Funktionsablauf wie SPOS[<n>]=DC(...).

SPOS[<n>]=ACP(...)

Fährt die Position aus positiver Richtung an.

Beim Positionieren aus negativer Drehrichtung wird bis auf Drehzahl Null gebremst und in die Gegenrichtung beschleunigt, um die positive Anfahrrichtung ausführen zu können.

SPOS[<n>]=ACN(...)

Führt die Position aus negativer Richtung an.

Beim Positionieren aus positiver Drehrichtung wird bis auf Drehzahl Null gebremst und in die Gegenrichtung beschleunigt, um die negative Anfahrrichtung ausführen zu können.

M19 (DIN 66025)

Mit M19 kann die Spindel positioniert werden. Die Position und der Positionsanfahrmodus werden dabei aus folgenden Settingdaten gelesen:

SD43240 \$SA_M19_SPOS[<n>] (Spindelposition für Spindelpositionieren mit M19)

SD43250 \$SA_M19_SPOSMODE[<n>] (Spindelpositionsanfahrmodus für Spindelpositionieren mit M19)

Die Positionierungsmöglichkeiten von M19 sind identisch mit denen von:

SPOS = <Anfahrmodus> <Position/Weg>

M19 wird als Hilfsfunktion an die NC/PLC-Nahtstelle alternativ zu M3, M4, M5 und M70 ausgegeben. Der M19-Satz bleibt im Interpolator für die Dauer der Positionierung aktiv (wie SPOS).

Teileprogramme, die M19 als Makro (z. B. `DEFINE M19 AS SPOS=0`) oder als Unterprogramm verwenden, bleiben weiterhin ablauffähig. Aus Gründen der Kompatibilität zu bisherigen Steuerungen kann die interne Behandlung von M19 (NC positioniert die Spindel) nach folgendem Beispiel ausgeschaltet werden:

```
MD22000 $MC_AUXFU_ASSIGN_GROUP[0] = 4      ; Hilfsfunktionsgruppe: 4
MD22010 $MC_AUXFU_ASSIGN_TYPE[0] = "M"      ; Hilfsfunktionsart: "M"
MD22020 $MC_AUXFU_ASSIGN_EXTENSION[0] = 0    ; Hilfsfunkt. Erweiterung: 0
MD22030 $MC_AUXFU_ASSIGN_VALUE[0] = 19      ; Hilfsfunktionswert: 19
```

Implizit erzeugte Hilfsfunktion M19

Um eine Durchgängigkeit von M19 und SPOS bzw. SPOSA bezüglich des Verhaltens an der NC/PLC-Nahtstelle zu erreichen, kann bei SPOS und SPOSA die Hilfsfunktion M19 an die NC/PLC-Nahtstelle ausgegeben werden.

Für die Aktivierung dieser Funktion gibt es zwei Möglichkeiten:

- Kanalspezifische Aktivierung für alle Spindeln im Kanal über das Maschinendatum:
MD20850 \$MC_SPOS_TO_VDI (Ausgabe von M19 an die PLC bei SPOS/SPOSA)

Bit	Wert	Bedeutung
0	0	Wenn auch im MD35035 \$MA_SPIND_FUNCTION_MASK das Bit 19 auf "0" gesetzt ist, dann wird bei SPOS und SPOSA keine Hilfsfunktion M19 erzeugt. Damit entfällt auch die Quittierungszeit der Hilfsfunktion.
	1	Bei der Programmierung von SPOS und SPOSA im Teileprogramm wird die Hilfsfunktion M19 erzeugt und an die PLC ausgegeben. Die Adresserweiterung entspricht der Spindelnummer.

- Spindelspezifische und kanalübergreifende Aktivierung über das Maschinendatum: MD35035 \$MA_SPIND_FUNCTION_MASK (Spindelfunktionen)

Bit	Wert	Bedeutung
19	0	Wenn auch im MD20850 \$MC_SPOS_TO_VDI das Bit 0 auf "0" gesetzt ist, dann wird bei SPOS und SPOSA keine Hilfsfunktion M19 erzeugt. Damit entfällt auch die Quittierungszeit der Hilfsfunktion.
	1	Bei der Programmierung von SPOS und SPOSA wird die implizite Hilfsfunktion M19 erzeugt und an die PLC ausgegeben. Die Adresserweiterung entspricht der Spindelnummer.

Hinweis

Die Aktivierung über MD35035 sollte bei Verwendung einer Spindel in mehreren Kanälen (Achs-/Spindeltausch) bevorzugt werden.

Die Hilfsfunktion M19 wird implizit erzeugt, wenn eine der beiden MD-Projektierungen = 1 ist.

Mit der Aktivierung erhöht sich die Mindestdauer eines SPOS/SPOSA-Satzes auf die Ausgabe- und Quittierungszeit der Hilfsfunktionen durch die PLC.

Die implizit erzeugte Hilfsfunktionsausgabe M19 hat die Eigenschaften "Quick" und "Ausgabe während der Bewegung". Diese Eigenschaften sind fest eingestellt und unabhängig von der M19-Projektierung in den Hilfsfunktionsspezifischen Maschinendaten (MD..._\$M..._AUXFU...).

Bei Spindelpositionieranweisungen über FC 18 wird **keine** Hilfsfunktion M19 implizit erzeugt.

Positionierende

Das Positionierende kann programmiert werden mit:

FINEA[S<n>]:	Bewegungsende beim Erreichen von "Genauhalt fein" (DB31, ... DBX60.7)
COARSEA[S<n>]:	Bewegungsende beim Erreichen von "Genauhalt grob" (DB31, ... DBX60.6)
IPOENDA[S<n>]:	Bewegungsende beim Erreichen von "IPO-Stopp"

Zusätzlich kann mit IPOBRKA für Einzelachsinterpolation ein Bewegungsendekriterium für Satzwechsel bereits in der Bremsrampe (100-0%) eingestellt werden.

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2)

Satzwechsel

Sind die Bewegungsendekriterien für alle im Satz programmierten Spindeln bzw. Achsen und außerdem das Satzwechselkriterium für die Bahninterpolation erfüllt, so erfolgt der Satzwechsel. Dies gilt sowohl für Teileprogrammsätze als auch für Technologiezyklussätze.

SPOS, M19 und SPOSA haben gleiche Funktionalität, unterscheiden sich aber im Satzwechselverhalten:

- SPOS und M19
Der Satzwechsel erfolgt, wenn alle im Satz programmierten Funktionen ihr Satzendeckriterium erreicht haben (z. B. alle Hilfsfunktionen von der PLC quittiert, alle Achsen Endpunkt erreicht) und die Spindel das Positionierende erreicht hat.
- SPOSA
Der Satzwechsel erfolgt, wenn alle im Satz programmierten Funktionen (außer der Spindel) ihr Satzendeckriterium erreicht haben. Steht *SPOSA* allein im Satz, wird der Satzwechsel sofort eingeleitet. Die Spindelpositionierung kann sich dabei über mehrere Sätze erstrecken (siehe WAITS).

Koordinierung

Eine Koordinierung im Bewegungsablauf kann erreicht werden durch:

- PLC
- MD-Projektierung
- Programmierung im Teileprogramm

PLC

Wenn das NC/PLC-Nahtstellensignal:

DB31, ... DBX83.5 (Spindel im Sollbereich)

nicht ansteht, kann das kanalspezifische NC/PLC-Nahtstellensignal:

DB21, ... DBX 6.1 (Einlesesperre)

gesetzt werden, um eine bestimmte Position einer Spindel abzuwarten.

MD-Projektierung

Mit der Einstellung:

MD35500 \$MA_SPIND_ON_SPEED_AT_IPO_START = 1

unter Berücksichtigung der Toleranz:

MD35150 \$MA_SPIND_DES_VELO_TOL

erfolgt die Bahninterpolation erst dann, wenn die Spindel auf vorgewählte Drehzahl hochgedreht hat.

Mit der Einstellung:

MD35500 \$MA_SPIND_ON_SPEED_AT_IPO_START = 2

werden fahrende Bahnachsen vor Bearbeitungsbeginn angehalten am letzten G0.

Die weitere Bearbeitung erfolgt:

- mit dem nächsten Fahrbefehl.
- wenn die Spindeldrehzahl hochgefahren ist.
- bei MD35510 \$MA_SPIND_STOPPED_AT_IPO_START = 1
(Bahn-Vorschubfreigabe, wenn Spindel steht).

Programmierung im Teileprogramm

Koordinierungsmaßnahmen im Teileprogramm haben folgende Vorzüge:

- Der Teileprogramm-Ersteller kann entscheiden, an welcher Programmstelle der Spindelhochlauf abgeschlossen sein muss, um z. B. eine Werkstückbearbeitung zu beginnen.
- Unnötige Wartezeiten werden vermieden.

Die Koordinierung im Teileprogramm erfolgt durch Programmierung der Anweisung `WAITS`:

<code>WAITS:</code>	für Hauptspindel (Masterspindel)
<code>WAITS [<n>]:</code>	für Spindel mit Nummer <n>
<code>WAITS [<n>, <m>, ...]:</code>	für mehrere Spindeln bis zur maximalen Spindelanzahl



VORSICHT

Koordinierungsfehler

Der Teileprogramm-Ersteller muss dafür sorgen, dass eine der folgenden Wartebedingungen für `WAITS` eintreten wird:

- Position erreicht
- Spindel steht
- Spindel auf programmierte Drehzahl hochgefahren

Bei kanalübergreifender Spindelnutzung muss der Teileprogramm-Ersteller sicherstellen, dass `WAITS` frühestens in der Phase startet, in der sich die Spindel eines anderen Kanals bereits in der Beschleunigungs- bzw. Bremsphase hin zur gewünschten neuen Drehzahl bzw. Drehrichtung befindet.

Mit der Abarbeitung nachfolgender Sätze wird solange gewartet, bis:

- mit `SPOSA` programmierte Position(en) erreicht sind.
- bei M5 Spindelstillstand erreicht ist:
DB31, ..., DBX 61.4 (Spindel steht)
unter Berücksichtigung der Toleranz:
MD36060 \$MA_STANDSTILL_VELO_TOL
Mit dem Erkennen des ersten Auftretens des Signals wird `WAITS` beendet und der nächste Satz eingewechselt.
- bei M3/M4 (Drehzahlsteuerbetrieb) die Drehzahl im Sollbereich ist:
DB31, ..., DBX83.5 (Spindel im Sollbereich)
unter Berücksichtigung der Toleranz:
MD35150 \$MA_SPIND_DES_VELO_TOL
Mit dem Erkennen des ersten Auftretens des Signals wird `WAITS` beendet und der nächste Satz eingewechselt.
Diese Funktion von `WAITS` wirkt im programmierten Kanal.
Mit `WAITS` kann auf alle Spindeln gewartet werden, die diesem Kanal bekannt sind, wobei Spindeln auch in anderen Kanälen gestartet worden sein können.

Sonderfälle

- **Toleranz für Spindelgeschwindigkeit:**
Bei Maschinendatum-Einstellung:
MD35150 \$MA_SPIND_DES_VELO_TOL = 0
wird das NC/PLC-Nahtstellensignal:
DB31, ... DBX83.5 (Spindel im Sollbereich)
immer auf 1 gesetzt.
WAITS wird beendet, sobald bei Drehzahl- oder bei Drehrichtungsänderung (M3/M4) die Spindel sollwertseitig die Vorgabe erreicht hat.
- **Fehlende Freigabesignale:**
Wartet WAITS im Drehzahlsteuerbetrieb auf das Signal "Spindel im Sollbereich" und die Spindel stoppt bzw. dreht nicht, weil ein Freigabesignal (axiale Vorschubfreigabe, Regler-, Impulsfreigabe, ...) fehlt, so wird der Satz erst beendet, wenn das Signal "Spindel im Sollbereich" ansteht, nachdem die Freigabesignale wieder gekommen sind.
- **Verhalten bei NC- und BAG-Stop:**
Wird während WAITS ein NC- oder BAG-Stop ausgelöst, so wird nach NC-Start der Wartevorgang mit allen oben genannten Randbedingungen fortgesetzt.

Hinweis

Insbesondere bei kanalübergreifender Spindelnutzung ist bei der Programmerstellung zu beachten, dass WAITS in einem Kanal nicht zu früh gestartet wird, also zu einem Zeitpunkt, wo die Spindel des anderen Kanals noch mit ihrer "alten" Drehzahl dreht.
In diesem Fall steht das Signal "Spindel im Sollbereich" an und WAITS wird zu früh beendet.
Um dies zu vermeiden, empfiehlt es sich, vor WAITS unbedingt ein WAITM zu setzen.

Vorschub

Die Positioniergeschwindigkeit wird projiziert im Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO (Lageregeleinschaltdrehzahl)

Die projizierte Positioniergeschwindigkeit kann durch Programmierung oder durch Synchronaktionen geändert werden:

FA[S<n>]=<Wert>

mit: <n>: Spindelnummer
 <Wert>: Positioniergeschwindigkeit in Grad/min

Bei FA[S<n>]=0 wird die projizierte Geschwindigkeit wirksam.

Beschleunigung

Die Beschleunigungen werden projiziert in den Maschinendaten:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL (Beschleunigung im Lageregelbetrieb)

MD35200 \$MA_GEAR_STEP_SPEEDCTRL_ACCEL (Beschleunigung im Drehzahlsteuerbetrieb)

Die projizierte Positionierdynamik kann durch Programmierung oder durch Synchronaktionen geändert werden:

ACC[S<n>]=<Wert>

mit: <n>: Spindelnummer

<Wert>: Beschleunigung in Prozent bezogen auf die projektierte Beschleunigung

Bei $ACC[S<n>]=0$ wird die projektierte Beschleunigung wirksam.

Abbruch Positioniervorgang

Der Positioniervorgang wird abgebrochen:

- durch das NC/PLC-Nahtstellensignal:
DB31, ... DBX2.2 (Restweg löschen / Spindel-Reset).
- mit jedem Reset (z. B. Bedientafelfront-Reset).
- durch NC-Stop.

Das Abbruchverhalten ist unabhängig vom Maschinendatum:

MD35040 \$MA_SPIND_ACTIVE_AFTER_RESET (eigener Spindel-Reset)

Besonderheiten

Der Spindel-Override-Schalter ist gültig.

17.2.5.2 Positionieren aus der Drehung

Ausgangssituation

Die Spindel kann sich zum Zeitpunkt des Positionierstarts (*SPOS*, *M19* oder *SPOSA* im Programm) im Drehzahlsteuerbetrieb oder Lageregelbetrieb befinden.

Folgende Fälle sind zu unterscheiden:

Fall 1:	Spindel im Drehzahlsteuerbetrieb, Gebergrenzfrequenz überschritten
Fall 2:	Spindel im Drehzahlsteuerbetrieb, Gebergrenzfrequenz nicht überschritten
Fall 3:	Spindel im Lageregelbetrieb
Fall 4:	Spindeldrehzahl < Lageregel-Einschaltdrehzahl

Ablauf

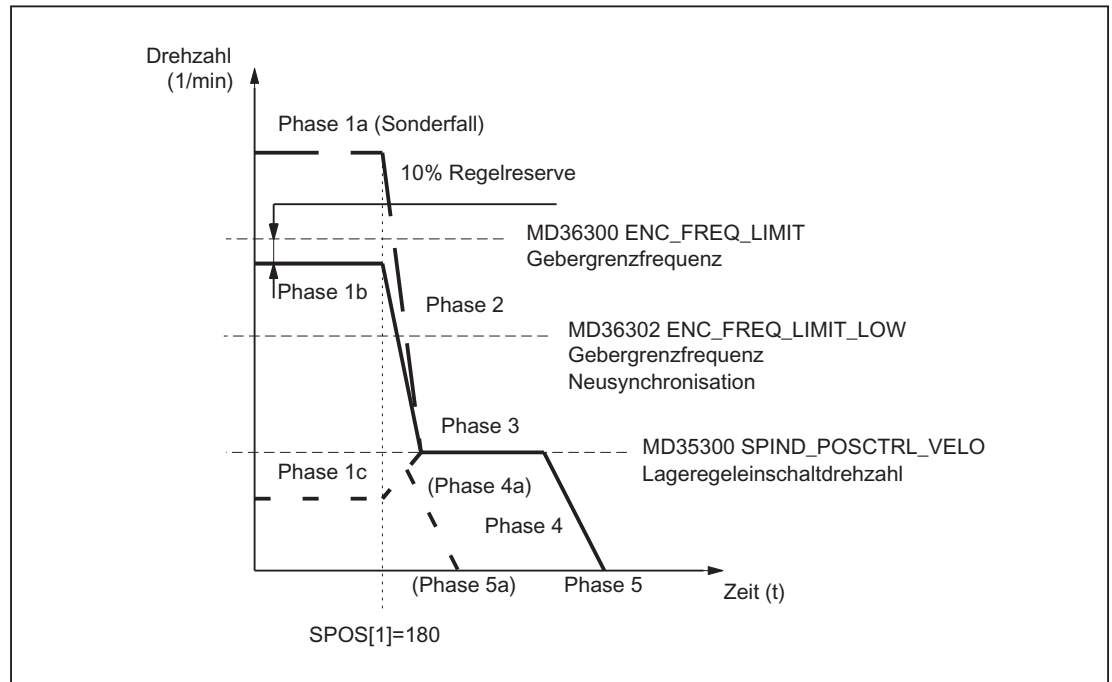


Bild 17-1 Positionieren aus der Drehung

Hinweis

Die Drehzahl, die sich aus der Projektierung der Gebergrenzfrequenz für die Geber-Neusynchronisation (MD36302 \$MA_ENC_FREQ_LOW) ergibt, muss größer als die Lageregeleinschalt Drehzahl (MD35300 \$MA_SPIND_POSCTRL_VELO) sein.

Phase 1

Positionieren aus Phase 1a:

Spindel dreht mit größerer Drehzahl als Gebergrenzfrequenz. Die Spindel ist nicht synchronisiert.

Positionieren aus Phase 1b:

Spindel dreht mit kleinerer Drehzahl als Gebergrenzfrequenz. Die Spindel ist synchronisiert.

Hinweis

Ist die Lageregelung eingeschaltet kann die Drehzahl nur 90% der Maximaldrehzahl der Spindel bzw. der Gebergrenzfrequenz betragen (10% Regelreserve notwendig).

Positionieren aus Phase 1c:

Die Spindel dreht sich mit der programmierten Spindeldrehzahl, wobei die Drehzahl kleiner ist als die projektierte Lageregeleinschalt Drehzahl:

MD35300 \$MA_SPIND_POSCTRL_VELO

Die Spindel ist synchronisiert.

Phase 2

Spindeldrehzahl > Lageregeleinschalt Drehzahl

Mit dem Wirksamwerden des Befehles *SPOS*, *M19* oder *SPOSA* beginnt das Abbremsen der Spindel bis auf die Lageregeleinschalt Drehzahl mit der projektierten Beschleunigung:

MD35200 \$MA_GEAR_STEP_SPEEDCTL_ACCEL

Mit Unterschreiten der Gebergrenzfrequenz wird die Spindel synchronisiert.

Spindeldrehzahl < Lageregeleinschalt Drehzahl

Mit der Programmierung von *SPOS*, *M19* oder *SPOSA* wird die Spindel in den Lageregelbetrieb geschaltet (wenn sie sich nicht schon im Lageregelbetrieb befindet).

Die projektierte Beschleunigung im Lageregelbetrieb wird aktiv:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Die Berechnung des Verfahrenswegs zur Zielposition wird ausgeführt.

Das Verfahren der Spindel bis zum programmierten Zielpunkt wird zeitoptimal durchgeführt. D. h., der Zielpunkt wird mit höchstmöglicher Geschwindigkeit angefahren (maximal MD35300 \$MA_SPIND_POSCTRL_VELO). Je nach den entsprechenden Randbedingungen werden die Phasen 2 - 3 - 4 - 5, bzw. 2 - 4a - 5a durchlaufen.

Phase 3

Spindeldrehzahl > Lageregeleinschalt Drehzahl

Mit Erreichen der projektierten Lageregeleinschalt Drehzahl (MD35300 \$MA_SPIND_POSCTRL_VELO) wird:

- die Lageregelung zugeschaltet (wenn kein Lageregelbetrieb ist)
- der Restweg (zur Zielposition) berechnet
- auf die projektierte Beschleunigung im Lageregelbetrieb umgeschaltet (bzw. diese Beschleunigung beibehalten):
MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Spindeldrehzahl < Lageregeleinschalt Drehzahl

Um den Zielpunkt zu erreichen, wurde bis auf die projektierte Lageregeleinschalt Drehzahl (MD35300 \$MA_SPIND_POSCTRL_VELO) beschleunigt. Diese wird nicht überschritten.

Die Bremseinsatzpunktberechnung erkennt, wann mit der projektierten Beschleunigung im Lageregelbetrieb (MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL) genau in die programmierte Spindelposition eingefahren werden kann.

Phase 4

Spindeldrehzahl > Lageregeleinschalt Drehzahl

Die Spindel bremst vom errechneten "Bremspunkt" mit Maschinendatum:
MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL
bis zur Zielposition ab.

Spindeldrehzahl < Lageregeleinschalt Drehzahl

Zu dem Zeitpunkt, den die Bremseinsatzpunktberechnung in Phase 3 erkannt hat, bremst die Spindel mit der projektierten Beschleunigung im Lageregelbetrieb (MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL) auf Stillstand ab.

Phase 4a:

Der Zielpunkt liegt bereits bei Wirksamwerden des Befehles `SPOS` so nah, dass die Spindel nicht mehr bis auf die projektierte Lageregeleinschalt Drehzahl (MD35300 \$MA_SPIND_POSCTRL_VELO) beschleunigt werden kann.

Die Spindel wird mit der projektierten Beschleunigung im Lageregelbetrieb (MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL) auf Stillstand abgebremst.

Phase 5

Spindeldrehzahl > Lageregeleinschalt Drehzahl

Die Lageregelung bleibt aktiv und hält die Spindel auf der programmierten Position.

Hinweis

Die max. Gebergrenzfrequenz des Spindellageistwertgebers wird von der Steuerung überwacht (Überschreiten möglich); im Lageregelbetrieb wird dann die Solldrehzahl auf 90% der Messsystem-Grenzdrehzahl verringert.

Dabei wird das folgende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX83.1 (Programmierte Drehzahl zu hoch)

Wird nach erfolgter Solldrehzahl-Verringerung immer noch "MS-Grenzfrequenz überschritten" festgestellt, so wird ein Alarm gemeldet.

Spindeldrehzahl < Lageregeleinschalt Drehzahl (Phase 5, 5a)

Die Spindel steht und hat die Position erreicht. Die Lageregelung ist aktiv und hält die Spindel auf der programmierten Position.

Wenn der Abstand zwischen der Spindelposition und der programmierten Position (Spindelsollposition) kleiner ist als projektierten Genauhaltgrenzen fein und grob, werden die folgenden NC/PLC-Nahtstellensignale gesetzt:

DB31, ... DBX60.6 (Position mit Genauhalt grob erreicht)

DB31, ... DBX60.7 (Position mit Genauhalt fein erreicht)

Die Genauhaltgrenzen werden festgelegt mit den Maschinendaten:

MD36010 \$MA_STOP_LIMIT_FINE (Genauhalt fein)

MD36000 \$MA_STOP_LIMIT_COARSE (Genauhalt grob)

Hinweis

Der Positioniervorgang gilt als beendet, wenn das Positionierendekriterium erreicht ist und gemeldet wird.

Die Voraussetzung ist "Genauhalt fein". Das gilt für SPOS, M19 oder SPOSA aus dem Teileprogramm, Synchronaktionen und Spindelpositionieren durch die PLC mit FC 18.

17.2.5.3 Positionieren aus dem Stillstand

Ablauf

Soll die Spindel aus dem Stillstand positioniert werden, sind zwei Fälle zu unterscheiden:

- Fall 1: Die Spindel ist nicht synchronisiert
Das ist dann der Fall, wenn die Spindel nach dem Einschalten der Steuerung und des Antriebs bzw. nach einem Getriebestufenwechsel positioniert werden soll (z. B. für einen Werkzeugwechsel).
MD31040 \$MA_ENC_IS_DIRECT = 0
- Fall 2: Die Spindel ist synchronisiert
Das ist dann der Fall, wenn die Spindel nach dem Einschalten der Steuerung und des Antriebs vor der ersten Positionierung mindestens eine Spindelumdrehung mit M3 oder M4 gedreht und dann mit M5 angehalten wurde (Synchronisation mit der Nullmarke).

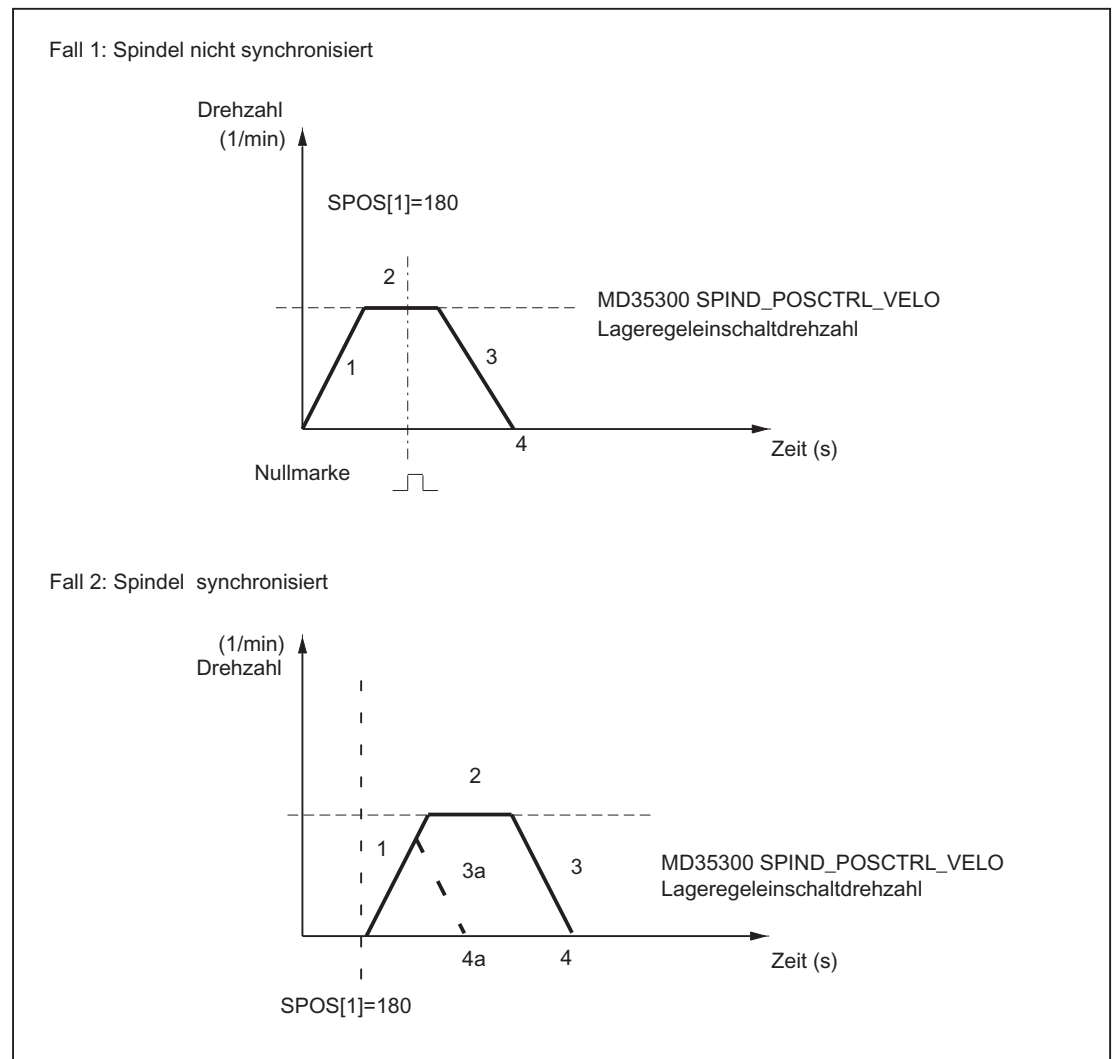


Bild 17-2 Positionieren bei stehender Spindel

Phase 1

Fall 1: Spindel nicht synchronisiert

Mit der Programmierung von `SPOS`, `M19` oder `SPOSA` beschleunigt die Spindel mit der Beschleunigung aus Maschinendatum:

MD35200 \$MA_GEAR_STEP_SPEEDCTRL_ACCEL (Beschleunigung im Drehzahlsteuerbetrieb)

Die Drehrichtung wird festgelegt durch das Maschinendatum:

MD35350 \$MA_SPIND_POSITIONING_DIR (Drehrichtung beim Positionieren aus dem Stillstand)

Ausnahme:

Wird mit `ACN`, `ACP`, `IC` positioniert, dann wird die programmierte Verfahrrichtung aktiviert.

Mit der nächsten Nullmarke des Spindellage-Istwertgebers wird die Spindel synchronisiert und geht in den Lageregelbetrieb.

Es wird überwacht, ob die Nullmarke im hinterlegten Weg gefunden wird (außer bei `IC`):

MD34060 \$MA_REFP_MAX_MARKER_DIST (Maximale Wegstrecke zur Referenzmarke)

Wird die im Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO (Positionierdrehzahl)

eingeebene Drehzahl erreicht, ohne dass die Spindel synchronisiert ist, dreht die Spindel mit der Lageregeleinschaltdrehzahl weiter (keine weitere Beschleunigung mehr).

Fall 2: Spindel synchronisiert

Mit der Programmierung von `SPOS`, `M19` oder `SPOSA` wird die Spindel in den Lageregelbetrieb geschaltet.

Die Beschleunigung aus dem folgenden Maschinendatum ist aktiv:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL (Beschleunigung im Lageregelbetrieb)

Die Drehrichtung wird durch die programmierte Bewegung (`ACP`, `ACN`, `IC`, `DC`) oder durch den anstehenden Restweg festgelegt.

Die im Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO (Lageregeleinschaltdrehzahl)

eingeebene Drehzahl wird nicht überschritten.

Die Berechnung des Verfahrenweges zur Zielposition wird ausgeführt.

Das Verfahren der Spindel bis zum programmierten Zielpunkt wird zeitoptimal durchgeführt. D. h., der Zielpunkt wird mit höchstmöglicher Geschwindigkeit angefahren (maximal MD35300 \$MA_SPIND_POSCTRL_VELO). Je nach den entsprechenden Randbedingungen werden die Phasen 1 - 2 - 3 - 4, bzw. 1 - 3a - 4a durchlaufen.

Phase 2

Fall 1: Spindel nicht synchronisiert

Ist die Spindel synchronisiert, wird die Lageregelung eingeschaltet.

Die Spindel dreht maximal mit der in Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO

hinterlegten Drehzahl so lange weiter, bis die Bremseinsatzpunktberechnung erkennt, wann mit der festgelegten Beschleunigung genau in die programmierte Spindelposition eingefahren werden kann.

Fall 2: Spindel synchronisiert

Um den Zielpunkt zu erreichen, wurde bis auf die im Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO

eingeegebene Drehzahl beschleunigt.

Diese wird nicht überschritten.

Die Bremseinsatzpunktberechnung erkennt, wann mit der festgelegten Beschleunigung im Maschinendatum:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

genau in die programmierte Spindelposition eingefahren werden kann.

Zu dem Zeitpunkt, den die Bremseinsatzpunktberechnung in Phase 1 erkannt hat, bremst die Spindel mit der Beschleunigung aus folgendem Maschinendatum auf Stillstand ab:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Phase 3

Zu dem Zeitpunkt, den die Bremseinsatzpunktberechnung in Phase 2 erkannt hat, bremst die Spindel mit der Beschleunigung aus folgendem Maschinendatum auf Stillstand ab:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Phase 3a:

Der Zielpunkt liegt bereits bei Wirksamwerden des Befehles `SPOS` so nah, dass die Spindel nicht mehr bis auf das Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO

beschleunigt werden kann.

Die Spindel wird mit der Beschleunigung aus dem folgenden Maschinendatum auf Stillstand abgebremst:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Phase 4, 4a

Die Spindel steht und hat die Position erreicht. Die Lageregelung ist aktiv und hält die Spindel auf der programmierten Position.

Die NC/PLC-Nahtstellensignale:

DB31, ... DBX60.6 (Position mit Genauhalt grob erreicht)

und

DB31, ... DBX60.7 (Position mit Genauhalt fein erreicht)

werden gesetzt, wenn der Abstand zwischen der Spindelposition und der programmierten Position (Spindelsollposition) kleiner als der Wert Genauhaltgrenze fein und grob ist.

Dies wird festgelegt in den Maschinendaten:

MD36010 \$MA_STOP_LIMIT_FINE

MD36000 \$MA_STOP_LIMIT_COARSE

Phase 3:

Zu dem Zeitpunkt, den die Bremseinsatzpunktberechnung in Phase 2 erkannt hat, bremst die Spindel mit der Beschleunigung aus folgendem Maschinendatum auf Stillstand ab:

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Phase 4:

Die Spindel steht und hat die Position erreicht. Die Lageregelung ist aktiv und hält die Spindel auf der programmierten Position.

Die NC/PLC-Nahtstellensignale:

DB31, ... DBX60.6 (Position mit Genauhalt grob erreicht)

und

DB31, ... DBX60.7 (Position mit Genauhalt fein erreicht)

werden gesetzt, wenn der Abstand zwischen der Spindelposition und der programmierten Position (Spindelsollposition) kleiner als der Wert Genauhaltgrenze fein und grob ist.

Dies wird festgelegt in den Maschinendaten:

MD36010 \$MA_STOP_LIMIT_FINE

MD36000 \$MA_STOP_LIMIT_COARSE

17.2.5.4 Signal "Spindel in Position" für Werkzeugwechsel

Funktion

Der Bewegungsablauf beim Werkzeugwechsel, vor allem bei Fräsmaschinen, besteht hauptsächlich aus dem Positionieren der Spindel und dem anschließenden (bei Optimierungen auch zeitparallelen) Anfahren der Werkzeugwechselposition mit den Bahnachsen. Als zwingende Voraussetzung gilt das Erreichen der Spindelposition vor dem Anfahren der Werkzeugwechselposition.

Wird der Werkzeugwechselzyklus durch den Maschinenbediener unterbrochen (z. B. mit NC-Stop, NC-Stop Achsen plus Spindeln, BAG-Stop, etc.), dann muss ausgeschlossen werden, dass die Spindel mit falscher Position in den Werkzeugwechsler hineinfährt.

Zur Positionskontrolle wird daher beim Spindelpositionieren beim Erreichen der zuletzt programmierten Spindelposition mit "Genauhalt fein" das folgende NC/PLC-Nahtstellensignal ausgegeben:

DB31, ... DBX85.5 (Spindel in Position)

Hinweis

Das Signal wird ausschließlich bei der Funktion "Spindelpositionieren" ausgegeben.

Dazu gehören:

- SPOS, SPOSA und M19 im Teileprogramm
 - SPOS und M19 in Synchronaktionen
 - Spindelpositionieren unter Verwendung des FC18
 - Spindelpositionieren über PLC-Schnittstelle (DB31, ... DBX30.4)
-

Setzen des Signals

Voraussetzungen für die Ausgabe des Signals DB31, ... DBX85.5 (Spindel in Position) sind:

- der Referenziertzustand der Spindel:
DB31, ... DBX60.4/5 (Referenziert/Synchronisiert 1/2) = 1

Hinweis

Da beim Positionieren der Spindel die Nullmarke selbständig gesucht wird, steht bei einem fehlerfreien Ablauf am Ende der Positionierbewegung das Referenziertsignal immer an.

- das Erreichen von "Genauhalt fein":
DB31, ... DBX60.7 (Genauhalt fein) = 1
Zusätzlich muss die zuletzt programmierte Spindelposition sollwertseitig erreicht worden sein.

Löschen des Signals

Mit der Wegnahme des Signals DB31, ... DBX60.7 (Genauhalt fein) wird immer auch das Signal DB31, ... DBX85.5 (Spindel in Position) zurückgesetzt.

Weitere Eigenschaften

- Steht die Spindel nach einer Positionierung bereits auf der programmierten Position, so bleibt das NC/PLC-Nahtstellensignal DB31, ... DBX85.5 (Spindel in Position) gesetzt.
- Wird die Spindel nach einer Positionierung (Signal "Spindel in Position" wurde ausgegeben) z. B. in der Betriebsart JOG verfahren, dann wird das NC/PLC-Nahtstellensignal DB31, ... DBX85.5 (Spindel in Position) gelöscht.
Wenn die Spindel in dieser Betriebsart auf ihre ursprüngliche Position zurückgefahren wird, dann wird das NC/PLC-Nahtstellensignal DB31, ... DBX85.5 (Spindel in Position) erneut gesetzt. Die letzte Positionsvorgabe bleibt erhalten.

17.2.6 Achsbetrieb

17.2.6.1 Allgemeine Funktionalität

Funktionalität

Ist es für bestimmte Bearbeitungsaufgaben, z. B. an Drehmaschinen mit Stirnflächenbearbeitung, nicht ausreichend, die Spindel ausschließlich drehzahlregelt über M3, M4, M5 zu verfahren bzw. und mit SPOS, M19 oder SPOSA zu positionieren, kann die Spindel in den lagegeregelten Achsbetrieb umgeschaltet und als Rundachse verfahren werden.

Beispielhafte Rundachsfunktionen:

- Programmierung mit Achsnamen
- Nullpunktverschiebungen (G54, G55, TRANS, ...)
- G90, G91, IC, AC, DC, ACP, ACN

- Kinematischen Transformationen (z. B. TRANSMIT)
- Bahninterpolation
- Verfahren als Positionierachse

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kapitel "Rundachsen (R2)"

Voraussetzungen

- Der Spindelmotor ist für den Spindelbetrieb und den Achsbetrieb derselbe.
- Das Lagemesssystem kann für den Spindelbetrieb und den Achsbetrieb dasselbe sein oder es können getrennte Lagemesssysteme benutzt werden.
- Für den Achsbetrieb ist zwingend ein Lageistwertgeber erforderlich.
- Die Spindel muss zur Verwendung des Achsbetriebs referenziert sein bzw. z.B. mit G74 referenziert werden.

Beispiel:

Programmcode	Kommentar
M70	; Spindel in den Achsbetrieb umschalten
G74 C1=0 Z100	; Achse referenzieren
G0 C180 X50	; Achse lagegeregelt verfahren

Projektierbare M-Funktion

Die M-Funktion, mit der die Spindel in den Achsbetrieb geschaltet wird, kann über das folgende Maschinendatum kanalspezifisch projiziert werden:

MD20094 \$MC_SPIND_RIGID_TAPPING_M_NR

Hinweis

Die Steuerung erkennt aufgrund der Programmierabfolge selbständig den Übergang in den Achsbetrieb (siehe "Impliziter Übergang in den Achsbetrieb (Seite 1320)"). Die explizite Programmierung der projizierten M-Funktion zum Schalten der Spindel in den Achsbetrieb im Teileprogramm ist daher nicht explizit notwendig. Die M-Funktion kann jedoch weiterhin programmiert werden, um z. B. die Lesbarkeit des Teileprogramms zu erhöhen.

Besonderheiten

- Der Vorschub-Override-Schalter ist gültig.
- Das NC/PLC-Nahtstellensignal beendet standardmäßig den Achsbetrieb nicht: DB21, ... DBX7.7 (Reset).
- Die NC/PLC-Nahtstellensignale:
DB31, ... DBB16 bis DBB19 und DBB82 bis DBB91
sind ohne Bedeutung, wenn:
DB31, ... DBX60.0 == 0 (Spindel / Rundachse)

- Der Achsbetrieb kann in jeder Getriebestufe eingeschaltet werden. Ist der Lageistwertgeber am Motor angebracht (indirektes Messsystem), können sich zwischen den Getriebestufen unterschiedliche Positionier- und Konturgenauigkeiten ergeben.
 - Ist der Achsbetrieb aktiv, kann die Getriebestufe nicht gewechselt werden. Dazu muss die Spindel zuerst mit M41 ... M45 bzw. M5, SPCOF in den Steuerbetrieb geschaltet werden.
 - Im Achsbetrieb wirkt der erste Parametersatz (Maschinendatenindex Null).
- Literatur**
Funktionshandbuch Grundfunktionen; Kapitel "Geschwindigkeiten, Soll-/Istwertsysteme, Regelung (G2)" > "Regelung" > "Parametersätze des Lagereglers"

Dynamik

Im Achsbetrieb gelten die dynamischen Grenzwerte der Achse. Z.B.:

- MD32000 \$MA_MAX_AX_VELO[<Achse>] (maximale Achsgeschwindigkeit)
- MD32300 \$MA_MAX_AX_ACCEL[<Achse>] (maximale Achsbeschleunigung)
- MD32431 \$MA_MAX_AX_JERK[<Achse>] (maximaler axialer Ruck bei Bahnbewegung)

Vorsteuerung

Der für die Achse aktive Vorsteuermodus wird beibehalten.

Eine ausführliche Beschreibung der Funktion "Dynamische Vorsteuerung" findet sich in:

Literatur

Funktionshandbuch Erweiterungsfunktionen; Kapitel "Kompensationen (K3)" > "Dynamische Vorsteuerung (Schleppfehler-Kompensation)"

Beispiel: Auflösungsumschaltung bei analogem Steller

Umschalten in den Achsbetrieb

Programmierung	Kommentar
SPOS=...	
M5	; Reglerfreigabe aus (von PLC) → wird ausgegeben an PLC
M70	; Steller umschalten (von PLC aufgrund von M70) Reglerfreigabe ein (von PLC)
C=...	; NC fährt mit Achsparametersatz

Umschalten in den Spindelbetrieb

Programmierung	Kommentar
C=...	
M71	; → wird ausgegeben an PLC Reglerfreigabe aus (von PLC) Steller umschalten (von PLC) NC-intern wird auf Spindelparametersatz (1-5) umgeschaltet, Reglerfreigabe ein (von PLC)

Programmierung	Kommentar
M3/4/5 oder SPOS=...	; NC fährt mit Spindelparametersatz

Wechsel in den Spindelbetrieb

Entsprechend der aktiven Getriebestufe wird der Parametersatz 1...5 angewählt.

Die Vorsteuerung wird, außer bei Gewindebohren mit Ausgleichsfutter, eingeschaltet, wenn gilt:

MD32620 \$MA_FFW_MODE (Vorsteuerungsart) \neq 0

Parametersatz	Achsbetrieb	Spindelbetrieb
1	gültig	-
2	-	gültig
3	-	gültig
4	-	gültig
5	-	gültig
6	-	gültig
Spindelbetrieb: Parametersatz entsprechend der Getriebestufe		

17.2.6.2 Impliziter Übergang in den Achsbetrieb

Funktion

Die Steuerung erkennt aufgrund der Programmierabfolge selbständig den Übergang in den Achsbetrieb und erzeugt steuerungsintern den notwendigen M70-Ablauf. Die Schritte werden situationsabhängig behandelt und umfassen maximal:

1. Stillsetzen der Spindel
2. Einschalten der Lageregelung, Behandlung von Vorsteuerung und Parametersatzumschaltung
3. Positionssynchronisation der Satzaufbereitung (interner Vorlaufstopp, wenn notwendig)

Die Funktion ist immer aktiv. Die explizite Programmierung von M70 im Teileprogramm ist damit grundsätzlich nicht notwendig.

Ablauf

Ablauf des impliziten Übergangs in den Achsbetrieb (keine Programmierung von M70 im Teileprogramm):

- Übergang vom Drehzahlsteuerbetrieb (M3, M4, M5, SPCOF, ...) in den Achsbetrieb: Von der Steuerung wird der Übergang intern erkannt und ein Zwischensatz vor dem Satz eingefügt, der Achsbetrieb erfordert. Der erzeugte Satz beinhaltet die M70-Funktionalität. Die Abarbeitungsdauer dieses Satzes entspricht in etwa dem eines programmierten M70-Satzes. Unterschiede können sich bei kurzen Umschaltungen bei stehender Spindel (keine Bremszeit) ergeben, wenn auf die implizite Erzeugung und Ausgabe der Hilfsfunktion M70 an die PLC verzichtet wird (siehe MD35035).
- Übergang vom Positionierbetrieb (M19, SPOS, SPOSA) in den Achsbetrieb: Der Übergang wird sofort und ohne die Erzeugung eines Zwischensatzes ausgeführt. Bei entsprechender Projektierung (siehe MD35035) wird die implizit erzeugte Hilfsfunktion M70 mit dem Einwechseln des Satzes, in dem die Spindel Achsbetrieb hat, an die PLC ausgegeben.

Hilfsfunktionsausgabe an PLC

Der implizite Übergang in den Achsbetrieb kann an die PLC in Form einer Hilfsfunktionsausgabe mitgeteilt werden.

Aktivierung / Deaktivierung

Die Aktivierung / Deaktivierung dieser Funktionalität erfolgt über das Maschinendatum:

MD35035 \$MA_SPIND_FUNCTION_MASK (Spindelfunktionen)

Bit	Wert	Bedeutung
20	0	Keine Hilfsfunktionsausgabe an die PLC bei steuerungsintern erzeugter M70-Funktionalität.
	1	Bei steuerungsintern erzeugter M70-Funktionalität wird implizit die Hilfsfunktion M70 erzeugt und an die PLC ausgegeben. Die Adresserweiterung entspricht der Spindelnummer.

Hinweis

Eine im Teileprogramm programmierte Hilfsfunktion M70 wird immer an die PLC ausgegeben.

Eigenschaften

Die implizit erzeugte Hilfsfunktionsausgabe M70 hat die Eigenschaften "Quick" und "Ausgabe während der Bewegung". Diese Eigenschaften sind fest eingestellt und unabhängig von der M70-Projektierung in den Hilfsfunktionsspezifischen Maschinendaten (MD..._\$M..._AUXFU...).

M70 wird nur einmal beim Übergang in den Achsbetrieb erzeugt. Bei anschließenden Sätzen, in denen die Spindel als Achse betrieben wird, wird keine weitere M70-Hilfsfunktion erzeugt und ausgegeben. Erst nach dem Verlassen des Achsbetriebs durch z. B. SPOS, M3, M4, M5, SPCOF, etc. und einem erneuten Übergang in den Achsbetrieb wird wieder implizit M70 erzeugt und ausgegeben.

Randbedingungen

Synchronaktionen

Bei der Programmierung der Spindel als Achse in Synchronaktionen muss weiterhin applikationsseitig dafür gesorgt werden, dass die Voraussetzungen für den Übergang in den Achsbetrieb bestehen.

Befindet sich die Spindel im Drehzahlsteuerbetrieb, dann muss vor der Programmierung als Achse die Anweisung `M70` oder `SPOS` programmiert werden. Anderenfalls kommt es bei der Achsprogrammierung zur Alarmmeldung.

FC 18

Wie bei Synchronaktionen muss auch bei FC 18 der Übergang in den Achsbetrieb applikationsseitig vorgenommen werden, z. B. durch eine vorbereitende Positionieranweisung. Anderenfalls wird der FC 18-Aufruf mit einem Fehlerbit im FC 18-Statuswort quittiert.

Bei Übergang in den Achsbetrieb durch Programmierung über FC 18 wird **keine** Hilfsfunktion M70 implizit erzeugt.

Beispiele

Beispiel 1:

Teileprogramm: Übergang von drehender Spindel in den Achsbetrieb

Projektionierung: MD35035 \$MA_SPIND_FUNCTION_MASK, Bit 20 = 1

Programmcode	Kommentar
N05 M3 S1000	
N10 ...	
N15 POS[C]=77	; Vor dem Einwechseln von N15 wird ein M70-Zwischensatz erzeugt, in dem die Spindel still gesetzt und M70 an die PLC ausgegeben wird.
...	

Beispiel 2:

Teileprogramm: Übergang aus dem Positionierbetrieb in den Achsbetrieb

Projektionierung: MD35035 \$MA_SPIND_FUNCTION_MASK, Bit 20 = 1

Programmcode	Kommentar
N05 SPOS=0	
N10 ...	
N15 C77	; Ausgabe des impliziten M70 an die PLC, kein Zwischensatz.
...	

Beispiel 3:

Synchronaktionen: Übergang vom Spindelpositionierbetrieb in den Achsbetrieb

Projektierung: MD35035 \$MA_SPIND_FUNCTION_MASK, Bit 20 = 1

Programmcode	Kommentar
WHEN COND1==TRUE DO SPOS=180	
WHEN COND2==TRUE DO POS[C]=270	; Ausgabe des impliziten M70 an die PLC.

Beispiel 4:

Synchronaktionen: Übergang vom Drehzahlsteuerbetrieb in den Achsbetrieb mit M70

Projektierung: MD35035 \$MA_SPIND_FUNCTION_MASK, Bit 20 = 1

Programmcode	Kommentar
WHEN COND11==TRUE DO M3 S1000	
WHEN COND12==TRUE DO M70	; Ausgabe von M70 an die PLC.
WHEN COND13==TRUE DO POS[C]=270	; Keine Erzeugung eines impliziten M70, da bereits Achsbetrieb vorliegt.

Beispiel 5:

Synchronaktionen: Unzulässiger Übergang vom Drehzahlsteuerbetrieb in den Achsbetrieb

Projektierung: MD35035 \$MA_SPIND_FUNCTION_MASK, Bit 20 = 1

Programmcode	Kommentar
WHEN COND21==TRUE DO M3 S1000	
WHEN COND22==TRUE DO POS[C]=270	; Alarm 20141!

17.2.7 Spindel-Grundstellung

Spindel-Grundstellung

Mit dem folgenden Maschinendatum wird eine Spindelbetriebsart als Grundstellung festgelegt:

MD35020 \$MA_SPIND_DEFAULT_MODE

Wert	Spindelgrundstellung
0	Drehzahlsteuerbetrieb, Lageregelung abgewählt
1	Drehzahlsteuerbetrieb, Lageregelung eingeschaltet
2	Positionierbetrieb
3	Achsbetrieb

Wirkungszeitpunkt der Spindel-Grundstellung

Der Wirkungszeitpunkt für die Spindel-Grundstellung wird eingestellt im Maschinendatum:

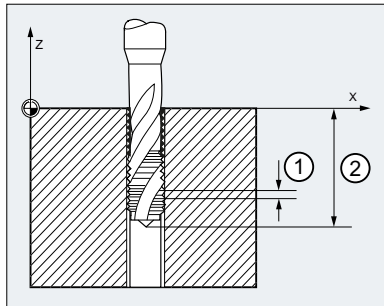
MD35030 \$MA_SPIND_DEFAULT_ACT_MASK

Wert	Wirkungszeitpunkt
0	POWER ON
1	POWER ON und Programmstart
2	POWER ON und RESET (M2 / M30)

17.2.8 Gewindebohren ohne Ausgleichsfutter

17.2.8.1 Funktion

Beim Gewindebohren ohne Ausgleichsfutter werden die Verfahrbewegungen der Linearachse und der Spindel lagegeregelt interpoliert. Voraussetzung dazu ist eine **lagegeregelter Spindel** mit **Wegmesssystem**.



- ① Gewindesteigung
- ② Gewindetiefe

Gewindedrehrichtung

Die Gewindedrehrichtung (Rechts- oder Linksgewinde) wird über das Vorzeichen der Steigung festgelegt:

- Positive Steigung → Rechtsgewinde (Drehrichtung entsprechend M3)
- Negative Steigung → Linksgewinde (Drehrichtung entsprechend M4)

Definierte Startposition der Spindel

Beim Gewindebohren ohne Ausgleichsfutter muss eine definierte Startposition der Spindel (SPOS) in folgenden Fällen programmiert werden:

- Gewinden, die in Mehrfachbearbeitung hergestellt werden
- Gewinden, bei denen eine definierte Gewindestartposition gefordert ist

SPOS = <Startposition>

G331 ...

17.2.8.2 Programmierung

Beim Gewindebohren ohne Ausgleichsfutter werden mit den Befehlen G331 und G332 folgenden Verfahrbewegungen ausgeführt:

- G331: Gewindebohren in Bohrrichtung bis zum Gewindeendpunkt
- G332: Rückzugsbewegung zum Gewindebohrsatz G331 mit automatischer Spindeldrehrichtungsumkehr

Syntax

G331 <Achse> <Gewindesteigung>
 G331 <Achse> <Gewindesteigung> S...
 G332 <Achse> <Gewindesteigung>

Bedeutung

G331:	Gewindebohren Die Gewindebohrung wird beschrieben durch die Verfahrbewegung der Achse (Bohrtiefe) und die Gewindesteigung.	
	Wirksamkeit:	modal
G332:	Rückzugsbewegung des Gewindebohrens Die Rückzugsbewegung muss mit derselben Steigung ausgeführt werden wie das Gewindebohren (G331). Die Richtungsumkehr der Spindel erfolgt automatisch.	
	Wirksamkeit:	modal
<Achse>:	Verfahrweg/position der Geometrieachse (X, Y oder Z) zum Gewindeendpunkt, z. B. Z50	
<Gewindesteigung>:	Gewindesteigung I (X), J (Y) oder K (Z): <ul style="list-style-type: none"> • Positive Steigung: Rechtsgewinde, z. B. K1,25 • Negative Steigung: Linksgewinde, z. B. K-1,25 	
	Wertebereich:	±0.001 bis ±2000.00 mm/Umdrehung
S...:	Spindeldrehzahl Ist keine Spindeldrehzahl angegeben, wird mit der zuletzt wirksamen Spindeldrehzahl verfahren.	

Hinweis

Zweiter Getriebestufen-Datensatz

Um beim Gewindebohren eine effektive Anpassung von Spindeldrehzahl und Motormoment zu erreichen und schneller beschleunigen zu können, kann in achsspezifischen Maschinendaten abweichend vom ersten Getriebestufen-Datensatz und auch unabhängig von diesen Drehzahlschaltsschwellen ein zweiter Getriebestufen-Datensatz für zwei weitere projektierbare Schaltschwellen (Maximaldrehzahl und Minimaldrehzahl) voreingestellt sein. Die Vorgaben des Maschinenherstellers sind zu beachten.

Literatur:

Funktionshandbuch Grundfunktionen; Spindeln (S1), Kapitel: "Projektierbare Getriebeanpassungen"

Beispiele

- Beispiel: Gewindebohren mit G331 / G332 (Seite 1326)
- Beispiel: Programmierte Bohrdrehzahl in der aktuellen Getriebestufe ausgeben (Seite 1326)
- Beispiel: Anwendung des zweiten Getriebestufen-Datensatzes (Seite 1327)
- Beispiel: Keine Drehzahlprogrammierung, Überwachung der Getriebestufe (Seite 1327)
- Beispiel: Getriebestufenwechsel nicht möglich, Überwachung der Getriebestufe (Seite 1328)
- Beispiel: Programmierung ohne SPOS (Seite 1328)

17.2.8.3 Beispiel: Gewindebohren mit G331 / G332

Programmcode	Kommentar
N10 SPOS[n]=0	; Spindel: Lageregelbetrieb ; Startposition 0 Grad
N20 G0 X0 Y0 Z2	; Achsen: Startposition anfahren
N30 G331 Z-50 K-4 S200	; Gewindebohren in Z, ; Steigung K-4 negativ => ; Spindeldrehrichtung: Linkslauf, ; Spindeldrehzahl 200 U/min
N40 G332 Z3 K-4	; Rückzugsbewegung in Z, ; Steigung K-4 negativ (Linkslauf), ; autom. Drehrichtungsumkehr => ; Spindeldrehrichtung Rechtslauf
N50 G1 F1000 X100 Y100 Z100 S300 M3	; Spindel im Spindelbetrieb.

17.2.8.4 Beispiel: Programmierte Bohrdrehzahl in der aktuellen Getriebestufe ausgeben

Programmcode	Kommentar
N05 M40 S500	; programmierte Spindeldrehzahl: 500 U/min => ; Getriebestufe 1 (20 bis 1028 U/min)
...	
N55 SPOS=0	; Spindel positionieren
N60 G331 Z-10 K5 S800	; Gewindebohren ; Spindeldrehzahl 800 U/min => Getriebestufe 1

Die zur programmierten Spindeldrehzahl S500 passende Getriebestufe bei M40 wird aus dem ersten Getriebestufen-Datensatz ermittelt. Die programmierte Bohrdrehzahl S800 wird in der aktuellen Getriebestufe ausgegeben und ist gegebenenfalls auf die Maximaldrehzahl der Getriebestufe begrenzt. Ein automatischer Getriebestufenwechsel nach erfolgtem SPOS ist

nicht möglich. Die Voraussetzung für den automatischen Getriebestufenwechsel ist der Drehzahlsteuerbetrieb der Spindel.

Hinweis

Soll bei einer Spindeldrehzahl von 800 U/min die Getriebestufe 2 angewählt werden, müssen die Schaltschwellen für die Maximal- und Minimaldrehzahl hierfür in den betreffenden Maschinendaten des zweiten Getriebestufen-Datensatzes projiziert sein (siehe nachfolgende Beispiele).

17.2.8.5 Beispiel: Anwendung des zweiten Getriebestufen-Datensatzes

Die Schaltschwellen des zweiten Getriebestufen-Datensatzes für die Maximaldrehzahl und Minimaldrehzahl werden bei G331/G332 und Programmierung eines S-Werts für die aktive Masterspindel ausgewertet. Automatischer Getriebestufenwechsel M40 muss aktiv sein. Die so ermittelte Getriebestufe wird mit der aktiven Getriebestufe verglichen. Besteht zwischen beiden ein Unterschied, wird ein Getriebestufenwechsel ausgeführt.

Programmcode	Kommentar
N05 M40 S500	; programmierte Spindeldrehzahl: 500 U/min
...	
N50 G331 S800	; Masterspindel: Getriebestufe 2 wird angewählt
N55 SPOS=0	; Spindel positionieren
N60 G331 Z-10 K5	; Gewindebohren
	; Spindel-Beschleunigung aus Getriebestufen-Datensatz 2

17.2.8.6 Beispiel: Keine Drehzahlprogrammierung, Überwachung der Getriebestufe

Wird bei Anwendung des zweiten Getriebestufen-Datensatzes mit G331 keine Drehzahl programmiert, wird das Gewinde mit der zuletzt programmierten Drehzahl gefertigt. Ein Getriebestufenwechsel erfolgt nicht. In diesem Fall wird jedoch überwacht, dass die zuletzt programmierte Drehzahl im vorgegebenen Drehzahlbereich (Schaltschwellen für die Maximal- und Minimaldrehzahl) der aktiven Getriebestufe liegt. Anderenfalls wird Alarm 16748 angezeigt.

Programmcode	Kommentar
N05 M40 S800	; programmierte Spindeldrehzahl: 800 U/min
...	
N55 SPOS=0	; Spindel positionieren
N60 G331 Z-10 K5	; Gewindebohren
	; Überwachung der Spindeldrehzahl 800 U/min
	; Getriebestufen 1 ist aktiv
	; Getriebestufe 2 müsste aktiv sein => Alarm 16748

17.2.8.7 Beispiel: Getriebestufenwechsel nicht möglich, Überwachung der Getriebestufe

Wird bei Anwendung des zweiten Getriebestufen-Datensatzes im G331-Satz zusätzlich zur Geometrie die Spindeldrehzahl programmiert, dann kann, falls die Drehzahl nicht im vorgegebenen Drehzahlbereich (Schaltschwellen für die Maximal- und Minimaldrehzahl) der aktiven Getriebestufe liegt, kein Getriebestufenwechsel durchgeführt werden, weil dann die Bahnbewegung von Spindel und Zustellachse(n) nicht eingehalten werden würde.

Wie im vorhergehenden Beispiel werden im G331-Satz die Drehzahl und die Getriebestufe überwacht und gegebenenfalls der Alarm 16748 angezeigt.

Programmcode	Kommentar
N05 M40 S500	; programmierte Spindeldrehzahl: 500 U/min => ; Getriebestufe 1
...	
N55 SPOS=0	; Spindel positionieren
N60 G331 Z-10 K5 S800	; Gewindebohren ; Getriebestufenwechsel nicht möglich, ; Überwachung der Spindeldrehzahl 800 U/min ; mit Getriebestufen-Datensatz 1: Getriebestufe 2 ; müsste aktiv sein => Alarm 16748

17.2.8.8 Beispiel: Programmierung ohne SPOS

Programmcode	Kommentar
N05 M40 S500	; programmierte Spindeldrehzahl: 500 U/min => ; Getriebestufe 1 (20 bis 1028 U/min)
...	
N50 G331 S800	; Masterspindel: Getriebestufe 2 wird ausgewählt
N60 G331 Z-10 K5	; Gewindebohren ; Spindel-Beschleunigung aus Getriebestufen-Datensatz 2

Die Gewindeinterpolation für die Spindel beginnt von der aktuellen Position, die von dem zuvor abgearbeiteten Teileprogrammabereich abhängig ist, z. B. wenn ein Getriebestufenwechsel ausgeführt wurde. Eine Nachbearbeitung des Gewindes ist deshalb ggf. nicht möglich.

Hinweis

Es ist darauf zu achten, dass bei der Bearbeitung mit mehreren Spindeln die Bohrspindel auch Masterspindel sein muss. Durch die Programmierung von SETMS (<Spindelnummer>) kann die Bohrspindel zur Masterspindel gemacht werden.

17.2.8.9 Sonderfall: Drehrichtungsumkehr über NC/PLC-Nahtstellensignal im NC-Programm

Im Normalfall muss eine Anforderung zur Drehrichtungsumkehr der Spindel über das NC/PLC-Nahtstellensignal "M3 / M4 invertieren" (DB31, ... DBX17.6) durch das PLC-Anwenderprogramm **vor** dem Start eines NC-Programms erfolgen. Nur beim Start wertet die NC den aktuellen Status des Nahtstellensignals automatisch aus und ändert gegebenenfalls die Drehrichtung der Spindel.

Ist während der Programmbearbeitung eine erneute Änderung der Spindeldrehrichtung durch das NC/PLC-Nahtstellensignal "M3 / M4 invertieren" (DB31, ... DBX17.6) erforderlich, kann dies innerhalb eines NC-Programms nur durch den nachfolgend beschriebenen prinzipiellen Ablauf erreicht werden:

N110 SPCOF oder M5	Spindel stoppen und in den Drehzahlregelbetrieb schalten
N111 Mxx oder Hxx	Ausgabe einer anwenderspezifische Hilfsfunktion zum Auslösen folgender Aktionen im PLC-Anwenderprogramm: <ul style="list-style-type: none"> • Setzen der Einleesperre im Kanal: DB21, ... DBX6.1 = 1 • Invertieren des Nahtstellensignals "M3/M4 invertieren" der Spindel: DB31, ... DBX17.6 • Rücksetzen der Einleesperre im Kanal: DB21, ... DBX6.1 = 0
N112 STOPRE	Vorlaufstop
N113 SPOS=IC(0.001)	Die Spindel im Lageregelbetrieb inkrementell verfahren. Durch das inkrementelle Verfahren der Spindel im Lageregelbetrieb wird von der NC das Nahtstellensignal DB31, ... DBX17.6 (M3/M4 invertieren) erneut ausgewertet. Ab diesem Satz wirkt beim Verfahren der Spindel die auf dem Nahtstellensignal basierende Drehrichtungsumkehr.
N114 SPOS=0	Optional: Positionieren der Spindel, falls technologisch eine definierte Ausgangsposition, z. B. 0°, erforderlich ist.

Beispiel

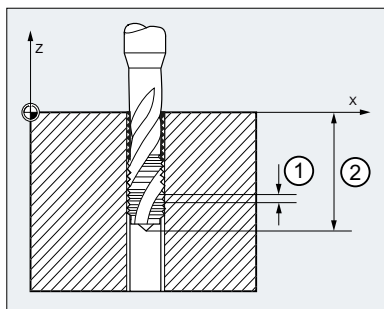
Programmcode	Kommentar
N010 G0 X0 Z0	; Grundstellung der Achsen
N020 G1 G94 F1000 Z10	; Z-Achse in Ausgangsposition verfahren
N030 M3 S100	; Spindel im Drehzahlregelbetrieb verfahren mit ; 100 U/min mit der Spindeldrehrichtung die beim ; Start des Programms ermittelt wurde
...	; beliebige Bearbeitung
N040 M5	; Spindel stoppen und in den Drehzahlregelbetrieb schalten
N050 H10	; Hilfsfunktionsausgabe: H0=10 ; Im PLC-Anwenderprogramm: ; - Einleesperre setzen: DB21, ... DBX6.1=1 ; - Drehrichtung invertieren: DB31, ... DBX17.6 ; - Einleesperre rücksetzen: DB21, ... DBX6.1=0
N060 STOPRE	; Vorlaufstop
N070 SPOS=IC(0.001)	; Spindel im Lageregelbetrieb inkrementell verfahren

Programmcode	Kommentar
N080 SPOS=0	; Spindel positionieren
N090 G331 Z-10 K1 S500	; Gewindebohren ohne Ausgleichsfutter
N100 G332 Z2 K1	; Gewindebohren ohne Ausgleichsfutter: Rückzugsbewegung
N110 G1 G94 F1000 Z10	; Zurückfahren in die Ausgangsposition der Z-Achse
N120 SPCOF	; Spindel in den Drehzahlregelbetrieb schalten
N130 H10	; siehe oben
N140 STOPRE	; Vorlaufstop
N150 SPOS=IC(.001)	; Spindel im Lageregelbetrieb inkrementell verfahren
N160 SPOS=0	; Spindel positionieren
N170 G331 Z-10 K1 S500	; Gewindebohren ohne Ausgleichsfutter
N180 G332 Z2 K1	; Gewindebohren ohne Ausgleichsfutter: Rückzugsbewegung
N190 G1 G94 F1000 Z10	; Z-Achse in Ausgangsposition verfahren
...	

17.2.9 Gewindebohren mit Ausgleichsfutter

17.2.9.1 Funktion

Beim Gewindebohren mit Ausgleichsfutter (G63) wird die Spindel nicht mit der Linearachse interpoliert, sondern drehzahl geregelt betrieben. Der von der Spindeldrehzahl und der Gewindesteigung abhängige Vorschub der Linearachse muss berechnet und explizit programmiert werden. Da Spindel und Linearachse nicht gemeinsam interpoliert werden, wird zum Ausgleich von z. B. Drehzahlschwankungen der Spindel, ein Gewindebohrer in einem Längenausgleichsfutter benötigt.



Rückzugsbewegung

Die Rückzugsbewegung wird ebenfalls mit G63 aber umgekehrter Spindeldrehrichtung programmiert.

Achs- und Spindel-Override-Wert

Während die Funktion "Gewindebohren **mit** Ausgleichsfutter" angewählt ist, wirken als Achs- und Spindel-Override-Wert 100 %.

Vorschub

Der zu programmierende Vorschub F der Linearachse ergibt sich aus dem Produkt von Spindeldrehzahl und der Gewindesteigung des Gewindebohrers:

$$F [\text{mm/min}] = \text{Spindeldrehzahl } S [\text{U/min}] * \text{Gewindesteigung} [\text{mm/U}]$$

17.2.9.2 Programmierung

Beim Gewindebohren mit Ausgleichsfutter werden mit dem Befehl G63 folgenden Verfahrbewegungen ausgeführt:

- G63: Gewindebohren in Bohrrichtung bis zum Gewindeendpunkt
- G63: Rückzugsbewegung mit programmierter Spindeldrehrichtungsumkehr

Hinweis

Nach einem G63 -Satz wird wieder die zuletzt wirksame Interpolationsart G0, G1, G2 aktiv.

Syntax

G63 <Achse> <Drehrichtung> <Drehzahl> <Vorschub>

Bedeutung

G63:	Gewindebohren mit Ausgleichsfutter	
	Wirksamkeit:	satzweise
<Achse>:	Verfahrweg/position der Geometrieachse (X, Y oder Z) zum Gewindeendpunkt, z. B. Z50	
<Drehrichtung>:	Spindeldrehrichtung: <ul style="list-style-type: none"> • M3: Drehrichtung im Uhrzeigersinn, Rechtsgewinde • M4: Drehrichtung gegen den Uhrzeigersinn, Linksgewinde 	
<Drehzahl>:	Maximale während des Gewindebohrens zulässige Spindeldrehzahl, z. B. S100	
<Vorschub>:	Vorschub der Bohrachse F, mit $F = \text{Spindeldrehzahl} * \text{Gewindesteigung}$	

Beispiel

Bohren eines M5-Gewindes:

- Gewindesteigung nach Norm: 0,8 mm/U
- Spindeldrehzahl S: 200 U/min
- Vorschub $F = 200 \text{ U/min} * 0,8 \text{ mm/U} = 160 \text{ mm/min}$.

Programmcode	Kommentar
N10 G1 X0 Y0 Z2 F1000 S200 M3	; Startpunkt anfahren
	; Spindel im Uhrzeigersinn, 200 U/min

Programmcode	Kommentar
N20 G63 Z-50 F160	; Gewindebohren mit Ausgleichsfutter ; Bohrtiefe: absolut Z=50mm ; Vorschub: 160 mm/min
N30 G63 Z3 M4	; Rückzugsbewegung: absolut Z=3mm ; Drehrichtungsumkehr ; Spindel gegen den Uhrzeigersinn, 200 U/min

17.3 Referenzieren / Synchronisieren

Warum Synchronisieren?

Damit nach dem Einschalten die Steuerung die Position der Spindel exakt kennt, muss die Steuerung mit dem Lagemesssystem der Spindel synchronisiert werden.

Erst mit einer synchronisierten Spindel sind folgende Funktionen möglich:

- Gewindeschneiden
- Gewindebohren ohne Ausgleichsfutter
- Achsprogrammierungen

Weitere Erläuterungen zum Synchronisieren der Spindel siehe Kapitel "R1: Referenzieren (Seite 1243)".

Warum Referenzieren?

Damit nach dem Einschalten die Steuerung den Maschinennullpunkt exakt kennt, muss die Steuerung mit dem Lagemesssystem der Rundachse synchronisiert werden. Diesen Vorgang nennt man Referenzieren. Den zeitlichen Ablauf, der eine Achse referenziert, nennt man Referenzpunktfahren.

Erst eine referenzierte Achse kann eine programmierte Position auf der Maschine genau anfahren.

Weitere Erläuterungen zum Referenzieren der Rundachse siehe Kapitel "R1: Referenzieren (Seite 1243)".

Montageorte der Lagemesssysteme

Die Lagemesssysteme können wie folgt montiert sein:

- direkt am Motor plus berührungsloser Näherungsschalter an der Spindel als Nullmarkengeber
- am Motor über ein Messgetriebe plus berührungsloser Näherungsschalter an der Spindel als Nullmarkengeber
- direkt an der Spindel
- an der Spindel über ein Messgetriebe plus berührungsloser Näherungsschalter an der Spindel als Nullmarkengeber (nur bei Übersetzungen ungleich 1:1)

Sind zwei Lagemesssysteme vorhanden, können beide Lagemesssysteme am gleichen Montageort oder an getrennten Montageorten angebracht sein.

Ablauf Synchronisieren

Nach dem Einschalten der Steuerung kann die Spindel wie folgt synchronisiert werden:

- Die Spindel wird mit einer Spindeldrehzahl (S-Wert) und einer Spindeldrehrichtung (M3 oder M4) gestartet und synchronisiert sich mit der nächsten Nullmarke des Lagemesssystems oder mit dem nächsten Signal des berührungslose Näherungsschalters.
- Die Spindel soll mit SPOS, M19 oder SPOSA aus dem Stillstand positioniert werden. Die Spindel synchronisiert sich mit der nächsten Nullmarke des Lagemesssystems oder mit dem nächsten Signal des berührungslose Näherungsschalters. Dann erfolgt die Positionierung auf die programmierte Position.

- Die Spindel kann mit SPOS, M19 oder SPOSA aus der Bewegung (nach M3 oder M4) synchronisiert werden.
Dabei liegt folgendes Verhalten vor:
 - Mit SPOS=<Pos>, SPOS=DC (<Pos>) und SPOS=AC (<Pos>) wird die Bewegungsrichtung beibehalten und die Position angefahren.
 - Mit SPOS=ACN (<Pos>) bzw. SPOS=ACP (<Pos>) wird die Position immer mit negativer bzw. positiver Bewegungsrichtung angefahren. Gegebenenfalls wird vor dem Positionieren die Bewegungsrichtung invertiert.
- Überfahren der Nullmarke in der Betriebsart JOG mittels den Richtungstasten im Drehzahlsteuerbetrieb.

Hinweis

Es ist gleichwertig, ob der Anstoß zum Synchronisieren aus dem Teileprogramm, FC 18 oder Synchronaktionen kommt.

Hinweis

Beim Synchronisieren der Spindel wirken alle vier möglichen Referenzpunktwerte des Referenzpunkts je nach angewähltem Messsystem. Die Verschiebung zum Messsystem wirkt gleichermaßen.

Zu beachten sind folgende Maschinendaten:

- MD34080 \$MA_REFP_MOVE_DIST
(Referenzpunktabstand/Zielpunkt bei abstandskodiertem System)
- MD34090 \$MA_REFP_MOVE_DIST_CORR
(Referenzpunktverschiebung/Absolutverschiebung abstandskodiert)
- MD34100 \$MA_REFP_SET_POS
(Referenzpunktwert, bei abstandskodiertem System ohne Bedeutung)

Wird eine nichtreferenzierte Spindel mit SPOS=IC (. . .) und einem Weg < 360 Grad positioniert, dann wird möglicherweise die Nullmarke nicht überfahren und die Spindelposition ist weiterhin nicht mit der Nullmarke synchronisiert. Dies kann vorkommen:

- nach POWER ON
 - durch Setzen der axialen NC/PLC-Nahtstellensignale:
DB31, ... DBX17.5 (Spindel neu synchronisieren beim Positionieren 2)
DB31, ... DBX17.4 (Spindel neu synchronisieren beim Positionieren 1)
-

Besonderheiten beim Synchronisieren mittels berührungslosem Näherungsschalter

Die durch die Verzögerung des Signals des berührungslosen Näherungsschalters hervorgerufene Positionsverfälschung kann NC-intern durch Eintrag einer Signallaufzeitkompensation korrigiert werden.

Die Einstellung der Signallaufzeitkompensation erfolgt über die Maschinendaten:

- MD31122 \$MA_BERO_DELAY_TIME_PLUS
(Verzögerungszeit des berührungslosen Näherungsschalters, positive Bewegungsrichtung)
- MD31123 \$MA_BERO_DELAY_TIME_MINUS
(Verzögerungszeit des berührungslosen Näherungsschalters, negative Bewegungsrichtung)

Die Auswirkung ist abhängig vom eingestellten Referenziermodus:

MD34200 \$MA_ENC_REFP_MODE = <Wert>

<Wert>	Bedeutung
2	Die Positionssynchronisation wird ohne Vorgabe einer bestimmten Anfahrgeschwindigkeit/drehzahl vorgenommen.
7	Als Anfahrgeschwindigkeit zur Positionssynchronisation wird die Geschwindigkeit aus folgendem Maschinendatum verwendet: MD34040 \$MA_REFP_VELO_SEARCH_MARKER (Abschaltgeschwindigkeit) Die Nullmarke wird nicht automatisch gesucht, sondern muss explizit angefordert werden mit der 0/1-Flanke des NC/PLC-Nahtstellensignals: DB31, ... DBX16.4 / 5 (Spindel neu synchronisieren, Messsystem 1 / 2)

Hinweis

Die Signallaufzeiten Signallaufzeitkompensation sind im Auslieferungszustand so vorbesetzt, dass in der Regel keine Änderung notwendig ist.

Ablauf Referenzieren

Soll direkt nach dem Steuerungshochlauf die Spindel im Achsbetrieb programmiert werden, muss sichergestellt sein, dass die Achse referenziert ist.

Nach dem Einschalten der Steuerung kann die Spindel referenziert werden (Voraussetzung ist eine Nullmarke pro Umdrehung).

Zum Ablauf des Referenzierens siehe Kapitel "R1: Referenzieren (Seite 1243)".

Mit der Synchronisation der Spindel (siehe Abschnitt "Ablauf Synchronisieren") wird auch die Rundachse gleichzeitig referenziert, wenn das Lagemesssystem der Spindel auch für die Rundachse verwendet wird.

Lagemesssysteme Spindel

Die Spindel kann vom Spindelbetrieb in den Achsbetrieb (Rundachse) geschaltet werden, wenn für Spindel- und Achsbetrieb ein Motor verwendet wird.

Die Spindel (Spindelbetrieb und Achsbetrieb) kann mit einem oder mit zwei Lagemesssystemen ausgerüstet sein. Bei zwei Lagemesssystemen kann ein Lagemesssystem der Spindel und ein Lagemesssystem der Rundachse zugeordnet sein, oder es können zwei Lagemesssysteme der Spindel zugeordnet sein. Sind zwei Lagemesssysteme vorhanden, werden beide von der Steuerung aktualisiert, aber es kann nur ein Lagemesssystem aktiv sein.

Ausgewählt wird das aktive Lagemesssystem mit dem NC/PLC-Nahtstellensignal:

DB31, ... DBX1.5 / 6 (Lagemesssystem 1 / 2)

Das aktive Lagemesssystem wird für folgende Funktionen benötigt:

- Lageregelung der Spindel (SPCON)
- Spindelpositionierung (SPOS, M19 und SPOSA)
- Gewindeschneiden (G33, G34, G35)
- Gewindebohren ohne Ausgleichsfutter (G331, G332)
- Umdrehungsvorschub (G95)
- Konstante Schnittgeschwindigkeit (G96, G961, G97, G971)
- Spindelstdrehzahlanzeige
- Achsbetrieb
- Synchronspindelsollwertkopplung

Lagemesssystem der Spindel neu synchronisieren

In folgenden Fällen muss das Lagemesssystem der Spindel neu synchronisiert werden:

- Der Lagemessgeber ist am Motor, ein berührungsloser Näherungsschalter ist an der Spindel montiert und es erfolgt ein Getriebestufenwechsel. Die Synchronisation wird intern angestoßen, wenn sich die Spindel in der neuen Getriebestufe dreht (siehe Ablauf Synchronisieren).
- Die Maschine hat eine Umschaltung zwischen vertikaler und horizontaler Spindel. Dabei werden zwei verschiedene Lagemessgeber (einer für die vertikale Spindel und einer für die horizontale Spindel) aber nur ein Istwerteingang an der Steuerung verwendet. Wird zwischen vertikaler und horizontaler Spindel umgeschaltet, muss neu synchronisiert werden.

Diese Synchronisation wird angestoßen mit dem NC/PLC-Nahtstellensignal:

DB31, ... DBX16.4 (Spindel neu synchronisieren 1)

oder

DB31, ... DBX16.5 (Spindel neu synchronisieren 2)

Dazu muss sich die Spindel im Steuerbetrieb befinden.

Restaurierung der Position nach Warmstart

Bei Spindeln mit inkrementellen Lagemesssystemen besteht die Möglichkeit, Istwerte über POWER OFF hinweg zu puffern und nach POWER ON die letzte vor dem Ausschalten gepufferte Position zu restaurieren, damit positionsabhängige Funktionen wie z. B. Transformationen wiederhergestellt werden können (siehe Kapitel "Automatisches Wiederherstellen des Maschinenbezugs (Seite 1287)"). Eine Anwendung ist z. B. der Werkzeugrückzug nach Warmstart bei einer Bearbeitung mit Werkzeugorientierung (siehe Kapitel "Werkzeugrückzug nach POWER ON mit Orientierungstransformation (Seite 662)").

Der Zustand des Lagemesssystems nach der Restaurierung der Position wird durch folgende NC/PLC-Nahtstellensignale angezeigt:

DB31, ... DBX71.4 ("Restauriert 1") für Lagemesssystem 1

DB31, ... DBX71.5 ("Restauriert 2") für Lagemesssystem 2

Nach dem Freifahren des Werkzeugs in der Betriebsart JOG müssen Achsen, deren Positionen restauriert wurden, referenziert werden. Dadurch werden die Signale DB31, ... DBX71.4/5 ("Restauriert 1/2") gelöscht und die Signale DB31, ... DBX60.4/5 ("Referenziert/Synchronisiert 1/2") gesetzt.

Hinweis

Wird das Maschinendatum MD20700 \$MC_REFP_NC_START_LOCK auf den Wert "2" gesetzt, so ist ein NC-Start auch mit "restaurierten" Achspositionen möglich (in der Betriebsart MDA oder beim Überspeichern).

17.4 Projektierbare Getriebearpassungen

17.4.1 Getriebestufen für Spindeln und Getriebestufenwechsel

Warum Getriebestufen?

Getriebestufen bei Spindeln dienen dazu, die Motordrehzahl zu untersetzen und so bei kleinen Spindeldrehzahlen ein hohes Drehmoment zu erzeugen, oder zu übersetzen, um eine hohe Drehzahl zu erhalten.

Getriebestufenanzahl

Für jede Spindel sind 5 Getriebestufen projektierbar.

Die Anzahl der verwendeten Getriebestufen wird festgelegt mit dem Maschinendatum:

MD35090 \$MA_NUM_GEAR_STEPS

Parametrierung der Getriebestufen

Die Getriebestufen 1 bis 5 sind parametrierbar über folgende Maschinendaten:

Maschinendatum	Bedeutung
MD35012 \$MA_GEAR_STEP_CHANGE_POSITION[<n>]	Getriebestufenwechsel-Position
MD35110 \$MA_GEAR_STEP_MAX_VELO[<n>]	Maximaldrehzahl für den automatischen Getriebestufenwechsel
MD35120 \$MA_GEAR_STEP_MIN_VELO[<n>]	Minimaldrehzahl für den automatischen Getriebestufenwechsel
MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[<n>]	Maximaldrehzahl der Getriebestufe
MD35135 \$MA_GEAR_STEP_PC_MAX_VELO_LIMIT[<n>]	Maximaldrehzahl der Getriebestufe bei Lageregelung
MD35140 \$MA_GEAR_STEP_MIN_VELO_LIMIT[<n>]	Minimaldrehzahl der Getriebestufe

Maschinendatum	Bedeutung
MD35200 \$MA_GEAR_STEP_SPEEDCTRL_ACCEL[<n>]	Beschleunigung im Drehzahlsteuerbetrieb
MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL[<n>]	Beschleunigung im Lageregelbetrieb
MD35300 \$MA_SPIND_POSCTRL_VELO[<n>]	Lageregeleinschalt Drehzahl
MD35310 \$MA_SPIND_POSIT_DELAY_TIME[<n>]	Positionierverzögerungszeit
MD35550 \$MA_DRILL_VELO_LIMIT[<n>]	Maximaldrehzahl für das Gewindebohren ohne Ausgleichsfutter

Art des Getriebestufenwechsels

Der Art des Getriebestufenwechsels wird eingestellt im Maschinendatum:

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE

Bit	Wert	Bedeutung
0	0	Der Spindelmotor ist direkt (1:1) oder mit einer nicht veränderbaren Übersetzung an die Spindel angebaut (Grundstellung). Es wirken die Maschinendaten der ersten Getriebestufe.
	1	Spindelmotor mit max. 5 Getriebestufen. Der Getriebestufenwechsel erfolgt: <ul style="list-style-type: none"> • Im Pendelbetrieb • Auf unbestimmter Wechsellposition
1	0	Bedeutung wie bei Bit 0 = 0.
	1	Bedeutung wie bei Bit 0 = 1, jedoch erfolgt der Getriebestufenwechsel auf projektierte Spindelposition. Die Wechsellposition wird eingestellt im Maschinendatum: MD35012 \$MA_GEAR_STEP_CHANGE_POSITION Die Position wird in der aktuellen Getriebestufe vor dem Getriebestufenwechsel angefahren. Wenn Bit 1 gesetzt ist, dann wird Bit 0 nicht beachtet!
3	1	Der Getriebestufenwechseldialog zwischen NC und PLC wird simuliert.
5	1	Beim Gewindebohren mit G331/G332 wird der zweite Getriebestufen-Datensatz verwendet (siehe nachfolgenden Absatz "Zweiter Getriebestufen-Datensatz"). Das Bit muss für die beim Gewindebohren verwendete Masterspindel gesetzt werden.

Voraussetzung für einen Getriebestufenwechsel

Prinzipiell wird ein Getriebestufenwechsel nur dann ausgeführt, wenn die angeforderte Getriebestufe ungleich der aktiven Getriebestufe ist.

Parametersatzanwahl beim Getriebestufenwechsel

Mit der Getriebestufe wird auch der Servo-Parametersatz umgeschaltet, wenn:

MD35590 \$MA_PARAMSET_CHANGE_ENABLE = 0 oder 1

Für weitere Informationen siehe Kapitel "Parametersatzanwahl beim Getriebestufenwechsel (Seite 1351)".

Getriebestufenwechsel anfordern

Ein Getriebestufenwechsel kann angefordert werden:

- im Teileprogramm durch:
 - M40 S . . .
Automatische Getriebestufenauswahl zur programmierten Drehzahl S . . .
 - M41 ... M45
Direkte Anwahl der Getriebestufe 1 ... 5
 - M70
Bei MD35014 \$MA_GEAR_STEP_USED_IN_AXISMODE = 1 ... 5
(siehe "Projektierbare Getriebestufe bei M70 (Seite 1368)")
 - G331 S . . .
Bei MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE, Bit 5 = 1
- in Synchronaktionen durch:
 - DO M40 S . . .
Automatische Getriebestufenauswahl zur programmierten Drehzahl S . . .
 - DO M41 . . . M45
Direkte Anwahl der Getriebestufe 1 ... 5
 - DO M70
Bei MD35014 \$MA_GEAR_STEP_USED_IN_AXISMODE = 1 ... 5
- durch die PLC unter Verwendung des Funktionsbausteins FC18
- im Reset-Zustand durch Beschreiben der NC/PLC-Nahtstelle:
DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)
Insbesondere nach einem POWER ON kann der NC die mechanisch aktive Getriebestufe mitgeteilt werden.

Hinweis

Wenn der Spindelmotor direkt (1:1) oder mit einer nicht veränderbaren Übersetzung an die Spindel angebaut ist (MD35010 = 0), dann haben die Hilfsfunktionen M40 und M41 ... M45 für diese Spindel keine Bedeutung.

Getriebestufenauswahl

Die Getriebestufenauswahl zwischen zwei Getriebestufen bei Vorgabe einer maximalen Spindeldrehzahl wird in folgendem Beispiel gezeigt:

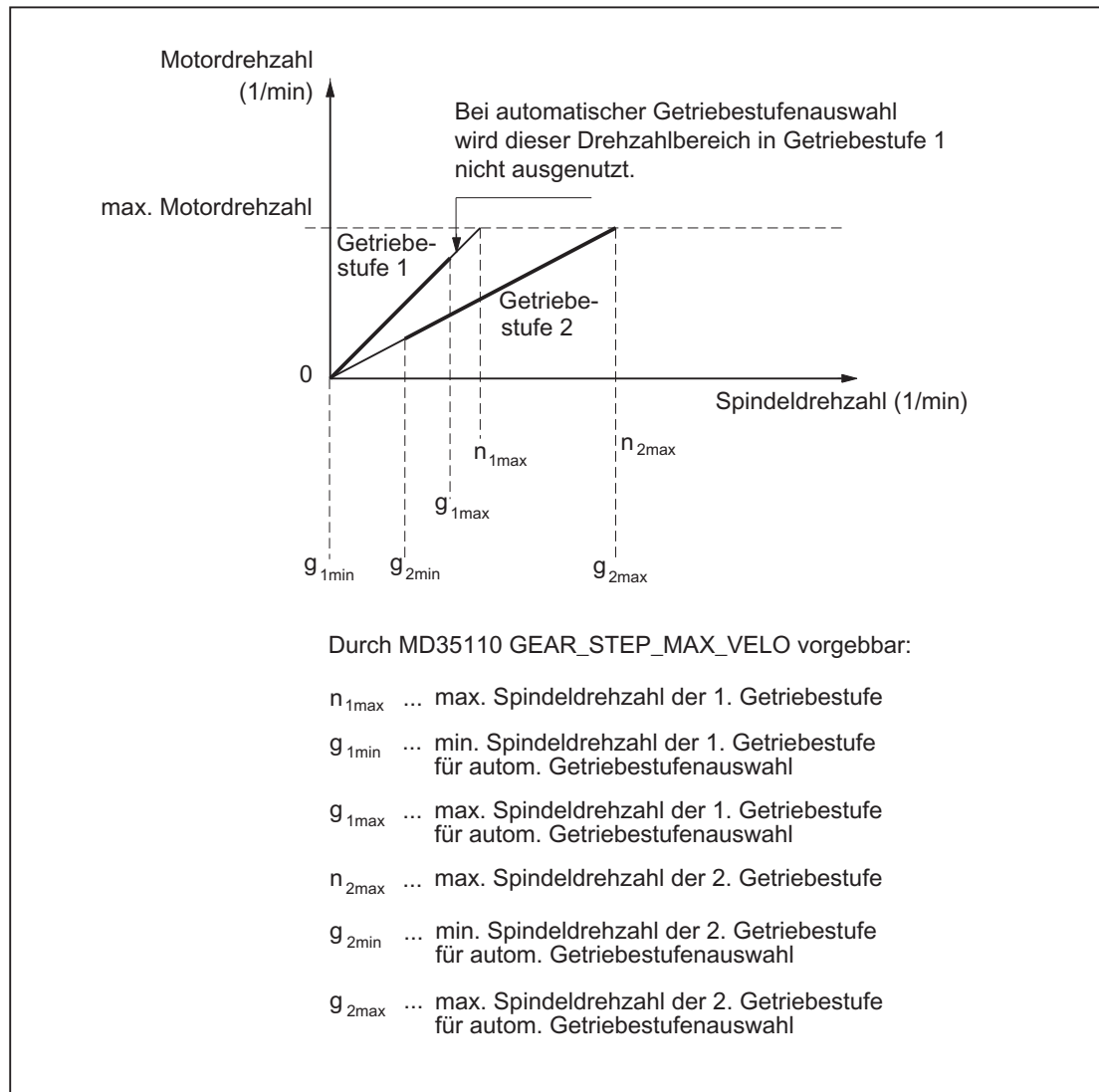


Bild 17-3 Getriebestufenwechsel mit Getriebestufenauswahl bei zwei Getriebestufen

Ablauf des Getriebestufenwechsels

Ist die neue Getriebestufe vorgewählt, dann findet folgender Ablauf statt:

1. Umschaltvorgang

Die beiden folgenden NC/PLC-Nahtstellensignale werden gesetzt:

DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)

DB31, ... DBX82.3 (Getriebe umschalten)

Je nachdem, zu welchem Zeitpunkt das NC/PLC-Nahtstellensignal:

DB31, ... DBX18.5 (Pendeldrehzahl)

gesetzt wird, bremst die Spindel mit der Beschleunigung für Pendeln oder mit der Beschleunigung für Drehzahlregelbetrieb / Lageregelbetrieb auf Stillstand ab.

Spätestens mit dem Stillstand der Spindel:

DB31, ... DBX61.4 (Achse/Spindel steht)

kann mit dem NC/PLC-Nahtstellensignal:

DB31, ... DBX18.5 (Pendeldrehzahl)

das Pendeln eingeschaltet werden.

Prinzipiell kann auch ohne Pendelvorgang die neue Getriebestufe eingelegt werden.

Ist die neue Getriebestufe eingelegt, dann werden vom PLC-Programm die beiden folgenden NC/PLC-Nahtstellensignale gesetzt:

DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

2. Getriebestufenwechsel-Ende

Der Getriebestufenwechsel gilt als beendet (Spindelbetriebsart "Pendelbetrieb" wird abgewählt), wenn das folgende NC/PLC-Nahtstellensignal gesetzt wird:

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

Im Motorstillstand wird auf den Servo- und Interpolationsparametersatz der neuen Istgetriebestufe umgeschaltet.

Mit dem NC/PLC-Nahtstellensignal:

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

wird der NC mitgeteilt, dass die neue Getriebestufe gültig ist, und der Pendelbetrieb beendet werden kann.

Das NC/PLC-Nahtstellensignal:

DB31, ... DBX82.3 (Getriebe umschalten)

wird durch den NC rückgesetzt,

worauf das PLC-Programm das NC/PLC-Nahtstellensignal:

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

rücksetzen muss.

Dabei ist es unwesentlich, ob das NC/PLC-Nahtstellensignal:

DB31, ... DBX18.5 (Pendeldrehzahl)

noch gesetzt ist.

Zur Anwahl des Parametersatzes ist die gemeldete Istgetriebestufe relevant, die der Sollgetriebestufe entsprechen sollte.

Ist dies nicht der Fall, wird bei:

MD11410 \$MN_SUPPRESS_ALARM_MASK, Bit 3 = 0

der Alarm 22010 gemeldet.

Nach der Quittierung des Getriebestufenwechsels durch die PLC mit dem NC/PLC-Nahtstellensignal:

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

befindet sich die Spindel im Drehzahlsteuerbetrieb (DB31, ... DBX84.7).

Weitere Hinweise zum Signalaustausch zwischen PLC und NC siehe Kapitel "A2: Diverse NC/PLC-Nahtstellensignale und Funktionen (Seite 39)".

Zweiter Getriebestufen-Datensatz

Der automatische Getriebestufenwechsel M40 kann durch einen zweiten projektierbaren Getriebestufen-Datensatz erweitert werden.

Der zweite Getriebestufen-Datensatz wird **ausschließlich** im Zusammenhang mit Gewindebohren ohne Ausgleichsfutter (G331, G332) verwendet, um eine effektive Anpassung von Spindeldrehzahl und Motormoment zu erreichen.

Die Aktivierung erfolgt, indem für die Masterspindel folgendes Bit gesetzt wird:

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE, Bit 5 = 1

Die Anzahl der verwendeten Getriebestufen des zweiten Getriebestufen-Datensatzes wird festgelegt mit dem Maschinendatum:

MD35092 \$MA_NUM_GEAR_STEPS2

Der zweite Getriebestufen-Datensatz wird deaktiviert, wenn:

MD35092 \$MA_NUM_GEAR_STEPS2 = 0 (Grundstellung)

Bei aktivem M40 erfolgt dann die Auswahl der Getriebestufe durch den ersten Getriebestufen-Datensatz.

Hinweis

Die Anzahl der Getriebestufen im zweiten Datensatz können sich vom ersten unterscheiden. Wird für eine programmierte Drehzahl bei M40 keine passende Getriebestufe gefunden, dann wird auch kein Getriebestufenwechsel ausgeführt (Ausnahmen siehe "M40: Automatische Getriebestufenauswahl bei Drehzahlen außerhalb projektierter Schaltschwellen (Seite 1399)").

Weitere Informationen zu einem typischen Programmablauf beim Gewindeschneiden ohne Ausgleichsfutter G331/G332 siehe:

Literatur:

Programmierhandbuch Grundlagen; Wegbefehle

Die Getriebestufen 1 bis 5 des zweiten Getriebestufen-Datensatzes sind parametrierbar über folgende Maschinendaten:

Maschinendatum	Bedeutung
MD35112 \$MA_GEAR_STEP_MAX_VELO2[n]	Maximaldrehzahl für den automatischen Getriebestufenwechsel
MD35122 \$MA_GEAR_STEP_MIN_VELO2[n]	Minimaldrehzahl für den automatischen Getriebestufenwechsel
MD35212 \$MA_GEAR_STEP_POSCTRL_ACCEL2[n]	Beschleunigung im Lageregelbetrieb

Hinweis

Die Anzahl der die mechanischen Gegebenheiten betreffenden Servo-Parametersätze bleiben unverändert. Weiterhin können fünf mechanische Getriebestufen für den Spindel- und eine für den Achsbetrieb projektiert werden.

Die Drehzahlbegrenzungen werden unabhängig von den verschiedenen Schaltschwellen nur einmal pro Getriebestufe mit folgenden Maschinendaten projiziert:

Maschinendatum	Bedeutung
MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[n]	Maximaldrehzahl der Getriebestufe
MD35140 \$MA_GEAR_STEP_MIN_VELO_LIMIT[n]	Minimaldrehzahl der Getriebestufe

Für das Gewindebohren ohne Ausgleichsfutter (G331, G332) kann die Drehzahl zusätzlich auf den linearen Beschleunigungsbereich des Motors begrenzt werden. Dazu wird im folgenden Maschinendatum die maximale Drehzahl des linearen Motorkennlinienbereichs getriebestufenabhängig angegeben:

MD35550 \$MA_DRILL_VELO_LIMIT[n]

Vorgabe einer Getriebestufe im Teileprogramm

Automatische Auswahl bei aktiven M40

Die Getriebestufe wird durch die Steuerung automatisch ausgewählt. Dabei wird kontrolliert, in welcher Getriebestufe die programmierte Spindeldrehzahl (S . . .) möglich ist. Wenn daraus eine Getriebestufe resultiert, die ungleich der aktuellen (Ist-)Getriebestufe ist, dann werden folgende NC/PLC-Nahtstellensignale gesetzt:

DB31, ... DBX82.3 (Getriebe umschalten)

DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)

Bei der Ermittlung der passenden Getriebestufe wird nur dann ein Getriebestufenwechsel angefordert, wenn die neue Drehzahl nicht im zulässigen Drehzahlbereich der aktiven Getriebestufe liegt.

Die Drehzahl wird gegebenenfalls auf die Maximaldrehzahl der aktuellen Getriebestufe begrenzt bzw. auf die Minimaldrehzahl der aktuellen Getriebestufe angehoben, und es wird das entsprechende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX83.1 (Soll-Drehzahl begrenzt)

DB31, ... DBX83.2 (Soll-Drehzahl erhöht)

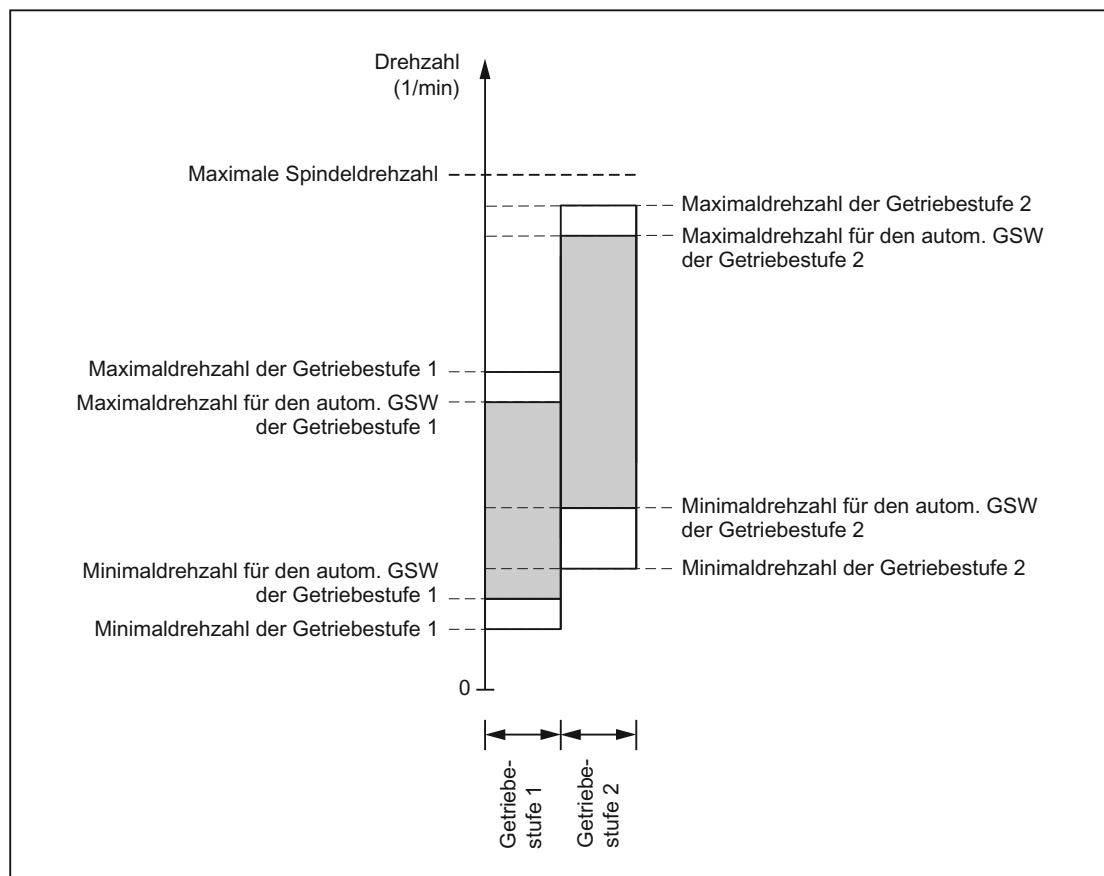


Bild 17-4 Beispiel für zwei Getriebestufen mit überlappenden Drehzahlbereichen bei automatischer Getriebestufenauswahl (M40)

Hinweis

Bei M40 muss sich die Spindel zur automatischen Getriebestufenauswahl bei einem S-Wort im Steuerbetrieb befinden. Andernfalls wird der Getriebestufenwechsel abgewiesen und folgender Alarm gesetzt:

Alarm 22000 "Kein Getriebestufenwechsel möglich"

Hinweis

Ein aktives Vorsatzgetriebe wird bei der Auswahl für den automatischen Getriebestufenwechsel nicht berücksichtigt.

Getriebestufe fest vorgeben mit M41 bis M45

Die Getriebestufe kann im Teileprogramm mit M41 bis M45 fest vorgegeben werden.

Wenn durch M41 bis M45 eine Getriebestufe vorgegeben wird, die ungleich der aktuellen (Ist-)Getriebestufe ist, dann werden folgende NC/PLC-Nahtstellensignale gesetzt:

DB31, ... DBX82.3 (Getriebe umschalten)

DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)

Die programmierte Spindeldrehzahl (s . . .) bezieht sich dann auf diese fest vorgegebene Getriebestufe:

- Wenn eine Spindeldrehzahl programmiert wird, die über der Maximaldrehzahl der fest vorgegebenen Getriebestufe (MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT) liegt, dann wird auf diese Maximaldrehzahl begrenzt und das folgende NC/PLC-Nahtstellensignal gesetzt:
DB31, ... DBX83.1 (Soll-Drehzahl begrenzt)
- Wenn eine Spindeldrehzahl programmiert wird, die unter der Minimaldrehzahl der fest vorgegebenen Getriebestufe (MD35140 \$MA_GEAR_STEP_MIN_VELO_LIMIT) liegt, dann wird auf diese Minimaldrehzahl angehoben und das folgende NC/PLC-Nahtstellensignal gesetzt:
DB31, ... DBX83.2 (Soll-Drehzahl erhöht)

Satzwechsel

Bei der Programmierung des Getriebestufenwechsels im Teileprogramm bleibt der Getriebestufenwechselsatz solange aktiv, bis der Getriebestufenwechsel von PLC beendet wurde.

Dies entspricht der Wirkung, als wäre folgendes NC/PLC-Nahtstellensignal gesetzt:

DB21, ... DBX6.1 (Einlesesperre)

Vorgabe einer Getriebestufe von PLC mit FC18

Der Getriebestufenwechsel ist durch Vorgabe mit dem Funktionsbaustein FC18 während eines Teileprogramms, im Reset-Zustand oder auch bei allen Betriebsarten möglich.

Bei der Drehzahl- und Drehrichtungsvorgabe mit FC18 kann die NC aufgefordert werden, die zur Drehzahl passende Getriebestufe auszuwählen. Dies entspricht einem automatischen Getriebestufenwechsel bei M40.

Die Getriebestufe wird nicht umgeschaltet, wenn:

- die Spindel durch FC18 positioniert wird.
- die Spindel im Achsbetrieb verfahren wird.

Für weitere Hinweise zum Funktionsbaustein FC18 siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)".

Vorgabe einer Getriebestufe in Synchronaktionen

Der Getriebestufenwechsel kann von Synchronaktionen angefordert werden durch:

- DO M40 S . . .
Automatische Getriebestufenauswahl zur programmierten Drehzahl s . . .
- DO M41 . . . M45
Direkte Anwahl der Getriebestufe 1 ... 5
- DO M70
Bei MD35014 \$MA_GEAR_STEP_USED_IN_AXISMODE = 1 ... 5
(siehe "Projektierbare Getriebestufe bei M70 (Seite 1368)")

Die Getriebestufe wird **nicht** umgeschaltet, wenn:

- die Spindel durch Synchronaktionen positioniert wird.
- die Spindel im Achsbetrieb verfahren wird.

Hinweis

Weitere Einzelheiten siehe Absatz "Vorgabe einer Getriebestufe im Teileprogramm".

Ausnahme:

Durch die Vorgabe einer Getriebestufe in Synchronaktionen wird der Satzwechsel nicht beeinflusst.

Getriebestufe manuell vorgeben

Außerhalb eines laufenden Teileprogramms kann die Getriebestufe auch ohne Aufforderung durch die NC an der Maschine gewechselt werden. Dies ist z. B. der Fall, wenn eine Getriebestufe direkt von Hand ausgewechselt wird.

Zur Auswahl der passenden Parametersätze muss der NC die aktuelle Getriebestufe mitgeteilt werden. Dazu muss sich die Steuerung oder das Teileprogramm im Reset-Zustand befinden.

Randbedingungen

Die Übernahme der Getriebestufe in die NC erfolgt mit Änderung der NC/PLC-Nahtstellensignale:

DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)

Diese drei Bits müssen während des Betriebs stabil anstehen.

Die vollzogene Übernahme wird mit den NC/PLC-Nahtstellensignalen:

DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)

zum PLC quittiert.

Das NC/PLC-Nahtstellensignal:

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

muss nicht gesetzt werden.

Ist zu dem Zeitpunkt der Vorgabe einer neuen Getriebestufe von PLC mit DB31, ... DBX16.0-16.2 die Lageregelung aktiv, dann wird diese für die Dauer des Umschaltvorgangs ausgeschaltet.

NC-Stop während Getriebestufenwechsel

Ein Spindelhalt durch das NC/PLC-Nahtstellensignal:

DB21, ... DBX7.4 (NC-Stop)

ist nicht möglich, solange:

- die Spindel im Pendelbetrieb für Getriebestufenwechsel noch nicht vorliegt.
- das NC/PLC-Nahtstellensignal:
DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
noch nicht vorliegt.

Hinweis

Abbruchmöglichkeiten:

DB31, ... DBX2.2 (Restweglöschen/Spindelreset)

oder

DB31, ... DBX16.3 (Getriebe ist umgeschaltet)

mit entsprechender Rückmeldung der Istgetriebestufe:

DB31, ... DBX16.0-16.2 (Istgetriebestufe)

Spindelverhalten nach einem Getriebestufenwechsel

Wie sich die Spindel nach beendetem Getriebestufenwechsel verhält, hängt von folgenden Anfangsbedingungen ab:

- Befand sich die Spindel vor dem Getriebestufenwechsel im Stopp-Zustand (M5, FC18: "Stopp Spindel drehen"), im Positionier- oder Achsbetrieb, dann wird nach Getriebestufenwechsel-Ende M5 (Spindel-Stopp) aktiv.
- War vor dem Getriebestufenwechsel eine Drehrichtung (M3, M4, FC18: "Start Spindel drehen") programmiert, dann werden nach dem Getriebestufenwechsel die letzte Drehrichtung und Drehzahl wieder aktiv. Die Spindel dreht in der neuen Getriebestufe auf die letzte programmierte Spindeldrehzahl (S . . .) hoch.
- War vor dem Getriebestufenwechsel Lageregelung aktiv (SPCON), dann wird diese nach dem Getriebestufenwechsel wieder aktiviert.
Der nächste Satz im Teileprogramm kann zur Ausführung kommen.

Besonderheiten

Beim Getriebestufenwechsel sind folgende Punkte zu beachten:

- Der Getriebestufenwechsel wird durch die Anwahl des NC/PLC-Nahtstellensignals:
DB31, ... DBX20.1 (Hochlaufumschaltung U/F Betrieb)
nicht beendet.
Es wird Sollwert 0 ausgegeben.
Die Quittierung des Getriebestufenwechsels erfolgt wie üblich durch das NC/PLC-Nahtstellensignal:
DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
- Das Signal "Hochlaufgeber Schnellstopp" muss von der PLC zurückgesetzt werden, bevor der Getriebestufenwechsel von der PLC beendet wird.
- Der Ablauf des Getriebestufenwechsels wird bei NC-Reset ohne Alarmausgabe beendet.
Die in den NC/PLC-Nahtstellensignalen:
DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)
gemeldete Getriebestufe wird von der NC übernommen.

Stern- / Dreieck-Umschaltung mit FC17

Digitale Hauptspindelantriebe können mittels FC17 auch bei laufender Spindel von Stern nach Dreieck in beiden Richtungen umgeschaltet werden. Dieser automatische Umschaltvorgang wird durch eine definierte Umschaltlogik im FC17 gesteuert, die eine vom Anwender parametrierbare Umschaltzeit für die betreffende Spindel zur Verfügung stellt.

Für weitere Hinweise zum Funktionsbaustein FC17 siehe Kapitel "P3: PLC-Grundprogramm für SINUMERIK 840D sl (Seite 883)".

17.4.2 Spindelgetriebestufe 0

Technischer Hintergrund

Bei Maschinen mit umschaltbarem Spindellastgetriebe können Situationen auftreten, bei denen der Getriebezug zwischen Motor und Last (Werkstück/Werkzeug) unterbrochen ist. Dieser Zustand kann z. B. durch Betätigen von Reset oder Not-Halt während der Durchführung eines Getriebestufenwechsels oder bei der Erstinbetriebnahme während der Montage einer Maschine auftreten. Die Steuerung muss diesen Zustand des offenen Getriebezugs erkennen und die nächste Getriebestufenwechselanforderung muss unbedingt ausgeführt werden.

Funktion

Bei ausgerücktem Getriebe wird von der PLC über die Nahtstellensignalbits DB31, ... DBX16.0-2 (Istgetriebestufe A bis C) der binärcodierte Wert "0" (\triangleq Getriebestufe 0) an die NC übermittelt:

DB31, ... DBX16.0-2 = 0

Dieser Wert dient der Steuerung als Kennzeichen für den Zustand des offenen Getriebezugs.

Auswirkungen auf den Getriebestufenwechsel

Getriebestufenwechsel im Teileprogramm

Die von der PLC gemeldete Istgetriebestufe wird von der NC beim Start eines Teileprogramms gelesen. Wird für die Istgetriebestufe zu diesem Zeitpunkt der Wert "0" gelesen, dann wird der nächste Getriebestufenwechsel ausgeführt und der Getriebestufenwechseldialog mit der PLC wird durchgeführt. Wenn ein Wert größer "0" gelesen wird, dann erfolgt bereits bei der Programmierung ein Vergleich zwischen angeforderter und aktiver Getriebestufe. Sind beide Getriebestufen gleich, dann wird kein Getriebestufenwechsel durchgeführt und eine eventuell programmierte Bahnbewegung wird nicht unterbrochen.

Getriebestufenwechsel in Synchronaktionen, FC18 und DBB30

Die von der PLC gemeldete Istgetriebestufe wird von der NC beim Getriebestufenwechsel immer ausgewertet. Wird der Wert "0" von der NC gelesen, dann wird der Getriebestufenwechsel immer ausgeführt. Beim Lesen eines Werts größer "0" erfolgt ein Vergleich zwischen angeforderter und aktiver Getriebestufe. Nur bei Ungleichheit beider Werte wird der Getriebestufenwechsel mit der PLC durchgeführt und das NC/PLC-Nahtstellensignal DB31, ... DBX82.3 (Getriebe umschalten) wird ausgegeben.

Randbedingungen

- **Ausgabe von DB31, ... DBX16.0-2 = 0**
Bei ausgerücktem Getriebe muss von der PLC die Getriebestufe 0 in die NC/PLC-Nahtstelle DB31, ... DBX16.0-2 (Istgetriebestufe A bis C) eingetragen werden.
- **Freigabe des Getriebestufenwechsels**
Voraussetzung für einen Getriebestufenwechsel nach Lesen der Getriebestufe 0 ist die generelle Freigabe des Getriebestufenwechsels über die Maschinendaten:
MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE (Getriebestufenwechsel parametrieren)
MD35090 \$MA_NUM_GEAR_STEPS (Anzahl eingerichteter Getriebestufen)
MD35092 \$MA_NUM_GEAR_STEPS2 (2. Getriebestufendatensatz: Anzahl eingerichteter Getriebestufen), wenn MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE, Bit 5 = 1 (Gewindebohren ohne Ausgleichsfutter)
- **PLC-Anwenderprogramm/ POWER ON-ASUP**
Vom PLC-Anwenderprogramm oder POWER ON-ASUP ist sicherzustellen, dass bei ausgerücktem Getriebe (Getriebestufe 0) vor einer Spindelbewegung eine Getriebestufenwechselanforderung programmiert wird. Das kann z. B. mit M41 im ASUP erfolgen. Spindelbewegungen wie z. B. in JOG oder im Achsbetrieb erzeugen selbst keinen Getriebestufenwechsel.

Beispiel

Beispiel für den Ablauf zum Schalten in die erste Getriebestufe nach POWER ON

1. POWER ON.
2. Das PLC-Anwenderprogramm ermittelt im mechanischen Umfeld den Zustand "Getriebe ist ausgerückt".
3. Die PLC übermittelt den Zustand "Getriebe ist ausgerückt" an die NC durch Setzen von:
DB31, ... DBX16.0-2 = 0

4. Teileprogrammstart oder POWER ON-ASUP.
5. N05 (Teileprogramm s. u.) wird abgearbeitet:
Ein Getriebestufenwechsel nach Getriebestufe 1 wird durchgeführt.
Von der NC:
 - wird das folgende NC/PLC-Nahtstellensignal gesetzt:
DB31, ... DBX82.3 (Getriebe umschalten)
 - wird die Sollgetriebestufe 1 an die PLC gemeldet:
DB31, ... DBX82.0 = 1
DB31, ... DBX82.1 = 0
DB31, ... DBX82.2 = 0
6. Mechanischer Wechsel der Getriebestufe, Quittung
Ist die Getriebestufe eingelegt, dann wird von der PLC:
 - das folgende NC/PLC-Nahtstellensignal gesetzt:
DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
 - Istgetriebestufe 1 an die NC gemeldet:
DB31, ... DBX16.0 = 1
DB31, ... DBX16.1 = 0
DB31, ... DBX16.2 = 0
7. N80 wird abgearbeitet:
Es findet aufgrund der Optimierung der Getriebestufenwechselhäufigkeit im
Teileprogramm kein Getriebestufenwechsel statt.

Teileprogramm:

Programmcode	Kommentar
N05 M41	; Anwahl 1.Getriebestufe
...	
N80 M41	; Kein Getriebestufenwechsel, wenn 1.Getriebestufe eingelegt ist.

Projektierungsdaten für Spindel 1 (AX5):

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE[AX5] = 1 (Getriebestufenwechsel freigeben)

17.4.3 Ermittlung der Spindelgetriebestufe

Die aktuelle Getriebestufe einer Spindel ist über Systemvariablen lesbar:

- Für die Anzeige in der Bedienoberfläche, in Synchronaktionen oder mit Vorlaufstopp im Teileprogramm über die Systemvariablen:

\$VC_SGEAR[<n>]	<p>Aktuell eingelegte Spindelgetriebestufe</p> <p>\$VC_SGEAR liest die von der PLC gemeldete Ist-Getriebestufe.</p> <p>Wertebereich: 0 ... 5</p>
\$AC_SGEAR[<n>]	<p>Aktive Spindelgetriebestufe</p> <p>\$AC_SGEAR liest die Soll-Getriebestufe im Hauptlauf.</p> <p>Wertebereich: 1 ... 5</p> <p>Entsprechend dieser Getriebestufe wird der Datensatz für die Spindel aktiviert.</p>

Hinweis

Bei Satzsuchlauf kann sich die Ist-Getriebestufe (\$VC_SGEAR[<n>]) von der Soll-Getriebestufe (\$AC_SGEAR[<n>]) unterscheiden, da während des Satzsuchlaufs kein Getriebestufenwechsel stattfindet. Mit Hilfe von \$VC_SGEAR[<n>] und \$AC_SGEAR[<n>] kann also abgefragt werden, ob ein Getriebestufenwechsel nach einem Satzsuchlauf erfolgen soll.

- Ohne Vorlaufstopp im Teileprogramm über die Systemvariablen:

\$P_SGEAR[<n>]	<p>Soll-Getriebestufe</p> <p>\$P_SGEAR liest die im Teileprogramm programmierte (M41 ... M45), bei M40 ausgewählte oder bei M70 projektierte Getriebestufe.</p>
\$P_SEARCH_SGEAR[<n>]	<p>Satzsuchlauf: Getriebestufen-M-Funktion</p> <p>\$P_SEARCH_SGEAR enthält die bei Satzsuchlauf aufgesammelte zuletzt programmierte Getriebestufen-M-Funktion.</p>

17.4.4 Parametersatzanwahl beim Getriebestufenwechsel

Servo-Parametersätze

Die Servo-Parametersätze 1 bis 6 dienen zur Anpassung des Lagereglers an veränderte Eigenschaften der Maschine bei einer Getriebeumschaltung der Spindel.

Parametersatzanwahl beim Getriebestufenwechsel

Beim Getriebestufenwechsel werden der Getriebestufen-Parametersatz (Interpolationsparameter) und, abhängig von der Einstellung im folgenden Maschinendatum, auch der Servo-Parametersatz umgeschaltet:

MD35590 \$MA_PARAMSET_CHANGE_ENABLE (Parametersatzwechsel möglich)

Wert	Bedeutung
0	Systeminterne Parametersatzanwahl Die Parametersätze des Servos sind fest zugeordnet. Dabei gilt: <ul style="list-style-type: none"> Bei Achsen und Spindeln im Achsbetrieb wirkt grundsätzlich der erste Parametersatz. Ausnahme: Bei G33, G34, G35, G331 und G332 wird für die beteiligten Achsen der Parametersatz mit folgender Nummer aktiviert: Masterspindel-Getriebestufe + 1 (entspricht Parametersatznummer 2 ... 6) Bei Spindeln im Spindelbetrieb wird der Parametersatz passend zur Getriebestufe eingestellt.
1	Außer der systeminternen Parametersatzanwahl besteht auch die Möglichkeit einer "externen" Parametersatzanwahl: <ul style="list-style-type: none"> durch die PLC (DB31, ... DBX 9.0 - 9.2) durch Programmierung von <code>SCPARA</code> im Teileprogramm oder in Synchronaktionen Die systeminterne Parametersatzanwahl hat jedoch Vorrang. Hinweis: Der Wert 1 ist nur für Achsen sinnvoll.
2	Der Servo-Parametersatz wird ausschließlich durch die PLC (DB31, ... DBX 9.0 - 9.2) oder durch Programmierung von <code>SCPARA</code> im Teileprogramm oder in Synchronaktionen vorgegeben (für Achsen und Spindeln). Nach POWER ON ist der 1. Parametersatz angewählt.

Spindelbetrieb

MD35590 \$MA_PARAMSET_CHANGE_ENABLE = 0 oder 1

Der Parametersatz wird entsprechend der Getriebestufe + 1 angewählt.

Die aktive Getriebestufe steht in:

DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)

Der aktive Parametersatz wird ausgegeben in:

DB31, ... DBX69.0-69.2 (Regler-Parametersatz A bis C)

Für jede der 5 Getriebestufen gibt es einen Parametersatz, der von der NC folgendermaßen zugeordnet ist:

Datensatz für ...	NC/PLC-Nahtstelle DBX 69.2 / 69.1 / 69.0	Parametersatz Nummer	Parametersatz Index [n]
Achsbetrieb	letzte aktive Getriebestufe	1	0
Getriebestufe 1	001	2	1
Getriebestufe 2	010	3	2

Datensatz für ...	NC/PLC-Nahtstelle DBX 69.2 / 69.1 / 69.0	Parametersatz Nummer	Parametersatz Index [n]
Getriebestufe 3	011	4	3
Getriebestufe 4	100	5	4
Getriebestufe 5	101 110 111	6	5

Spindel im Achsbetrieb

Beindet sich die Spindel im Achsbetrieb, dann wird im Servo der Parametersatz-Index "0" angewählt (MD35590 \$MA_PARAMSET_CHANGE_ENABLE beachten!).

Das Getriebestufenwechsel-Verhalten ist abhängig von der Einstellung im Maschinendatum:

MD35014 \$MA_GEAR_STEP_USED_IN_AXISMODE (Getriebestufe für den Achsbetrieb bei M70)

Wenn keine Getriebestufe für den Achsbetrieb projektiert ist (MD35014 = 0), dann findet kein impliziter Getriebestufenwechsel bei M70 statt (Standardeinstellung!). Die letzte Getriebestufe wird intern abgespeichert und wird mit dem dazugehörenden Parametersatz mit der nächsten Spindelprogrammierung wieder aktiv.

Ist dagegen eine Getriebestufe für den Achsbetrieb projektiert (MD35014 = 1 ... 5), dann findet ein Getriebestufenwechsel in die Getriebestufe 1 ... 5 während der Abarbeitung von M70 statt. Beim Wechsel vom Achsbetrieb in den Spindelbetrieb bleibt die mit M70 eingewechselte Getriebestufe weiterhin aktiv. Die im Spindelbetrieb vor M70 aktive Getriebestufe wird nicht automatisch wieder eingewechselt.

Siehe auch "Projektierbare Getriebestufe bei M70 (Seite 1368)".

Lastgetriebeübersetzung

Es ist möglich, für jede Getriebestufe und im Achsbetrieb positive oder negative **Lastgetriebefaktoren** zu projektieren.

Die Einstellung erfolgt getrennt für Zähler und Nenner über die Maschinendaten:

MD31050 \$MA_DRIVE_AX_RATIO_DENOM[n] (Nenner Lastgetriebe)

MD31060 \$MA_DRIVE_AX_RATIO_NUMERA[n] (Zähler Lastgetriebe)

Der Einstellbereich ist für positive und negative Lastgetriebefaktoren gleich groß.

Die Eingabe des Werts "0" ist nicht möglich.

Hinweis

Wenn ein indirekter Geber projektiert ist und sich die Lastgetriebeübersetzung ändert, dann geht die Referenz verloren und das NC/PLC-Nahtstellensignal: DB31, ... DBX60.4/60.5 (Referenziert / Synchronisiert 1 bzw. 2) wird für das betreffende Messsystem zurückgesetzt.

Literatur

Weitere Informationen zur Regelung und zum Servo-Parametersatz siehe:

- Funktionshandbuch Grundfunktionen; Geschwindigkeiten, Soll-/Istwertsystem, Regelung (G2)
- Programmierhandbuch Arbeitsvorbereitung; Kap.: Programmierbarer Servo-Parametersatz

17.4.5 Vorsatz-Getriebe

Anwendung und Funktionalität

Mit einem projektierbaren Vorsatz-Getriebe können verschiedene angetriebene Werkzeuge angepasst werden. Das an der Werkzeugseite befindliche Vorsatz-Getriebe wirkt multiplikativ zum Motor-/Last-Getriebe.

Es wird eingestellt über die Maschinendaten:

MD31066 \$MA_DRIVE_AX_RATIO2_NUMERA (Zähler Vorsatzgetriebe)

MD31064 \$MA_DRIVE_AX_RATIO2_DENOM (Nenner Vorsatzgetriebe)

Ein für das Vorsatz-Getriebe vorhandener werkzeugseitiger Geber wird mit dem Maschinendatum:

MD31044 \$MA_ENC_IS_DIRECT2 (Geber am Vorsatzgetriebe) projektiert.

Eine geänderte Parametrierung dieser Maschinendaten kann mit der Funktion "Maschinendaten wirksam setzen" entweder mit Hilfe der Inbetriebnahme-Software SinuCOM-NC oder über die Bedientafel HMI über einen Softkey aktiviert werden. Die bereits existierenden Motor-/Last-Getriebe sind dagegen über Power-On wirksam.

Werkzeugwechsel

Wird beim Werkzeugwechsel das Vorsatzgetriebe mit gewechselt, dann muss vom Anwender das Übersetzungsverhältnis vom Zähler und Nenner über die Maschinendaten des Vorsatz-Getriebes auch umprojektiert werden.

Beispiel:

 VORSICHT
Projektierungsfehler
Es bleibt Aufgabe des Anwenders, in der betreffenden Zeitspanne anzuhalten, um in unmittelbarem zeitlichem Zusammenhang eine Änderung der Maschinendaten vorzunehmen und danach die Funktion "Maschinendaten wirksam setzen" zu aktivieren.

Für ein eingewechseltes Werkzeug mit einem Übersetzungsverhältnis von 2:1 wird ein passendes Vorsatzgetriebe projiziert und im Teileprogramm mit dem Befehl `NEWCONF` sofort wirksam.

Programmcode
N05 \$MA_DRIVE_AX_RATIO2-NUMERA[AX5] = 2
M10 \$MA_DRIVE_AX_RATIO2-DENOM[AX5] = 1
N15 NEWCONF

Umschaltung

Die Umschaltung auf die neue Übersetzung wird mit Hilfe der Funktion "Maschinendaten wirksam setzen" sofort durchgeführt. Technologisch hat primär der zugehörige mechanische Umschaltvorgang eine zeitliche Ausdehnung, da mechanisch ein anderer Vorsatzkopf mit angetriebenen Werkzeug eingewechselt wird.

Hinweis

Im Stillstand ist die Umschaltung ruckfrei. Es liegt daher in der Verantwortung des Anwenders hierfür entsprechende Vorsorgemaßnahmen zu ergreifen.

Anwendungen bei denen in Bewegung umgeschaltet wird und die einen geglätteten oder weichen Drehzahl-Übergang benötigen, sind mit existierenden antriebsseitigen Drehzahl-sollwert-Filtern beherrschbar.

Weitere Erläuterungen hinsichtlich regelungstechnischer Abhängigkeiten siehe Kapitel "G2: Geschwindigkeiten, Soll-/Istwertsysteme, Regelung (Seite 341)".

17.4.6 Nicht quittierter Getriebestufenwechsel

Betriebsartenwechsel

Ein nicht quittierter Getriebestufenwechsel kann nicht durch einen Betriebsartenwechsel (z.B. Umschaltung nach JOG) unterbrochen werden.

Die Umschaltung wird maximal um die im Maschinendatum:
MD10192 \$MN_GEAR_CHANGE_WAIT_TIME
eingetragene Zeitspanne verzögert.

Wird der Getriebestufenwechsel nicht innerhalb dieser Zeitdauer quittiert, so löst die NC einen Alarm aus.

Weitere Ereignisse

Ereignisse, die ein Reorganisieren auslösen, warten ebenso das Ende eines Getriebestufenwechsels ab.

Die im Maschinendatum:
MD10192 \$MN_GEAR_CHANGE_WAIT_TIME
eingestellte Zeit bestimmt nun wie lange auf den Getriebestufenwechsel gewartet wird.

Läuft diese Zeit ab, ohne dass ein Getriebestufenwechsel beendet wurde, so reagiert die NC mit einem Alarm.

Folgende Ereignisse besitzen analoges Verhalten:

- Anwender-ASUP
- Mode-Wechsel
- Restweg löschen
- Achstausch
- PI-Anwenderdaten wirksam setzen
- PI-Dienst Maschinendaten wirksam setzen
- Ausblenden umschalten, DryRun umschalten
- Editieren in der Betriebsart
- Korrektursatzalarme
- Overstore
- Schnellrückzug bei G33, G34, G35
- Unterprogrammebenenabbruch, Unterprogrammabbruch

Verhalten nach Power On

Nach Power On und im RESET-Zustand des NC kann die an der Maschine aktive Getriebestufe von der PLC Steuerung vorgegeben werden.

Die NC wählt daraufhin den entsprechenden Parametersatz an und meldet die NC/PLC-Nahtstellensignale:
DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)
an die PLC zurück.

17.4.7 Getriebestufenwechsel mit Pendelbetrieb

Die Spindel befindet sich im Pendelbetrieb, wenn durch M40 (automatische Getriebestufenauswahl) oder M41 bis M45 eine neue Getriebestufe vorgegeben wurde.

Beim Pendeln wird die Drehrichtung des Spindelmotors abwechselnd in zeitlich kurzen Abständen umgekehrt. Durch die Pendelbewegung des Motors wird z. B. das Einrücken einer neuen Getriebestufe erleichtert. Prinzipiell kann auch ohne Pendelvorgang eine neue Getriebestufe eingelegt werden.

Folgende Pendelarten sind möglich:

- Pendeln durch die NC
- Pendeln durch die PLC
- Pendeln mit FC 18

Literatur:

Funktionshandbuch Grundfunktionen; PLC–Grundprogramm (P3)

Inbetriebnahme: Maschinendaten

- MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE (Getriebestufenwechsel parametrieren)
- MD35410 \$MA_SPIND_OSCILL_ACCEL (Beschleunigung beim Pendeln)
- MD35430 \$MA_SPIND_OSCILL_START_DIR (Startrichtung beim Pendeln)
- MD35400 \$MA_SPIND_OSCILL_DES_VELO (Pendeldrehzahl)
- MD35450 \$MA_SPIND_OSCILL_TIME_CCW (Pendelzeit für M4-Richtung)
- MD35440 \$MA_SPIND_OSCILL_TIME_CW (Pendelzeit für M3-Richtung)

Inbetriebnahme: NC/PLC-Nahtstellensignale

- DB31, ... DBX16.0 - 2 (Istgetriebestufe)
- DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
- DB31, ... DBX18.4 (Pendeln durch die PLC)
- DB31, ... DBX18.5 (Pendelfreigabe)
- DB31, ... DBX18.6 (Solldrehrichtung rechts)
- DB31, ... DBX18.7 (Solldrehrichtung links)
- DB31, ... DBX61.4 (Spindel steht)
- DB31, ... DBX82.0 - 2 (Sollgetriebestufe)
- DB31, ... DBX82.3 (Getriebe umschalten)
- DB31, ... DBX83.5 (Spindel im Sollbereich)
- DB31, ... DBX84.6 (Aktive Spindelbetriebsart: Pendelbetrieb)

Ablaufbeschreibung: Getriebestufenwechsel

Parametrierung: MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE, Bit 0 = 1

Ablauf:

- Abbremsen der Spindel.
Der Bremsvorgang entspricht dabei einer M5-Bewegung.
- NC/PLC-Nahtstellensignale NC → PLC:
 - DB31, ... DBX84.6 == 1 (Pendelbetrieb)
 - DB31, ... DBX82.3 == 1 (Getriebe umschalten)
 - DB31, ... DBX82.0 - 2 == <Sollgetriebestufe>
 - DB31, ... DBX61.5 == 0 (Lageregler aktiv)
- Das Lastgetriebe kann jetzt "ausrücken".
- Das Pendel kann jetzt von der NC oder der PLC (siehe unten) durchgeführt werden.
- Nach der Rückmeldung der PLC an die NC, dass das Getriebe umgeschaltet und die neue Getriebestufe aktiv ist, wird die Programmbearbeitung fortgesetzt:
- DB31, ... DBX16.3 = 1 (Getriebe ist umgeschaltet)

- DB31, ... DBX16.0 - 2 (Istgetriebestufe)
- Bei Messsystemen mit indirektem Geber (Motorgeber) wird der Referenziertstatus gelöscht:
DB31, ... DBX60.4 / 5 = 0

Pendeln durch die NC

Anforderung

- DB31, ... DBX18.5 = 1 (Pendelfreigabe)

Status-Rückmeldungen

- DB31, ... DBX84.6 == 1 (Pendelbetrieb)
- DB31, ... DBX82.3 == 1 (Getriebe umschalten)
- DB31, ... DBX82.0 - 2 == <Sollgetriebestufe>
- DB31, ... DBX61.5 == 0 (Lageregler aktiv)

Das Lastgetriebe kann jetzt "ausrücken".

Pendeln starten

Vom PLC-Anwenderprogramm kann jetzt die Pendelfreigabe gesetzt werden:

- DB31, ... DBX18.5 = 1 (Pendelfreigabe)

Der Motor beschleunigt mit der Pendelbeschleunigung in Startdrehrichtung für die drehrichtungsabhängige Pendelzeit auf die Pendeldrehzahl:

- MD35410 \$MA_SPIND_OSCILL_ACCEL (Beschleunigung beim Pendeln)
- MD35430 \$MA_SPIND_OSCILL_START_DIR (Startrichtung beim Pendeln)
- MD35400 \$MA_SPIND_OSCILL_DES_VELO (Pendeldrehzahl)
- MD35450 \$MA_SPIND_OSCILL_TIME_CCW (Pendelzeit für M4-Richtung)
- MD35440 \$MA_SPIND_OSCILL_TIME_CW (Pendelzeit für M3-Richtung)

Drehrichtung umschalten

Nach Ablauf der drehrichtungsabhängigen Pendelzeit wird der Motor mit der Pendelbeschleunigung abgebremst und für die jeweils andere Pendelzeit in entgegengesetzter Drehrichtung auf die Pendeldrehzahl beschleunigt.

Pendeln durch die PLC

Anforderung

- DB31, ... DBX18.4 = 1 (Pendeln durch die PLC)
- DB31, ... DBX18.5 = 1 (Pendelfreigabe)

Status-Rückmeldungen

- DB31, ... DBX84.6 == 1 (Pendelbetrieb)
- DB31, ... DBX82.3 == 1 (Getriebe umschalten)

- DB31, ... DBX82.0 - 2 == <Sollgetriebestufe>
- DB31, ... DBX61.5 == 0 (Lageregler aktiv)

Das Lastgetriebe kann jetzt "ausrücken".

Pendeln starten

Vom PLC-Anwenderprogramm kann jetzt die "Pendelfreigabe" gesetzt werden:

- DB31, ... DBX18.7 = 1 (Solldrehrichtung links) bzw. DB31, ... DBX18.6 = 1 (Solldrehrichtung rechts)

Der Motor beschleunigt mit der Pendelbeschleunigung in die vorgegebene Solldrehrichtung auf die Pendeldrehzahl:

- MD35410 \$MA_SPIND_OSCILL_ACCEL (Beschleunigung beim Pendeln)
- MD35400 \$MA_SPIND_OSCILL_DES_VELO (Pendeldrehzahl)

Drehrichtung umschalten

Die Pendelzeiten sowie die Umschaltung der Drehrichtung müssen im PLC-Anwenderprogramm realisiert werden.

Ende des Pendelbetriebes

Die Spindel befindet sich nach Beendigung des Pendelbetriebes wieder im Steuerbetrieb und geht automatisch in den durch `SPCON` bzw. `SPCOF` festgelegten Betrieb.

Alle getriebespezifischen Grenzwerte (min./max. Drehzahl der Getriebestufe, etc.) entsprechen den parametrisierten Werten der Istgetriebestufe.

Satzwechsel

Wurde die Spindel in den Pendelbetrieb geschaltet (DB31, ... DBX82.3 = 1 (Getriebe umschalten)), wird die Bearbeitung des aktiven NC-Programms angehalten.

Ist der Getriebewechsel abgeschlossen (DB31, ... DBX16.3 == 1 (Getriebe ist umgeschaltet)), wird die Bearbeitung des NC-Programms fortgesetzt.

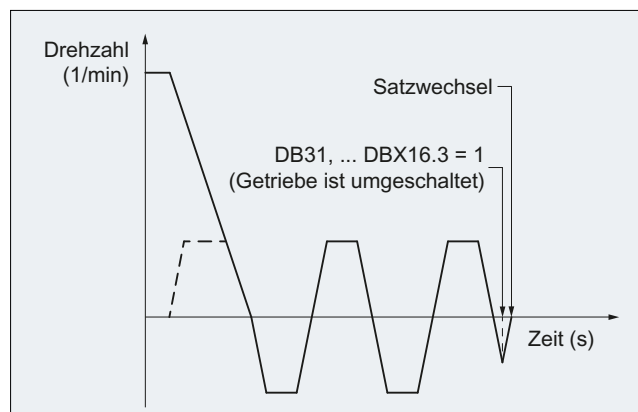


Bild 17-5 Satzwechsel nach Pendelbetrieb

- t_2 : Die NC meldet **Spindeldrehzahl == 0** an die PLC:
- DB31, ... DBX61.4 = 1 (Spindel steht)
- Das PLC-Anwenderprogramm erteilt die Pendelfreigabe.
- PLC → NC: DB31, ... DBX18.5 = 1 (Pendelfreigabe)
- Hinweis**
Die Pendelfreigabe muss spätestens zum Zeitpunkt t_2 gesetzt sein.
- Die NC beginnt mit dem Pendeln.
- t_3 : Der Getriebestufenwechsel wurde an der Maschine durchgeführt.
- Das PLC-Anwenderprogramm meldet die Umschaltung an die NC zurück und löscht die Pendelfreigabe:
- PLC → NC: DB31, ... DBX16.0 - 2 = 2 (Istgetriebestufe)
 - PLC → NC: DB31, ... DBX16.3 = 1 (Getriebe ist umgeschaltet)
 - PLC → NC: DB31, ... DBX18.5 = 0 (Pendelfreigabe)
- t_4 : Die NC beendet den Getriebestufenwechsel durch Rücksetzen der Anforderung an die PLC und das Pendeln durch Umschalten der Spindelbetriebsart von Pendelbetrieb nach Steuerbetrieb::
- NC → PLC: DB31, ... DBX82.3 = 0 (Getriebe umschalten)
 - DB31, ... DBX84.6 = 0 (Aktive Spindelbetriebsart: Pendelbetrieb)
 - DB31, ... DBX84.7 = 1 (Aktive Spindelbetriebsart: Steuerbetrieb)
- Die NC gibt den nächsten Satz zur Bearbeitung frei und beschleunigt die Spindel auf den neuen S-Wert S1300.
- *: Die interne Vorschubsperrung wird gesetzt, wenn:
- Getriebestufenwechsel aufgrund der Programmierung des S-Wertes im NC-Programm **UND**
 - Ein **Bearbeitungssatz** ist aktiv (nicht Eilgang G0)
- Hinweis**
Bei einem Getriebestufenwechsel aus Synchronaktionen oder bei Vorgaben über die PLC mit FC 18 wird die interne Vorschubsperrung **nicht** gesetzt.

Bild 17-6 Getriebestufenwechsel bei stehender Spindel

Randbedingungen

- Zum Abbremsen der Spindel muss das NC/PLC-Nahtstellensignal nicht gesetzt werden: DB31, ... DBX4.3 (Spindel-Halt)
Die Spindel wird steuerungsintern stillgesetzt, wenn ein Getriebestufenwechsel angefordert wird.
- Der Getriebestufenwechsel ist immer mit dem Setzen des NC/PLC-Nahtstellensignals zu beenden:
DB31, ... DBX16.3 = 1 (Getriebe ist umgeschaltet)

- Mit dem Rücksetzen des NC/PLC-Nahtstellensignals wird die Pendelbewegung in der NC gestoppt:
DB31, ... DBX18.5 (Pendelfreigabe)
Die Spindel bleibt aber in der Betriebsart "Pendelbetrieb".
- Bei Getriebeumschaltung und indirektem Messsystem (MD31050 \$MA_ENC_IS_DIRECT = 0 Motorgeber) geht die Synchronisation der Spindel verloren.
Getriebeumschaltung \Rightarrow DB31, ... DBX60.4 / 5 = 0 (Referenziert/synchronisiert)
Beim nächsten Überfahren der Nullmarke nach der Getriebeumschaltung wird die Spindel wieder automatisch synchronisiert.

17.4.8 Getriebestufenwechsel auf Festposition

Anwendung und Vorteile

An Werkzeugmaschinen werden zunehmend standardisierte Spindelgetriebe eingesetzt, um einerseits technologische Totzeit beim Getriebestufenwechsel zu sparen und um andererseits den Kostenvorteil aus der Verwendung standardisierter mechanischer Bauteile zu nutzen.

Die Funktion Getriebestufenwechsel auf Festposition unterstützt den "gerichteten Getriebestufenwechsel" von Lastgetrieben, welche von der NC eine andere Ansteuerung erfordern. Der Getriebestufenwechsel kann dabei nur auf einer definierten Spindelposition ausgeführt werden. Eine Pendelbewegung, wie bei herkömmlichen Lastgetrieben, wird nicht mehr benötigt.

Ablauf zum GSW auf Festposition

Der Getriebestufenwechsel auf Festposition

Die Maschinendatum-Projektierung:

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE = 2

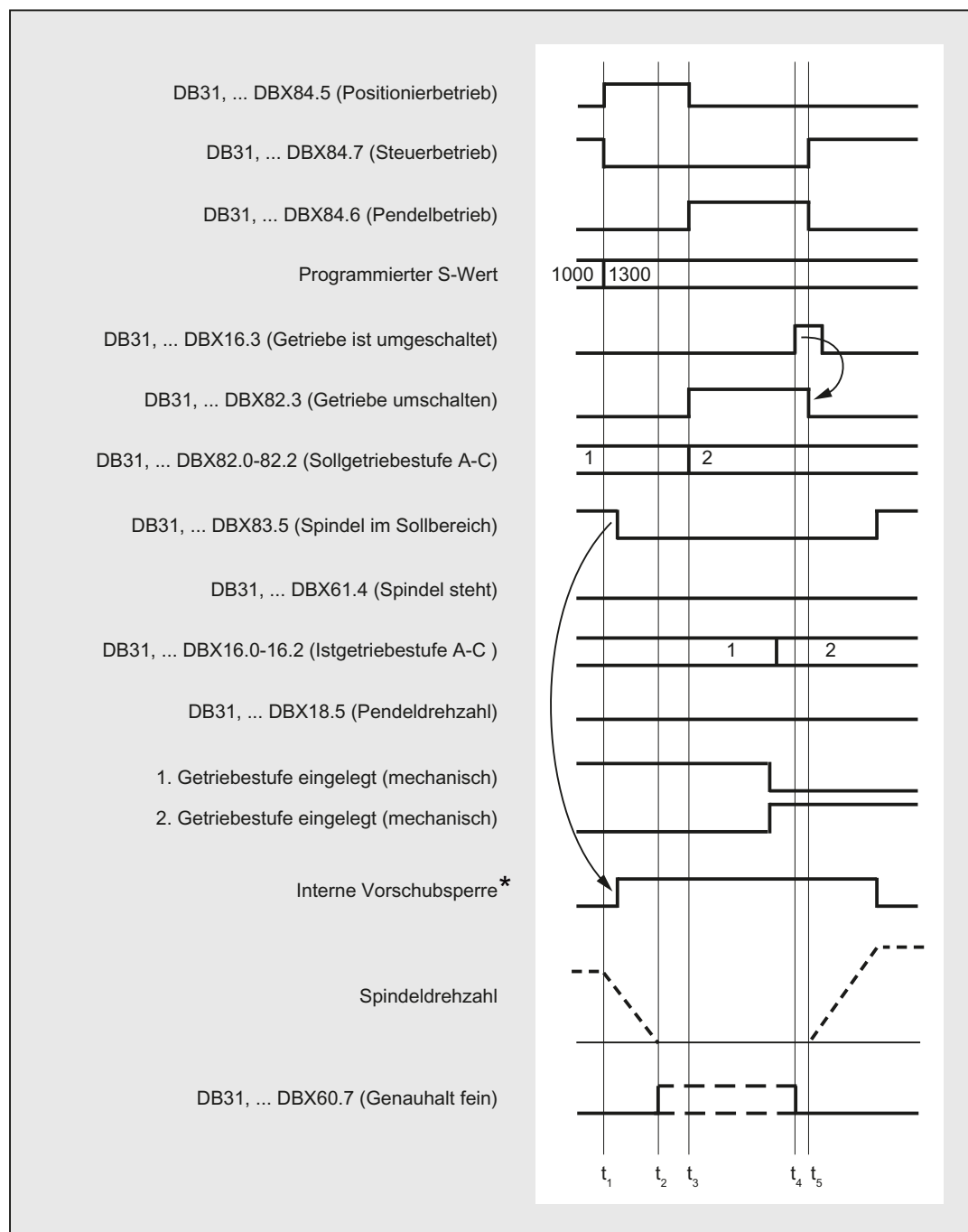
führt folgenden Ablauf aus:

- Positionierung der Spindel aus dem Stillstand oder einer Bewegung auf die in Maschinendatum:
MD35012 \$MA_GEAR_STEP_CHANGE_POSITIONprojektierte Position.
Erfolgt der Getriebestufenwechsel aus der Bewegung, dann wird die aktuelle Drehrichtung beibehalten. Die Spindel befindet sich während des Positioniervorganges im Positionierbetrieb.
Es wird das NC/PLC-Nahtstellensignal:
DB31, ... DBX84.5 (Positionierbetrieb)
ausgegeben.
Ist keine Referenz vorhanden:
DB31, ... DBX60.4/5 = 0
oder das NC/PLC-Nahtstellensignal:
DB 31, ... DBX17.4/5 (Beim Positionieren neu Synchronisieren MS 1/2)
gesetzt, dann verlängert sich der Positioniervorgang um die Nullmarkensuche.
- Nach dem Erreichen der im Maschinendatum:
MD35012\$MA_GEAR_STEP_CHANGE_POSITION
projektierten GSW-Position wird die Zeit aus Maschinendatum:
MD35310 \$MA_SPIND_POSIT_DELAY_TIME
abgewartet, bevor auf Pendelbetrieb umgeschaltet wird,
und der bekannte GSW-Dialog beginnt.
- Ausgabe der NC/PLC-Nahtstellensignale:
DB31, ... DBX84.6 (Pendelbetrieb)
DB31, ... DBX82.3 (Getriebe umschalten)
DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)
- Die Lageregelung wird bei aktivem Messsystem mit indirektem Geber (Motorgeber) nicht ausgeschaltet:
MD31040 \$MA_ENC_IS_DIRECT = 0
Bei aktivem Messsystem mit direktem Geber (Lastgeber) wird die Lageregelung ausgeschaltet:
DB31, ... DBX61.5 = 0,
da der Kraftfluss zur Last unterbrochen wird und eine Positionsregelung dadurch nicht mehr möglich ist.
- Ist der lageregelte Betrieb nicht möglich, dann kann dieser durch
Rücksetzen der "Reglerfreigabe":
DB31, ... DBX2.1 = 0
abgeschaltet werden.
- Mechanisches Umschalten der Getriebestufe an der Maschine.
Dazu ist keine Pendelbewegung des Motors erforderlich.
Die NC/PLC-Nahtstellensignale:
DB31, ... DBX18.5 (Pendelfreigabe)
und
DB31, ... DBX18.4 (Pendeln durch die PLC)
sollen **nicht** gesetzt werden.
Die Pendelbewegung ist jetzt noch prinzipiell möglich.

- Beschreiben der NC/PLC-Nahtstellensignale:
DB31, ... DBX16.0-16.2 (Istgetriebestufe A bis C)
durch die PLC.
- Nach der Meldung:
DB31, ... DBX16.3 (Getriebe ist umgeschaltet)
wird die zuletzt aktive Bewegung, falls vorhanden, fortgesetzt.
Für indirekte Geber (Motorgeber) wird der Referenzierstatus gelöscht:
DB31, ... DBX60.4/5 = 0.
Die Spindel befindet sich im Drehzahlsteuerbetrieb und das NC/PLC-Nahtstellensignal:
DB31, ... DBX84.7 (Steuerbetrieb)
wird ausgegeben.

GSW Festposition

Typischer zeitlicher Ablauf für den Getriebestufenwechsel auf Festposition:



t_1 : NC erkennt durch Programmierung von S1300 eine neue Getriebebestufe (2. Getriebebestufe), setzt das NST DB31, ... DBX84.5 (Positionierbetrieb) und sperrt die Bearbeitung für den nächsten Teileprogrammsatz (= interne Vorschubsperr*).

t_2 : Die Spindel steht und es wird Genauhalt gemeldet.

t_3 : Getriebestufenwechsel - Wartezeit

- t₄: Die neue Getriebestufe ist eingelegt. Der PLC-Anwender übergibt die neue (Ist-)Getriebestufe an die NC und setzt das NST DB31, ... DBX16.3 (Getriebe ist umgeschaltet).
- t₅: Die NC nimmt daraufhin das NST DB31, ... DBX82.3 (Getriebe umschalten) zurück, gibt den nächsten Teileprogrammsatz zur Bearbeitung frei und beschleunigt die Spindel auf den neuen S-Wert (S1300).
- *: Die interne Vorschubsperrung wird gesetzt, wenn:
- der Spindel-Getriebestufenwechsel über das Teileprogramm programmiert wurde **und**
 - ein Bearbeitungssatz aktiv ist (d. h. G0 ist nicht aktiv).
- Bei einem Getriebestufenwechsel aus Synchronaktionen oder bei Vorgaben über die PLC mit FC 18 wird die interne Vorschubsperrung **nicht** gesetzt.

Bild 17-7 Getriebestufenwechsel bei stehender Spindel

GSW-Position MD35012

Die Getriebestufenwechselposition wird im Maschinendatum:
MD35012 \$MA_GEAR_STEP_CHANGE_POSITION
für jede Getriebestufe projiziert.

GSW-Wartezeit MD35310

Nach dem Positioniervorgang wird um die im Maschinendatum:
MD35310 \$MA_SPIND_POSIT_DELAY_TIME
projizierte Zeit gewartet, bis die Getriebeumschaltanforderung:
DB31, ... DBX84.6 (Pendelbetrieb)
DB31, ... DBX82.3 (Getriebe umschalten)
und
DB31, ... DBX82.0-82.2 (Sollgetriebestufe A bis C)
ausgegeben wird.

Positionierkennwerte / Position

Die Position wird immer auf kürzesten Weg (entspricht DC) angefahren.

Ist keine Referenz vorhanden und befindet sich die Spindel im Stillstand (z.B. nach Power On), dann wird die Anfahrrichtung durch das folgende Maschinendatum bestimmt:
MD35350 \$MA_SPIND_POSITIONING_DIR

Wird eine veränderliche GSW-Position benötigt, dann kann dies durch Beschreiben des Maschinendatums mit anschließendem "Maschinendaten wirksam setzen" erreicht werden. Die Änderung des MD-Wertes kann über das Teileprogramm oder HMI erfolgen.

Sollte die projizierte Position nicht erreicht werden, dann wird der Alarm 22020 gemeldet und der GSW-Dialog zwischen NC und PLC findet in diesem Fall nicht statt, um das Getriebe nicht zu zerstören. Da der Alarm schwerwiegend ist, kann das Teileprogramm nicht fortgesetzt werden und die Ursache muss unbedingt beseitigt werden. Die Ursache für den Abbruch der Positionierung dürfte erfahrungsgemäß auf falsche MD-Einstellungen oder unverträgliche PLC-Signale zurückführbar sein.

Geschwindigkeit

Die Positioniergeschwindigkeit wird entnommen aus dem getriebestufenabhängig projektierten Maschinendatum:

MD35300 \$MA_SPIND_POSCTRL_VELO

Die NC/PLC-Nahtstellensignale "Spindel-Override"/"Vorschub-Override" bei:

DB31, ... DBX17.0=0: DB31, ... DBB19

sowie bei:

DB31, ... DBX17.0=1: DB31, ... DBB0

sind wie beim Positionieren üblich wirksam.

Die Positioniergeschwindigkeit kann durch die Programmanweisung `OVRA [Sn]` prozentual verändert werden.

Hinweis

`OVRA [Sn]` ist modal gültig. Nach dem Getriebestufenwechsel sollte wieder ein für die Bearbeitung sinnvoller Wert eingestellt werden.

Die Teileprogrammanweisung `FA [Sn]` ändert die Positioniergeschwindigkeit beim Getriebestufenwechsel nicht.

Beschleunigung

Die Beschleunigungswerte werden bestimmt durch die getriebestufenabhängig projektierten Maschinendaten:

MD35200 \$MA_GEAR_STEP_SPEEDCTRL_ACCEL

und

MD35210 \$MA_GEAR_STEP_POSCTRL_ACCEL

Die Beschleunigung kann durch Programmierung von `ACC [Sn]` prozentual verändert werden.

Hinweis

`ACC [Sn]` ist modal gültig. Nach dem Getriebestufenwechsel sollte wieder ein für die Bearbeitung sinnvoller Wert eingestellt werden.

Drehzahlabhängige Beschleunigung

Die "geknickte Beschleunigungskennlinie" ist wie beim Positionieren mit `SPOS` oder `FC18` wirksam.

Ruck

Gegenwärtig ist die Begrenzung der Beschleunigungsänderung nicht möglich.

Positionierende

Der Übergang zwischen dem Ende des Positioniervorganges (DB31, ... DBX84.5) und dem Beginn des Pendelbetriebes (DB31, ... DBX84.6) wird bestimmt durch das Erreichen von "Genauhalt fein" (DB31, ... DB60.7) und der Größe der in Maschinendatum: MD3510 \$MA_SPIND_POSIT_DELAY_TIME eingetragenen Zeit.

Die Festlegung der Übergangsbedingung wirkt sich einerseits auf die GSW-Zeit und andererseits auf die Genauigkeit des Erreichens der projektierten GSW-Position aus.

Satzwechsel

Der Satzwechsel wird angehalten und die Bearbeitungssätze werden bis zur Beendigung des Getriebestufenwechsel durch die PLC (DB31, ... DBX16.3) nicht begonnen.

GSW-Ende

Die Spindel befindet sich nach Beendigung des Getriebestufenwechsels wieder im Steuerbetrieb und geht automatisch in den durch `SPCON` bzw. `SPCOF` festgelegten Reglermodus.

Alle getriebespezifischen Grenzwerte (min./max. Drehzahl der Getriebestufe, etc.) entsprechen der zurückgemeldeten Istgetriebestufe.

Randbedingungen

- Die Spindel muss mindestens ein Messsystem besitzen.
- Der lagegeregelter Betrieb muss möglich und in Betrieb genommen worden sein.
- Grundsätzlich muss `SPOS` aus dem Teileprogramm, Synchronaktion oder über FC18: "Start Spindel positionieren" fehlerfrei ausführbar sein.

Können nicht alle Voraussetzungen erfüllt werden, dann kann die beschriebene Funktion nicht erfolgreich angewendet werden.

Aktivierung

Die Funktion des Getriebestufenwechsel auf Festposition wird durch die Projektierung aktiviert:
MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE = 2

17.4.9 Projektierbare Getriebestufe bei M70

Technischer Hintergrund

Bei manchen Maschinen ist es erforderlich, dass sich die Spindel während des Achsbetriebs in einer bestimmten Getriebestufe befindet.

Gründe dafür können sein:

- Im Servo-Parametersatz für den Achsbetrieb (Index 0) kann nur eine Optimierung (K_v , Vorsteuerung, Filter) passend zu einer Getriebestufe gefunden werden. Die Maschinendaten dieses Parametersatzes sollen nicht umgeschrieben werden.
- Es gibt nur eine mechanische Getriebeübersetzung, die im Gegensatz zu den restlichen eine geringe oder keine Umkehrlose besitzt. Nur in dieser Getriebestufe kann die Spindel zusammen mit anderen Achsen einer Bahnbewegung oder Transformationen (z. B. TRANSMIT) folgen.

Funktion

Bei aktiver Funktion kommt es beim Übergang in den Achsbetrieb zum automatischen Einwechseln einer vorbestimmten Getriebestufe.

Der Getriebestufenwechsel ist in den M70-Ablauf integriert und findet nach dem Abbremsen der Spindel und vor dem Einwechseln des Servo-Parametersatzes mit Index 0 statt (MD35590 \$MA_PARAMSET_CHANGE_ENABLE beachten!).

Der beim Getriebestufenwechsel typische Dialog zwischen NC und PLC wird gleichermaßen wie bei einem programmierten Getriebestufenwechsel (M41 ... M45) ausgeführt.

Voraussetzungen

Voraussetzung für einen Getriebestufenwechsel beim Übergang in den Achsbetrieb ist die generelle Freigabe des Getriebestufenwechsels über die Maschinendaten:

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE (Getriebestufenwechsel parametrieren)

MD35090 \$MA_NUM_GEAR_STEPS (Anzahl eingerichteter Getriebestufen)

Aktivierung / Deaktivierung

Die Funktion wird aktiviert / deaktiviert über das Maschinendatum:

MD35014 \$MA_GEAR_STEP_USED_IN_AXISMODE (Getriebestufe für den Achsbetrieb bei M70)

Wert	Bedeutung
0	Es findet kein impliziter Getriebestufenwechsel bei M70 statt. Die aktuelle Getriebestufe wird beibehalten (Standardeinstellung!).
1 ... 5	Es findet ein Getriebestufenwechsel in die Getriebestufe 1 ... 5 während der Abarbeitung von M70 statt.

Randbedingungen

Getriebestufenwechsel auf Festposition (MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE = 2)

Die Funktion "Getriebestufenwechsel auf Festposition" wird unterstützt. Der Ablauf bei M70 verlängert sich dann um die Spindelpositionierungsdauer. Die Position wird dabei mit der aktuellen Getriebestufe angefahren.

Übergang in den Achsbetrieb ohne Programmierung von M70

Die Steuerung erkennt aufgrund der Programmierabfolge selbständig den Übergang in den Achsbetrieb (siehe "Impliziter Übergang in den Achsbetrieb (Seite 1320)") und erzeugt steuerungsintern den notwendigen M70-Ablauf inklusive des Getriebestufenwechsels.

Übergang in den Achsbetrieb mit FC 18

Bei dem Übergang in den Achsbetrieb mit dem FC 18 ("Start Achse") wird der implizite Getriebestufenwechsel nicht unterstützt. Hier muss von der PLC-Applikation vor der Umschaltung in den Achsbetrieb die passende Getriebestufe eingelegt werden. Der Getriebestufenwechsel ist auch mit dem FC 18 möglich ("Start Getriebestufenwechsel").

Wechsel vom Achsbetrieb in den Spindelbetrieb

Beim Wechsel vom Achsbetrieb in den Spindelbetrieb bleibt die mit M70 eingewechselte Getriebestufe weiterhin aktiv. Die im Spindelbetrieb vor M70 aktive Getriebestufe wird nicht automatisch wieder eingewechselt. Der Servo-Parametersatz wird von Parametersatz 1 (Index 0) auf den zu der Getriebestufe passenden Parametersatz 2 ... 6 (Index 1 ... 5) gewechselt (bei MD35590 \$MA_PARAMSET_CHANGE_ENABLE < 2).

Beispiel

Beim Übergang der Spindel in den Achsbetrieb soll die Getriebestufe 4 eingewechselt werden.

Projektion: MD35014 \$MA_GEAR_STEP_USED_IN_AXISMODE[<Spindelname>] = 4

Programmcode	Kommentar
N05 M3 S1000	
N10 G1 X100 F1000	
N15 M70	; Getriebestufe 4 wird eingewechselt.
N20 POS[C]=77	
N25 ...	

Hinweis

MD35014 ist durch die Funktion "Maschinendaten wirksam setzen" veränderbar. Damit kann die einzuwechselnde Getriebestufe im Teileprogramm vor dem Übergang in den Achsbetrieb bei Bedarf noch verändert werden.

17.4.10 Unterdrückung des Getriebestufenwechsels bei DryRun, Programmtest und SERUPRO

Funktion

Bei Probelaufvorschub (DryRun), Programmtest und SERUPRO ist normalerweise kein Getriebestufenwechsel notwendig. Er kann deshalb für diese Funktionen unterdrückt werden. Die entsprechende Projektierung erfolgt mit den Bits 0 ... 2 im Maschinendatum:

MD35035 \$MA_SPIND_FUNCTION_MASK

Probelaufvorschub (DryRun)	
Bit 0 = 0	Getriebestufen werden bei aktivierter Funktion DryRun für Teileprogrammsätze mit M40, M41 bis M45, Programmierung über FC18 und Synchronaktionen eingewechselt.
Bit 0 = 1	Getriebestufenwechsel werden bei aktivierter Funktion DryRun für Teileprogrammsätze mit M40, M41 bis M45, Programmierung über FC18 und Synchronaktionen unterdrückt.
Programmtest und SERUPRO	
Bit 1 = 0	Getriebestufen werden bei aktivierter Funktion Programmtest / SERUPRO für Teileprogrammsätze mit M40, M41 bis M45, Programmierung über FC18 und Synchronaktionen eingewechselt.
Bit 1 = 1	Getriebestufenwechsel werden bei aktivierter Funktion Programmtest / SERUPRO für Teileprogrammsätze mit M40, M41 bis M45, Programmierung über FC18 und Synchronaktionen unterdrückt.
DryRun, Programmtest und SERUPRO	
Bit 2 = 0	Getriebestufenwechsel für programmierte Getriebestufe wird nach Abwahl der Funktionen DryRun, Programmtest und SERUPRO bei REPOS nicht nachgeholt.
Bit 2 = 1	Getriebestufenwechsel für programmierte Getriebestufe wird nach Abwahl der Funktionen DryRun und SERUPRO wenn möglich nachgeholt.

Ablauf

Wird ein Getriebestufenwechsel unterdrückt, so wird die programmierte Spindeldrehzahl gegebenenfalls vom Interpolator auf den zulässigen Drehzahlbereich der aktiven Getriebestufe begrenzt.

Die bei einer solchen Begrenzung erzeugten NC/PLC-Nahtstellensignale DB31, ... DBX83.2 (Solldrehzahl erhöht) und DB31, ... DBX83.1 (Solldrehzahl begrenzt) werden unterdrückt.

Überwachungen vom PLC-Programm sind während DryRun und im Probelaufvorschub überflüssig.

Bei der Unterdrückung des Getriebestufenwechsels wird keine neue Sollgetriebestufe (DB31,... DBX82.0-82.2) an die PLC ausgegeben.

Auch die GSW-Aufforderung DB31, ... DBX82.3 (Getriebe umschalten) wird unterdrückt.

Damit wird sichergestellt, dass keine GSW-Information vom PLC-Programm verarbeitet werden muss.

Ermittlung der zuletzt aktiven Getriebestufe

Die Systemvariable \$P_GEAR gibt die durch das Teileprogramm programmierte (und gegebenenfalls nicht an die PLC ausgegeben) Getriebestufe zurück.

Mit der Systemvariablen \$AC_SGEAR kann die zuletzt aktive Getriebestufe vom Teileprogramm, Synchronaktionen und auf der Bedienoberfläche gelesen werden.

Verhalten nach Abwahl

Die Funktion DryRun kann innerhalb eines laufenden Teileprogramms abgewählt werden. Nach dieser Abwahl muss die durch das Teileprogramm geforderte korrekte Getriebestufe ermittelt und eingewechselt werden.

Erst nach korrekter aktiver Getriebestufe ist die fehlerfreie Abarbeitung des restlichen Teileprogramms sichergestellt. Ein eventuell notwendiger Getriebestufenwechsel wird in dem bei der Abwahl gestarteten System-REPOS ausgeführt, wenn sich die Spindel im Drehzahlsteuerbetrieb befindet. Dabei findet ein vollständiger GSW-Dialog mit der PLC statt und die zuletzt programmierte Getriebestufe wird eingewechselt.

Stimmen bei REPOS die programmierte Getriebestufe des Teileprogramms und die durch die NC/PLC-Schnittstelle gelieferte aktuelle Getriebestufe überein, wird kein Getriebestufenwechsel ausgeführt.

Gleiches gilt für die Funktion SERUPRO.

Weitere Erläuterungen zum Satzsuchlauf SERUPRO siehe:

Literatur:

Funktionshandbuch Grundfunktionen; BAG, Kanal, Programmbetrieb, Reset-Verhalten (K1)

Randbedingungen

Bei unterdrücktem Getriebestufenwechsel bewegt sich die ausgegebene Spindeldrehzahl innerhalb des durch die aktuelle Getriebestufe vorgegebenen Drehzahlbereichs.

Einschränkungen für das Nachholen eines GSW bei REPOS sind:

- Der GSW wird nicht nachgeholt, wenn die Spindel im Abwahl- bzw. Zielsatz eine Kommandospindel (Synchronaktion) oder PLC-Spindel (FC18) ist.
- Wenn die Getriebestufe nicht eingelegt werden kann, weil sich die Spindel im Positions- oder Achsbetrieb befindet oder eine Kopplung aktiv ist, wird der Alarm 22011 "Kanal%1 Satz %3 Spindel2% Wechsel in programmierte Getriebestufe nicht möglich" gemeldet.

Beispiel

Getriebestufenwechsel bei Probelaufvorschub (DryRun)

```
; 1. Getriebestufe (GS) für Ausgangszustand aktivieren
N00 M3 S1000 M41                ; 1.GS wird eingewechselt
M0                               ; Teileprogramm halt

; PI-Dienst: Probelaufvorschub (DryRun) aktivieren
                                ; (Projektierung)
```



```

N10 M42                                ; 2.GS gefordert, es findet kein GSW statt
N11 G0 X0 Y0 Z0                        ; Achsen positionieren
N12 M0                                  ; Teileprogramm halt

; PI-Dienst: Probelaufvorschub (DryRun) deaktivieren
                                ; REORG und REPOS werden durchgeführt
                                ; jetzt findet GSW auf die 2. Getriebestufe statt

N20 G1 Z100 F1000
...
N99 M30                                ; Teileprogramm Ende

```

17.5 Weitere projektierbare Anpassungen der Spindelfunktionalität

Mit dem Maschinendatum können folgende Spindelfunktionen aktiviert werden::

MD35035 \$MA_SPIND_FUNCTION_MASK, <Bit> = <Wert>

<Bit>	<Wert>	Bedeutung
0 ... 2	1	Aktivierung: Getriebestufenwechselverhalten bei Probelaufvorschub (DryRun), Programmtest und SERUPRO Siehe "Unterdrückung des Getriebestufenwechsels bei DryRun, Programmtest und SERUPRO (Seite 1371)".
4	1	Die im NC-Programm programmierte Spindeldrehzahl S... wird nach SD43200 \$SA_SPIND_S übernommen. Das gilt auch für Drehzahlvorgaben über das PLC-Anwenderprogramm mit FC18 und Synchronaktionen. Siehe "Funktion (Seite 1383)".
5	0	Beim Verfahren der Spindel in der Betriebsart JOG wirkt die aktuell gültige Drehzahlvorgabe. Siehe SD41200 \$SN_JOGSPIND_SET_VELO
	1	Beim Verfahren der Spindel in der Betriebsart JOG wirkt die in SD43200 \$SA_SPIND_S parametrisierte Drehzahl-
8	1	Die im NC-Programm programmierte Schnittgeschwindigkeit S... wird nach SD43202 \$SA_SPIND_CONSTCUT_S übernommen. Das gilt auch für Schnittgeschwindigkeitsvorgaben über das PLC-Anwenderprogramm mit FC18. Siehe "Funktion (Seite 1383)".
10	0	SD 43206 \$SA_SPIND_SPEED_TYPE wird nicht durch NC-Programm- oder Kanaleinstellungen verändert.
	1	Ist die Spindel Masterspindel wird der Vorschubtyp (aktiver Wert der 15. G-Gruppe) nach SD43206 \$SA_SPIND_SPEED_TYPE übernommen. Ist die Spindel keine Masterspindel bleibt SD43206 \$SA_SPIND_SPEED_TYPE unverändert. Siehe "Funktion (Seite 1383)".
12	Wirksamkeit des Spindeloverrides bei Nullmarkensuche, M19, SPOS, SPOSA = 0:	
	0	Der Spindeloverride wirkt nicht .
	1	Der Spindeloverride wirkt.

<Bit>	<Wert>	Bedeutung
19	Verhalten bezüglich impliziter Hilfsfunktionsausgabe M19:	
	0	UND MD20850 \$MC_SPOS_TO_VDI == 0: Bei SPOS und SPOSA wird keine Hilfsfunktion M19 an die PLC ausgegeben.
	1	Bei SPOS und SPOSA wird die Hilfsfunktion M19 an die PLC ausgegeben. Die Adresserweiterung entspricht der Spindelnummer. Siehe "Positionierbetrieb (Seite 1301)"
20	Verhalten bezüglich impliziter Hilfsfunktionsausgabe M70:	
	0	Es wird keine implizite Hilfsfunktion M70 an die PLC ausgegeben. Hinweis: Eine programmierte Hilfsfunktion M70 wird immer an die PLC ausgegeben.
	1	Beim Übergang vom Spindel- in den Achsbetrieb wird die Hilfsfunktion M70 an die PLC ausgegeben. Die Adresserweiterung entspricht der Spindelnummer. Siehe "Impliziter Übergang in den Achsbetrieb (Seite 1320)".
22	Ausschalten der Wirksamkeit des achsspezifischen NC/PLC-Nahtstellensignals DB31, ... DBX17.6 (M3/M4 invertieren) zur Drehrichtungsumkehr der Spindel bei Gewindebohren ohne Ausgleichsfutter (G331, G332).	
	0	Nahtstellensignal DB31, ... DBX17.6 wirkt . Hinweis <ul style="list-style-type: none"> • Standardfall: Der Wert des Nahtstellensignals wird beim Start des NC-Programms ermittelt. Eine anschließende Änderung des Nahtstellensignals hat innerhalb des NC-Programms keine Auswirkung. • Sonderfall: Soll eine Änderung des Nahtstellensignals auch während der Bearbeitung des NC-Programms berücksichtigt werden, muss dies anwenderspezifisch realisiert werden. Siehe Kapitel "Sonderfall: Drehrichtungsumkehr über NC/PLC-Nahtstellensignal im NC-Programm (Seite 1329)".
	1	Nahtstellensignal DB31, ... DBX17.6 wirkt nicht .

17.6 Umschaltbare Spindeln

Funktion

Die Funktion "Umschaltbare Spindeln" ermöglicht, bezüglich der Kanalspindeln allgemeine NC-Programme zu schreiben, die in unterschiedlichen Kanälen verwendet werden können. Die im NC-Programm verwendete logische Spindelnummer wird dabei durch einen Spindelnummernumsetzer auf die physikalische Spindelnummer umgesetzt. Sodass bei der gleichen im NC-Programm programmierten Spindelnummer in unterschiedlichen Kanälen unterschiedliche physikalische Spindeln (Maschinenachsen) verfahren.

Die durch "Achstausch" eingewechselten oder abgegebenen physikalischen Spindeln müssen im Teileprogramm nicht mehr explizit angegeben werden.

Maschinendaten

Physikalische Spindelnummer

Jede Spindel muss mit einer eindeutigen projektierbaren Nummer, der physikalischen Spindelnummer, einer Maschinenachse zugeordnet werden:

MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX[<Maschinenachse>] = <physikalische Spindelnummer>

Aktivierung des Spindelnummernumsetzers

Die Aktivierung des Spindelnummernumsetzers erfolgt kanalspezifisch über das Maschinendatum:

MD20092 \$MC_SPIND_ASSIGN_TAB_ENABLE = TRUE

Settingdaten

Spindelnummernumsetzer

Zur Parametrierung des kanalspezifischen Spindelnummernumsetzers wird jeder logischen Spindelnummer bzw. programmierten Adresserweiterung, mit der im NC-Programm eine Spindel (Kanalspindel) angesprochen wird, eine physikalische Spindelnummer (Maschinenachse) zugeordnet:

SD42800 \$SC_SPIND_ASSIGN_TAB[<logische Spindelnummer>] = <physikalische Spindelnummer>

Weitere Informationen:

- Aus SD42800 \$SC_SPIND_ASSIGN_TAB[0] kann die logische Spindelnummer der aktuellen Masterspindel gelesen werden. Das Datum darf nicht beschrieben werden. Bei Programmierung von SETMS wird das Datum aktualisiert. Für nicht benutzte Spindeln ist in SD42800 der Wert 0 einzutragen.
- Bei einer Hilfsfunktionsausgabe wird als Adresserweiterung die physikalische Spindelnummer ausgegeben.
- Der Spindelnummernumsetzer wirkt bei Spindelprogrammierung in NC-Programmen und Synchronaktionen.
- Der Spindelnummernumsetzer wirkt **nicht** bei Spindelvorgaben über die NC/PLC-Nahtstelle unter Verwendung des Funktionsbausteins FC18, da dabei direkt die physikalischen Spindeln adressiert werden.
- Systemvariablen, die von der Spindelumsetzung betroffen sind: \$P_S, \$P_SDIR, \$P_SMODE, \$P_GWPS, \$AC_SDIR, \$AC_SMODE, \$AC_MSNUM, \$AA_S

Randbedingungen

- Die umschaltbaren Kanalspindeln ersetzen **nicht** die Funktion "Achstausch".
- Die Spindelumsetzung kann durch Schreiben des Settingdatums (SD42800 \$SC_SPIND_ASSIGN_TAB) über NC-Programm, PLC-Anwenderprogramm oder PI-Dienst, geändert werden. Die Änderung ist sofort wirksam.

- Im Spindelnummernumsetzer können nur die physikalischen Spindeln verwendet werden, die dem Kanal zugeordnet sind (MD20070 \$MC_AXCONF_MACHAX_USED). Werden im Spindelnummernumsetzer physikalische Spindeln angegeben, die gerade in einem anderen Kanal aktiv sind, wird je nach Einstellung in MD30552 \$MA_AUTO_GET_TYPE entweder die physikalische Spindel für den Kanal angefordert oder der Alarm 16105 "Zugewiesene Spindel nicht vorhanden" angezeigt.
- Wird SD42800 \$SC_SPIND_ASSIGN_TAB[<n>] von der PLC oder von HMI beschrieben, dann sollte sich der Kanal, dessen Tabelle geändert wird, im Resetzustand befinden bzw. die umzuschaltende Spindel sollte nicht in dem laufenden Teileprogramm verwendet werden.
Synchrones Verhalten kann über einen Vorlaufstopp `STOPRE` erreicht werden.
- Die Umsetzung einer logischen auf mehrere physikalische Spindeln ist nicht verriegelt. Bei der Anzeige der logischen Spindel an der Bedienoberfläche ergeben sich jedoch entsprechend der Umsetztabelle Mehrdeutigkeiten.

Beispiel

Zuordnung von Spindelnummer und Maschinenachse:

```
MD35000 $MA_SPIND_ASSIGN_TO_MACHAX [AX4] = 1
MD35000 $MA_SPIND_ASSIGN_TO_MACHAX [AX5] = 2
MD35000 $MA_SPIND_ASSIGN_TO_MACHAX [AX6] = 3
MD35000 $MA_SPIND_ASSIGN_TO_MACHAX [AX7] = 5
```

Maschinenachse in Kanal aufnehmen:

```
MD20070 $MC_AXCONF_MACHAX_USED[0] = 4
MD20070 $MC_AXCONF_MACHAX_USED[1] = 5
MD20070 $MC_AXCONF_MACHAX_USED[2] = 6
MD20070 $MC_AXCONF_MACHAX_USED[3] = 7
```

Festlegung der Masterspindel:

```
MD20090 $MC_SPIND_DEF_MASTER_SPIND = 1
```

Spindelnummerumsetzer

MD20092 \$MC_SPIND_ASSIGN_TAB_ENABLE=1	; Spindelnummerumsetzer aktivieren
SD42800 \$SC_SPIND_ASSIGN_TAB[0]=1	; Masterspindel wie projiziert
SD42800 \$SC_SPIND_ASSIGN_TAB[1]=1	; Grundstellung der Tabelle
SD42800 \$SC_SPIND_ASSIGN_TAB[2]=2	
SD42800 \$SC_SPIND_ASSIGN_TAB[3]=3	

```

SD42800 $SC_SPIND_AS-          ; logische Spindel nicht zugewiesen
SIGN_TAB[4]=0
; Adresserweiterung ist 1 Ausgabe: M1=3 S1=1000
; Spindel 1 (physikalische Masterspindel) dreht
M3 S1000
...
; Zuweisung der logischen Spindel 1 zur physikalischen Spindel 5
SD42800 $SC_SPIND_AS-
SIGN_TAB[1]=5

; Zuweisung der logischen Spindel 2 zur physikalischen Spindel 3
SD42800 $SC_SPIND_AS-
SIGN_TAB[2]=3
; Achtung: die physikalische Spindel 3 ist jetzt zweimal vergeben.
; Bei Programmierung der logischen Spindel 2 und 3 wird immer die
; physikalische Spindel 3 angesprochen.
; In den Maschinengrundbildern drehen sich beide Spindeln.

SETMS(2)                      ; SD42800 $SC_SPIND_ASSIGN_TAB[0] = 2
; Masterspindel = Adresserweiterung = 2, ausgegeben wird ungesetzte Spindelnr. M3 =
5
; Die physikalische Spindel mit der projektierten Nummer "3" stoppt.
M5
GET(S4)                        ; Alarm 16105: logische Spindel 4
                                ; kann nicht umgesetzt werden
RELEASE(S1)                    ; Kanalspindel 1 = physik. Spindel 5
                                ; wird freigegeben.

```

17.7 Programmierung

17.7.1 Programmierung aus dem Teileprogramm

Programmieranweisungen

Anweisung	Beschreibung																		
SETMS:	Masterspindel ist die im folgenden Maschinendatum angegebene Spindel: MD20090 \$MC_SPIND_DEF_MASTER_SPIND (Löschstellung Masterspindel im Kanal)																		
SETMS (<n>) :	<p>Spindel mit der Nummer <n> ist die Masterspindel (kann abweichend von der Löschstellung sein: MD20090 \$MC_SPIND_DEF_MASTER_SPIND)</p> <p>Die Masterspindel muss für folgende Funktionen festgelegt werden:</p> <table> <tr> <td>• G95:</td><td>Umdrehungsvorschub</td></tr> <tr> <td>• G96 S.../G961 S...:</td><td>konstante Schnittgeschwindigkeit in m/min bzw. ft/min</td></tr> <tr> <td>• G97/G971:</td><td>G96/G961 aufheben und letzte Spindeldrehzahl einfrieren</td></tr> <tr> <td>• G63:</td><td>Gewindebohren mit Ausgleichsfutter</td></tr> <tr> <td>• G33/G34/G35:</td><td>Gewindeschneiden</td></tr> <tr> <td>• G331/G332:</td><td>Gewindebohren ohne Ausgleichsfutter</td></tr> <tr> <td>• G4 S...:</td><td>Verweilzeit in Spindelumdrehungen</td></tr> <tr> <td colspan="2"> <ul style="list-style-type: none"> • Programmierung von M3, M4, M5, S, SVC, SPOS, M19, SPOSA, M40, M41 bis M45 und WAITS ohne Angabe der Spindelnummer. </td></tr> <tr> <td colspan="2"> <p>Die aktuelle Einstellung der Masterspindel kann über RESET/Teileprogrammende und Teileprogrammstart erhalten bleiben. Die Einstellung erfolgt über die Maschinendaten:</p> <ul style="list-style-type: none"> • MD20110 \$MC_RESET_MODE_MASK • MD20112 \$MC_START_MODE_MASK </td></tr> </table>	• G95:	Umdrehungsvorschub	• G96 S.../G961 S...:	konstante Schnittgeschwindigkeit in m/min bzw. ft/min	• G97/G971:	G96/G961 aufheben und letzte Spindeldrehzahl einfrieren	• G63:	Gewindebohren mit Ausgleichsfutter	• G33/G34/G35:	Gewindeschneiden	• G331/G332:	Gewindebohren ohne Ausgleichsfutter	• G4 S...:	Verweilzeit in Spindelumdrehungen	<ul style="list-style-type: none"> • Programmierung von M3, M4, M5, S, SVC, SPOS, M19, SPOSA, M40, M41 bis M45 und WAITS ohne Angabe der Spindelnummer. 		<p>Die aktuelle Einstellung der Masterspindel kann über RESET/Teileprogrammende und Teileprogrammstart erhalten bleiben. Die Einstellung erfolgt über die Maschinendaten:</p> <ul style="list-style-type: none"> • MD20110 \$MC_RESET_MODE_MASK • MD20112 \$MC_START_MODE_MASK 	
• G95:	Umdrehungsvorschub																		
• G96 S.../G961 S...:	konstante Schnittgeschwindigkeit in m/min bzw. ft/min																		
• G97/G971:	G96/G961 aufheben und letzte Spindeldrehzahl einfrieren																		
• G63:	Gewindebohren mit Ausgleichsfutter																		
• G33/G34/G35:	Gewindeschneiden																		
• G331/G332:	Gewindebohren ohne Ausgleichsfutter																		
• G4 S...:	Verweilzeit in Spindelumdrehungen																		
<ul style="list-style-type: none"> • Programmierung von M3, M4, M5, S, SVC, SPOS, M19, SPOSA, M40, M41 bis M45 und WAITS ohne Angabe der Spindelnummer. 																			
<p>Die aktuelle Einstellung der Masterspindel kann über RESET/Teileprogrammende und Teileprogrammstart erhalten bleiben. Die Einstellung erfolgt über die Maschinendaten:</p> <ul style="list-style-type: none"> • MD20110 \$MC_RESET_MODE_MASK • MD20112 \$MC_START_MODE_MASK 																			
M3:	Spindeldrehrichtung rechts für die Masterspindel																		
M<n>=3:	Spindeldrehrichtung rechts für die Spindel mit der Nummer <n>																		
M4:	Spindeldrehrichtung links für die Masterspindel																		
M<n>=4:	Spindeldrehrichtung links für die Spindel mit der Nummer <n>																		
M5:	Spindel Halt ohne Orientierung für die Masterspindel																		
M<n>=5:	Spindel Halt ohne Orientierung für die Spindel mit der Nummer <n>																		
S...:	Spindeldrehzahl in 1/min für die Masterspindel																		
S<n>=...:	Spindeldrehzahl in 1/min für die Spindel mit der Nummer <n>																		
SVC=...:	Schnittgeschwindigkeit in m/min bzw. ft/min für die Masterspindel																		
SVC [<n>]=...:	Schnittgeschwindigkeit in m/min bzw. ft/min für die Spindel mit der Nummer <n>																		
SPOS=...:	Spindelpositionierung für die Masterspindel																		
SPOS [<n>]=...:	<p>Spindelpositionierung für die Spindel mit der Nummer <n></p> <p>Der Satzwechsel erfolgt erst, wenn die Spindel in Position ist.</p>																		

Anweisung	Beschreibung
SPOSA=...: SPOSA [<n>]=...:	Spindelpositionierung für die Masterspindel Spindelpositionierung für die Spindel mit der Nummer <n> Der Satzwechsel erfolgt sofort. Die Spindel positioniert unabhängig von der weiteren Teileprogrammbearbeitung so lange, bis sie die Position erreicht hat.
SPOS=DC (...): SPOS [<n>]=DC (...): SPOSA=DC (...): SPOSA [<n>]=DC (...):	Die Bewegungsrichtung wird beim Positionieren aus der Bewegung beibehalten und die Position angefahren. Beim Positionieren aus dem Stillstand wird auf kürzestem Weg die Position angefahren.
SPOS=ACN (...): SPOS [<n>]=ACN (...): SPOSA=ACN (...): SPOSA [<n>]=ACN (...):	Die Position wird immer mit negativer Bewegungsrichtung angefahren. Gegebenfalls wird vor dem Positionieren die Bewegungsrichtung invertiert.
SPOS=ACP (...): SPOS [<n>]=ACP (...): SPOSA=ACP (...): SPOSA [<n>]=ACP (...):	Die Position wird immer mit positiver Bewegungsrichtung angefahren. Gegebenfalls wird vor dem Positionieren die Bewegungsrichtung invertiert.
SPOS=IC (...): SPOS [<n>]=IC (...): SPOSA=IC (...): SPOSA [<n>]=IC (...):	Es wird der Fahrweg vorgegeben. Die Fahrrichtung ergibt sich aus dem Vorzeichen des Fahrwegs. Ist die Spindel bereits in Bewegung, dann wird gegebenenfalls die Fahrrichtung invertiert, um in die programmierte Richtung fahren zu können. Wird während der Fahrbewegung die Nullmarke überfahren, dann wird die Spindel automatisch mit der Nullmarke synchronisiert, falls noch keine Referenz vorhanden ist oder diese per NST-Signal neu angefordert wurde.
M19: M [<n>]=19:	Spindelpositionierung für die Masterspindel auf die Position in SD43240 Spindelpositionierung für die Spindel mit der Nummer <n> auf die Position in SD43240 Der Satzwechsel erfolgt erst, wenn die Spindel in Position ist.
M70: M<n>=70:	Spindel stillsetzen und Lageregelung einschalten, Parametersatz Null anwählen, Achsbetrieb aktivieren für die Masterspindel für die Spindel mit der Nummer <n>
SPCON: SPCON (<n>): SPCON (<n>, <m>):	Spindellageregelung EIN für die Masterspindel für die Spindel mit der Nummer <n> für die Spindeln mit den Nummern <n> und <m>
PCOF: SPCOF (<n>): SPCOF (<n>, <m>):	Spindellageregelung AUS, Drehzahlsteuerbetrieb aktivieren für die Masterspindel für die Spindel mit der Nummer <n> für die Spindeln mit den Nummern <n> und <m>
FPRAON (S<n>):	Umdrehungsvorschub für Spindel <n> EIN, abgeleitet von der Masterspindel
FPRAON (S<n>, S<m>):	Umdrehungsvorschub für Spindel <n> EIN, abgeleitet von der Spindel <m> Der Umdrehungsvorschubwert muss mit FA [S<m>] vorgegeben werden.
FPRAOF (S<n>):	Umdrehungsvorschub für Spindel <n> AUS
C30 G90 G1 F3600	Rundachse C (Spindel im Achsbetrieb) fährt auf die Position 30 Grad mit einer Geschwindigkeit von 3600 Grad/min = 10 U/min

Anweisung	Beschreibung
G25 S...: G25 S<n>:	<p>Programmierbare minimale Spindeldrehzahlbegrenzung für die Masterspindel für die Spindel mit der Nummer <n></p>
G26 S...: G26 S<n>:	<p>Programmierbare maximale Spindeldrehzahlbegrenzung für die Masterspindel für die Spindel mit der Nummer <n></p>
LIMS=...: LIMS [<n>]=...:	<p>Programmierbare maximale Spindeldrehzahlbegrenzung bei G96, G961, G97 für die Masterspindel für die Spindel mit der Nummer <n></p>
VELOLIM [<Spindel>]=...:	<p>Programmierbare Begrenzung der projektierten getriebestufenabhängigen Maximaldrehzahl</p> <p>Über Maschinendatum (MD30455 \$MA_MISC_FUNCTION_MASK, Bit 6) kann für die Programmierung im Teileprogramm eingestellt werden, ob VELOLIM unabhängig von der aktuellen Verwendung als Spindel oder Achse wirkt (Bit 6 = 1) oder getrennt für jede Betriebsart programmierbar sein soll (Bit 6 = 0). Ist eine getrennte Wirkung projektiert, dann wird die Auswahl über den Namen bei der Programmierung getroffen:</p> <ul style="list-style-type: none"> • Spindelname S<n> für Spindelbetriebsarten • Achsname, z. B. "C", für den Achsbetrieb <p>Der Korrekturwert bezieht sich:</p> <ul style="list-style-type: none"> • Bei Spindeln im Achsbetrieb (wenn MD30455 Bit 6 = 0): auf die projektierte maximale Achsgeschwindigkeit (MD32000 \$MA_MAX_AX_VELO). • Bei Spindeln im Spindel- oder Achsbetrieb (wenn MD30455 Bit 6 = 1): auf die Maximaldrehzahl der aktiven Getriebestufe (MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT [<n>]) <p>Weitere Erläuterungen zur Programmierung von VELOLIM siehe: Literatur: Programmierhandbuch Arbeitsvorbereitung</p>
WAITS:	<p>Synchronisationsbefehl für Masterspindel</p> <p>Mit der Abarbeitung nachfolgender Sätze wird solange gewartet, bis die in einem früheren NC-Satz mit SPOSA programmierte Spindel ihre Position erreicht hat (mit Genauhalt fein).</p> <p>WAITS nach M5: Warten, bis die Spindel steht.</p> <p>WAITS nach M3/M4: Warten, bis die Spindel ihre Solldrehzahl erreicht hat.</p>
WAITS (<n>): WAITS (<n>, <m>):	<p>Synchronisationsbefehl für die Spindel mit der Nummer <n> Synchronisationsbefehl für die Spindeln mit den Nummern <n> und <m></p>
FA[S<n>]:	<p>Programmierung der Positioniergeschwindigkeit (axialer Vorschub) für die Spindel <n> in [Grad/min]</p> <p>Mit FA[S<n>]=0 wird der projektierte Wert wieder gültig: MD35300 \$MA_SPIND_POSCTRL_VELO</p>
OVRA[S<n>]:	Programmierung des axialen Override-Werts für die Spindel <n> in [%]
ACC[S<n>]:	Programmierung des axialen Beschleunigungsvermögens der Spindel <n> in [%]

Anweisung	Beschreibung
SPI (<n>):	<p>Mit SPI (<n>) wird eine Spindelnummer entsprechend dem Maschinendatum: MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX[] in den Datentyp AXIS konvertiert.</p> <p>SPI wird verwendet, wenn mit der Spindel Achsfunktionen programmiert werden sollen. Folgende Anweisungen sind mit SPI möglich:</p> <ul style="list-style-type: none"> • Frameanweisungen: <ul style="list-style-type: none"> – CTRANS () – CFINE () – CMIRROR () – CSCALE () • Geschwindigkeits- und Beschleunigungswerte für Folgespindeln: <ul style="list-style-type: none"> – FA [SPI (<n>)] – ACC [SPI (<n>)] – OVRA [SPI (<n>)] • Systemvariablen: <ul style="list-style-type: none"> – \$P_PFRAME [SPI (<n>), TR] = <Wert> – \$P_PFRAME = CTTRANS (X, <Achswert>, Y, <Achswert>, SPI (<n>), <Achswert>) – \$P_PFRAME = CSCALE (X, <Maßstab>, Y, <Maßstab>, SPI (<n>), <Maßstab>) – \$P_PFRAME = CMIRROR (S<n>, Y, Z) – \$P_UBFR = CTRANS (A, 10) : CFINE (19, 0.1) <p>Weitere Erläuterungen zur Programmierung von SPI siehe: Literatur: Programmierhandbuch Arbeitsvorbereitung</p>
M40:	Automatische Getriebestufenauswahl für die Masterspindel
M<n>=40:	Automatische Getriebestufenauswahl für die Spindel mit der Nummer <n>
M41 bis M45:	Getriebestufe 1 bis 5 auswählen für die Masterspindel
M<n>=41 bis M<n>=45:	Getriebestufe 1 bis 5 auswählen für die Spindel mit der Nummer <n>

Hinweis

Die M-Funktionen M3, M4, M5 und M70 werden nicht im DB21, ... DBB194 und DBB202 ausgegeben, wenn in einem Kanal eine Spindel konfiguriert ist. Diese M-Funktionen werden als erweiterte M-Funktionen im DB21, ... DBB68 ff. und im jeweiligen Achs-DB, DB31, ... DBB86 ff., angeboten.

Literatur

Genauere Erläuterungen zur Programmierung der Spindel siehe:

- Programmierhandbuch Grundlagen

17.7.2 Programmierung über Synchronaktionen

Die M-Funktionen M40 bis M45 sind auch in Synchronaktionen programmierbar.

Dabei ist Folgendes zu beachten:

- Die Programmierung von M40 ... M45 im Teileprogramm hat keine Auswirkungen auf den aktuellen Zustand des automatischen Getriebestufenwechsels der Synchronaktionen und umgekehrt.
- Der automatische Getriebestufenwechsel wirkt bei S-Wert-Programmierung mit M40 für Synchronaktionen und Teileprogramm getrennt.
- Nach POWER ON ist M40 ausgeschaltet.
Es wird keine Getriebestufenanpassung bei einer S-Wert-Vorgabe aus Synchronaktionen vorgenommen.
- Ein durch Synchronaktionen programmiertes M40 bleibt für Synchronaktionen immer aktiv (modal wirksam) und wird bei Reset nicht zurückgesetzt.
- M41 ... M45 wählt entsprechend der Programmierung im Teileprogramm die erste bis fünfte Getriebestufe an.
Zur Ausführung der Funktion wird ein Achstausch notwendig.
Nach Ausführung des Getriebestufenwechsels befindet sich die Spindel im neutralen Zustand (analoges Verhalten zur M3-, M4-, M5-Programmierung).

Literatur

Weitere Erläuterungen zur Programmierung der Spindel sowie Spindelbewegungen aus Synchronaktionen siehe:

- Programmierhandbuch Arbeitsvorbereitung
- Funktionshandbuch Synchronaktionen

17.7.3 Programmierung von Spindelsteuerungen über PLC mit FC18 - nur 840D sl

Bei der Vorgabe von Drehrichtung und Drehzahl durch die PLC mittels FC18 kann die zur Drehzahl passende Getriebestufe vom NC ermittelt und eingelegt werden. Dies entspricht der M40-Funktionalität bei der Programmierung über das Teileprogramm.

Zur Aktivierung der Getriebestufenauswahl muss beim Aufruf des FC18 bei PLC-Anwenderprogrammen der entsprechende Startcode gesetzt werden.

Literatur

Genauere Erläuterungen zur Programmierung von Spindelsteuerungen über PLC mit FC18 siehe:

- Funktionshandbuch Grundfunktionen; PLC-Grundprogramm (P3)

17.7.4 Programmierung über NC/PLC-Nahtstellensignale

17.7.4.1 Funktion

Hinweis

Die Funktion steht nur bei Verwendung von SINUMERIK Operate zur Verfügung!

Einer Spindel können vom PLC-Anwenderprogramm aus über die Auftragsschnittstelle DB31, ... DBX30.0 - 4 folgende Aufträge erteilt werden:

- Spindel-Stopp
- Spindel-Start Rechtslauf
- Spindel-Start Linkslauf
- Getriebestufe auswählen
- Spindel-Start Positionieren

Auftragserkennung

Ein Spindelauftrag wird NC-seitig aufgrund eines positiven **Flankenwechsels** des entsprechenden Nahtstellensignals (0 → 1) erkannt.

Voraussetzungen

Zum Zeitpunkt eines Spindelauftrags müssen bezüglich des Kanal-, Programm- und Spindelzustandes folgende Voraussetzungen erfüllt sein.

- Kanalzustand:
 - DB21, ... DBX35.6 == 1 (Kanalzustand "unterbrochen") **ODER**
 - DB21, ... DBX35.7 == 1 (Kanalzustand "Reset")
- Programmzustand:
 - DB21, ... DBX35.3 = 1 (Programmzustand "unterbrochen") **ODER**
 - DB21, ... DBX35.4 = 1 (Programmzustand "abgebrochen")
- Spindelzustand
 - ("Kanalachse" **ODER** "Neutrale Achse") **UND KEINE**
 - (Positionierbewegung durch FC18 **ODER** Synchronaktion)

Hinweis

Verhalten im Fehlerfall

Im Fehlerfall, d. h., wenn ein Auftragssignal während eines ungültigen Zustands gesetzt wird, wird der entsprechende Auftrag **ohne Rückmeldung** ignoriert.

Kanalzuordnung

Ein Spindelauftrag wird in dem Kanal bearbeitet, dem die Spindel zum Zeitpunkt des Auftrags zugeordnet ist.

Der Kanal, dem die Spindel zugeordnet ist, kann über folgendes NC/PLC-Nahtstellensignal ermittelt werden:

DB31, ... DBX68.0 - 3 (Kanalzuordnung der NC-Achse/Spindel)

Betriebsartenwechsel

Die im Rahmen der Spindelaufträge gemachten Vorgaben bleiben über Betriebsartenwechsel hinaus erhalten.

Programmstart

Mit dem Start eines NC-Programms im Kanal, dem die Spindel zugeordnet ist, bleiben die aktuellen Spindelvorgaben erhalten. Die Spindelvorgaben können durch Befehle im NC-Programm selbst oder über Synchronaktionen verändert werden.

Siehe auch

M40: Automatische Getriebestufenauswahl bei Drehzahlen außerhalb projektierter Schaltschwellen (Seite 1399)

G2: Geschwindigkeiten, Soll-/Istwertsysteme, Regelung (Seite 341)

17.7.4.2 Inbetriebnahme: Maschinendaten**Automatische Übernahme von Spindelvorgaben in Settingdaten**

Durch folgende Maschinendateneinstellungen werden Spindelvorgaben aus NC-Programmen, Synchronaktionen und der Spindelsteuerung von PLC (FC18) in die entsprechenden Settingdaten übernommen:

MD35035 \$MA_SPIND_FUNCTION_MASK, Bit x = <Wert>

Bit	Wert	Bedeutung
4	1	<p>Eine Drehzahlvorgabe über NC-Programm, Synchronaktionen oder FC18 wird in folgendes spindelspezifische Settingdatum übernommen:</p> <p>SD43200 \$SA_SPIND_S[<Spindel>]</p> <p>Hinweis</p> <p>S-Werte, die keine Drehzahlprogrammierungen sind, werden nicht übernommen. Z. B.:</p> <ul style="list-style-type: none"> • S-Wert bei konstanter Schnittgeschwindigkeit (G96, G961) • S-Wert bei umdrehungsbezogener Verweilzeit (G4)
	0	Es werden keine Drehzahlvorgaben in das Settingdatum übernommen.

Bit	Wert	Bedeutung
8	1	Eine Vorgabe der Schnittgeschwindigkeit bei aktiver Funktion "Konstante Schnittgeschwindigkeit" (G96, G961, G962) über NC-Programm, Synchronaktionen oder FC18 wird in folgendes spindelspezifische Settingdatum übernommen: SD43202 \$SA_SPIND_CONSTCUT_S[<Spindel> Hinweis S-Werte, die keine Schnittgeschwindigkeitsprogrammierungen sind, werden nicht übernommen. Z. B.: <ul style="list-style-type: none"> S-Wert bei umdrehungsbezogener Verweilzeit (G4)
	0	Es werden keine Schnittgeschwindigkeitvorgaben in das Settingdatum übernommen.
10	1	Für die Masterspindel wird der Vorschubtyp in folgendes Settingdatum übernommen: SD43206 \$SA_SPIND_SPEED_TYPE[<Spindel>]
	0	Es werden keine Vorschubtypvorgaben für die Masterspindel in das Settingdatum übernommen.

Mit den in die Settingdaten übernommenen Spindelvorgaben wird die Spindel dann im Rahmen der Spindelaufträge verfahren.

17.7.4.3 Inbetriebnahme: NC/PLC-Nahtstellensignale

Auftragsschnittstelle

- DB31, ... DBX30.0 (Spindel-Stopp)
- DB31, ... DBX30.1 (Spindel-Start Rechtslauf)
- DB31, ... DBX30.2 (Spindel-Start Linkslauf)
- DB31, ... DBX30.3 (Getriebestufe auswählen)
- DB31, ... DBX30.4 (Spindel-Start Positionieren)

Relevante Zustandssignale

- DB21, ... DBX35.3 (Programmzustand "unterbrochen")
- DB21, ... DBX35.4 (Programmzustand "abgebrochen")
- DB21, ... DBX35.7 (Kanalzustand "Reset")
- DB21, ... DBX35.6 (Kanalzustand "unterbrochen")
- DB31, ... DBX68.0 - 3 (Kanalzuordnung der NC-Achse/Spindel)
- DB31, ... DBX 84.0 (Konstante Schnittgeschwindigkeit)

Literatur:

Eine ausführliche Beschreibung der NC/PLC-Nahtstellensignale findet sich im Listenhandbuch NC-Variablen und Nahtstellensignale.

17.7.4.4 Drehzahlvorgabe (SD43200)

Beim Start einer Spindel über einen Spindelauftrag wird als Spindeldrehzahl der aktuell im Settingdatum eingetragene spindelspezifische Wert wirksam:

SD43200 \$SA_SPIND_S[<Spindel>] (Drehzahl bei Spindelstart über die PLC-Auftragsschnittstelle)

Schreiben eines neuen Drehzahlwertes

Ein neuer Drehzahlwert wird in folgenden Situationen in das Settingdatum geschrieben:

- Drehzahlvorgabe über NC-Programm, Synchronaktion oder FC18
Randbedingungen:
 - MD35035 \$MA_SPIND_FUNCTION_MASK (Seite 1384), Bit 4 == 1
 - DB31, ... DBX84.0 == 0 (Konstante Schnittgeschwindigkeit)
- Schreiben des Settingdatums im NC-Programm oder Synchronaktion
- Schreiben des Settingdatums über HMI (BTSS) @@@

Randbedingungen

Getriebestufenwechsel

Es wird kein Getriebestufenwechsel ausgelöst, wenn die vorgegebene Drehzahl außerhalb des Drehzahlbereichs der aktuellen Getriebestufe liegt.

Ausnahmen, siehe Kapitel "M40: Automatische Getriebestufenauswahl bei Drehzahlen außerhalb projektierter Schaltschwellen (Seite 1399)".

17.7.4.5 Vorgabe der konstanten Schnittgeschwindigkeit (SD43202)

Beim Start einer Masterspindel über einen Spindelauftrag und aktivem Spindeldrehzahltyp "Konstante Schnittgeschwindigkeit", wird als konstante Schnittgeschwindigkeit der aktuell im Settingdatum eingetragene spindelspezifische Wert wirksam:

SD43202 \$SA_SPIND_CONSTCUT_S[<Spindel>] (Schnittgeschwindigkeit bei Spindelstart über die PLC-Auftragsschnittstelle)

Schreiben eines neuen Schnittgeschwindigkeitswertes

Ein neuer Wert für die konstante Schnittgeschwindigkeit der Masterspindel wird in folgenden Situationen in das spindelspezifische Settingdatum geschrieben:

- Vorgabe der konstanten Schnittgeschwindigkeit über NC-Programm, Synchronaktion oder FC18
Randbedingungen:
 - MD35035 \$MA_SPIND_FUNCTION_MASK (Seite 1384), Bit 8 == 1
 - DB31, ... DBX84.0 == 1 (Konstante Schnittgeschwindigkeit aktiv)
- Schreiben des Settingdatums im NC-Programm oder Synchronaktion
- Schreiben des Settingdatums über HMI (BTSS) @@@

Maßsystem

Beim Schreiben des Settingdatums wird der Wert entsprechend den folgenden Randbedingungen interpretiert:

- Schreiben über NC-Programm oder Synchronaktion:
 - G700 aktiv: feet/min
 - G710 aktiv: m/min
 - G70, G71 aktiv: Abhängig von der Einstellung in MD10240 \$MN_SCALING_SYSTEM_IS_METRIC
- Schreiben über SINUMERIK Operate:
 - Abhängig von der Einstellung in MD10240 \$MN_SCALING_SYSTEM_IS_METRIC
- Schreiben über FC18:
 - Funktionsnummer 10: "konstanter Schnittgeschwindigkeit (m/min)"
 - Funktionsnummer 11: "konstanter Schnittgeschwindigkeit (feet/min)"

Lesen über System- oder BTSS-Variable

Der aktuell vorgegebene Wert für die konstante Schnittgeschwindigkeit kann im NC-Programm, Synchronaktion oder von HMI über folgende Systemdaten gelesen werden:

Systemvariable	BTSS-Variable	Bedeutung
\$P_CONSTCUT_S	---	Zuletzt programmierte Schnittgeschwindigkeit
\$AC_CONSTCUT_S	acConstCutS	Aktuelle konstante Schnittgeschwindigkeit

17.7.4.6

Vorgabe des Spindeldrehzahltyps für die Masterspindel (SD43206)

Vorgaben des Spindeldrehzahltyps für die **Masterspindel** aus Teileprogramm, FC18 oder Synchronaktionen werden aus allen üblichen Quellen in das folgende Settingdatum geschrieben:

SD43206 \$SA_SPIND_SPEED_TYPE (Spindeldrehzahltyp für Spindelstart über PLC-Schnittstelle)

Der Wertebereich und die Funktionalität entsprechen der 15. G-Gruppe (Vorschubtyp).

Zulässige Werte sind die G-Werte: 93, 94, 95, 96, 961, 97, und 971.

Je nach Einstellung wird bei DB31, ... DBX30.1/2 (Spindel-Start Rechts-/Linkslauf) entweder die Drehzahl aus SD43200 \$SA_SPIND_S oder die Schnittgeschwindigkeit aus SD43202 \$SA_SPIND_CONSTCUT_S wirksam:

- | | |
|-------------------------|--|
| 93, 94, 95, 97 und 971: | Die Masterspindel wird mit der Drehzahl aus SD43200 gestartet. |
| 96 und 961: | Die Drehzahl der Masterspindel ergibt sich aus der vorgegebenen Schnittgeschwindigkeit (SD43202) und dem Radius der Planachse. |

17.7.5 Programmierung von extern (PLC, HMI)

SD43300 und SD42600

Über das axiale Settingdatum:

SD43300 \$SA_ASSIGN_FEED_PER_REV_SOURCE (Umdrehungsvorschub für Spindeln)
kann in der Betriebsart JOG über das kanalspezifische Settingdatum:

SD42600 \$SC_JOG_FEED_PER_REV_SOURCE (Steuerung Umdrehungsvorschub in JOG)
das Verfahren mit Umdrehungsvorschub von extern angewählt werden.

Über das Settingdatum sind folgende Einstellungen möglich:

>0:	die Maschinenachsnnummer der Rundachse/Spindel, von der Umdrehungsvorschub abgeleitet werden soll.
-1:	der Umdrehungsvorschub wird abgeleitet von der Masterspindel des Kanals, in dem die Achse/Spindel jeweils aktiv ist.
0:	Funktion ist abgewählt.

FPRAON (S2)

Umdrehungsvorschub für Spindel S2 ein, abgeleitet von der Masterspindel.

FPRAON (S2, A)

Umdrehungsvorschub für Spindel S2 ein, abgeleitet von Achse A.

Der Umdrehungsvorschubwert muss mit FA[Sn] vorgegeben werden.

FPRAOF (S2)

Umdrehungsvorschub für Spindel S2 aus.

SPI(n)

Es ist auch die Programmierung von SPI (n) anstelle von SPI (Sn) zulässig.

17.8 Spindelüberwachungen

17.8.1 Zulässige Drehzahlbereiche

Aus den parametrisierten bzw. programmierten Drehzahlgrenzwerten sowie der aktiven Spindelfunktion (G94, G95, G96, G961, G97, G971, G33, G34, G35, G331, G332, etc.) ergibt sich der zulässige Drehzahlbereich einer Spindel.

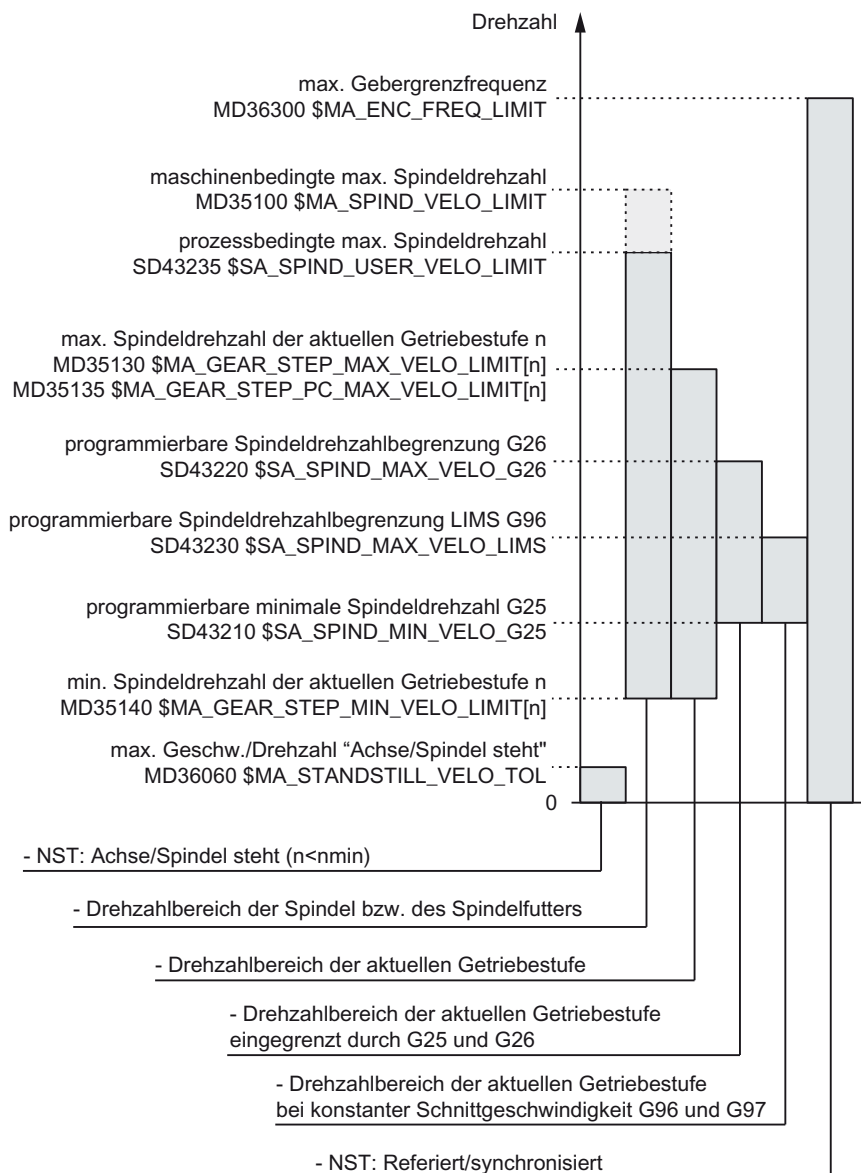


Bild 17-8 Bereiche der Spindelüberwachungen / Drehzahlen

17.8.2 Achse/Spindel steht

Erst wenn die Spindel steht, sind an der Maschine Funktionen wie Werkzeugwechsel, Maschinentüre öffnen, Bahnvorschub freigeben, etc. möglich.

Funktion

Der Zustand "Achse/Spindel steht" wird erreicht, wenn keine Sollwerte mehr erzeugt werden und die Spindel-Istdrehzahl den projektierten Schwellwert für "Achse/Spindel steht" unterschreitet:

MD36060 \$MA_STANDSTILL_VELO_TOL (max. Geschwindigkeit/Drehzahl "Achse/Spindel steht")

Ist die Spindel zum Stillstand gekommen, wird das folgende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX61.4 (Achse/Spindel steht)

Wirksamkeit

Die Überwachung des Spindelstillstandes wirkt in allen Spindelbetriebsarten und im Achsbetrieb.

Bahnvorschub sperren

Wird eine Spindel im Steuerbetrieb gestoppt (M5), dann wird der Bahnvorschub gesperrt, wenn das folgende Maschinendatum gesetzt ist:

MD35510 \$MA_SPIND_STOPPED_AT_IPO_START (Vorschubfreigabe bei Spindel steht)

Ist die Spindel zum Stillstand gekommen, wird der Bahnvorschub wieder freigegeben.

17.8.3 Spindel im Sollbereich

Funktion

Die Spindelüberwachung "Spindel im Sollbereich" kontrolliert, ob:

- die programmierte Spindeldrehzahl erreicht ist.
- die Spindel steht:
DB31, ... DBX61.4 (Achse/Spindel steht) = 1
- sich die Spindel noch in der Beschleunigungs- oder Bremsphase befindet.

In der Spindelbetriebsart Steuerbetrieb wird die Solldrehzahl mit der Istdrehzahl verglichen. Weicht die Istdrehzahl um mehr als die über MD vorgebbare Spindeldrehzahltoleranz (s. u.) ab, wird:

- das folgende axiale NC/PLC-Nahtstellensignal auf "0" gesetzt:
DB31, ... DBX83.5 (Spindel im Sollbereich) = 0
- der nächste Bearbeitungssatz nicht freigegeben (abhängig von der Einstellung in MD35500 \$MA_SPIND_ON_SPEED_AT_IPO_START, siehe "Achse/Spindel steht (Seite 1390)").

Spindelsolldrehzahl

Die Spindelsolldrehzahl ergibt sich aus der programmierten Drehzahl unter Berücksichtigung der Spindel-Override und der aktiven Begrenzungen.

Eine Begrenzung oder Anhebung der programmierten Drehzahl wird durch DB31, ... DBX83.1 (Soll-Drehzahl begrenzt) oder DB31, ... DBX83.2 (Soll-Drehzahl erhöht) angezeigt (siehe auch "Minimale/Maximale Drehzahl der Getriebestufe (Seite 1392)"). Das Erreichen des Toleranzbereichs der Solldrehzahl wird dadurch **nicht** verhindert.

Toleranzbereich der Solldrehzahl

Der Toleranzbereich der Solldrehzahl wird festgelegt durch den Spindeldrehzahl-Toleranzfaktor:

MD35150 \$MA_SPIND_DES_VELO_TOL

Beispiel:

MD35150 \$MA_SPIND_DES_VELO_TOL = 0,1

⇒ Die Spindel-Istdrehzahl darf $\pm 10\%$ von der Solldrehzahl abweichen.

Liegt die Spindel-Istdrehzahl innerhalb des Toleranzbereichs, wird das folgende NC/PLC-Nahtstellensignal auf "1" gesetzt:

DB31, ... DBX83.5 (Spindel im Sollbereich) = 1

Sonderfall:

Wird die Spindeldrehzahltoleranz auf "0" gesetzt, dann wird DB31, ... DBX83.5 (Spindel im Sollbereich) permanent auf "1" gesetzt und es findet keine Bahnbeeinflussung statt.

Drehzahländerung

Die Bahnbeeinflussung findet nur zu Beginn des Verfahrsatzes und nur bei programmierter Drehzahländerung statt. Wird der Drehzahltoleranzbereich z. B. durch Überlastung verlassen, dann führt dies nicht automatisch zum Stillstand der Bahnbewegung.

17.8.4 Minimale/Maximale Drehzahl der Getriebestufe

Minimale Drehzahl

Die minimale Drehzahl der Getriebestufe einer Spindel wird projiziert im Maschinendatum:

MD35140 \$MA_GEAR_STEP_MIN_VELO_LIMIT[<n>]

Die unter Berücksichtigung des Overrides erzeugten Drehzahlsollwerte unterschreiten die minimale Drehzahl nicht.

Wird ein S-Wert programmiert, der kleiner als die minimale Drehzahl ist, so wird die Soll-drehzahl auf die Minimaldrehzahl angehoben und das folgende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX83.2 (Soll-Drehzahl erhöht)

Die minimale Drehzahl der Getriebestufe wirkt nur im Drehzahlbetrieb und kann nur unterschritten werden durch:

- Spindel-Override 0%
- M5
- S0
- DB31, ... DBX4.3 (Spindel Halt)
- DB31, ... DBX2.1 (Reglerfreigabe) wegnehmen
- DB21, ... DBX7.7 (Reset)
- DB31, ... DBX2.2 (Restweglöschen/Spindel-Reset)
- DB31, ... DBX18.5 (Pendeldrehzahl)
- DB21, ... DBX7.4 (NC-Stop Achsen plus Spindeln)
- DB31, ... DBX1.3 (Achs-/Spindelsperre)
- DB31, ... DBX16.7 (S-Wert löschen)

Maximale Drehzahl

Die maximale Drehzahl der Getriebestufe einer Spindel wird projiziert im Maschinendatum:

MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[<n>]

Die unter Berücksichtigung des Overrides erzeugten Drehzahlsollwerte werden auf diese Drehzahl begrenzt.

Im Falle der Begrenzung der Drehzahl wird das folgende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX83.1 (Soll-Drehzahl begrenzt)

17.8.5 Diagnose von Spindeldrehzahlbegrenzungen

Systemvariablen

Die wirksamen/begrenzenden Spindelparameter können über folgende Systemvariablen gelesen werden. Die Systemvariablen müssen mit der Spindelnummer indexiert werden und liefern nur Werte, die im Drehzahlsteuer- und Spindelpositionierbetrieb relevant sind.

Systemvariable	Bedeutung
\$AC_SMAXVELO[<n>]	Maximal mögliche Spindeldrehzahl [U/min].
\$AC_SMAXVELO_INFO[<n>]	Kennung für das drehzahlbegrenzende Datum.
\$AC_SMINVELO[<n>]	Minimal mögliche Spindeldrehzahl [U/min].
\$AC_SMINVELO_INFO[<n>]	Kennung für das drehzahlhebende Datum.
\$AC_SMAXACC[<n>]	Maximale mögliche Spindelbeschleunigung [U/s ²].
\$AC_SMAXACC_INFO[<n>]	Kennung für das beschleunigungsbegrenzende Datum.
\$AC_SPIND_STATE[<n>]	Status der Spindel im Drehzahlsteuerbetrieb.
<n>: 0, 1, 2, ... maximale Spindelnummer (0: aktuelle Masterspindel des Kanals)	

Literatur

Die ausführliche Beschreibung der Systemvariablen findet sich in:

Listenhandbuch Systemvariablen

NC/PLC-Nahtstellensignale

Die Begrenzung bzw. Anhebung der Spindeldrehzahl wird durch folgende NC/PLC-Nahtstellensignale angezeigt:

- DB31, ... DBX83.1 (Soll-Drehzahl begrenzt)
- DB31, ... DBX83.2 (Soll-Drehzahl erhöht)

Randbedingungen

Spindelbetriebsart

Die von den Systemvariablen gelieferten Werte sind von der Spindelbetriebsart abhängig:

- Drehzahlsteuerbetrieb:
Alle Systemvariablen liefern aktuelle Werte.
- Positionierbetrieb:
Die Systemvariablen \$AC_SMAXVELO, \$AC_SMAXACC und \$AC_SPIND_STATE liefern gültige Werte. Die Systemvariablen \$AC_SMINVELO und \$AC_SMINVELO_INFO liefern die Daten, die beim Wechsel in den Drehzahlsteuerbetrieb wirksam werden.
- Achsbetrieb (z. B. wenn die Spindel von einer Transformation TRANSMIT, TRACYL,... verwendet wird oder als Zusatzachse einer Bahnbewegung folgt):
Die Systemvariable \$AC_SPIND_STATE kann auch im Achsbetrieb verwendet werden. Für die Dynamikdaten stehen im Achsbetrieb eigene Systemvariable zur Verfügung: \$AA_VMAXM, \$AA_VMAXB und \$AA_VLFCT.

Satzsuchlauf SERUPRO

Bei Satzsuchlauf SERUPRO ergibt sich folgendes Steuerungsverhalten:

- Die Systemvariable \$AC_SMAXVELO / \$AC_SMAXACC liefert die maximal darstellbare Geschwindigkeit / Beschleunigung.
- \$AC_SMAXVELO_INFO und \$AC_SMAXACC_INFO liefern den Wert "0" (keine Limitierungen aktiv).
- \$AC_SMINVELO und \$AC_SMINVELO_INFO liefern Daten wie bei der normalen Programmabarbeitung.
- \$AC_SPIND_STATE gibt die Zustände zurück, die sich bei SERUPRO einstellen.

Beispiel

Die wirksamen/begrenzenden Spindelparameter von Spindel 1, werden zyklisch in R-Parameter geschrieben und auf der HMI-Bedienoberfläche im Bedienbereich "Parameter" > "Anwendervariable" als R-Parameter angezeigt.

Programmcode

```
N05 IDS=1 WHENEVER TRUE DO $R10=$AC_SMAXVELO[1]
N10 IDS=2 WHENEVER TRUE DO $R11=$AC_SMAXVELO_INFO[1]
N15 IDS=3 WHENEVER TRUE DO $R12=$AC_SMINVELO[1]
N20 IDS=4 WHENEVER TRUE DO $R13=$AC_SMINVELO_INFO[1]
N25 IDS=5 WHENEVER TRUE DO $R14=$AC_SPIND_STATE[1]
```

Siehe auch

Spindel im Sollbereich (Seite 1390)

17.8.6 Maximale Drehzahl der Spindel**Maximale Spindeldrehzahl: parametrierbarer maschinenbedingter Grenzwert**

Die maximale maschinenbedingte Spindeldrehzahl, z.B. zum Schutz des Spindelfutters oder des Werkzeugs, wird über folgendes Maschinendatum parametrier:

MD35100 \$MA_SPIND_VELO_LIMIT (Maximale Spindeldrehzahl)

Die Spindeldrehzahl wird von der NC istwertseitig, d.h. unter Berücksichtigung der aktuellen Getriebestufe, überwacht.

Hinweis

Maschinenhersteller

Es wird empfohlen, eine Änderung der maximalen Spindeldrehzahl im Maschinendatum MD35100 nur bei stillgesetzter Spindel vorzunehmen. Dies ist insbesondere bei Spindeln zu beachten, die über NC-Reset hinaus aktiv sind (siehe auch MD35040 SPIND_ACTIVE_AFTER_RESET). Andernfalls kann es zum Auftreten des Alarms 22100 kommen.

Literatur

- Herstellerdokumentation: Listenhandbuch "Ausführliche Maschinendaten Beschreibung"
 - Anwenderdokumentation: Diagnosehandbuch
-

Reaktionen bei Überschreitung

Überschreitet die Istdrehzahl einer Spindel die parametriert maximale Spindeldrehzahl um mehr als den Toleranzwert,

```
"Istdrehzahl der Spindel" > MD35100 $MA_SPIND_VELO_LIMIT (Maximale
Spindeldrehzahl) +
MD35150 $MA_SPIND_DES_VELO_TOL (Spindeldrehzahl-Toleranzwert)
```

erfolgen NC-seitig folgende Reaktionen:

- DB31, ... DBX83.0 = 1 (Drehzahlgrenze überschritten)
 - Alarm 22100 "Futterdrehzahl überschritten" wird angezeigt
 - Alle Achsen und Spindeln des Kanals werden stillgesetzt
-

Hinweis

Rundachse / Spindel

Wird eine Spindel zeitweilig auch als Rundachse betrieben, wird der Alarm 22100 "Futterdrehzahl überschritten" angezeigt, wenn gilt:

```
"Istdrehzahl der Rundachse" > MD35100 $MA_SPIND_VELO_LIMIT (Maximale
Spindeldrehzahl) +
MD35150 $MA_SPIND_DES_VELO_TOL (Spindeldrehzahl-Toleranzwert)
```

Abhilfe: Angleich der maximalen Spindeldrehzahl auf maximale Rundachsdrehzahl während des Rundachsbetriebs:

```
MD35100 $MA_SPIND_VELO_LIMIT = MD32000 $MA_MAX_AX_VELO (Maximale
Drehzahl der Rundachse)
```

Damit die Änderung wirksam wird, muss ein Reset ausgelöst werden.

Maximale Spindeldrehzahl: Einstellbarer prozessbedingter Grenzwert

Die maximale prozessbedingte Spindeldrehzahl wird über folgendes Settingdatum eingestellt:

SD43235 \$SA_SPIND_USER_VELO_LIMIT (Maximale Spindeldrehzahl)

Eine Änderung kann durch Schreiben des Settingdatums vorgenommen werden über:

- Programmierung in einem NC-Programm
- Manuell vom Maschinenbediener über die Bedienoberfläche

Eine Änderung wird sofort wirksam.

Begrenzung der Solldrehzahl

Die Steuerung begrenzt gegebenenfalls eine programmierte Spindelsolldrehzahl auf den im Settingdatum angegebenen Wert. Die Begrenzung wird dann über folgende Systemvariable angezeigt:

\$AC_SMAXVELO_INFO[<Spindelnummer>] == 21

17.8.7 Maximale Gebergrenzfrequenz



VORSICHT

Grenzwertüberschreitung

Die max. Gebergrenzfrequenz des Spindellageistwertgebers wird von der Steuerung überwacht (Überschreiten möglich). Der Werkzeugmaschinenhersteller muss durch die Auslegung der Komponenten Spindelmotor, Getriebe, Messgetriebe und Geber und der dazugehörigen Maschinendaten sicherstellen, dass die max. Drehzahl (mechanische Grenzdrehzahl) des Spindellageistwertgebers nicht überschritten werden kann.

Max. Geberfrequenz überschritten

Erreicht die Spindel in der Spindelbetriebsart Steuerbetrieb eine Drehzahl (großer S-Wert programmiert), die über der max. Gebergrenzfrequenz liegt (die max. Drehzahl des Gebers darf dabei nicht überschritten sein), geht die Synchronisation verloren. Die Spindel dreht weiter, jedoch mit verminderter Funktionalität.

Bei folgenden Funktionen wird die Spindeldrehzahl reduziert, bis das aktive Messsystem wieder unterhalb der Gebergrenzfrequenz arbeitet:

- Gewindeschneiden (G33, G34, G35)
- Gewindebohren ohne Ausgleichsfutter (G331, G332)
- Umdrehungsvorschub (G95)
- Konstante Schnittgeschwindigkeit (G96, G961, G97, G971)
- SPCON (Lagegeregelter Betrieb der Spindel)

Bei Überschreitung der Gebergrenzfrequenz werden für das betreffende Messsystem die NC/PLC-Nahtstellensignale:

DB31, ... DBX60.4 (Referenziert / Synchronisiert 1)

bzw.

DB31, ... DBX60.5 (Referenziert / Synchronisiert 2)

zurückgesetzt und die NC/PLC-Nahtstellensignale:
 DB31, ... DBX60.2 (Gebergrenzfrequenz 1 überschritten)
 bzw.
 DB31, ... DBX60.3 (Gebergrenzfrequenz 2 überschritten)
 gesetzt.

Befindet sich die Spindel im Achsbetrieb, darf die max. Gebergrenzfrequenz nicht überschritten werden.

Die max. Geschwindigkeit (MD32000 \$MA_MAX_AX_VELO) muss unterhalb der max. Gebergrenzfrequenz liegen, ansonsten wird der Alarm 21610 ausgegeben und die Achse stillgesetzt.

Maximale Gebergrenzfrequenz unterschritten

Wurde die max. Gebergrenzfrequenz überschritten und wird anschließend wieder eine Drehzahl erreicht, die unterhalb der max. Gebergrenzfrequenz liegt (kleineren S-Wert programmiert, Spindel-Override-Schalter verändert, etc.), synchronisiert sich die Spindel automatisch mit der nächsten Nullmarke bzw. dem nächsten berührungslosen Näherungsschalter-Signal. Die neue Synchronisation läuft immer für das aktive Lagemesssystem, das seine Synchronisation verloren hat und dessen max. Gebergrenzfrequenz momentan unterschritten ist.

Besonderheiten

Sind folgende Funktionen aktiv, kann die max. Gebergrenzfrequenz nicht überschritten werden:

- Spindelbetriebsart Positionierbetrieb, Achsbetrieb
- Gewindeschneiden (G33, G34, G35)
- Gewindebohren ohne Ausgleichsfutter G331, G332 (gilt nicht für G63)
- Umdrehungsvorschub (G95)
- Konstante Schnittgeschwindigkeit (G96, G961, G97, G971)
- SPCON

17.8.8 Zielpunktüberwachung

Zielpunktüberwachung

Beim Positionieren (Spindel befindet sich in der Spindelbetriebsart Positionierbetrieb) wird überwacht, wie weit die Spindel (mit ihrer Istposition) von der programmierten Spindelsollposition (Zielpunkt) entfernt ist.

Dazu können in den Maschinendaten:
 MD36000 \$MA_STOP_LIMIT_COARSE (Genauhaltgrenze grob)
 und
 MD36010 \$MA_STOP_LIMIT_FINE (Genauhaltgrenze fein)

zwei Grenzwerte als inkrementeller Weg von der Spindelsollposition ausgehend vorgegeben werden.

Die Genauigkeit der Spindelpositionierung ist unabhängig von den zwei Grenzwerten immer so gut, wie durch den angeschlossenen Spindelmessgeber, die Lose, die Getriebeübersetzung etc. vorgegeben wird.

Genauhaltfenster parametersatzabhängig

Parametersatzabhängig unterschiedliche Genauhaltfenster können projiziert werden. Damit ist es möglich, beim Achsbetrieb und beim Spindelpositionieren mit unterschiedlichen Genauigkeiten zu arbeiten. Für das Spindelpositionieren kann die Projektierung der Genauhaltfenster für jede Getriebestufe getrennt vorgenommen werden.

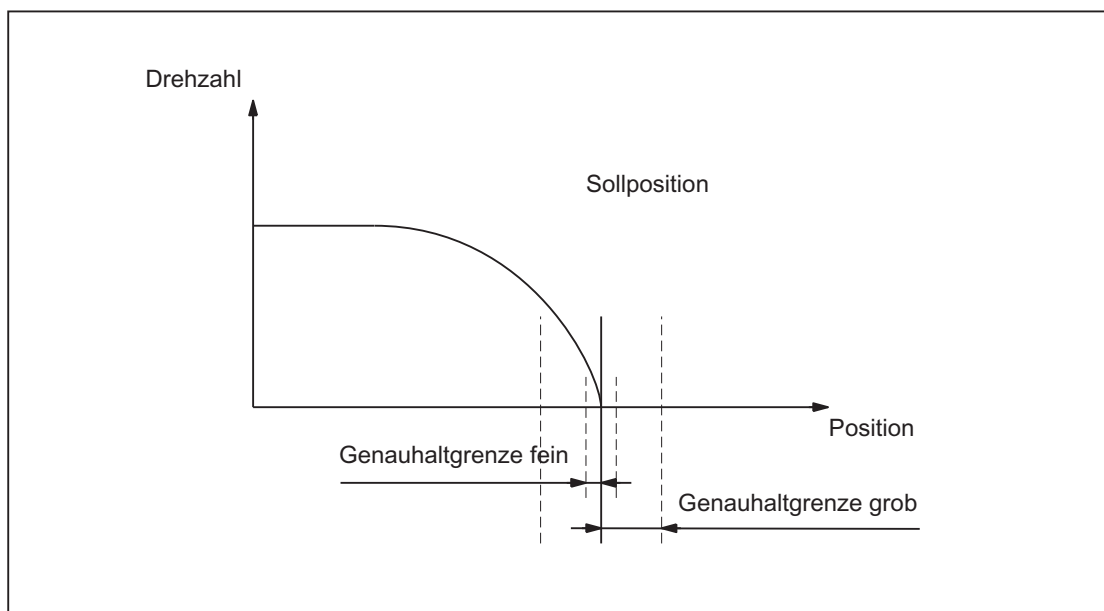


Bild 17-9 Genauhalt-Zonen einer Spindel

DB31, ... DBX60.7 und DB31, ...DBX60.6 (Position erreicht mit Genauhalt grob/fein)

Die zwei durch die Maschinendaten:

MD36000 \$MA_STOP_LIMIT_COARSE (Genauhaltgrenze grob)

und

MD36010 \$MA_STOP_LIMIT_FINE (Genauhaltgrenze fein)

festgelegten Grenzwerte werden mit den NC/PLC-Nahtstellensignalen:

DB31, ... DBX60.7 (Position erreicht mit Genauhalt grob)

und

DB31, ... DBX60.6 (Position erreicht mit Genauhalt fein)

an die PLC ausgegeben.

Satzwechsel bei SPOS und M19

Beim Positionieren der Spindel mit SPOS oder M19 erfolgt der Satzwechsel abhängig von der Zielpunktüberwachung mit dem NC/PLC-Nahtstellensignal: DB31, ... DBX60.6 (Position erreicht mit Genauhalt fein).

Dabei müssen auch alle anderen im Satz programmierten Funktionen ihr Satzendekriterium erreicht haben (z.B. alle Hilfsfunktionen von der PLC quittiert).

Bei SPOSA ist der Satzwechsel unabhängig von der Zielpunktüberwachung.

17.8.9 M40: Automatische Getriebestufenauswahl bei Drehzahlen außerhalb projektierter Schaltschwellen

Funktion

Bei aktivem M40 erfolgt eine automatische Getriebestufenauswahl auch dann, wenn die programmierte Spindeldrehzahl S... außerhalb der projektieren Schaltschwellen liegt.

Dabei sind folgende Fälle zu unterscheiden:

- **Programmierte Drehzahl zu groß**
Die programmierte Spindeldrehzahl ist höher als die projektierter Maximaldrehzahl der zahlenmäßig größten Getriebestufe:
S... > MD35110 \$MA_GEAR_STEP_MAX_VELO[<n>]
In diesem Fall wird in die **höchste Getriebestufe** (gemäß MD35090 \$MA_NUM_GEAR_STEPS) geschaltet.
- **Programmierte Drehzahl zu klein**
Die programmierte Spindeldrehzahl ist kleiner als die projektierter Minimaldrehzahl der ersten Getriebestufe:
S... < MD35120 \$MA_GEAR_STEP_MIN_VELO[1]
In diesem Fall wird in die **erste Getriebestufe** geschaltet.
- **Programmierte Drehzahl = 0**
Bei der Programmierung der Drehzahl 0 (S0) ist das Verhalten abhängig von der Projektierung der Minimaldrehzahl der ersten Getriebestufe MD35120 \$MA_GEAR_STEP_MIN_VELO[1]:
 - Ist MD35120 \$MA_GEAR_STEP_MIN_VELO[1] = 0 projektierter, dann wird bei Programmierung von S0 in die **erste Getriebestufe** geschaltet.
 - Ist MD35120 \$MA_GEAR_STEP_MIN_VELO[1] > 0 projektierter, dann wird bei Programmierung von S0 **kein Getriebestufenwechsel** durchgeführt und die letzte Getriebestufe bleibt weiterhin aktiv. Damit bleibt es möglich, die Spindel mit S0 (anstatt von M5) zu stoppen ohne einen Getriebestufenwechsel auszulösen.

Wirksamkeit

Das Schalten in die höchste Getriebestufe oder in die erste Getriebestufe bei automatischer Getriebestufenauswahl (M40) ist bei Programmierungen von Spindeldrehzahlen S... über das Teileprogramm, in Synchronaktionen und bei Vorgaben über PLC FC18 wirksam.

Bei Drehzahlprogrammierungen aus dem Teileprogramm für Gewindebohren mit G331 wird das Verhalten auch für den zweiten Datensatz zur Getriebestufenauswahl unterstützt (Voraussetzung: MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE, Bit 5 = 1).

Randbedingungen

Freigabe des Getriebestufenwechsels

Voraussetzung für die Funktion ist die generelle Freigabe des Getriebestufenwechsels über die Maschinendaten:

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE (Getriebestufenwechsel parametrieren)

MD35090 \$MA_NUM_GEAR_STEPS (Anzahl eingerichteter Getriebestufen)

Beispiel

Automatische Getriebestufenauswahl M40 ist Grundstellung nach NC-Reset.

Teileprogramm:

Programmcode	Kommentar
...	
N15 S3500 M3	; S3500 ist größer als MD35110 der 2.Getriebestufe. Es wird in die 2. Getriebestufe geschaltet.
...	
N50 S0 M3	; Spindel wird gestoppt, S0 fordert keinen Getriebestufenwechsel an (Sonderbehandlung S0).
...	
N95 S5 M3	; S5 ist kleiner als MD35120 der 1.Getriebestufe. Es wird in die 1.Getriebestufe geschaltet.
...	

Projektierungsdaten für Spindel 1 (AX5):

MD35010 \$MA_GEAR_STEP_CHANGE_ENABLE[AX5] = 1	; Getriebestufenwechsel freigeben
MD35090 \$MA_NUM_GEAR_STEPS[AX5] = 2	; Anzahl vorhandener Getriebestufen
MD35110 \$MA_GEAR_STEP_MAX_VELO[1,AX5] = 500	; Obere Schaltschwelle für Getriebestufe 1
MD35120 \$MA_GEAR_STEP_MIN_VELO[1,AX5] = 10	; Untere Schaltschwelle für Getriebestufe 1
MD35110 \$MA_GEAR_STEP_MAX_VELO[2,AX5] = 2000	; Obere Schaltschwelle für Getriebestufe 2
MD35120 \$MA_GEAR_STEP_MIN_VELO[2,AX5] = 500	; Untere Schaltschwelle für Getriebestufe 2

17.9 Spindel mit SMI 24 (Weiss-Spindel)

17.9.1 Allgemeine Informationen

Um die Sensordaten der Spindel in der Steuerung verarbeiten zu können, mussten bisher die Sensoren an I/O-Baugruppen verdrahtet und per Feldbus (PROFIBUS DP bzw. PROFINET IO) an die PLC übertragen werden.

Bei einer Spindel mit SMI 24 (Weiss-Spindel) werden die Sensordaten per DRIVE-CLiQ zum Antrieb übertragen und stehen dort in Antriebsparametern zur Verfügung. Bei Verwendung des zyklischen Antriebstelegramms 139 werden die Sensordaten vom Antrieb weiter an die Steuerung übertragen. Dort stehen sie dann in folgenden Systemdaten zur Verfügung:

- Systemvariable
- BTSS-Variable
- NC/PLC-Nahtstellensignale

Voraussetzung

- Die Spindel ist über das Sensormodul SMI 24 per DRIVE-CLiQ mit dem Antrieb verbunden.
- Für die Spindel ist Antriebstelegramm 139 projektiert.

Hinweis

Antriebstelegramm 139

Eine Spindel mit Sensormodul SMI 24 kann prinzipiell auch mit einem anderen Antriebstelegramm betrieben werden. Die Sensordaten werden aber nur mit dem Antriebstelegramm 139 zur Steuerung übertragen.

17.9.2 Sensordaten

Sensoren des Spindelmotors

Die Sensoren der Spindel liefern Informationen über die Spannvorrichtung und die Winkellage der Motorwelle:

- Analog-Sensor S1: Spannzustand
Spannungswert im Bereich von 0 - 10 V abhängig von der Lage der Zugstange.
- Digital-Sensor S4: Kolbenendlage
 - 0 = Kolben nicht in Position
 - 1 = Kolben in Position, d. h. Kolben ist frei
- Digital-Sensor S5: Winkellage der Motorwelle
 - 0 = Motorwelle nicht gerichtet
 - 1 = Motorwelle steht auf Position (Voraussetzung: Stillstand der Spindel)

Hinweis**Spindel mit Sensormodul SMI 24 und Achscontainer**

Eine Spindel mit Sensormodul SMI 24 und Antriebstelegramm 139 zur Übertragung der Sensordaten an die Steuerung darf nicht Bestandteil eines Achscontainers sein, dessen Achsen per NCU-Link über mehrere NCUs verteilt sind.

Übertragung der Sensordaten

Die Sensordaten werden vom Sensormodul SMI 24 im zyklischen Antriebstelegramm 139 als Prozessdaten 11 - 14 zur Steuerung übertragen. Das Antriebstelegramm 139 basiert auf dem Antriebstelegramm 136, bei dem statt der Daten des 2. Gebers die Sensordaten übertragen werden. Eine detaillierte Beschreibung des Antriebstelegramms 139 findet sich in:

Literatur:

SINAMICS S120/S150 Listenhandbuch; Funktionspläne, Kapitel: PROFIdrive

Systemdaten: Sensordaten

Die Sensordaten können in der Steuerung über folgende Systemdaten gelesen werden:

Bedeutung	Systemvariable \$VA_	NC/PLC- Nahtstelle DB31, ...	BTSS-Variable	Antriebspa- rameter
Sensorkonfiguration	MOT_SENSOR_CONF[<Achse>]	DBB132, DBB133	vaMotSensorConf	r5000
Spannzustand ¹⁾	MOT_CLAMPING_STATE[<Achse>]	DBW134	vaMotClampingState	r5001
Messwert Analogsens- sor S1 ^{1) 2)}	MOT_SENSOR_ANA[<Achse>]	DBW136	vaMotSensorAna	r5002
Status Digitalsensoren	MOT_SENSOR_DIGI[<Achse>]	DBB138, DBB139	vaMotSensorDigi	r5003
<Achse>: Maschinenachse: AX1 ... AXn oder Spindelname: S1 ... Sm				

¹⁾ siehe Kapitel "Spannzustand (Seite 1403)"

²⁾ Sensor S1: 0 - 10 V

Analoger Istwert: 10000 Inkremente, Auflösung 1 mV

Beispiel:

SIMATIC S7-Eingangsbaugruppe: 27648 Inkremente, Auflösung 0,36 mV

Anpassfaktor bei Umstellung auf eine Spindel mit SMI 24: $(10000 \text{ Inkr.} \cdot 1 \text{ mV}) / (27648 \text{ Inkr.} \cdot 0,36 \text{ mV}) = 1,00469393$

Ausführliche Systemdatenbeschreibung

NC/PLC-Nahtstellensignale: Listenhandbuch NC-Variablen und Nahtstellensignale

Systemvariable: Listenhandbuch Systemvariable

BTSS-Variable:	Listenhandbuch 2; Variablen
Antriebsparameter:	SINAMICS S120/S150 Listenhandbuch

17.9.3 Spannzustand

Der Sensor S1 liefert einen analogen Spannungswert 0 V - 10 V abhängig von der Lage der Spannvorrichtung. Der Spannungswert steht in den Systemdaten zur anwenderseitigen Auswertung des Spannzustandes zur Verfügung.

Hinweis

Die nachfolgend beschriebene Auswertung des Sensors S1 zur Erzeugung des Zustandswertes für den Spannzustand und die Begrenzung der Spindeldrehzahl, erfolgen nur, wenn im Antriebsparameter r5000 folgende Zustandswerte angezeigt werden:

- r5000.0 == 1: Sensorik vorhanden
- r5000.1 == 1: Sensor S1 (Spannzustand) vorhanden
- r5000.10 == 1: Zustandswerte werden gebildet, Drehzahlbegrenzungen p5043 aktiv

Siehe auch Kapitel "Sensordaten (Seite 1401)", Absatz: "Systemdaten: Sensordaten"

Zustandswert

Zur Vereinfachung der Auswertung steht der Spannzustand in den Systemdaten auch als Zustandswert 0 - 11 zu Verfügung.

Ein Spannungsbereich entspricht einem bestimmten Spannzustand. Die Spannungsbereiche können über den Antriebsparameter p5041[0...5] eingestellt werden.

Für die Spannungsbereiche kann über den Antriebsparameter p5040 zusätzlich eine Spannungstoleranz eingestellt werden.

Hinweis

Die Spannungsbereiche \pm Spannungstoleranz dürfen sich nicht überschneiden.

Drehzahlgrenzen

Für die Spannzustände mit den Zustandswerten 3 - 10 können über den Antriebsparameter p5043[0...6] Drehzahlgrenzwerte vorgegeben werden. In den anderen Spannzuständen (Zustandswerte 1, 2 und 11) ist als Drehzahlgrenzwert 0 [1/min] fest vorgegeben.

Die Steuerung begrenzt in den verschiedenen Spannzuständen die Spindeldrehzahl auf den jeweiligen Grenzwert.

Hinweis

Ändern der Drehzahlgrenzen

Eine Änderung der Drehzahlgrenzen in den Antriebsparametern p5043[0...6] wird in der Steuerung erst wirksam (Begrenzung der Spindeldrehzahl auf die neue Drehzahlgrenze) nach:

- Power On-Reset oder Aus/Einschalten der Steuerung
- Abwahl des Zustandes "Parken" für die Spindel (siehe Kapitel "Parken einer Maschinenachse (Seite 122)")

Zusammenhang: Zustandswert, Spannungsbereich und Drehzahlgrenze

Zustandswert ¹⁾	Spannzustand	Spannungsbereich ²⁾		Drehzahl-grenze
		Obergrenze	Untergrenze	
0	Sensor S1 nicht vorhanden oder Zustandswerte inaktiv	---	---	---
1	Zustandsinitialisierung läuft	---	---	³⁾
2	Gelöst mit Meldung (Fehlerzustand)	---	p5041[0] + p5040	³⁾
3	Gelöst	p5041[0]	p5041[1]	p5043[0]
4	Spannend mit Werkzeug	---	---	p5043[1]
5	Lösend mit Werkzeug	---	---	p5043[2]
6	Lösend ohne Werkzeug	---	---	p5043[3]
7	Gespannt mit Werkzeug UND S4 == 0	p5041[2]	p5041[3]	p5043[4]
8	Gespannt mit Werkzeug UND S4 == 1			p5043[4]
9	Spannend ohne Werkzeug	---	---	p5043[5]
10	Gespannt ohne Werkzeug	p5041[4]	p5041[5]	p5043[6]
11	Gespannt mit Meldung (Fehlerzustand)	p5041[5] - p5040	---	³⁾

¹⁾ Der Zustandswert kann in der Steuerung über folgende Systemdaten gelesen werden:

- Systemvariable: \$VA_MOT_CLAMPING_STATE[<Achse>]
- NC/PLC-Nahtstelle: DB31, ... DBW134
- BTSS-Variable: vaMotClampingState
- Antriebsparameter: r5001

²⁾ p5041[0...5]: Spannungsschwellwerte, p5040: Spannungsschwellwerte Toleranz

³⁾ Drehzahlgrenzwert fest vorgegeben: 0 [1/min]

17.9.4 Weitere Antriebsparameter

p5042:Transitionszeit

Im Antriebsparameter p5042 können für die Spannzustandserkennung folgende Zeiten eingestellt werden:

- p5042[0]: Stabilisierungszeit für "Gespannt mit Werkzeug"
Der Spannzustand "Gespannt mit Werkzeug" muss im Spindelmotor mindesten die eingestellte Stabilisierungszeit anstehen, bevor der Zustand an die Steuerung gemeldet wird.
- p5042[1]: Maximalzeit zum Spannen
Der Übergang vom Zustand "Gelöst" zum Zustand "Gespannt mit Werkzeug" oder "Gespannt ohne Werkzeug" darf maximal die eingestellte Zeit dauern.

r5044: Drehzahlbegrenzung aus Spannzzyklus

Im Antriebsparameter r5044 wird die im Spannzustand "Gespannt ohne Werkzeug" wirkende Drehzahlgrenze aus p5043[6] angezeigt.

Ein Wert von 65535 bedeutet, dass die Drehzahlgrenze nicht wirksam ist.

17.10 Randbedingungen

17.10.1 Ändern von Regelparametern

Bei Spindeln, die sich nicht im lagegeregelten Betrieb befinden, werden mit dem Befehl NEWCONF Maschinendatenänderungen auch dann wirksam, wenn die Spindel nicht steht.

Bei Änderungen von Regelparametern kann es nach dem Wirksam werden der neuen Werte zu Drehzahlsollwertsprüngen kommen. Regelparameter sind z.B.:

- MD32200 \$MA_POSCTRL_GAIN (KV-Faktor)
- MD32210 \$MA_POSCTRL_INTEGR_TIME (Nachstellzeit Lageregler)
- MD32410 \$MA_AX_JERK_TIME (Zeitkonstante für den axialen Ruckfilter)

17.11 Beispiele

17.11.1 Automatische Getriebestufenanwahl (M40)

Beispiel

Veranschaulichung der Inhalte der neuen Satzsuchlaufvariablen:
Annahmen zur automatischen Getriebestufenanwahl (M40):

S0...500	1. Getriebestufe
S501..1000	2. Getriebestufe
S1001..2000	3. Getriebestufe

Inhalte der Systemvariablen:

\$P_SEARCH_S	; Aufgesammelte S-Wert
\$P_SEARCH_DIR	; Aufgesammelte Drehrichtung
\$P_SEARCH_GEAR	; Aufgesammelte Getriebestufe

Aufgesammelte	S-Wert:	Drehrichtung:	Getriebestufe:
	; 0/letzte Drehzahl	-5	40/letzte GS
N05 G94 M40 M3 S1000	; 1000	3	40
N10 G96 S222	; 222	3	40
N20 G97	; f (PlanAxPosPCS)*	3	40
N30 S1500	; 1500	3	40
N40 SPOS=0**	; 1500	-19	40
N50 M19**	; 1500	-19	40
N60 G94 G331 Z10 S300	; 300	-19	40
N70 M42	; 300	-19	42
N80 M4	; 300	4	42
N90 M70	; 300	70	42
N100 M3 M40	; 300	3	40
N999 M30			

* f (PlanAxPosPCS): Die Drehzahl ist abhängig von der aktuellen Position der Planachse im Werkstückkoordinatensystem.

** (\$P_SEARCH_SPOS und \$P_SEARCH_SPOSMODE werden beschrieben)

17.12 Datenlisten

17.12.1 Maschinendaten

17.12.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10192	GEAR_CHANGE_WAIT_TIME	Wartezeit auf Quittierung eines Getriebestufenwechsel beim Reorganisieren
10714	M_NO_FCT_EOP	M-Funktion für Spindel aktive nach RESET
12060	OVR_SPIND_IS_GRAY_CODE	Spindel-Override Gray-codiert
12070	OVR_FACTOR_SPIND_SPEED	Bewertung des Spindel-Override-Schalters
12080	OVR_REFERENCE_IS_PROG_FEED	Override-Bezugsgeschwindigkeit
12082	OVR_REFERENCE_IS_MIN_FEED	Festlegung des Bezugs des Bahn-Override
12090	OVR_FUNCTION_MASK	Auswahl von Override-Spezifikationen

17.12.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20090	SPIND_DEF_MASTER_SPIND	Löschstellung Masterspindel im Kanal
20092	SPIND_ASSIGN_TAB_ENABLE	Freigabe/Sperren des Spindelumsetzers
20850	SPOS_TO_VDI	Ausgabe von M19 an die PLC bei SPOS/SPOSA
22400	S_VALUES_ACTIVE_AFTER_RESET	S-Funktion über RESET wirksam

17.12.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30300	IS_ROT_AX	Rundachse
30310	ROT_IS_MODULO	Modulo-Wandlung
31044	ENC_IS_DIRECT2	Geber am Vorsatz-Getriebe
31050	DRIVE_AX_RATIO_DENOM	Nenner Lastgetriebe
31060	DRIVE_AX_RATIO_NUMERA	Zähler Lastgetriebe
31064	DRIVE_AX_RATIO2_DENOM	Nenner Vorsatz-Getriebe
31066	DRIVE_AX_RATIO2_NUMERA	Zähler Vorsatz-Getriebe
31070	DRIVE_ENC_RATIO_DENOM	Nenner Messgetriebe
31080	DRIVE_ENC_RATIO_NUMERA	Zähler Messgetriebe
31122	BERO_DELAY_TIME_PLUS	BERO-Verzögerungszeit in Plus-Richtung
31123	BERO_DELAY_TIME_MINUS	BERO-Verzögerungszeit in Minus-Richtung
32200	POSCTRL_GAIN	K _v -Faktor

Nummer	Bezeichner: \$MA_	Beschreibung
32800	EQUIV_CURRCTRL_TIME	Ersatzzeitkonstante Stromregelkreis fuer Vorsteuerung
32810	EQUIV_SPEEDCTRL_TIME	Ersatzzeitkonstante Drehzahlregelkreis fuer Vorsteuerung
32910	DYN_MATCH_TIME	Zeitkonstante der Dynamikanpassung
34040	REFP_VELO_SEARCH_MARKER	Referenzpunktabschaltgeschwindigkeit
34060	REFP_MAX_MARKER_DIST	Überwachung der Nullmarkenstrecke
34080	REFP_MOVE_DIST	Referenzpunkt Abstand/Zielpunkt bei abstandskodiertem System
34090	REFP_MOVE_DIST_CORR	Referenzpunktverschiebung/Absolutverschiebung abstandskodiert
34100	REFP_SET_POS	Referenzpunkt wert
34200	ENC_REFP_MODE	Referenziermodus
35000	SPIND_ASSIGN_TO_MACHAX	Zuordnung Spindel zu Maschinenachse
35010	GEAR_STEP_CHANGE_ENABLE	Art des Getriebestufenwechsels
35012	GEAR_STEP_CHANGE_POSITION	Getriebestufenwechselposition
35014	GEAR_STEP_USED_IN_AXISMODE	Getriebestufe für den Achsbetrieb bei M70
35020	SPIND_DEFAULT_MODE	Grundstellung der Spindel
35030	SPIND_DEFAULT_ACT_MASK	Aktivieren Spindel Grundstellung
35035	SPIND_FUNCTION_MASK	Einstellen von spindelspezifischen Funktionen
35040	SPIND_ACTIVE_AFTER_RESET	Spindel über Reset aktiv
35090	NUM_GEAR_STEPS	Anzahl eingerichteter Getriebestufen
35092	NUM_GEAR_STEPS2	2. Getriebestufendatensatz: Anzahl eingerichteter Getriebestufen
35100	SPIND_VELO_LIMIT	Maximale Spindeldrehzahl
35110	GEAR_STEP_MAX_VELO[n]	Maximaldrehzahl für den automatischen Getriebestufenwechsel
35112	GEAR_STEP_MAX_VELO2[n]	2. Getriebestufendatensatz: Maximaldrehzahl für den automatischen Getriebestufenwechsel
35120	GEAR_STEP_MIN_VELO[n]	Minimaldrehzahl für den automatischen Getriebestufenwechsel
35122	GEAR_STEP_MIN_VELO2[n]	2. Getriebestufendatensatz: Minimaldrehzahl für den automatischen Getriebestufenwechsel
35130	GEAR_STEP_MAX_VELO_LIMIT[n]	Maximaldrehzahl der Getriebestufe
35135	GEAR_STEP_PC_MAX_VELO_LIMIT[n]	Maximaldrehzahl der Getriebestufe bei Lageregelung
35140	GEAR_STEP_MIN_VELO_LIMIT[n]	Minimaldrehzahl der Getriebestufe
35150	SPIND_DES_VELO_TOL	Spindeldrehzahltoleranz
35160	SPIND_EXTERN_VELO_LIMIT	Spindeldrehzahlbegrenzung von PLC
35200	GEAR_STEP_SPEEDCTRL_ACCEL[n]	Beschleunigung im Drehzahlregelbetrieb
35210	GEAR_STEP_POSCTRL_ACCEL[n]	Beschleunigung im Lageregelbetrieb
35212	GEAR_STEP_POSCTRL_ACCEL2[n]	2. Getriebestufendatensatz: Beschleunigung im Lageregelbetrieb
35220	ACCEL_REDUCTION_SPEED_POINT	Drehzahlgrenze reduzierte Beschleunigung
35230	ACCEL_REDUCTION_FACTOR	reduzierte Beschleunigung

Nummer	Bezeichner: \$MA_	Beschreibung
35300	SPIND_POSCTRL_VELO	Lageregeleinschalt Drehzahl
35350	SPIND_POSITIONING_DIR	Positionierdrehrichtung bei nicht synchronisierter Spindel
35400	SPIND_OSCILL_DES_VELO	Pendeldrehzahl
35410	SPIND_OSCILL_ACCEL	Beschleunigung beim Pendeln
35430	SPIND_OSCILL_START_DIR	Startrichtung beim Pendeln
35440	SPIND_OSCILL_TIME_CW	Pendelzeit für M3-Richtung
35450	SPIND_OSCILL_TIME_CCW	Pendelzeit für M4-Richtung
35500	SPIND_ON_SPEED_AT_IPO_START	Vorschubfreigabe bei Spindel im Sollbereich
35510	SPIND_STOPPED_AT_IPO_START	Vorschubfreigabe bei Spindel steht
35550	DRILL_VELO_LIMIT[n]	Maximaldrehzahlen für das Gewindebohren
35590	PARAMSET_CHANGE_ENABLE	Parametersatzvorgabe durch PLC möglich
36060	STANDSTILL_VELO_TOL	Schwellgeschwindigkeit "Achse/Spindel steht"
36200	AX_VELO_LIMIT	Schwellwert für Geschwindigkeitsüberwachung.

17.12.2 Settingdaten

17.12.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42600	JOG_FEED_PER_REF_SOURCE	Steuerung Umdrehungsvorschub in JOG
42800	SPIND_ASSIGN_TAB	Spindelnummernumsetzer
42900	MIRROR_TOOL_LENGTH	Werkzeuglängenkorrektur spiegeln
42910	MIRROR_TOOL_WEAR	Verschleißwerte der Werkzeuglängenkorrektur spiegeln
42920	WEAR_SIGN_CUTPOS	Verschleißwerte der Bearbeitungsebene spiegeln
42930	WEAR_SIGN	Vorzeichen aller Verschleißwerte invertieren
42940	TOOL_LENGTH_CONST	Beim Wechsel der Bearbeitungsebene (G17 bis G19) die Zuordnung der Werkzeuglängenkomponenten beibehalten

17.12.2.2 Achs-/Spindel-spezifische Settingdaten

Nummer	Bezeichner: \$SA_	Beschreibung
43200	SPIND_S	Drehzahl für Spindelstart über PLC-Schnittstelle
43202	SPIND_CONSTCUT_S	Schnittgeschwindigkeit für Spindelstart über PLC-Schnittstelle
43206	SPIND_SPEED_TYPE	Spindeldrehzahltyp für Spindelstart über PLC-Schnittstelle
43210	SPIND_MIN_VELO_G25	progr. Spindeldrehzahlbegrenzung G25

Nummer	Bezeichner: \$SA_	Beschreibung
43220	SPIND_MAX_VELO_G26	progr. Spindeldrehzahlbegrenzung G26
43230	SPIND_MAX_VELO_LIMS	progr. Spindeldrehzahlbegrenzung G96/G961
43235	SPIND_USER_VELO_LIMIT	maximale Spindeldrehzahl
43240	M19_SPOS	Spindelposition für Spindelpositionieren mit M19
43250	M19_SPOSMODE	Spindelpositionieranfahrmodus für Spindelpositionieren mit M19
43300	ASSIGN_FEED_PER_REF_SOURCE	Umdrehungsvorschub für Positionierachsen/Spindeln

17.12.3 Signale

17.12.3.1 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Vorschub-Override A bis H	DB31,DBX0.0-7	DB380x.DBX0.0-7
Achsen-/Spindelsperre	DB31,DBX1.3	DB380x.DBX1.3
Nachfuhrbetrieb	DB31,DBX1.4	DB380x.DBX1.4
Lagemesssystem 1	DB31,DBX1.5	DB380x.DBX1.5
Lagemesssystem 2	DB31,DBX1.6	DB380x.DBX1.6
Korrektur wirksam	DB31,DBX1.7	DB380x.DBX1.7
Reglerfreigabe	DB31,DBX2.1	DB380x.DBX2.1
Spindel-Reset/Restweg löschen	DB31,DBX2.2	DB380x.DBX2.2
Geschwindigkeits-/Spindeldrehzahlbegrenzung	DB31,DBX3.6	DB380x.DBX3.6
Programmtest Achs-/Spindel Freigabe	DB31,DBX3.7	DB380x.DBX3.7
Istgetriebestufe A bis C	DB31,DBX16.0-2	DB380x.DBX2000.0-2
Getriebe ist umgeschaltet	DB31,DBX16.3	DB380x.DBX2000.3
Spindel neu synchronisieren 1	DB31,DBX16.4	DB380x.DBX2000.4
Spindel neu synchronisieren 2	DB31,DBX16.5	DB380x.DBX2000.5
keine n-Überwachung bei Getriebeumschaltung	DB31,DBX16.6	DB380x.DBX2000.6
S-Wert löschen	DB31,DBX16.7	DB380x.DBX2000.7
Vorschub-Override für Spindel gültig	DB31,DBX17.0	DB380x.DBX2001.0
Spindel neu synchronisieren beim Positionieren 1	DB31,DBX17.4	DB380x.DBX2001.4
Spindel neu synchronisieren beim Positionieren 2	DB31,DBX17.5	DB380x.DBX2001.5
M3/M4 invertieren	DB31,DBX17.6	DB380x.DBX2001.6
Pendeln durch die PLC	DB31,DBX18.4	DB380x.DBX2002.4
Pendelfreigabe (Pendeldrehzahl)	DB31,DBX18.5	DB380x.DBX2002.5
Pendeln Drehrichtung rechts (Solldrehrichtung rechts)	DB31,DBX18.6	DB380x.DBX2002.6
Pendeln Drehrichtung links (Solldrehrichtung links)	DB31,DBX18.7	DB380x.DBX2002.7
Spindel-Override A bis H	DB31,DBX19.0-7	DB380x.DBX2003.0-7

17.12.3.2 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Spindel / Rundachse	DB31,DBX60.0	DB390x.DBX0.0
Gebergrenzfrequenz überschritten 1	DB31,DBX60.2	DB390x.DBX0.2
Gebergrenzfrequenz überschritten 2	DB31,DBX60.3	-
Referenziert/Synchronisiert 1	DB31,DBX60.4	DB390x.DBX0.4
Referenziert/Synchronisiert 2	DB31,DBX60.5	DB390x.DBX0.5
Position erreicht mit Genauhalt grob	DB31,DBX60.6	DB390x.DBX0.6
Position erreicht mit Genauhalt fein	DB31,DBX60.7	DB390x.DBX0.7
Achse/Spindel steht ($n < n_{\min}$)	DB31,DBX61.4	DB390x.DBX1.4
Lageregler aktiv	DB31,DBX61.5	DB390x.DBX1.5
Drehzahlregler aktiv	DB31,DBX61.6	DB390x.DBX1.6
Stromregler aktiv	DB31,DBX61.7	DB390x.DBX1.7
Restauriert 1	DB31,DBX71.4	DB390x.DBX11.4
Restauriert 2	DB31,DBX71.5	DB390x.DBX11.5
Sollgetriebestufe A bis C	DB31,DBX82.0-2	DB390x.DBX2000.0-2
Getriebe umschalten	DB31,DBX82.3	DB390x.DBX2000.3
Drehzahlgrenze überschritten	DB31,DBX83.0	DB390x.DBX2001.0
Soll-Drehzahl begrenzt	DB31,DBX83.1	DB390x.DBX2001.1
Soll-Drehzahl erhöht	DB31,DBX83.2	DB390x.DBX2001.2
Spindel im Sollbereich	DB31,DBX83.5	DB390x.DBX2001.5
Ist-Drehrichtung rechts	DB31,DBX83.7	DB390x.DBX2001.7
Gewindebohren ohne Ausgleichsfutter aktiv	DB31,DBX84.3	DB390x.DBX2002.3
aktive Spindelbetriebsart Synchronbetrieb	DB31,DBX84.4	DB390x.DBX2002.4
aktive Spindelbetriebsart Positionierbetrieb	DB31,DBX84.5	DB390x.DBX2002.5
aktive Spindelbetriebsart Pendelbetrieb	DB31,DBX84.6	DB390x.DBX2002.6
aktive Spindelbetriebsart Steuerbetrieb	DB31,DBX84.7	DB390x.DBX2002.7
Spindel in Position wirklich erreicht	DB31,DBX85.5	DB390x.DBX2003.5
M-Funktion für Spindel	DB31,DBB86-87	DB370x.DBD0000
S-Funktion für Spindel	DB31,DBB88-91	DB370x.DBD0004
Sensorik vorhanden	DB31,DBX132.0	DB390x.DBX7000.0
Sensor S1 vorhanden (Spannzustand)	DB31,DBX132.1	DB390x.DBX7000.1
Sensor S4 vorhanden (Kolbenendlage)	DB31,DBX132.4	DB390x.DBX7000.4
Sensor S5 vorhanden (Winkellage der Motorwelle)	DB31,DBX132.5	DB390x.DBX7000.5
Zustandswert wird gebildet, Drehzahlbegrenzung p5043 aktiv	DB31,DBX133.2	DB390x.DBX7001.2
Spannzustand	DB31,DBW134	DB390x.DBW7002
Analogwert: Spannzustand	DB31,DBW136	DB390x.DBW7004
Sensor S4 (Kolbenendlage)	DB31,DBX138.4	DB390x.DBX7006.4
Sensor S5 (Winkellage der Motorwelle)	DB31,DBX138.5	DB390x.DBX7006.5

V1: Vorschübe

18.1 Kurzbeschreibung

Vorschubarten

Der Vorschub bestimmt die Bearbeitungsgeschwindigkeit (Achs- bzw. Bahngeschwindigkeit) und wird bei jeder Interpolationsart auch unter Berücksichtigung von Werkzeugkorrekturen auf der Kontur oder auf der Werkzeugmittelpunktbahn (abhängig von G-Befehle) eingehalten.

Zur optimalen Anpassung an die verschiedenen technologischen Anwendungen (Drehen, Fräsen, Bohren, ...) sind folgende Vorschubarten möglich:

- Eilgangvorschub (G0)
- Zeitreziproker Vorschub (G93)
- Linearvorschub (G94)
- Umdrehungsvorschub (G95)
- Konstante Schnittgeschwindigkeit (G96, G961)
- Konstante Drehzahl (G97, G971)
- Vorschub bei Gewindeschneiden (G33, G34, G35)
- Vorschub bei Gewindebohren ohne Ausgleichsfutter (G331, G332)
- Vorschub bei Gewindebohren mit Ausgleichsfutter (G63)
- Vorschub für Fase/Rundung FRC, FRCM
- Satzweiser Vorschub FB

Achszuordnung der Vorschübe

Zur Anpassung an die verschiedenen technologischen Anforderungen können Vorschübe den Achsen variabel zugeordnet werden.

Folgende Varianten sind möglich:

- Getrennte Vorschübe für Arbeitsebene und Zustellachse
- Variable Achszuordnung für Bahnvorschub
- Vorschub für Positionierachsen

Vorschubbeeinflussung

Zur Anpassung an geänderte technologische Gegebenheiten während der Bearbeitung oder für Testzwecke kann der programmierte Vorschub verändert werden:

- Über die Maschinensteuertafel
- Über die Bedientafelfront

- Über die PLC
- Per Programmbefehl

18.2 Bahnvorschub F

Bahnvorschub F

Der Bahnvorschub stellt die geometrische Summe der Geschwindigkeitskomponenten der beteiligten Achsen dar. Er ergibt sich also aus den Einzelbewegungen der miteinander interpolierenden Achsen.

In der Grundeinstellung werden die axialen Geschwindigkeiten der Geometrieachsen, die programmiert sind, verwendet. Mit dem Befehl `FGROUP` kann man erreichen, dass andere Geometrie- und/oder Synchronachsen bei der Berechnung des Bahnvorschubs berücksichtigt werden.

Der Bahnvorschub F bestimmt die Bearbeitungsgeschwindigkeit und wird bei jeder Interpolationsart auch unter Berücksichtigung von Werkzeugkorrekturen eingehalten. Der unter der Adresse F programmierte Wert bleibt in einem Programm erhalten, bis ein neuer F-Wert bzw. eine neue Vorschubart programmiert wird.

Wertebereich für Bahnvorschub F

Siehe Funktionsbeschreibung G2: "Geschwindigkeiten, Soll-/ Istwertsysteme, Regelung", Kapitel: "Geschwindigkeiten (Seite 341)".

F-Wert an PLC-Nahtstelle

Der F-Wert des aktuellen Bahnvorschubs wird in die kanalspezifische PLC-Nahtstelle für Hilfsfunktionen (DB21, ... DBB158 bis 193) eingetragen.

Erläuterungen der zugehörigen Nahtstellensignale (Änderungssignal, F-Wert) siehe Funktionsbeschreibung "H2: Hilfsfunktionsausgaben an PLC (Seite 399)".

Vorschub bei Übergangskreis

Literatur:
Programmierhandbuch Grundlagen

Vorschub für innen- und außengekrümmte Bahnstücke

Bei Kreissätzen oder gleichsinnig gekrümmten Spline-Sätzen und eingeschalteter Werkzeugradiuskorrektur (G41/G42) kann der programmierte Vorschub an der Mittelpunktbahn oder an der Kontur (in Abhängigkeit von innen- und außengekrümmten Bahnstücken) wirksam werden.

Dazu existiert eine Gruppe von G-Befehle:

- CFTCP
Programmierter Vorschub an der Mittelpunktsbahn wirksam.
- CFC
Programmierter Vorschub an der Kontur wirksam.
- CFCIN
Programmierter Vorschub an der Kontur nur bei Innenkrümmung wirksam.

Literatur:

Programmierhandbuch Grundlagen

Maximale Bahngeschwindigkeit

Die maximale Bahngeschwindigkeit ergibt sich aus den Maximalgeschwindigkeiten der beteiligten Linear- bzw. Rundachsen (MD32000 \$MA_MAX_AX_VELO), d. h. die Achse mit der niedrigsten Maximalgeschwindigkeit bestimmt die maximale Bahngeschwindigkeit. Diese kann nicht überschritten werden.

Ist G0 programmiert, so wird mit der Bahngeschwindigkeit gefahren, die sich durch die Begrenzung von MD32000 \$MA_MAX_AX_VELO ergibt.

Grenzgeschwindigkeit für Bahnachsen

Zusätzlich kann für Bahnachsen (Geometrie- und Synchronachsen) mit der Anweisung FL[<Achse>] eine Grenzgeschwindigkeit für die jeweilige Achse programmiert werden.

Damit können getrennte Vorschübe für Arbeitsebene und Zustellachse programmiert werden. Das bedeutet, dass sowohl für die bahnbezogene Interpolation als auch für die Zustellachse eine Vorschubgeschwindigkeit vorgegeben wird. Als Zustellachse wird die senkrecht zur angewählten Bearbeitungsebene wirkende Achse bezeichnet. Die zustellachsspezifische Vorschubangabe begrenzt programmierbar die Achsgeschwindigkeit und damit die Bahngeschwindigkeit. Es ist dabei zu beachten, dass keine Koordinatendrehungen durch Frames berücksichtigt werden. Dies bedeutet, dass die Zustellachse eine Achse des Basiskoordinatensystems sein muss. Mit der Funktion kann z. B. berücksichtigt werden, dass ein Fräser auf der Stirnseite eine geringere Schnittleistung aufweist als über den Fräserumfang.

Programmierbeispiel:

Programmcode	Kommentar
... G94 ...	; Anwahl des Vorschubtyps (mm/min)
X30 Y20 F200	; Bahnvorschub = 200 mm/s
FL[Z]=50 Z-30	; Max. Vorschub für Z-Achse: 50 mm/s

Niedrig auflösende Geber

Bei der Verwendung von niedrig auflösenden Gebern können mit geglätteten Istwerten stetigere Bahn- bzw. Achsbewegungen erzielt werden. Je größer dabei die Zeitkonstante eingestellt wird, desto besser ist die Glättung der Istwerte und umso größer ist der Nachlauf.

MD34990 \$MA_ENC_ACTVAL_SMOOTH_TIME[<Achse>] (Glättungszeitkonstante für Istwerte)

Geglättete Istwerte werden verwendet bei:

- Gewindeschneiden (G33, G34, G35)
- Umdrehungsvorschub (G95, G96, G97, FPRAON)
- Anzeige von Drehzahl, Istposition und -geschwindigkeit

18.2.1 Vorschubart G93, G94, G95

Wirksamkeit

Die Vorschubarten G93, G94, G95 wirken bei den G-Befehlen der Gruppe 1 (außer bei G0) in den Automatikbetriebsarten.

Im JOG-Betrieb kann mit G94 oder G95 verfahren werden.

Literatur:

Funktionshandbuch Erweiterungsfunktionen; manuelles Verfahren und manuelles Verfahren per Handrad (H1)

Zeitreziproker Vorschub (G93)

Der zeitreziproke Vorschub wird verwendet, wenn es einfacher ist, anstatt des Vorschubs die Zeitdauer für das Abfahren eines Satzes zu programmieren.

Der zeitreziproke Vorschub errechnet sich nach folgender Formel:

$$F = v / s$$

mit F: zeitreziproker Vorschub in 1/min
 v: Gewünschte Bahngeschwindigkeit in mm/min bzw. inch/min
 s: Bahnlänge in mm bzw. inch

Programmierbeispiel

Programmcode	Kommentar
N10 G1 G93 X100 Y200 F2	; Der programmierte Bahnweg wird in 0,5 min abgefahren.
...	

Hinweis

Bei aktivem G41 / G42 darf G93 nicht verwendet werden. Ist die Satzlänge von Satz zu Satz sehr unterschiedlich, sollte bei G93 in jedem Satz ein neu bestimmter F-Wert programmiert werden.

Linearvorschub (G94)

Der Linearvorschub wird in folgenden Einheiten bezogen auf eine Linearachse bzw. Rundachse programmiert:

- [mm/min, Grad/min] bei metrischem Grundsystem
- [inch/min, Grad/min] bei Inch-Grundsystem

Umdrehungsvorschub (G95)

Der Umdrehungsvorschub wird in folgenden Einheiten bezogen auf eine Masterspindel programmiert:

- [mm/Umdrehung] bei metrischem Grundsystem
- [inch/Umdrehung] bei Inch-Grundsystem
- [Grad/Umdrehung] bei einer Rundachse

Die Bahngeschwindigkeit ergibt sich aus der Ist-Drehzahl der Spindel nach folgender Formel:

$$V = n * F$$

mit V: Bahngeschwindigkeit in mm/min bzw. inch/min
 n: Drehzahl der Masterspindel in U/min
 F: programmierter Umdrehungsvorschub in mm/U bzw. inch/U

Hinweis

Wird zwischen den Vorschubarten G93, G94, G95 umgeschaltet, so wird der programmierte F-Wert gelöscht.

Zahnvorschub

Vorrangig für Fräsbearbeitungen kann anstelle des Umdrehungsvorschubwerts F auch der in der Praxis gebräuchlichere Zahnvorschubwert FZ (Vorschubweg pro Zahn) programmiert werden.

Über den Werkzeugparameter \$TC_DPNT (Anzahl der Zähne pro Umdrehung) des aktiven Werkzeugkorrekturdatensatzes berechnet die Steuerung aus dem programmierten Zahnvorschub für jeden Verfahrssatz den wirksamen Umdrehungsvorschub:

$$F = FZ * \$TC_DPNT$$

mit F: Umdrehungsvorschub in mm/U bzw. inch/U
 FZ: Zahnvorschub in mm/Zahn bzw. inch/Zahn
 \$TC_DP Werkzeugparameter: Anzahl der Zähne/U
 NT:

Beispiel: Fräser mit 5 Zähnen (\$TC_DPNE = 5)

Programmcode	Kommentar
N10 G0 X100 Y50	
N20 G1 G95 FZ=0.02	; Zahnvorschub 0,02 mm/Zahn

Programmcode	Kommentar
N30 T3 D1	; Werkzeug einwechseln und Werkzeugkorrekturdatensatz aktivieren.
M40 M3 S200	; Spindeldrehzahl 200 U/min
N50 X20	; Fräsen mit FZ = 0,02 mm/Zahn
	; wirksamer Umdrehungsvorschub:
	; $F = 0,02 \text{ mm/Zahn} * 5 \text{ Zähne/U} = 0,1 \text{ mm/U}$
	; bzw.:
	; $F = 0,1 \text{ mm/U} * 200 \text{ U/min} = 20 \text{ mm/min}$
...	

Settingdaten

- Umdrehungsvorschub in der Betriebsart JOG
Das Verhalten einer Achse bezüglich des Umdrehungsvorschubs bezogen auf die Masterspindel des Kanals dem die Achse aktuell zugeordnet ist in der Betriebsart JOG ist abhängig von den Einstellungen im NC-spezifischen Settingdatum:
SD41100 \$SN_JOG_REV_IS_ACTIVE, Bit <x>

Bit	Wert	Bedeutung
0	0	Das Verhalten ist von weiteren Settingdaten abhängig, abhängig von der Funktion der Achse: <ul style="list-style-type: none"> Achse / Spindel: SD43300 \$SA_ASSIGN_FEED_PER_REV_SOURCE Geometrieachse, auf die ein Frame mit Rotation wirkt: SD42600 \$SC_JOG_FEED_PER_REV_SOURCE Orientierungsachse: SD42600 \$SC_JOG_FEED_PER_REV_SOURCE
	1	Die Achse wird mit Umdrehungsvorschub verfahren. Randbedingung Es wird ohne Umdrehungsvorschub verfahren, wenn gilt: <ul style="list-style-type: none"> Achse / Spindel: Die Masterspindel steht UND SD43300 \$SA_ASSIGN_FEED_PER_REV_SOURCE == -3 Geometrieachse, auf die ein Frame mit Rotation wirkt: Die Masterspindel steht UND SD42600 \$SC_JOG_FEED_PER_REV_SOURCE == -3 Orientierungsachse: Die Masterspindel steht UND SD42600 \$SC_JOG_FEED_PER_REV_SOURCE == -3
1 ¹⁾	0	Bei Eilgang wird mit Umdrehungsvorschub verfahren.
	1	Bei Eilgang wird ohne Umdrehungsvorschub verfahren.
2 ¹⁾	0	Bei Handradfahren in JOG wird mit Umdrehungsvorschub verfahren.
	1	Bei Handradfahren in JOG wird ohne Umdrehungsvorschub verfahren.
3 ¹⁾	0	Bei DRF-Handradfahren wird mit Umdrehungsvorschub verfahren.
	1	Bei DRF-Handradfahren wird ohne Umdrehungsvorschub verfahren.
1) Die Funktion wirkt zusätzlich zur Einstellung in Bit 1.		

NC/PLC-Nahtstellensignale

- Umdrehungsvorschub aktiv (**kanalspezifisch**)
Mit dem Nahtstellensignal wird angezeigt, daß Achsen im Kanal mit Umdrehungsvorschub verfahren werden:
 – Betriebsart AUTOMATIK: Bahn- oder Synchronachsen
 – Betriebsart JOG: Geometrieachsen
 DB31, ... DBX62.2 == 1 (Umdrehungsvorschub aktiv)
- Umdrehungsvorschub aktiv (**achsspezifisch**)
Mit dem Nahtstellensignal wird angezeigt, daß die Achse mit Umdrehungsvorschub verfahren wird:
 DB31, ... DBX33.2 == 1 (Umdrehungsvorschub aktiv)

18.2.2 Vorschubart G96, G961, G962, G97, G971**Konstante Schnittgeschwindigkeit (G96, G961)**

Die konstante Schnittgeschwindigkeit wird bei Drehmaschinen benutzt, um die Schnittbedingungen unabhängig vom Arbeitsdurchmesser des Werkstücks konstant zu halten. Damit kann das Werkzeug immer im optimalen Schnittleistungsbereich gefahren werden und hält deshalb länger.

Anwahl von G96, G961

Beim Programmieren von G96, G961 wird der zugehörige S-Wert als Schnittgeschwindigkeit in m/min bzw. ft/min in der Planachse interpretiert. Verringert sich bei der Bearbeitung der Werkstückdurchmesser in der Planachse, so wird die Drehzahl solange erhöht, bis die konstante Schnittgeschwindigkeit erreicht wird.

Bei Erstanwahl von G96, G961 im Teileprogramm muss, bei Wiederanwahl kann eine konstante Schnittgeschwindigkeit in m/min bzw. ft/min eingegeben werden.

Bei G96 schaltet die Steuerung automatisch auf Umdrehungsvorschub (wie bei G95) um, d. h. der programmierte Vorschub F wird in mm/Umdrehung bzw. inch/Umdrehung interpretiert.

Bei der Programmierung von G961 wird automatisch Linearvorschub angewählt (wie bei G94). Ein programmierter Vorschub F wird in mm/min bzw. inch/min interpretiert.

Ermittlung der Spindeldrehzahl

Ausgehend von der programmierten Schnittgeschwindigkeit (entweder S_{G96} oder S_{G961}) und der aktuellen kartesischen Position der Planachse (Radius) am TCP ermittelt die Steuerung die Spindeldrehzahl nach folgender Formel:

$$n = \frac{S_{\text{Speed}}}{2 * \pi * r}$$

n: Spindeldrehzahl
 S_{Speed} : Programmierte Schnittgeschwindigkeit
 π : Kreiskonstante
 r: Radius (Abstand Drehmitte zum TCP)

Hierbei wird zur Ermittlung des Radius Folgendes angenommen:

- Die Planachsposition 0 im WKS stellt die Drehmitte dar.
- Positionskorrekturen (wie Online-Werkzeugkorrektur, externe Nullpunktverschiebung, \$AA_OFF, DRF-Verschiebung und Compile-Zyklen) und Positionsanteile durch Kopplungen (z. B. Folgeachse bei TRAIL) werden bei der Radiusermittlung nicht berücksichtigt.

Frames (z. B. programmierbare Frames wie SCALE, TRANS oder ROT) werden bei der Ermittlung der Spindeldrehzahl berücksichtigt und können eine Drehzahländerung bewirken, sofern sich der wirksame Durchmesser am TCP ändert.

Durchmesserprogrammierung und Bezugsachse für mehrere Planachsen im Kanal:

Eine oder mehrere Planachsen sind gleichzeitig oder getrennt zugelassen und aktivierbar:

- Programmierung und Anzeige in der Bedienoberfläche im Durchmesser
- Zuweisung der angegebenen Bezugsachse mit `SCC [<Achse>]` für konstante Schnittgeschwindigkeit G96, G961, G962

Weitere Informationen siehe Funktionsbeschreibung "P1: Planachsen (Seite 871)".

Beispiel

$S_{G96} = 230 \text{ m/min}$

- bei $r = 0,2 \text{ m} \rightarrow n = 183,12 \text{ U/min}$
- bei $r = 0,1 \text{ m} \rightarrow n = 366,24 \text{ U/min}$

⇒ Je kleiner der Werkstückdurchmesser, desto höher die Drehzahl.

Für G96, G961 oder G962 muss eine Geometrieachse als Planachse definiert werden.

Die Planachse, deren Position die Drehzahl der Masterspindel beeinflusst, wird bestimmt über das kanalspezifische Maschinendatum:

MD20100 \$MC_DIAMETER_AX_DEF (Geometrieachse mit Planachsfunktion)

Die Funktion G96, G961 oder G962 setzt voraus, dass der Maschinennullpunkt und der Werkstücknullpunkt der Planachse sich in Drehmitte der Spindel befinden.

Konstante Drehzahl (G97, G971)

Mit G97, G971 wird die Funktion "Konstante Schnittgeschwindigkeit" (G96, G961) ausgeschaltet und die zuletzt berechnete Spindeldrehzahl abgespeichert. Bei G97 wird der Vorschub als Umdrehungsvorschub (wie bei G95) interpretiert. Bei der Programmierung von G971 wird Linearvorschub (wie bei G94) angewählt. Der Vorschub F wird in mm/min bzw. inch/min interpretiert.

Bei aktivem G97, G971 kann durch ein erneutes Programmieren eines S-Werts eine neue Spindeldrehzahl vorgegeben werden. Die unter G96, G961 programmierte Schnittgeschwindigkeit wird dadurch nicht verändert.

Mit G97, G971 können Drehzahlveränderungen bei Bewegungen in der Planachse ohne Bearbeitung (z. B. Werkzeugmeißel) vermieden werden.

Hinweis

G96, G961 ist nur beim Bearbeiten am Werkstück aktiv (G1, G2, G3, Spline-Interpolation usw., wo Vorschub F wirkt).

Das Verhalten der Spindeldrehzahl bei aktivem G96, G961 und G0-Sätzen kann definiert werden in dem kanalspezifischen Maschinendatum:

MD20750 ALLOW_G0_IN_G96 (G0-Logik bei G96, G961)

Bei konstanter Schnittgeschwindigkeit G96, G961 kann kein Getriebestufenwechsel stattfinden.

Der Spindel-Override-Schalter wirkt auf die ermittelte Spindeldrehzahl.

Eine DRF-Verschiebung in der Planachse beeinflusst die Spindeldrehzahlsollwertberechnung nicht.

Zu Beginn der Bearbeitung (nach G0) und nach NC-Stop, G60, G09, ... wird für den Bahnstart "nIst = nSoll" abgewartet.

Die Nahtstellensignale "nIst = nSoll" und "Solldrehzahl begrenzt" werden nicht durch interne Drehzahlvorgaben verändert.

Bei Unterschreitung der Minimaldrehzahl oder beim Erkennen des Signals "Achse /Spindel steht" wird "nIst = nSoll" zurückgesetzt.

Eine begonnene Bahnbearbeitung (G64, Überschleifen) wird nicht unterbrochen.

Drehzahlbegrenzung bei G96, G961

Für die Funktion "Konstante Schnittgeschwindigkeit" kann eine maximale Spindeldrehzahl vorgegeben werden:

- im Settingdatum:
SD43230 \$SA_SPIND_MAX_VELO_LIMS (Spindeldrehzahlbegrenzung bei G96/G961)
- im Teileprogramm (für die Masterspindel) mit dem Programmierbefehl LIMS

Der zuletzt geänderte Wert (LIMS oder SD) ist wirksam.

LIMS wirkt bei G96, G961, G97 und kann auf bis zu vier Drehzahlbegrenzungen im Teileprogramm in einem Satz vorgegeben werden. Hierfür kann bei der

Teileprogrammanweisung LIMS[<Sn>] die Spindelnummer <Sn> = 1, 2, 3 oder 4 der jeweils möglichen Maserspindel angegeben werden.

Beim Einwechseln des Satzes in den Hauptlauf werden alle programmierten Werte in die Settingdaten SD43230 \$SA_SPIND_MAX_VELO_LIMS übernommen.

Abhängig von dem Maschinendatum:

MD10710 \$MN_PROG_SD_RESET_SAVE_TAB[n] (zu aktualisierende Settingdaten)

bleibt die geschriebene Drehzahlbegrenzung mit LIMS nach dem Ausschalten der Steuerung gespeichert.

Mit dem erneuten Aktivieren von G96, G961, G97 wird diese Drehzahlbegrenzung wieder aktiviert.

Die maximal zulässige Spindeldrehzahl, die über G26 bzw. über das Settingdatum:

SD43220 \$SA_SPIND_MAX_VELO_G26 (Maximale Spindeldrehzahl)

vorgegeben wurde, kann nicht überschritten werden.

Bei falscher Programmierung, die zu einer Überschreitung einer der Drehzahlgrenzen (G26 bzw. SD43220 \$SA_SPIND_MAX_VELO_G26) führen würde, wird das folgende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX83.1 (Programmierte Drehzahl zu hoch)

Um einen Rundlauf bei großen Teiledurchmessern sicher zu stellen, darf eine minimale Spindeldrehzahl nicht unterschritten werden.

Diese Drehzahl ist über das Settingdatum:

SD43210 \$SA_SPIND_MIN_VELO_G25 (Minimale Spindeldrehzahl)

und je Getriebestufe mit dem Maschinendatum:

MD35140 \$MA_GEAR_STEP_MIN_VELO_LIMIT (Minimaldrehzahl der Getriebestufe) einstellbar.

Die minimale Spindeldrehzahl kann im Teileprogramm mit G25 verändert werden. Bei Programmierung, die zu einer Unterschreitung einer der Drehzahlgrenzen (G25 bzw. SD43210 \$SA_SPIND_MIN_VELO_G25) führen würde, wird das folgende NC/PLC-Nahtstellensignal gesetzt:

DB31, ... DBX83.2 (Solldrehzahl zu niedrig)

Weitere Informationen zu den Spindeldrehzahlbegrenzungen siehe Funktionsbeschreibung S1: "Spindeln", Kapitel: "Spindelüberwachungen (Seite 1389)".

Hinweis

Die im Teileprogramm mit G25/G26/LIMS geänderten Drehzahlgrenzen werden in die Settingdaten übernommen und bleiben somit auch über das Programmende hinaus gespeichert.

Sollen die mit G25/G26/LIMS geänderten Drehzahlgrenzen nach Programmende dagegen nicht mehr gelten, müssen folgende Definitionen in den GUD-Baustein des Maschinenherstellers eingefügt werden:

REDEF \$SA_SPIND_MIN_VELO_G25 PRLOC

REDEF \$SA_SPIND_MAX_VELO_G26 PRLOC

REDEF \$SA_SPIND_MAX_VELO_LIMS PRLOC

Masterspindel Umschaltung bei G96, G961

Beim Wechsel der Masterspindel während aktivem G96, G961 wird die Drehzahl der ehemaligen Masterspindel beibehalten. Dies entspricht einem Übergang von G96 nach G97. Die mit SETMS neu definierte Masterspindel führt die so erzeugte Funktion "Konstante Schnittgeschwindigkeit" aus.

Alarme

Konstante Schnittgeschwindigkeit G96, G961, G962

- Wird kein F-Wert programmiert, so wird der Alarm 10860 "Kein Vorschub programmiert" ausgegeben. Der Alarm wird nicht bei G0-Sätzen erzeugt.
- Bei Programmierung einer negativen Bahngeschwindigkeit wird der Alarm 14800 "Programmierte Bahngeschwindigkeit kleiner oder gleich Null" ausgegeben.
- Ist bei aktivem G96, G961 oder G962 keine Planachse im Maschinendatum: MD20100 \$MC_DIAMETER_AX_DEF (Geometrieachsen mit Planachsenfunktion) definiert, so wird der Alarm 10870 "keine Planachse definiert" ausgegeben.
- Wird bei aktivem G96, G961 eine negative maximale Spindeldrehzahl mit dem Programmierbefehl `LIMS` programmiert, so wird der Alarm 14820 "negative maximale Spindeldrehzahl für G96, G961 programmiert" ausgegeben.
- Ist bei erstmaliger Anwahl von G96, G961 keine konstante Schnittgeschwindigkeit programmiert, so wird der Alarm 10900 "kein S-Wert für konstante Schnittgeschwindigkeit programmiert" ausgegeben.

18.2.3 Vorschub bei Gewindschneiden (G33, G34, G35, G335, G336)

18.2.3.1 Vorschub bei G33

G33

Mit der Funktion G33 können Gewinde mit konstanter Steigung bearbeitet werden.

Drehzahl S, Vorschub F, Gewindesteigung

Bei G33-Gewinden wirkt ein Umdrehungsvorschub [mm/Umdrehung], der durch die Programmierung der Gewindesteigung [mm/Umdrehung] vorgegeben wird.

Die Geschwindigkeit der Achsen für die Gewindelänge errechnet sich aus der programmierten Spindeldrehzahl S und der Gewindesteigung:

Vorschub F [mm/min] = Drehzahl S [U/min] * Gewindesteigung [mm/U]

Am Ende der Beschleunigungsrampe wird die Lagekopplung vom Spindelwert (Spindelsollwert bei SPCON der Masterspindel) mit dem Achssollwert hergestellt. Die Achsposition steht zu diesem Zeitpunkt in Bezug auf die Nullmarke (Nullmarkenverschiebungen berücksichtigt) der Spindel so, als wäre die Achse am

Satzanfangspunkt bei Überfahren der Gewindestartposition (Nullmarke plus SF) schlagartig beschleunigt worden. Der Schleppabstand der Achse wird dabei kompensiert.

Minimale Spindeldrehzahl

Um einen Rundlauf bei geringen Drehzahlen sicher zu stellen, darf eine minimale Spindeldrehzahl nicht unterschritten werden.

Diese Drehzahl ist einstellbar:

- mit dem Settingdatum:
SD43210 \$SA_SPIND_MIN_VELO_G25 (Minimale Spindeldrehzahl)
- je Getriebestufe mit dem Maschinendatum:
MD35140 \$MA_GEAR_STEP_MIN_VELO_LIMIT (Minimaldrehzahl für
Getriebestufenwechsel)

Die minimale Spindeldrehzahl kann im Teileprogramm mit G25 verändert werden.

NC-Stop, Einzelsatz

NC-Stop und Einzelsatz (auch an Satzgrenze) wirken erst am Ende einer Gewindekettung. Dabei werden alle aufeinander folgenden G33-Sätze und der erste darauf folgende Nicht-G33-Satz wie ein Satz abgefahren.

Vorzeitiger zerstörungsfreier Abbruch

Es ist möglich, das Gewindeschneiden vor Erreichen des Endpunkts zerstörungsfrei zu unterbrechen. Dies kann durch die Aktivierung einer Rückzugsbewegung erfolgen.

Gewindeschneiden mit ROT-Frame

Bei ROT-Frame und G33, G34, G35 wird Alarm 10607 "Gewinde mit Frame nicht ausführbar" gemeldet, wenn durch die Rotation die Gewindelänge und damit die Steigung verändert wird. Rotationen um die Gewindeachse sind zulässig.

Der Alarm 10607 "Gewinde mit Frame nicht ausführbar" kann durch Setzen von Bit 12 im Maschinendatum MD11410 \$MN_SUPPRESS_ALARM_MASK unterdrückt werden, falls in der Anwendung bewusst die ROT-Anweisung verwendet wird.

Alle weiteren Frames werden von der NC ohne Alarm akzeptiert. Auf die steigungsändernde Wirkung von SCALE wird hingewiesen.

18.2.3.2 Linear progressive/degressive Gewindesteigungsänderung bei G34 und G35

Funktion

Die Gewindesteigungszunahme (G34) beschreibt die zahlenmäßige Vergrößerung des Steigungswerts. Eine größere Steigung bewirkt einen größeren Abstand zwischen den Gewindegängen auf dem Werkstück. Die Geschwindigkeit der Gewindeachse nimmt also bei angenommener konstanter Spindeldrehzahl zu.

Für die Gewindesteigungsabnahme (G35) gilt sinngemäß das Gegenteil.

Folgende Begriffsbestimmungen werden für die Gewindesteigungsänderung getroffen:

- G34: Gewindesteigungszunahme entspricht progressiver Änderung
- G35: Gewindesteigungsabnahme entspricht degressiver Änderung

Beide Funktionen G34 und G35 implizieren die Funktionalität von G33 und bieten zusätzlich die Möglichkeit unter F eine Steigungsänderung des Gewindes betragsmäßig zu programmieren. Ist die Anfangs- und Endsteigung eines Gewindes bekannt, dann kann die zu programmierende Gewindesteigungsänderung nach folgender Gleichung ermittelt werden:

$$F = \frac{|k_e^2 - k_a^2|}{2 * l_G}$$

Dabei bedeuten:

- F: die zu programmierende Gewindesteigungsänderung [mm/U²]
 k_e : Gewindesteigung der Achszielpunktkoordinate Gewindeachse [mm/U]
 k_a : Gewindeanfangssteigung (unter I, J oder K programmiert) [mm/U]
 l_G : Gewindelänge [mm]

Der Betrag von F ist entsprechend der gewünschten Steigungszunahme oder Steigungsabnahme auf G34 bzw. G35 anzuwenden.

Bei bekannter Gewindelänge l_G , Steigungsänderung F und Anfangssteigung k_a kann die Gewindesteigung am Satzende k_e durch Umstellen der Gleichung wie folgt ermittelt werden:

- Bei G34 (zunehmende Steigerung):

$$k_e = \sqrt{k_a^2 + F * 2 * l_G}$$

- Bei G35 (abnehmende Steigung):

$$k_e = \sqrt{k_a^2 - F * 2 * l_G}$$

Hinweis

Ergibt sich bei der Formel ein negativer Wurzelausdruck, dann kann das Gewinde nicht gefertigt werden!

Die NC meldet in diesem Fall den Alarm 10605 bzw. den Alarm 22275.

Anwendung

Die Funktionen G34 und G35 können zur Realisierung von selbstscherenden Gewinden eingesetzt werden.

Beispiel**Gewindeschneiden G33 mit abnehmender Gewindesteigung G35**

Programmcode	Kommentar
N1608 M3 S10	; Spindeldrehzahl
N1609 G0 G64 Z40 X216	; Startpunkt anfahren
N1610 G33 Z0 K100 SF=R14	; Gewinde mit konstanter Steigung 100 mm/U
N1611 G35 Z-220 K100 F17.045455	; Gewindesteigungsabnahme 17.045455 mm/U2
	; Gewindesteigung am Satzende 50 mm/U
N1612 G33 Z-240 K50	; Gewindegang ohne Ruck fahren
N1613 G0 X218	
N1614 G0 Z40	
N1616 M17	

Überwachung während der Satzaufbereitung

Steigungsänderungen, welche bei G34 die Gewindeachse überlasten oder bei G35 einen Achsstillstand bewirken würden, werden frühzeitig bei der Satzaufbereitung erkannt. Es wird der Alarm 10604 "Gewindesteigungszunahme zu hoch" bzw. 10605 "Gewindesteigungsabnahme zu hoch" gemeldet.

Bestimmte praxisrelevante Anwendungen erfordern beim Gewindeschneiden eine Korrektur der Spindeldrehzahl. Der Bediener orientiert sich in diesem Fall an der zulässigen Geschwindigkeit der Gewindeachse.

Dazu ist es möglich, die Ausgabe der Alarme 10604 und 10605 wie folgt zu unterdrücken:

MD11410 \$MN_SUPPRESS_ALARM_MASK Bit 10 = 1

Die Satzaufbereitung wird dann normal fortgesetzt.

Überwachung während der Abarbeitung

Während der Abarbeitung (Interpolation) des Gewindes werden folgende Situationen zyklisch überwacht:

- Überschreiten der Maximalgeschwindigkeit der Gewindeachse.
- Erreichen des Achsstillstandes bei G35.

Bei Ansprechen der Überwachung wird folgender Alarm gemeldet:

- Alarm 22270 "Maximaler Geschwindigkeit der Gewindeachse erreicht" bzw.
- Alarm 22275 "Geschwindigkeit Null der Gewindeachse erreicht"

18.2.3.3 Beschleunigungsverhalten der Achse bei G33, G34 und G35

Das Beschleunigungsverhalten der Vorschubachse beim Gewindeschneiden mit G33, G34 oder G35 ist einstellbar über das kanalspezifische Settingdatum:

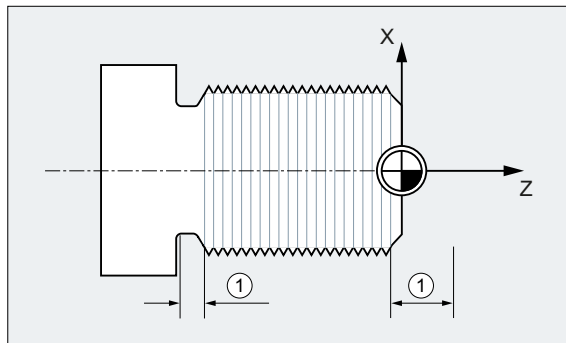
SD42010 \$SC_THREAD_RAMP_DISP[<n>] = <Wert>

<n>	Bedeutung	<Wert>	Bedeutung
0	Beschleunigungsverhalten beim Gewindeeinlauf	< 0 (Default: -1)	Das Beschleunigen der Gewindeachse beim Einlauf erfolgt entsprechend der aktuellen Programmierung BRISK / SOFT .
		= 0	Das Beschleunigen der Gewindeachse beim Einlauf erfolgt sprungförmig (Δ BRISK).
		> 0	Es wird der maximale Einlaufweg vorgegeben. Die Vorgabe des maximalen Einlaufwegs ist über die Adresse DITS auch im Teileprogramm möglich (siehe "Programmierter Einlauf- und Auslaufweg bei G33, G34 und G35 (DITS, DITE) (Seite 1428)"). Mit dem Einwechseln des Satzes in den Hauptlauf wird der programmierte Einlaufweg in das Settingdatum übernommen. Hinweis Ein zu kurzer Weg kann zu einer Beschleunigungsüberlastung der Achse führen.
1	Beschleunigungsverhalten beim Gewindeauslauf	< 0 (Default: -1)	Das Bremsen der Gewindeachse beim Auslauf erfolgt entsprechend der aktuellen Programmierung BRISK / SOFT .
		= 0	Das Bremsen der Gewindeachse beim Auslauf erfolgt sprungförmig (Δ BRISK).
		> 0	Es wird der maximale Auslaufweg vorgegeben. Die Vorgabe des maximalen Auslaufwegs ist über die Adresse DITE auch im Teileprogramm möglich (siehe "Programmierter Einlauf- und Auslaufweg bei G33, G34 und G35 (DITS, DITE) (Seite 1428)"). Mit dem Einwechseln des Satzes in den Hauptlauf wird der programmierte Auslaufweg in das Settingdatum übernommen. Hinweis Ein zu kurzer Weg kann zu einer Beschleunigungsüberlastung der Achse führen.
2	Beschleunigungsverhalten bei Gewindeübergängen innerhalb einer Gewindekette	= -1 (Default)	Erlaubt das Überschleifen an Ecken zwischen zwei Gewindesätzen zur Einhaltung der Dynamikgrenzen. Es wird nicht durch Überschleifen verrundet, wenn die Dynamik der Maschine einen harten Übergang erlaubt, z. B. aufgrund der Wirksamkeit des Überlastfaktors für axiale Geschwindigkeitssprünge (MD32310 \$MA_MAX_ACCEL_OVL_FACTOR).
		= 0	Ecken zwischen zwei Gewindesätzen werden hart ausgefahren, die Achsen folgen nach Vorgabe der Regelkreise.
		> 0	reserviert

18.2.3.4 Programmierter Einlauf- und Auslaufweg bei G33, G34 und G35 (DITS, DITE)

Mit den Adressen `DITS` und `DITE` kann im Teileprogramm der Einlauf- und Auslaufweg des Gewindes vorgegeben werden.

Die Gewindeachse wird innerhalb des vorgegebenen Wegs beschleunigt bzw. gebremst.



① Ein- bzw. Auslaufweg, je nach Bearbeitungsrichtung

Kurzer Einlaufweg

Durch den Bund am Gewindeeinlauf ist wenig Platz für die Werkzeug-Startrampe. Diese muss deshalb über `DITS` kürzer vorgegeben werden.

Kurzer Auslaufweg

Durch den Bund am Gewindeauslauf ist wenig Platz für die Werkzeug-Bremsrampe, wodurch Kollisionsgefahr zwischen Werkstück und Schneide besteht. Die Bremsrampe kann über `DITE` kürzer vorgegeben werden. Wegen der Trägheit der Mechanik kann es dennoch zur Kollision kommen.

Abhilfe: Gewinde kürzer programmieren, Spindeldrehzahl reduzieren.

Hinweis

`DITE` wirkt am Gewindeende als Überschleifabstand. Damit wird eine stoßfreie Änderung der Achsbewegung erreicht.

Auswirkungen

Der programmierte Einlauf- und Auslaufweg wirkt auf die Bahn ausschließlich beschleunigungssteigernd. Wird einer der beiden Wege größer vorgegeben, als er für die Gewindeachse mit aktiver Beschleunigung benötigt wird, wird die Gewindeachse mit maximaler Beschleunigung beschleunigt bzw. gebremst.

Syntax

`DITS=<Wert> DITE=<Wert>`

Bedeutung

DITS:	Gewinde-Einlaufweg festlegen
DITE:	Gewinde-Auslaufweg festlegen
<Wert>:	Unter DITS und DITE werden ausschließlich Wege programmiert, keine Positionen! Der programmierte Ein-/Auslaufweg wird entsprechend der aktuellen Maßeinstellung (Inch, metrisch) behandelt.

Beispiel

Programmcode	Kommentar
...	
N40 G90 G0 Z100 X10 SOFT M3 S500	
N50 G33 Z50 K5 SF=180 DITS=1 DITE=3	; Überschleifbeginn bei Z=53.
N60 G0 X20	

Weitere Informationen

SD42010 \$SC_THREAD_RAMP_DISP

Mit dem Einwechseln eines Satzes mit DITS und/oder DITE in den Hauptlauf wird der programmierte Ein- bzw. Auslaufweg in das Settingdatum SD42010 \$SC_THREAD_RAMP_DISP (Seite 1426) übernommen:

- SD42010 \$SC_THREAD_RAMP_DISP[0] = programmierter Wert von DITS
- SD42010 \$SC_THREAD_RAMP_DISP[1] = programmierter Wert von DITE

Wird vor oder im ersten Gewindesatz kein Ein- bzw. Auslaufweg programmiert, wird jeweils der aktuelle Wert des Settingdatums verwendet.

Verhalten nach Kanal-/BAG-/Programmende-Reset

Die durch DITS und/oder DITE überschriebenen Werte von SD42010 bleiben auch nach einem Kanal-/BAG-/Programmende-Reset wirksam.

Verhalten nach Warmstart

Bei einem Warmstart wird das Settingdatum auf die Werte zurückgesetzt, die vor dem Überschreiben durch DITS und/oder DITE wirksam waren (Standardverhalten).

Sollen die mit DITS und DITE programmierten Werte dagegen auch nach einem Warmstart wirksam sein, muss das Settingdatum SD42010 \$SC_THREAD_RAMP_DISP im Maschinendatum MD10710 \$MN_PROG_SD_RESET_SAVE_TAB aufgelistet sein:

MD10710 \$MN_PROG_SD_RESET_SAVE_TAB[<n>] = 42010

Verhalten bei sehr kleinem Ein- und/oder Auslaufweg

Bei sehr kleinem Ein- und/oder Auslaufweg wird die Gewindeachse stärker beschleunigt, als es die Projektierung vorsieht. Die Achse wird dann beschleunigungsmäßig überlastet.

Für den Gewindeeinlauf wird dann der Alarm 22280 "Programmierter Einlaufweg zu kurz" gemeldet (bei entsprechender Projektierung im MD11411 \$MN_ENABLE_ALARM_MASK). Der Alarm ist rein informativ und hat keine Auswirkungen auf die Teileprogrammabarbeitung.

18.2.3.5 Schnelrückzug während Gewindeschneiden

Funktion

Die Funktion "Schnelrückzug während Gewindeschneiden" ermöglicht eine zerstörungsfreie Unterbrechung des Gewindeschneidens bei:

- NC-Stop (NC/PLC-Nahtstellensignal)
- Alarme, die implizit NC-Stop auslösen
- Schalten eines schnellen Eingangs

Literatur

Programmierhandbuch Arbeitsvorbereitung, Kapitel "Schnellabheben von der Kontur"

Die Rückzugsbewegung ist programmierbar über:

- Rückzugweg und Rückzugsrichtung (relativ)
- Rückzugsposition (absolut)

Hinweis

Gewindebohren

Die Funktion "Schnelrückzug" ist **nicht** anwendbar beim Gewindebohren (G331 / G332).

Programmierung

Syntax

Schnelrückzug freigeben, Rückzugsbewegung über Rückzugweg und Rückzugsrichtung:

```
G33 ... LFON DILF=<Wert> LFTXT/LFWP ALF=<Wert>
```

Schnelrückzug freigeben, Rückzugsbewegung über Rückzugsposition:

```
POLF[<Achsname>]=<Wert> LFPOS  
POLFMASK/POLFMLIN(<Achsname1>,<Achsname2>,...)  
G33 ... LFON
```

Schnelrückzug für Gewindeschneiden sperren:

```
LFOF
```

Bedeutung

LFON: Schnelrückzug für Gewindeschneiden (G33) freigeben

LFOF: Schnelrückzug für Gewindeschneiden (G33) sperren

DILF= : Länge des Rückzugswegs festlegen

Der durch MD-Projektierung (MD21200 \$MC_LIFTFAST_DIST) voreingestellte Wert kann im Teileprogramm durch Programmierung von DILF verändert werden.

Hinweis:

Nach NC-RESET ist immer der projektierte MD-Wert aktiv.

LFTXT	Die Rückzugsrichtung wird in Verbindung mit ALF mit den G-Befehlen
LFWP:	LFTXT und LFWP gesteuert.
LFTXT:	Die Ebene, in welcher die Rückzugsbewegung ausgeführt wird, wird aus der Bahntangente und der Werkzeugrichtung errechnet (Standardeinstellung).
LFWP:	Die Ebene, in welcher die Rückzugsbewegung ausgeführt wird, ist die aktive Arbeitsebene.
ALF= :	In der Ebene der Rückzugsbewegung wird mit ALF die Richtung in diskreten Schritten von Grad programmiert. Bei LFTXT ist für ALF=1 der Rückzug in Werkzeugrichtung festgelegt. Bei LFWP ergibt sich die Richtung in der Arbeitsebene nach folgender Zuordnung:
	<ul style="list-style-type: none"> • G17 (X/Y-Ebene) ALF=1 ; Rückzug in X-Richtung ALF=3 ; Rückzug in Y-Richtung • G18 (Z/X-Ebene) ALF=1 ; Rückzug in Z-Richtung ALF=3 ; Rückzug in X-Richtung • G19 (Y/Z-Ebene) ALF=1 ; Rückzug in Y-Richtung ALF=3 ; Rückzug in Z-Richtung
	Literatur: Zu den Programmiermöglichkeiten mit ALF siehe auch Kapitel "Verfahr- richtung beim Schnellabheben von der Kontur" im Programmierhandbuch Arbeits- vorbereitung.
LFPOS:	Rückzug der mit POLFMASK oder POLFMLIN bekannt gemachten Achse auf die mit POLF programmierte absolute Achsposition
POLFMASK:	Freigabe der Achsen (<Achsname1>, <Achsname1>, . . .) für den unab- hängigen Rückzug auf Absolutposition
POLFMLIN:	Freigabe der Achsen für den Rückzug auf Absolutposition im linearen Zusam- menhang
	Hinweis: Der lineare Zusammenhang kann je nach dynamischen Verhalten aller betei- ligten Achsen bis zum Erreichen der Abhebe- position nicht immer hergestellt werden.
POLF[]:	Absolute Rückzugsposition für die im Index angegebene Geometrieachse bzw. Maschinenachse festlegen
	Wirksamkeit: modal
=<Wert>:	Bei Geometrieachsen wird der zugewiesene Wert als Positi- on im Werkstückkoordinatensystem (WKS) interpretiert, bei Maschinenachsen als Position im Maschinenkoordinatensys- tem (MKS). Die Wertzuweisung ist auch als Kettenmaßangabe program- mierbar: =IC<Wert>
<Achsname>:	Name einer Geometrie- oder Maschinenachse

Hinweis

LFON bzw. LFOF können immer programmiert werden, die Auswertung erfolgt jedoch ausschließlich beim Gewindeschneiden (G33).

Hinweis

POLF mit POLFMASK/POLFMLIN sind nicht auf den Einsatz bei Gewindeschneiden beschränkt.

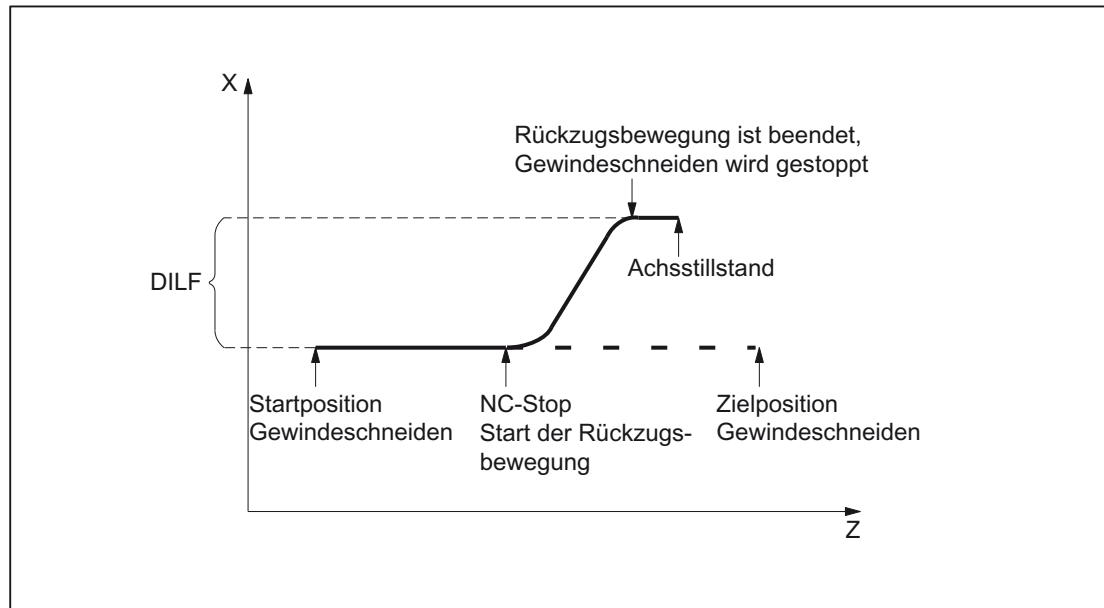


Bild 18-1 Unterbrechung von G33 durch Rückzugsbewegung

Dynamik der Rückzugsbewegung

Die Rückzugsbewegung wird mit maximaler Achsdynamik ausgeführt:

- MD32000 \$MA_MAX_AX_VELO[<Achse>] (Geschwindigkeit)
- MD32300 \$MA_MAX_AX_ACCEL[<Achse>] (Beschleunigung)
- MD32431 \$MA_MAX_AX_JERK[<Achse>] (Ruck)

Beispiel

Programmcode	Kommentar
N55 M3 S500 G90 G18	; Aktive Bearbeitungsebene setzen.
...	
N65 MSG ("Gewindeschneiden")	
MM_THREAD:	
N67 \$AC_LIFTFAST=0	; Vor Beginn des Gewindes zurücksetzen.
N68 G0 Z5	

```

N69 X10
N70 G33 Z30 K5 LFON DILF=10 LFWP ; Schnellrückzug für Gewindeschneiden
ALF=7                             freigegeben.
                                   ; Rückzugsweg=10 mm
                                   ; Rückzugsebene Z/X (wegen G18)
                                   ; Rückzugsrichtung -X (mit ALF=3
                                   Rückzugsrichtung +X)

N71 G33 Z55 X15
N72 G1                             ; Gewindeschneiden abwählen.
N69 IF $AC_LIFTFAST GOTOB MM_THREAD ; Wenn Gewindeschneiden unterbrochen
                                   wurde.

N90 MSG ("")
...
N70 M30
N55 M3 S500 G90 G0 X0 Z0
...
N87 MSG ("Gewindebohren")
N88 LFOF                           ; Schnellrückzug vor Gewindebohren
                                   ausschalten.

N89 CYCLE...                       ; Gewindebohrzyklus mit G33.
N90 MSG ("")
...
N99 M30

```

Verhalten bei Power On und Reset

Nach Power On und Reset werden folgende Einstellungen aktiv:

- Löschstellungen für die Rückzugsbewegung (LFON / LFOF) und Rückzugsrichtung (LFTXT / LFWP): MD20150 \$MC_GCODE_RESET_VALUES
- Rückzugsweg: MD21200 \$MC_LIFTFAST_DIST

18.2.3.6 Ballige Gewinde (G335, G336)

Funktion

Mit den G-Befehle G335 und G336 besteht die Möglichkeit, ballige (= von der zylindrischen Form abweichende) Gewinde zu drehen. Anwendung ist die Bearbeitung extrem großer Bauteile, die durch ihr Eigengewicht in der Maschine durchhängen. Achsparallele Gewinde würden dazu führen, dass der Gewindegang in der Mitte des Bauteils zu gering ist. Mit den balligen Gewinden kann dies ausgeglichen werden.

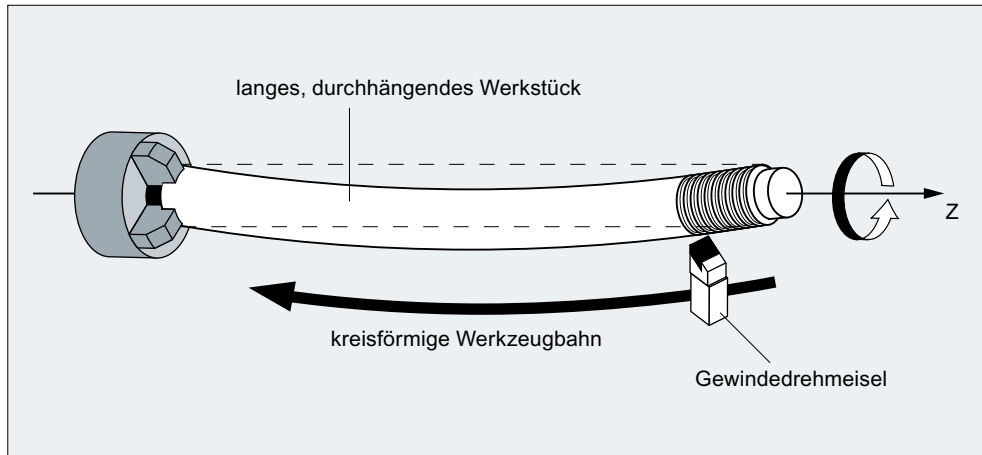


Bild 18-2 Drehen eines balligen Gewindes

Programmierung

Das Drehen balliger Gewinde wird mit G335 oder G336 programmiert:

G335:	Drehen eines balligen Gewindes auf einer im Uhrzeigersinn verlaufenden kreisförmigen Werkzeugbahn
G336:	Drehen eines balligen Gewindes auf einer gegen den Uhrzeigersinn verlaufenden kreisförmigen Werkzeugbahn

Die Programmierung erfolgt zunächst wie bei einem linearen Gewinde über die Angabe der axialen Satzpunkte und der Steigung über die Parameter I, J, und K.

Zusätzlich wird noch ein Kreisbogen mit angegeben. Dieser kann wie bei G2/G3 über die Mittelpunkt-, Radius-, Öffnungswinkel- oder Zwischenpunktangabe programmiert werden. Bei Programmierung des balligen Gewindes mit Mittelpunktprogrammierung ist dabei Folgendes zu beachten: Da I, J, und K bei Gewindeschneiden für die Steigung benutzt werden, müssen die Kreisparameter bei Mittelpunktprogrammierung mit IR=..., JR=... und KR=... programmiert werden.

IR=...:	Kartesische Koordinate für den Kreismittelpunkt in X-Richtung
JR=...:	Kartesische Koordinate für den Kreismittelpunkt in Y-Richtung
KR=...:	Kartesische Koordinate für den Kreismittelpunkt in Z-Richtung

Hinweis

IR, JR und KR sind die Standardwerte der über das Maschinendatum MD10651 \$MN_IPO_PARAM_THREAD_NAME_TAB einstellbaren Namen der Interpolationsparameter für ballige Gewinde.

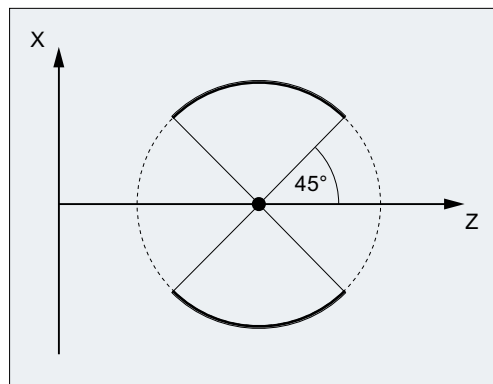
Optional kann noch ein Startpunktversatz SF mit angegeben werden.

Syntax

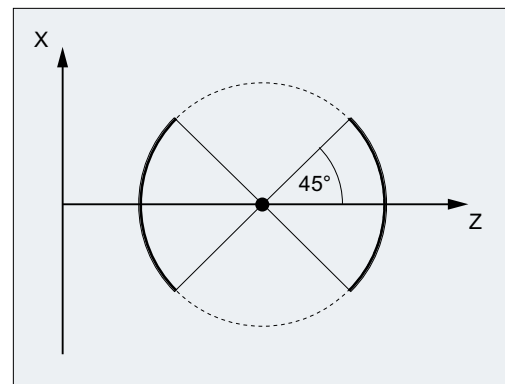
Die Syntax zur Programmierung eines balligen Gewindes hat also folgende allgemeine Form:
G335/G336 <Achszieldpunktcoordinate(n)> <Steigung> <Kreisbogen>
[<Startpunktversatz>]

Erlaubte Kreisbogenbereiche

Der unter G335/G336 programmierte Kreisbogen muss in einem Bereich liegen, in dem die spezifizierte Gewindehauptachse (I, J oder K) über den gesamten Kreisbogen den Hauptachsanteil am Kreisbogen hat:



Erlaubte Bereiche für die Z-Achse (Steigung mit K programmiert)



Erlaubte Bereiche für die X-Achse (Steigung mit I programmiert)

Ein Wechsel der Gewindehauptachse, wie im folgenden Bild dargestellt, ist **nicht** erlaubt:

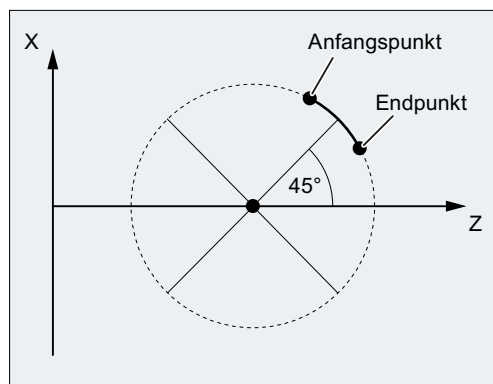


Bild 18-3 Ballige Gewinde: Nicht erlaubter Bereich

Randbedingungen

Frames

Auch bei aktiven Frames ist G335 und G336 möglich. Es ist allerdings darauf zu achten, dass im Basiskoordinatensystem (BKS) die erlaubten Kreisbogenbereiche eingehalten werden.

Verhalten bei ...

Das Verhalten von G335/G336 bei:

- Power On / Power Off
- Betriebsartenwechsel
- NCK- / BAG- / Kanal- / Teilprogrammende-Reset
- Satzsuchlauf / REPOS / ASUP
- Alarmen / Not-Halt / Störfälle

entspricht dem Verhalten von G33/G34/G35.
Es gibt keine spezifischen Einschränkungen.

Beispiele

Beispiel 1: Balliges Gewinde im Uhrzeigersinn mit End- und Mittelpunktprogrammierung

Programmcode	Kommentar
N5 G0 G18 X50 Z50	; Anfangspunkt anfahren.
N10 G335 Z100 K=3.5 KR=25 IR=-20 SF=90	; Balliges Gewinde im Uhrzeiger-sinn drehen.

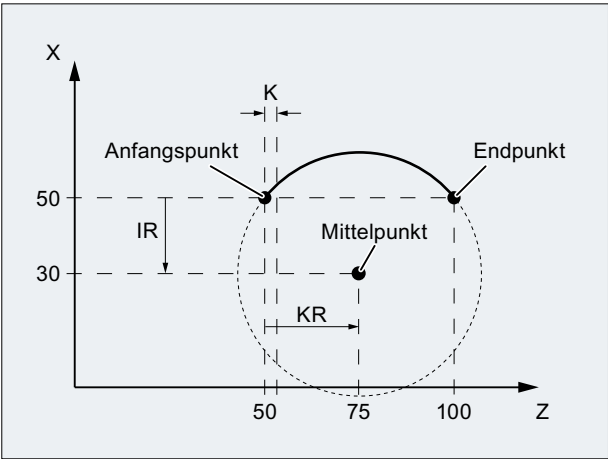


Bild 18-4 Balliges Gewinde im Uhrzeigersinn mit End- und Mittelpunktprogrammierung

Beispiel 2: Balliges Gewinde gegen den Uhrzeigersinn mit End- und Mittelpunktprogrammierung

Programmcode	Kommentar
N5 G0 G18 X50 Z50	; Anfangspunkt anfahren.

Programmcode	Kommentar
N10 G336 Z100 K=3.5 KR=25 IR=20 SF=90	; Balliges Gewinde gegen den Uhrzeigersinn drehen.

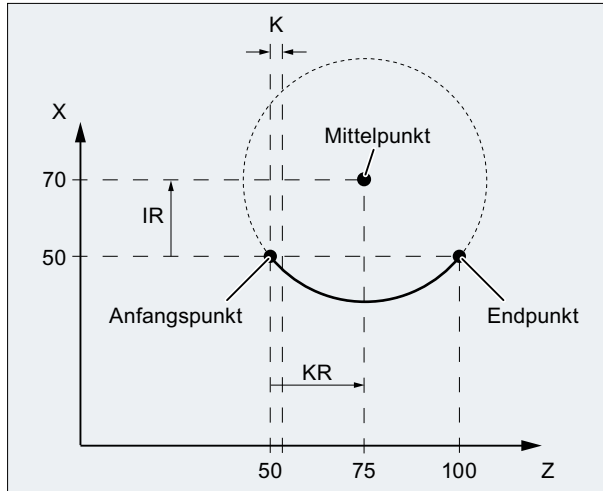


Bild 18-5 Balliges Gewinde gegen den Uhrzeigersinn mit End- und Mittelpunktprogrammierung

Beispiel 3: Balliges Gewinde im Uhrzeigersinn mit Endpunkt- und Radiusprogrammierung

Programmcode
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 CR=32 SF=90

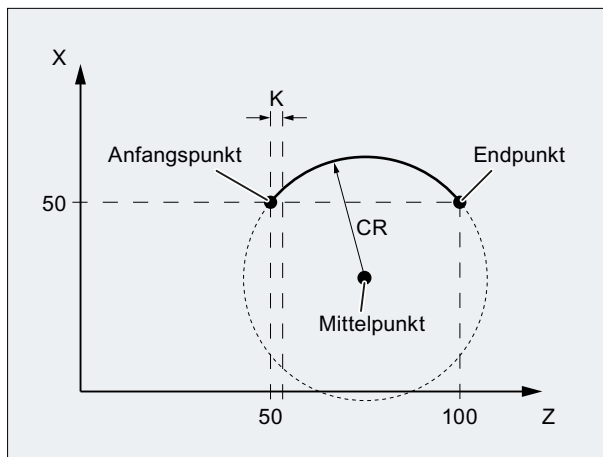


Bild 18-6 Balliges Gewinde im Uhrzeigersinn mit Endpunkt- und Radiusprogrammierung

Beispiel 4: Balliges Gewinde im Uhrzeigersinn mit Endpunkt- und Öffnungswinkelprogrammierung

Programmcode
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 AR=102.75 SF=90

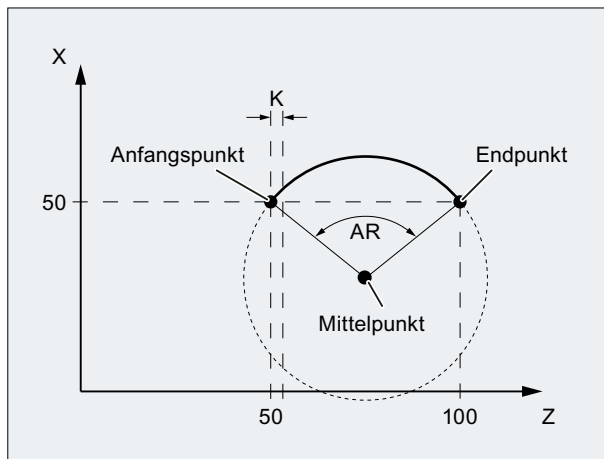


Bild 18-7 Balliges Gewinde im Uhrzeigersinn mit Endpunkt- und Öffnungswinkelprogrammierung

Beispiel 5: Balliges Gewinde im Uhrzeigersinn mit Mittelpunkt- und Öffnungswinkelprogrammierung**Programmcode**

```
N5 G0 G18 X50 Z50
N10 G335 K=3.5 KR=25 IR=-20 AR=102.75 SF=90
```

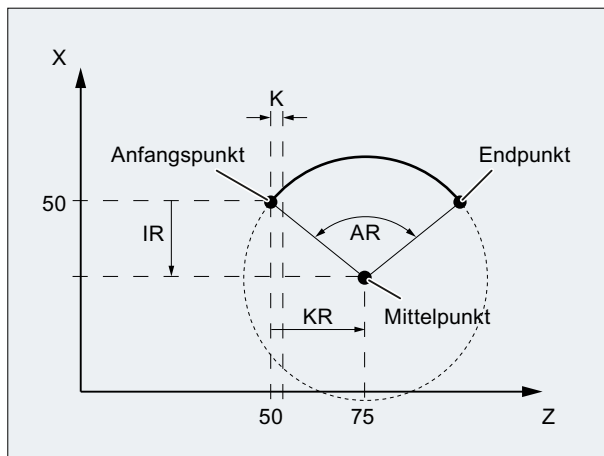


Bild 18-8 Balliges Gewinde im Uhrzeigersinn mit Mittelpunkt- und Öffnungswinkelprogrammierung

Beispiel 6: Balliges Gewinde im Uhrzeigersinn mit End- und Zwischenpunktprogrammierung**Programmcode**

```
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 I1=60 K1=64
```

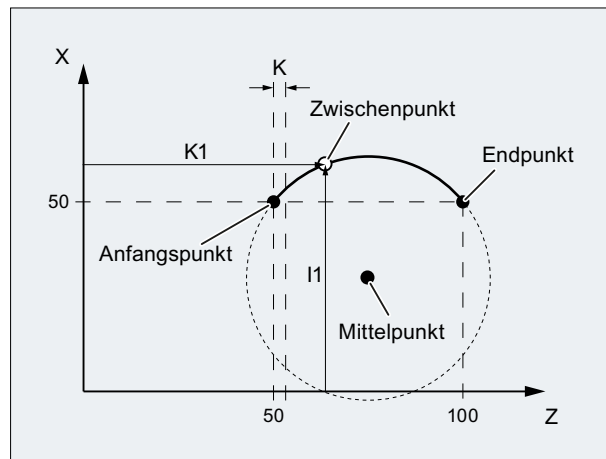


Bild 18-9 Balliges Gewinde im Uhrzeigersinn mit End- und Zwischenpunktprogrammierung

18.2.4 Vorschub bei Gewindebohren ohne Ausgleichsfutter (G331, G332)

Funktion

Mit den Funktionen G331 (Gewindebohren) und G332 (Gewindebohren Rückzug) kann ein Gewinde ohne Ausgleichsfutter gebohrt werden.

Bahnvorschub

Bei G331 / G332 ergibt sich der Bahnvorschub F für die am Gewindebohren beteiligten Achsen aus der wirksamen Spindeldrehzahl S und der programmierten Gewindesteigung:

$$F \text{ [mm/min]} = S \text{ [U/min]} * \text{Gewindesteigung [mm/U]}$$

Voraussetzung

Voraussetzung für das Gewindebohren ohne Ausgleichsfutter ist eine **lagegeregelte Spindel** mit Wegmesssystem.

Maschinendaten

- Stopp-Ereignisse verhindern
Über das Maschinendatum wird das Stopp-Verhalten bei aktivem G331 / G332 festgelegt:
MD11550 \$MN_STOP_MODE_MASK, Bit <x> = <Wert>

Bit	Wert	Bedeutung
0	0	Bei aktivem G331 / G332 wird während einer Bahnbewegung oder Verweilzeit (G4) nicht angehalten.
	1	Bei aktivem G331 / G332 wird während einer Unterbrechung des Bahnsteuerbetriebs durch z.B. G60 oder G4, angehalten.

Soll in jedem Fall ein Stopp der Verfahrbewegung bei aktivem G331 / G332 verhindert werden, muss dieser Bereich mit den Befehlen DELAYFSTON und DELAYFSTOF zu einem Stop-Delay-Bereich deklariert werden.

Literatur:

Funktionshandbuch Grundfunktionen; Kapitel "Beeinflussung von Stopp-Ereignissen durch Stop-Delay-Bereiche (Seite 584)"

Hinweis

Einzelsatz

Schaltet man den Einzelsatz im Stop-Delay-Bereich ein, stoppt die NC am Ende des ersten Satzes außerhalb des Stop-Delay-Bereichs. Ist der Einzelsatz bereits vor dem Stop-Delay-Bereich angewählt, hält die NC an jeder Satzgrenze, d. h. auch im Stop-Delay-Bereich! Der Stop-Delay-Bereich ist damit abgewählt.

Overrideänderungen

Wird der Override **vor** dem Stop-Delay-Bereich verändert, wird der Override **im** Stop-Delay-Bereich wirksam.

Wird der Override **im** Stop-Delay-Bereich verändert, wird die Änderung erst **nach** dem Stop-Delay-Bereich wirksam.

- Override
Welcher Override beim Gewindebohren ohne Ausgleichsfutter wirksam ist, wird eingestellt über:
MD12090 \$MN_OVR_FUNCTION_MASK, Bit <x> = <Wert>

Bit	Wert	Bedeutung
0	0	Bei G331/G332 ist der Spindeloverride wirksam Abhängig von der Einstellung im nachfolgenden Maschinendatum bezieht sich der Override entweder auf die programmierte Spindeldrehzahl oder auf die projektierte Spindeldrehzahlbegrenzung: MD12080 \$MN_OVR_REFERENCE_IS_PROG_FEED
	1	Bei G331/G332 ist der Bahnoverride wirksam Abhängig von der Einstellung im nachfolgenden Maschinendatum bezieht sich der Override entweder auf den programmierten Bahnvorschub oder auf die projektierte Bahnvorschubbegrenzung: MD12082 \$MN_OVR_REFERENCE_IS_MIN_FEED Hinweis Dies gilt nur, wenn der Begrenzungswert niedriger ist als der programmierte Bahnvorschub.

Randbedingungen

Unwirksame Korrekturen

Folgende Korrekturen sind während des Gewindebohrens ohne Ausgleichsfutter unwirksam:

- Programmierbarer Bahnvorschub-Override `OVR`
- Eilgangkorrektur

18.2.5 Vorschub bei Gewindebohren mit Ausgleichsfutter (G63)

Funktion

G63 ist eine Teilfunktion zum Bohren von Gewinden mit einem Gewindebohrer mit Ausgleichsfutter. Ein Geber (Wegmesssystem) ist hierbei nicht erforderlich.

Drehzahl S, Vorschub F, Gewindesteigung

Bei G63 muss für die Spindel eine Drehzahl S und für die Zustellachse (Achse für Gewindelänge) ein Vorschub F programmiert werden.

Der Vorschub F ist vom Programmierer aus der Drehzahl S und der Gewindesteigung zu errechnen:

$$\text{Vorschub F [mm/min]} = \text{Drehzahl S [U/min]} * \text{Gewindesteigung [mm/U]}$$

Literatur

Weitere Informationen zu G63 siehe Programmierhandbuch Grundlagen.

18.2.6 FGROUP und FGREF

Programmierung

Für Bearbeitungsvorgänge, bei denen das Werkzeug oder das Werkstück oder beide von einer Rundachse bewegt werden (z.B. Laser-Bearbeitung von drehenden Rohren), soll der wirksame Bearbeitungsvorschub in gewohnter Weise als Bahnvorschub über den F-Wert programmiert werden können.

Dazu muss für jede der beteiligten Rundachsen ein effektiver Radius (Bezugsradius) angegeben werden. Dies wird ermöglicht durch die Programmierung der modal wirksamen NC-Adresse:

`FGREF [<Rundachse>] = <Bezugsradius>`

Die Einheit des Bezugsradius ist abhängig von der `G70/G71/G700/G710`-Einstellung.

Um überhaupt zur Berechnung des Bahnvorschubs beizutragen, müssen alle mitwirkenden Achsen wie bisher in den `FGROUP`-Befehl aufgenommen werden.

Um kompatibel zum Verhalten ohne FGREF-Programmierung zu bleiben, wird nach dem System-Hochlauf und bei RESET die Bewertung 1 Grad entspricht 1 mm wirksam.

Dies entspricht einem Bezugsradius von:

$$FGREF = 360\text{mm} / (2\pi) = 57.296 \text{ mm}$$

Diese Voreinstellung ist unabhängig vom aktiven Grundsystem (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) und von der aktuell wirksamen G70/G71/G700/G710-Einstellung.

Besonderheiten der Vorschubbewertung von Rundachsen in FGROUP:

Programmcode
N100 FGROUP(X,Y,Z,A)
N110 G1 G91 A10 F100
N120 G1 G91 A10 X0.0001 F100

Bei angegebener Programmierung wird der programmierte F-Wert in Satz N110 als Rundachsvorschub in Grad/min bewertet, während die Vorschubbewertung in Satz N120 abhängig von der aktuell wirksamen inch/metrisch-Einstellung entweder 100 inch/min oder 100 mm/min ist.

ACHTUNG

Unterschiedliche Maßsysteme

Die FGREF-Bewertung wirkt auch, wenn nur Rundachsen im Satz programmiert sind. Die gewohnte F-Wert-Interpretation als Grad/min gilt in diesem Fall nur, wenn der Radiusbezug entsprechend der FGREF-Voreinstellung ist:

- Bei G71/G710: FGREF[A]=57.296
- Bei G70/G700: FGREF[A]=57.296/25.4

Beispiel

Das folgende Beispiel soll die Wirkung von FGROUP auf den Bahnweg und Bahnvorschub verdeutlichen. Die Variable \$AC_TIME enthält die Zeit vom Satzanfang in Sekunden. Sie ist nur in Synchronaktionen verwendbar.

Programmcode	Kommentar
N100 G0 X0 A0	
N110 FGROUP(X,A)	
N120 G91 G1 G710 F100	; Vorschub= 100mm/min bzw. 100Grad/min
N130 DO \$R1=\$AC_TIME	
N140 X10	; Vorschub= 100mm/min, Bahnweg= 10mm, R1= ca.6s
N150 DO \$R2=\$AC_TIME	
N160 X10 A10	; Vorschub= 100mm/min, Bahnweg= 14.14mm, R2= ca.8s
N170 DO \$R3=\$AC_TIME	
N180 A10	; Vorschub= 100Grad/min, Bahnweg= 10Grad, R3= ca.6s
N190 DO \$R4=\$AC_TIME	
N200 X0.001 A10	; Vorschub= 100mm/min, Bahnweg= 10mm, R4= ca.6s
N210 G700 F100	; Vorschub= 2540mm/min bzw. 100Grad/min

Programmcode	Kommentar
N220 DO \$R5=\$AC_TIME	
N230 X10	; Vorschub= 2540mm/min, Bahnweg= 254mm, R5= ca.6s
N240 DO \$R6=\$AC_TIME	
N250 X10 A10	; Vorschub= 2540mm/min, Bahnweg= 254,2mm, R6= ca.6s
N260 DO \$R7=\$AC_TIME	
N270 A10	; Vorschub= 100Grad/min, Bahnweg= 10Grad, R7= ca.6s
N280 DO \$R8=\$AC_TIME	
N290 X0.001 A10	; Vorschub= 2540mm/min, Bahnweg= 10mm, R8= ca.0.288s
N300 FGREF[A]=360/(2*\$PI)	; 1 Grad = 1 inch über den effektiven Radius einstellen.
N310 DO \$R9=\$AC_TIME	
N320 X0.001 A10	; Vorschub= 2540mm/min, Bahnweg= 254mm, R9= ca.6s
N330 M30	

Diagnose

Bezugsradius lesen

Der Wert des Bezugsradius einer Rundachse kann über Systemvariablen gelesen werden:

- Für die Anzeige in der Bedienoberfläche, in Synchronaktionen oder mit Vorlaufstopp im Teileprogramm über die Systemvariable:

\$AA_FGREF[<Achse>] Aktueller Hauptlauf-Wert

- Ohne Vorlaufstopp im Teileprogramm über die Systemvariable:

\$PA_FGREF[<Achse>] Programmierter Wert

Sind keine Werte programmiert, liest man in beiden Variablen für Rundachsen die Voreinstellung $360 \text{ mm} / (2\pi) = 57.296 \text{ mm}$ (entspricht 1 mm pro Grad).

Für Linearachsen liest man in beiden Variablen immer den Wert 1 mm.

Geschwindigkeitsbestimmende Bahnachsen lesen

Die an der Bahninterpolation beteiligten Achsen können über Systemvariablen gelesen werden:

- Für die Anzeige in der Bedienoberfläche, in Synchronaktionen oder mit Vorlaufstopp im Teileprogramm über die Systemvariablen:

\$AA_FGROUP[<Achse>] Liefert den Wert "1", wenn die angegebene Achse per Grundeinstellung oder durch FGROUP-Programmierung einen Einfluss auf die Bahngeschwindigkeit im aktuellen Hauptlaufsatz hat. Wenn nicht, dann liefert die Variable den Wert "0".

\$AC_FGROUP_MASK Liefert einen Bitschlüssel der mit FGROUP programmierten Kanalachsen, die zur Bahngeschwindigkeit beitragen sollen.

- Ohne Vorlaufstopp im Teileprogramm über die Systemvariablen:

<code>\$PA_FGROUP[<Achse>]</code>	Liefert den Wert "1", wenn die angegebene Achse per Grundeinstellung oder durch <code>FGROUP</code> -Programmierung einen Einfluss auf die Bahngeschwindigkeit hat. Wenn nicht, dann liefert die Variable den Wert "0".
<code>\$P_FGROUP_MASK</code>	Liefert einen Bitschlüssel der mit <code>FGROUP</code> programmierten Kanalachsen, die zur Bahngeschwindigkeit beitragen sollen.

18.3 Vorschub für Positionierachsen (FA)

Funktion

Über den achsspezifischen Vorschub `FA` wird die Geschwindigkeit einer Positionierachse programmiert.

`FA` ist modal wirksam.

Es gilt immer die Vorschubart `G94`.

Hinweis

Die maximale Achsgeschwindigkeit (MD32000 `$MA_MAX_AX_VELO`) wird nicht überschritten.

Programmierung

Pro Teileprogrammsatz können max. 5 achsspezifische Vorschübe programmiert werden.

Syntax:

`FA[<Positionierachse>]=<Vorschubwert>`

<code><Positionierachse>:</code>	Name der Kanalachse (MD20080 <code>\$MC_AXCONF_CHANAX_NAME_TAB</code>)
<code><Vorschubwert>:</code>	Vorschubgeschwindigkeit
	Wertebereich: 0,001...999 999,999 mm/min, Grad/min bzw. 0,001...39 999,9999 inch/min

Voreinstellung

Wird kein axialer Vorschub `FA` programmiert, wirkt die axiale Voreinstellung:

MD32060 `$MA_POS_AX_VELO` (Löschstellung für Positionierachsgeschwindigkeit)

Ausgabe an die PLC

Der Vorschubwert kann an die PLC ausgegeben werden:

- an die Kanal-spezifische NC/PLC-Nahtstelle über:
DB21, ... DBB158 - DBB193
- an die Achs-spezifisch NC/PLC-Nahtstelle über:
DB31, ... DBB78 - DBB81

Der Ausgabezeitpunkt wird festgelegt mit dem Maschinendatum:

MD22240 \$MC_AUXFU_F_SYNC_TYPE (Ausgabezeitpunkt der F-Funktionen)

In der Grundeinstellung ist die Ausgabe unterdrückt (MD22240 = 3), da es durch die Ausgabe der F-Funktionen an die NC/PLC-Nahtstelle im Bahnsteuerbetrieb zu Geschwindigkeitseinbrüchen kommen kann.

Weitere Informationen siehe Funktionsbeschreibung "H2: Hilfsfunktionsausgaben an PLC (Seite 399)".

Reset-Verhalten

Das Verhalten nach Programmende oder NC-Reset wird festgelegt durch das Maschinendatum:

MD22410 \$MC_F_VALUES_ACTIVE_AFTER_RESET (F-Funktion über Reset hinaus wirksam)

Wert	Bedeutung
0	Nach NC-Reset sind die Standardwerte wirksam.
1	Nach NC-Reset sind die zuletzt programmierten FA-Werte wirksam.

18.4 Vorschubbeeinflussung

18.4.1 Vorschubsperrung und Vorschub/Spindel Halt

Funktion

Bei "Vorschubsperrung" oder "Vorschub/Spindel Halt" werden die Achsen unter Einhaltung der Bremskennlinie zum Stillstand gebracht. Die programmierte Kontur wird außer bei Gewindebearbeitung G33, eingehalten.

Kanalspezifische Vorschubsperrung

Mit dem kanalspezifischen NC/PLC-Nahtstellensignal werden alle Achsen (Geometrie- und Zusatzachsen) eines Kanals in allen Betriebsarten stillgesetzt:

DB21, ... DBX6.0 (Vorschubsperrung)

Kanalspezifischer "Vorschub Halt" für Geometrieachsen im JOG-Betrieb

Mit den kanalspezifischen NC/PLC-Nahtstellensignalen wird achsspezifisch die Verfahrbewegung der jeweiligen Geometrieachse des Kanals angehalten:

- DB21, ... DBX12.3 (Vorschub Halt, Geometrieachse 1)
- DB21, ... DBX16.3 (Vorschub Halt, Geometrieachse 2)
- DB21, ... DBX20.3 (Vorschub Halt, Geometrieachse 3)

Die Nahtstellensignale wirken nur in der Betriebsart **JOG**.

Achsspezifischer "Vorschub Halt / Spindel Halt" für Maschinenachsen

Mit dem achsspezifischen NC/PLC-Nahtstellensignal wird die Verfahrbewegung der Maschinenachse angehalten:

DB31, ... DBX4.3 (Vorschub Halt / Spindel Halt)

Betriebsarten AUTOMATIK und MDA

In den Betriebsarten AUTOMATIK und MDA gilt:

- Erfolgt "Vorschub Halt" für eine **Bahnachse**, werden alle im aktuellen Satz bewegten und am Bahnverbund beteiligten Achsen stillgesetzt.
- Erfolgt "Vorschub Halt" für eine **Positionierachse**, wird nur diese Achse stillgesetzt.

Gewindebearbeitung

Gewindebearbeitung	Wirksamkeit
G33, G34, G35	wirksam Achtung Konturabweichung
G331, G332	wirksam
G63	Achse: wirksam Spindel: nicht wirksam

Achsspezifische Achsen-/Spindelsperre für Maschinenachsen

Mit dem achsspezifischen NC/PLC-Nahtstellensignal wird die Verfahrbewegung der Maschinenachse angehalten:

DB31, ... DBX1.3 (Achsen- / Spindelsperre)

18.4.2 Vorschub-Override über Maschinensteuertafel

Funktion

Mit dem "Vorschub-Override über Maschinensteuertafel" kann der Bediener vor Ort und mit sofortiger Wirkung an der Maschine den Bahnvorschub prozentual verringern bzw. erhöhen. Dazu werden die programmierten Vorschübe mit den entsprechenden an der NC/PLC-Nahtstelle vorliegenden Korrekturwerten multipliziert.

Für Positionierachsen ist der Vorschub achsspezifisch veränderbar.

Mit der "Spindel-Override" kann die Spindeldrehzahl und die Schnittgeschwindigkeit (G96, G961) verändert werden.

Bei einer Vorschubänderung werden die axialen Beschleunigungs- und Geschwindigkeitsgrenzen eingehalten. Auf der Bahn entstehen keine Konturfehler.

Der Vorschub-Override kann für Bahn- und Positionierachsen getrennt verändert werden.

Die Korrekturen wirken auf die programmierten Werte bzw. auf die Begrenzungen (z. B. G26, LIMS bei Spindeldrehzahl).

Kanalspezifischer Vorschub- und Eilgang-Override

Für Vorschub- und Eilgangkorrektur stehen in der NC/PLC-Nahtstelle eigene Freigabesignale und Korrekturfaktoren zur Verfügung:

DB21, ... DBX6.7 (Vorschub-Override wirksam)

DB21, ... DBB4 (Vorschub-Override)

DB21, ... DBX6.6 (Eilgangkorrektur wirksam)

DB21, ... DBB5 (Eilgangkorrektur)

Die Korrekturfaktoren können von der PLC binär- oder graycodiert vorgegeben werden. Das vorliegende Format wird der NC über folgende Maschinendaten bekannt gemacht:

MD12020 \$MN_OVR_FEED_IS_GRAY_CODE (Bahnvorschub-Override-Schalter graycodiert)

MD12040 \$MN_OVR_RAPID_IS_GRAY_CODE (Eilgang-Override-Schalter graycodiert)

Bei Binärcode gilt folgende feste Zuordnung:

Binärcode	Dezimal	Korrekturfaktor
00000000	0	0,00 \pm 0%
00000001	1	0,01 \pm 1%
00000010	2	0,02 \pm 2%
00000011	3	0,03 \pm 3%
00000100	4	0,04 \pm 4%
...
01100100	100	1,00 \pm 100%
...
11001000	200	2,00 \pm 200%

Bei Graycodierung sind in den folgenden Maschinendaten die der Schalterstellung entsprechenden Korrekturfaktoren einzutragen:

MD12030 \$MN_OVR_FACTOR_FEEDRATE [<n>] (Bewertung des Bahnvorschub-Override-Schalters)

MD12050 \$MN_OVR_FACTOR_RAPID_TRA [<n>] (Bewertung des Eilgang-Override-Schalters)

Eine wirksame Vorschub-Override wirkt auf alle Bahnachsen, die dem aktuellen Kanal zugeordnet sind. Eine wirksame Eilgangkorrektur wirkt auf alle Achsen, die mit Eilgang verfahren und dem aktuellen Kanal zugeordnet sind.

Kein Eilgang-Override-Schalter vorhanden

Ist kein eigener Eilgang-Override-Schalter vorhanden, kann zwischen Eilgang- und Vorschub-Override umgeschaltet werden. Welche Korrektur wirksam sein soll, kann über PLC oder Bedientafelfront angewählt werden. Bei aktiver Eilgangkorrektur werden die Vorschub-Override-Werte auf 100% begrenzt.

- Bei Aktivierung der Eilgangkorrektur über die Bedientafelfront wird durch das PLC-Grundprogramm:
 - die Anwahl des Vorschub-Overrides für Eilgang auf das Aktivierungssignal für die Eilgangkorrektur übertragen:
DB21, ... DBX6.6 = DB21, ... DBX25.3
 - der Vorschub-Override-Wert in den Eilgang-Override-Wert übertragen:
DB21, ... DBB5 = DB21, ... DBB4
- Bei Anwahl der Eilgangkorrektur über die PLC ist durch das PLC-Anwenderprogramm:
 - das Aktivierungssignal für die Eilgangkorrektur zu setzen:
DB21, ... DBX6.6 = 1
 - der Vorschub-Override-Wert in den Eilgangkorrekturwert zu übertragen:
DB21, ... DBB5 = DB21, ... DBB4

Wirksamkeit der kanalspezifischen Vorschub- und Eilgangkorrektur:	
• Bei aktivem G33, G34, G35:	nicht wirksam
• Bei aktivem G63:	nicht wirksam
• Bei aktivem G331, G332:	nicht wirksam

Bezugsgeschwindigkeit für Bahnvorschub-Override

Die Bezugsgeschwindigkeit für die "Bahnvorschub-Override über Maschinensteuertafel" kann abweichend vom Standard (= programmierter Vorschub) gesetzt werden:

MD12082 \$MN_OVR_REFERENCE_IS_MIN_FEED

Achsspezifischer Vorschub-Override

Für jede Positionierachse sind in der NC/PLC-Nahtstelle ein Freigabesignal und ein Byte für den Vorschub-Override-Faktor vorhanden:

DB31, ... DBX1.7 (Korrektur wirksam)

DB31, ... DBB0 (Vorschub-Override)

Der Korrekturfaktor kann von der PLC binär- oder graycodiert vorgegeben werden. Das vorliegende Format wird der NC über folgendes Maschinendatum bekannt gemacht:

MD12000 \$MN_OVR_AX_IS_GRAY_CODE (Achsvorschub-Override-Schalter graycodiert)

Bei Binärcode gilt folgende feste Zuordnung:

Binärcode	Dezimal	Korrekturfaktor
00000000	0	0,00 ± 0%
00000001	1	0,01 ± 1%
00000010	2	0,02 ± 2%
00000011	3	0,03 ± 3%
00000100	4	0,04 ± 4%
...
01100100	100	1,00 ± 100%
...
11001000	200	2,00 ± 200%

Bei Graycodierung sind in das folgende Maschinendatum die der Schalterstellung entsprechenden Korrekturfaktoren einzutragen:

MD12010 \$MN_OVR_FACTOR_AX_SPEED [<n>] (Bewertung des Achsvorschub-Override-Schalters)

Wirksamkeit der achsspezifischen Vorschub-Override:	
• Bei aktivem G33, G34, G35:	nicht wirksam
• Bei aktivem G63:	nicht wirksam (die Korrektur wird in der NC fest auf 100% gesetzt)
• Bei aktivem G331, G332:	nicht wirksam (die Korrektur wird in der NC fest auf 100% gesetzt)

Spindel-Override

Für jede Spindel stehen in der NC/PLC-Nahtstelle ein Freigabesignal und ein Byte für den Spindel-Override-Faktor zur Verfügung:

DB31, ... DBX1.7 (Korrektur wirksam)

DB31, ... DBB19 (Spindel-Override)

Der Korrekturfaktor kann von der PLC binär- oder graycodiert vorgegeben werden. Das vorliegende Format wird der NC über folgendes Maschinendatum bekannt gemacht:

MD12060 \$MN_OVR_SPIND_IS_GRAY_CODE (Spindel-Override-Schalter graycodiert)

Bei Binärcode gilt folgende feste Zuordnung:

Binärcode	Dezimal	Korrekturfaktor
00000000	0	0,00 ± 0%
00000001	1	0,01 ± 1%
00000010	2	0,02 ± 2%

Binärcode	Dezimal	Korrekturfaktor
00000011	3	0,03 \pm 3%
00000100	4	0,04 \pm 4%
...
01100100	100	1,00 \pm 100%
...
11001000	200	2,00 \pm 200%

Bei Graycodierung sind in das folgende Maschinendatum die der Schalterstellung entsprechenden Korrekturfaktoren einzutragen:

MD12070 \$MN_OVR_FACTOR_SPIND_SPEED [<n>] (Bewertung des Spindel-Override-Schalters)

Wirksamkeit der "Spindel-Override":	
• Bei aktivem G33, G34, G35:	wirksam
• Bei aktivem G63:	nicht wirksam
• Bei aktivem G331, G332:	wirksam

Bezug des Spindel-Overrides

Der Spindel-Override kann sich auf die durch Maschinen- oder Settingdatum begrenzte Drehzahl oder die programmierte Drehzahl beziehen. Die Einstellung erfolgt über:

MD12080 \$MN_OVR_REFERENCE_IS_PROG_FEED (Override-Bezugsgeschwindigkeit)

Begrenzung des Korrekturfaktors

Bei binärcodierten Korrekturfaktoren können die maximal möglichen Korrekturen für Bahnvorschub, Achsvorschub und Spindeldrehzahl begrenzt werden:

MD12100 \$MN_OVR_FACTOR_LIMIT_BIN (Begrenzung bei binärcodiertem Override-Schalter)

Korrektur wirksam

Bei freigegebenen Korrekturen wirken die über Maschinensteuertafel vorgegebenen Korrekturwerte ohne Verzögerung in allen Betriebsarten und Maschinenfunktionen.

Korrektur unwirksam

Bei nicht wirksam gesetzter Korrektur wirkt intern ein Korrekturfaktor von 100%. Der an der NC/PLC-Nahtstelle anstehende Korrekturfaktor wird nicht ausgewertet.

Eine Ausnahme bildet die Nullstellung bei binärer Schnittstelle und die 1. Schalterstellung für eine graycodierte Schnittstelle. Hier werden die Korrekturfaktoren, die in die NC/PLC-Nahtstelle eingetragen sind, ausgewertet. Bei binärer Schnittstelle ist der Korrekturfaktor immer 0%. Bei graycodierter Schnittstelle wird der in den Maschinendaten für die

1. Schalterstellung eingetragene Wert als Korrekturwert ausgegeben. Dieser sollte mit "0" belegt sein.

18.4.3 Programmierbarer Vorschub-Override

Funktion

Mit der Funktion "Programmierbarer Vorschub-Override" kann das Geschwindigkeitsniveau von Bahn- und Positionierachsen über das Teileprogramm verändert werden.

Programmierung

Syntax	Bedeutung
OVR=<Wert>	Vorschubänderung für den Bahnvorschub F
OVR[<Achse>=<Wert>	Vorschubänderung für Positioniervorschub FA

Der programmierbare Bereich beträgt 0 - 200%.

Standardeinstellung: 100%

Wirksamkeit

Die NC/PLC-Nahtstellensignale DB21, ... DBB6 (Eilgang- bzw. Vorschub-Override wirksam) und DB31, ... DBX1.7 (achsspezifische Korrektur wirksam) beziehen sich **nicht** auf den programmierbaren Vorschub-Override. Bei Deaktivierung dieser Signale bleibt der programmierbare Vorschub-Override weiterhin wirksam.

Die wirksame Korrektur errechnet sich aus dem Produkt des "Programmierbaren Vorschub-Override" und der "Vorschub-Override über Maschinensteuertafel (Seite 1447)".

Die Standardeinstellung für den "Programmierbaren Vorschub-Override" ist 100%.

Die Standardeinstellung wirkt:

- wenn kein Vorschub-Override programmiert ist bzw.
- nach Reset, wenn das Maschinendatum:
MD22410 \$MC_F_VALUES_ACTIVE_AFTER_RESET (F-Funktion über Reset hinaus wirksam)
nicht gesetzt ist.

Hinweis

OVR wirkt nicht bei G33, G34, G35.

18.4.4 Probelaufvorschub

Funktion

Der Probelaufvorschub wird beim Einfahren von Teileprogrammen verwendet, um das Programm oder Programmteile mit erhöhtem Bahnvorschub ablaufen zu lassen, z. B. bei einem Testlauf ohne Werkstückbearbeitung.

Aktivierung

Der Probelaufvorschub kann in den Automatikbetriebsarten angewählt und über PLC oder Bedientafelfront aktiviert werden.

Bei Aktivierung über die Bedientafelfront wird das Nahtstellensignal:
DB21, ... DBX24.6 (Probelaufvorschub angewählt)
gesetzt und vom PLC-Grundprogramm auf das Nahtstellensignal:
DB21, ... DBX0.6 (Probelaufvorschub aktivieren)
übertragen.

Bei Anwahl über PLC ist das Nahtstellensignal DB21, ... DBX0.6 (Probelaufvorschub aktivieren) vom PLC-Anwenderprogramm zu setzen.

Wirksamkeit

Solange das Nahtstellensignal "Probelaufvorschub aktivieren" ansteht, wirkt anstelle des programmierten Vorschubs der über SD42100 \$SC_DRY_RUN_FEED eingestellte Vorschubwert in der über SD42101 \$SC_DRY_RUN_FEED_MODE festgelegten Wirkungsweise (siehe Parametrierung).

Der Probelaufvorschub wird immer als Linearvorschub (G94) interpretiert.

Parametrierung

Aktivierung des Probelaufvorschubs

Der Zeitpunkt der Aktivierung ist abhängig von der Einstellung im Maschinendatum:

MD10704 \$MN_DRYRUN_MASK (Aktivierung des Probelaufvorschubs)

Wert	Bedeutung
0	Der Probelaufvorschub darf nur am Satzende ein- und ausgeschaltet werden (Standardeinstellung)
1	Der Probelaufvorschub kann auch während der Programmbearbeitung (im Teileprogrammsatz) aktiviert werden. Achtung: Das Einschalten während der Bearbeitung führt steuerungsintern zu einem Reorganisationsvorgang, bei dem die Achsen kurzfristig gestoppt werden. Dies kann Einflüsse auf die Oberflächengüte des bearbeiteten Werkstücks bewirken.
2	Der Probelaufvorschub kann in jeder Phase ein- und ausgeschaltet werden, ohne dass die Achsen gestoppt werden. Allerdings wird die Funktion erst mit einem im Programmablauf "späteren" Satz wirksam.

Änderung des Probelaufvorschubs

Der Vorschub für den Probelauf wird eingetragen in das Settingdatum:

SD42100 \$SC_DRY_RUN_FEED (Probelaufvorschub)

Das Settingdatum kann über die Bedientafelfront im Bedienbereich "Parameter" verändert werden.

Wenn die Anwahl vom NC akzeptiert wurde, wird das folgende NC/PLC-Nahtstellensignal gesetzt:

DB21, ... DBX318.6 (Probelauf Vorschub aktiv)

Ein aktiver Probelaufvorschub wird in der Statuszeile der Bedientafelfront mit "DRY" angezeigt, wenn:

- die Anwahl bei Programmstopp am Satzende erfolgte oder
- das Maschinendatum MD10704 \$MN_DRYRUN_MASK während der Programmbearbeitung auf "1" gesetzt wurde.

Wirkungsweise des Probelaufvorschubs

Die Wirkungsweise des in SD42100 eingetragenen Probelaufvorschubs kann eingestellt werden über das Settingdatum:

SD42101 \$SC_DRY_RUN_FEED_MODE

Wert	Bedeutung
0	Es wird der programmierte Vorschub mit dem Probelaufvorschub in SD42100 verglichen und danach mit dem größeren der beiden Vorschübe verfahren (Standardeinstellung).
1	Es wird der programmierte Vorschub mit dem Probelaufvorschub in SD42100 verglichen und danach mit dem kleineren der beiden Vorschübe verfahren.
2	Es wird direkt der in SD42100 eingetragene Probelaufvorschub wirksam, unabhängig von der programmierten Geschwindigkeit.
3 -9	reserviert
10	Wie Projektierung 0, außer Gewindeschneiden (G33, G34, G35) und Gewindebohren (G331, G332, G63). Diese Funktionen werden wie programmiert ausgeführt.
11	Wie Projektierung 1, außer Gewindeschneiden (G33, G34, G35) und Gewindebohren (G331, G332, G63). Diese Funktionen werden wie programmiert ausgeführt.
12	Wie Projektierung 2, außer Gewindeschneiden (G33, G34, G35) und Gewindebohren (G331, G332, G63). Diese Funktionen werden wie programmiert ausgeführt.

18.4.5 Mehrere Vorschubwerte in einem Satz

Funktion

Mit der Funktion "Mehrere Vorschubwerte in einem Satz" können abhängig von externen digitalen und/oder analogen Eingängen 6 verschiedene Vorschubwerte eines NC-Satzes, eine Verweilzeit sowie ein Rückzug bewegungssynchron aktiviert werden.

Wenn der Eingang für Ausfeuerzeit bzw. Rückzugsweg aktiviert wird, wird der Restweg für Bahnachsen oder die betreffenden Einzelachsen gelöscht und die Verweilzeit bzw. der Rückzug gestartet.

Die Einleitung des Rückzugs erfolgt innerhalb eines IPO-Takts.

Signale

Die Eingangssignale werden für die Funktion in einem Eingangsbyte zusammengefasst. Innerhalb des Bytes existiert eine feste funktionelle Zuordnung.

Tabelle 18-1 Eingangsbyte für die Funktion "Mehrere Vorschubwerte in einem Satz"

	Bit							
	7	6	5	4	3	2	1	0
Eingangs-Nr.	E7	E6	E5	E4	E3	E2	E1	E0
Vorschubadresse	F7	F6	F5	F4	F3	F2	ST	SR

E7 bis E2: Aktivierung der Vorschübe F7 bis F2

E1: Aktivierung der Verweilzeit ST/STA (in Sekunden)

E0: Aktivierung der Rückzugsbewegung SR/SRA

Priorität der Signale

Die Abfragereihenfolge der Signale erfolgt von E0 aufsteigend. Somit besitzt die Rückzugsbewegung (SR) die höchste und der Vorschub F7 die niedrigste Priorität.

SR und ST beenden die Vorschubbewegungen, die mit F2 bis F7 aktiviert wurden. SR beendet auch ST, d. h. die komplette Funktion.

Das Signal mit der höchsten Priorität bestimmt den aktuellen Vorschub.

Das Verhalten bei Abfall des jeweils höchstpriorien Eingangs (F2 - F7) kann festgelegt werden mit dem Maschinendatum:

MD21230 \$MC_MULTFEED_STORE_MASK (Speicherverhalten für die Funktion "Mehrere Vorschubwerte in einem Satz")

Bit	Wert	Bedeutung
2 ... 7	0	Bei Abfall des jeweils höchstpriorien Eingangs wird der zugehörige Vorschub nicht beibehalten (Standardeinstellung)
	1	Gesetztes Bit 2 ... 7 bewirkt, dass der zugehörige Vorschub (F2 bis F7), der vom jeweils höchstpriorien Eingangssignal angewählt wurde, auch dann beibehalten wird, wenn das Eingangssignal abfällt und ein niederpriorier Eingang ansteht.

Das Satzendekriterium ist erfüllt bei:

- Erreichen der programmierten Endposition
- Ende der Rückzugsbewegung (SR)
- Nach Ablauf der Verweilzeit (ST)

Hardware-Zuordnung

Dem Eingangsbyte für die Funktion "Mehrere Vorschubwerte in einem Satz" können maximal zwei digitale Eingangsbytes bzw. Komparator-Eingangsbytes der NC-Peripherie zugeordnet werden:

MD21220 \$MC_MULTFEED_ASSIGN_FASTIN (Zuordnung der Eingangsbytes der NC-Peripherie für "Mehrere Vorschubwerte in einem Satz"), Bit 0 ... 15

Weiterhin können die Eingangsbits invertiert werden:

MD21220 \$MC_MULTFEED_ASSIGN_FASTIN, Bit 16 ... 31

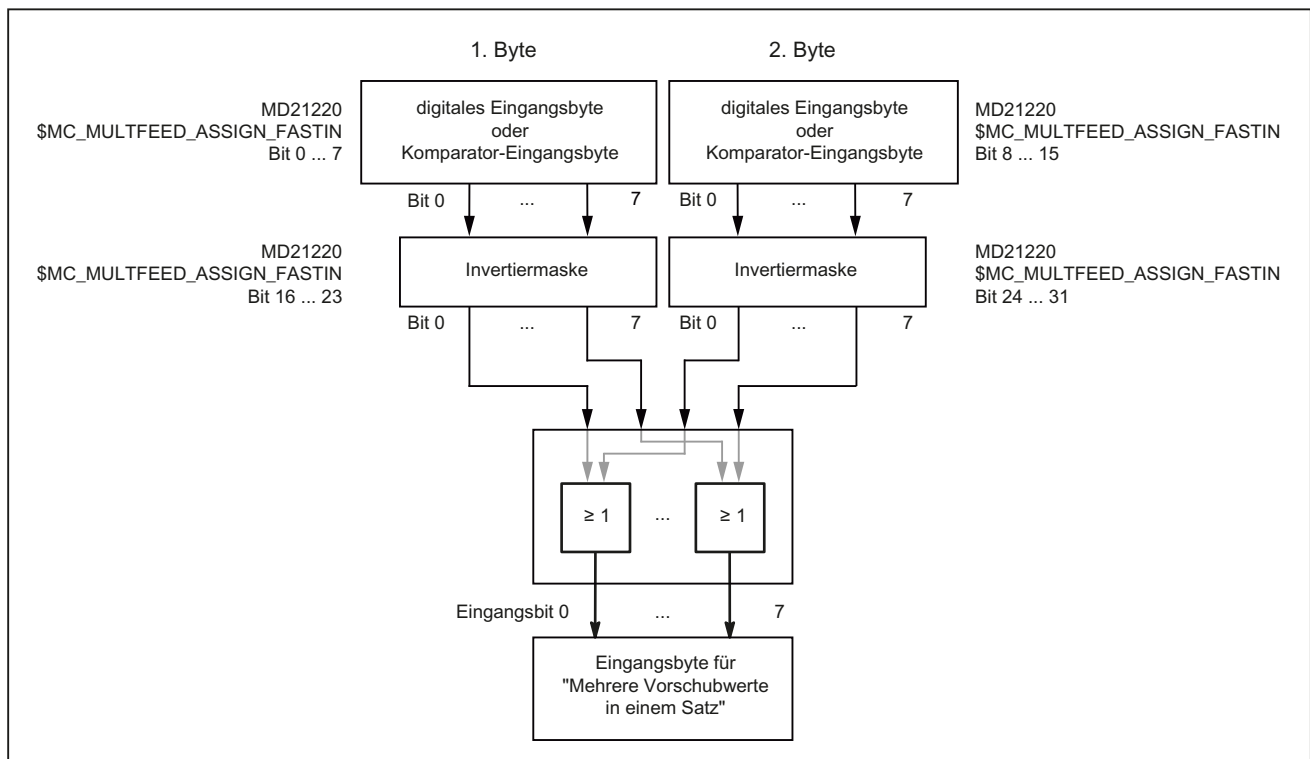


Bild 18-10 Rangierung der Signale für die Funktion "Mehrere Vorschubwerte in einem Satz"

Die Rangierung der digitalen Eingangsbytes und Parametrierung der Komparatoren sind beschrieben in:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Digitale und analoge NC-Peripherie (A4)

Programmierung

Bahnbewegung

Unter der Adresse F wird der Bahnvorschub programmiert, der gültig ist, solange kein Eingangssignal ansteht. Dieser ist modal wirksam.

Mit $F2 = \dots$ bis $F7 = \dots$ können zusätzlich zum Bahnvorschub bis zu 6 weitere Vorschübe im Satz programmiert werden. Die numerische Erweiterung gibt die Bitnummer des Eingangs an, mit dessen Änderung der Vorschub wirksam wird.

Beispiel:

$F7 = 1000$; 7 entspricht Eingangsbit 7

Die programmierten Werte wirken satzweise. Im Folgesatz gilt der unter F programmierte Bahnvorschub.

Verweilzeit (Ausfeuerzeit) und Rückzugsweg werden unter zusätzlichen Adressen im Satz programmiert:

$ST = \dots$ Verweilzeit (bei Schleifen Ausfeuerzeit)
 $SR = \dots$ Rückzugsweg

Diese Adressen wirken satzweise.

Axiale Bewegung

Unter der Adresse FA werden axiale Vorschübe programmiert, die gültig sind, solange kein Eingangssignal ansteht. Sie sind modal wirksam.

Mit $FMA[2, <Achse>] = \dots$ bis $FMA[7, <Achse>] = \dots$ können zusätzlich bis zu 6 weitere Vorschübe pro Achse im Satz programmiert werden.

Der erste Ausdruck in eckigen Klammern gibt die Bitnummer des Eingangs an, mit dessen Änderung der Vorschub wirksam wird. Der zweite Ausdruck gibt an, für welche Achse der Vorschub gilt.

Beispiel:

$FMA[3, Y] = 1000$; axialer Vorschub für Y-Achse, entspricht Eingangsbit 3

Die unter FMA programmierten Werte wirken satzweise. Im Folgesatz gilt der unter FA programmierte Vorschub.

Verweilzeit (Ausfeuerzeit) und Rückzugsweg können zusätzlich für einzelne Achsen vorgegeben werden:

$STA[<Achse>] = \dots$ axiale Verweilzeit (Ausfeuerzeit)
 $SRA[<Achse>] = \dots$ axialer Rückzugsweg

Der Ausdruck in eckiger Klammer gibt an, für welche Achse Ausfeuerzeit bzw. Rückzugsweg gelten.

Beispiele:

STA[X]=2.5 ; Die Ausfeuerzeit für Achse X beträgt 2,5 Sekunden.
SRA[X]=3.5 ; Der Rückzugsweg für Achse X beträgt 3,5 (Einheit z. B. mm).

Diese Adressen wirken satzweise.

Hinweis

Rückzugsweg

Die Einheit für den Rückzugsweg bezieht sich auf die aktuell gültige Maßeinheit (mm oder inch).

Der Rückhub erfolgt immer in Gegenrichtung zur aktuellen Bewegung. Mit SR/SRA wird immer der Betrag des Rückhubs programmiert. Es wird kein Vorzeichen programmiert.

Hinweis

POS statt POSA

Wenn für eine Achse Vorschübe, Ausfeuerzeit (Verweilzeit) oder Rückzugsweg aufgrund eines externen Eingangs programmiert sind, darf diese Achse in diesem Satz nicht als POSA-Achse (Positionierachse über Satzgrenze hinweg) programmiert werden.

Hinweis

Statusabfrage

Der Status eines Eingangs kann auch für Synchronbefehle unterschiedlicher Achsen abgefragt werden.

Hinweis

LookAhead

LookAhead ist auch bei mehreren Vorschüben in einem Satz wirksam. Damit kann der aktuelle Vorschub durch LookAhead begrenzt werden.

Anwendung

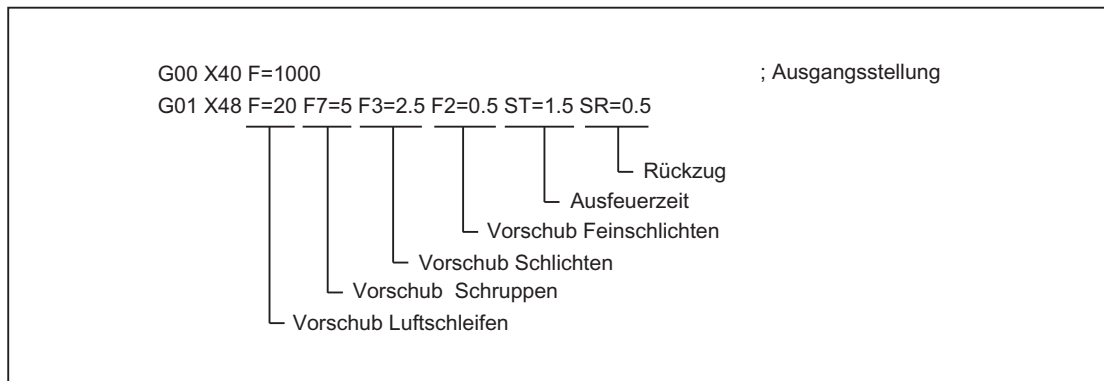
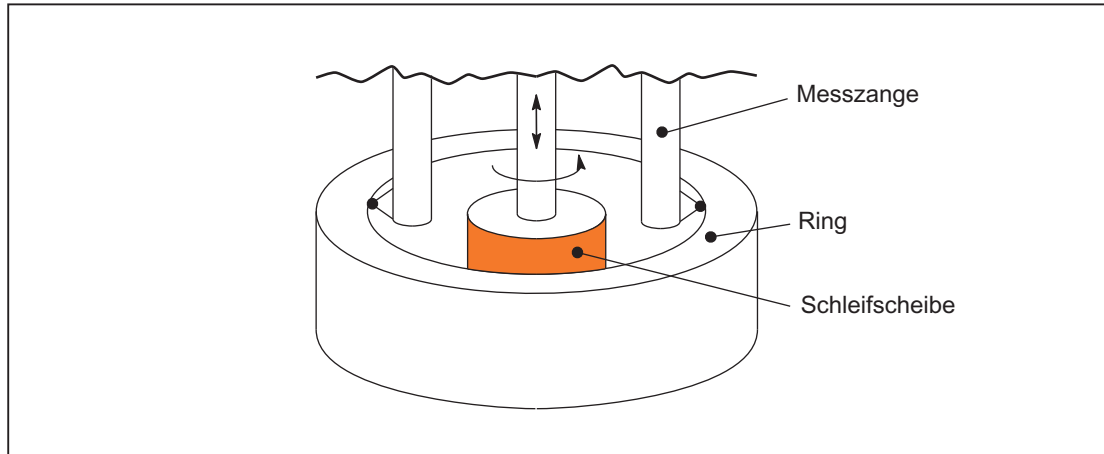
Die Funktion "Mehrere Vorschubwerte in einem Satz" wird schwerpunktmäßig bei der Technologie Schleifen eingesetzt. Sie ist jedoch nicht darauf beschränkt.

Typische Anwendungsfälle sind z. B.:

- Analoge oder digitale Messzange
Abhängig von externen analogen oder digitalen Eingängen können verschiedene Vorschubwerte, eine Verweilzeit sowie ein Rückzugsweg aktiviert werden, wobei die Vorgabe der Schwellwerte über Settingdaten erfolgt.
- Umschalten von Zustell- auf Arbeitsvorschub über Näherungsschalter

Beispiel

Innenschleifen eines Kugellagerringes, wobei der Ist-Durchmesser über eine Messzange erfasst wird und, abhängig von Schwellwerten, der jeweils notwendige Vorschubwert für Schruppen, Schlichten oder Feinschlichten aktiviert wird. Die Messzangenposition liefert auch die Endposition. Somit wird das Satzendeckriterium nicht nur von der programmierten Achsposition der Zustellachse, sondern auch von der Messzange bestimmt.



Einschränkungen für die Funktion bei SINUMERIK 828D

MD21220 =	1H	1.Byte (4 Bits auf X242 und 4 Bits über Nahstelle)
	2H	2.Byte (4 Bits auf X252 und 4 Bits über Nahstelle)
	102H	8 Bits nur über Nahstelle

Bei SINUMERIK 828D kann aufgrund der eingeschränkten Hardware-Eingangsklemmen kein ganzes Byte angesprochen werden. Das bedeutet wiederum deshalb Folgendes:

- Hardwaremäßig können nur 4 verschiedene Vorschübe programmiert werden (4 Bits auf X242 oder 4 Bits auf X252 an der PPU)
- Die verbleibenden 4 Bits werden über die Nahtstelle realisiert.
- Es wird „nur“ die Nahtstelle verwendet, keine Hardware-Eingänge. Dadurch laufen die Vorschübe nicht mehr in einem IPO-Takt.

Tabelle der Zuordnung Vorschubadresse zu Eingang bzw. Nahtstelle:

Vorschubadresse	1. Byte	\$A_IN[x]	Nahtstelle	2. Byte	\$A_IN[x]	Nahtstelle
SR	0	1	DB2800.DBX1.0	0	9	DB2800.DBX1001.0
ST	1	2	DB2800.DBX1.1	1	10	DB2800.DBX1001.1
F2	2	3	DB2800.DBX1.2	2	11	DB2800.DBX1001.2
F3	3	4	DB2800.DBX1.3	3	12	DB2800.DBX1001.3
F4	4	5	DB2800.DBX1.4	4	13	DB2800.DBX1001.4
F5	5	6	DB2800.DBX1.5	5	14	DB2800.DBX1001.5
F6	6	7	DB2800.DBX1.6	6	15	DB2800.DBX1001.6
F7	7	8	DB2800.DBX1.7	7	16	DB2800.DBX1001.7

Beispiel:

G90 G0 X600

POS[X]=700 FA[X]=1000

POS[X]=700 FA[3,X]=2000 -> (wird der \$A_IN[4]=1 -> so beträgt der Vorschubwert 2000, andernfalls gilt der Wert 1000 für den Vorschub.)

Inbetriebnahme:

4 Vorschubadressen durch die Hardware-Klemmen und 4 durch die Nahtstelle

- -X242 = IN1 bis IN4 verdrahten (PIN 3 – 6) -> SR, ST, F2, F3
- Nahtstelle: DB2800.DBX1.4 – 1.7 = IN5 – IN8 -> F4, F5, F6, F7
- MD21220 = 1H

18.4.6 Feste Vorschubwerte

Funktion

Mit der Funktion "Feste Vorschubwerte" können anstelle des programmierten Vorschubs bzw. der projektierten JOG-Geschwindigkeiten über Maschinendatum definierte Festvorschübe (max. 4) aktiviert werden.

Die Funktion steht in den Betriebsarten AUTOMATIK und JOG zur Verfügung.

Verhalten in der Betriebsart AUTOMATIK

Anstelle des programmierten Vorschubs wird die Kontur mit dem aktivierten Festvorschub verfahren.

Verhalten in der Betriebsart JOG

Anstelle der projektierten JOG-Geschwindigkeit/JOG-Eilgang-Geschwindigkeit wird die Achse mit dem aktivierten Festvorschub verfahren. Die Verfahrrichtung wird über Nahtstellensignale vorgegeben.

Parametrierung

Die Einstellung der Festvorschübe erfolgt:

- für Linearachsen mit dem Maschinendatum:
MD12202 \$MN_PERMANENT_FEED[<n>]
- für Rundachsen mit dem Maschinendatum:
MD12204 \$MN_PERMANENT_ROT_AX_FEED[<n>]

mit <n> = 0, 1, 2, 3 (für Festvorschub 1, 2, 3, 4)

Hinweis

Die Festvorschübe sind immer Linearvorschubwerte. Auch bei Umdrehungsvorschub wird intern auf Linearvorschub umgestellt.

Aktivierung

Die Aktivierung der Festvorschübe erfolgt über NC/PLC-Nahtstellensignale:

- in der Betriebsart AUTOMATIK für Bahn-/Geometrieachsen über die kanalspezifischen Nahtstellensignale:
DB21, ... DBX29.0 (Festvorschub 1 aktivieren)
DB21, ... DBX29.1 (Festvorschub 2 aktivieren)
DB21, ... DBX29.2 (Festvorschub 3 aktivieren)
DB21, ... DBX29.3 (Festvorschub 4 aktivieren)
- in der Betriebsart JOG für Maschinenachsen über die achsspezifischen Nahtstellensignale:
DB31, ... DBX3.2 (Festvorschub 1 aktivieren)
DB31, ... DBX3.3 (Festvorschub 2 aktivieren)
DB31, ... DBX3.4 (Festvorschub 3 aktivieren)
DB31, ... DBX3.5 (Festvorschub 4 aktivieren)

Randbedingungen

Wirksamkeit

Die Funktion "Feste Vorschubwerte" ist **nicht** wirksam:

- Bei Spindeln
- Bei Positionierachsen
- Beim Gewindebohren

Override = 0

Das Fahrverhalten bei Override = 0 ist abhängig von der Einstellung im Maschinendatum:

MD12200 \$MN_RUN_OVERRIDE_0

DRF-Verschiebung

Bei angewähltem Festvorschub kann die DRF-Verschiebung nicht aktiviert werden.

18.4.7 Programmierbare Vorschubverläufe

Funktion

Zur flexiblen Vorgabe des Vorschubverlaufs wird die Vorschubprogrammierung nach DIN 66025 um lineare und kubische Verläufe erweitert.

Die kubischen Verläufe können direkt oder als interpolierender Spline programmiert werden.

Programmierung

Folgende Vorschubprofile können programmiert werden:

- **FNORM**
Verhalten gemäß DIN 66025 (Standardeinstellung).
Ein im Satz programmierter F-Wert wird über den gesamten Bahnweg des Satzes konstant vorgegeben und gilt danach als fester modaler Wert.
- **FLIN**
Ein im Satz programmierter F-Wert wird vom aktuellen Wert am Satzanfang bis zum Satzende linear über den Bahnweg eingefahren und gilt danach als modaler Wert.
- **FCUB**
Die satzweise programmierten F-Werte werden - bezogen auf den Satzendpunkt - durch einen Spline verbunden. Der Spline beginnt und endet tangential zur vorhergehenden bzw. nachfolgenden Vorschubvorgabe. Fehlt die F-Adresse in einem Satz, wird der zuletzt programmierte F-Wert dafür verwendet.
- **FPO**
Die F-Adresse [Syntax: $F=FPO(\dots, \dots, \dots)$] bezeichnet den Vorschubverlauf über ein Polynom vom aktuellen Wert bis zum Ende des Satzes, in dem sie programmiert wurde. Der Endwert gilt von da an modal.

Parametrierung

Wenn FLIN und FCUB in Verbindung mit Kompression COMPON verwendet wird, kann für den Bahnvorschub eine Toleranz festgelegt werden:

MD20172 \$MC_COMPRESS_VELO_TOL (maximal erlaubte Abweichung des Bahnvorschubs bei Kompression)

Randbedingungen

FLIN/FCUB

Das mit FLIN oder FCUB programmierte Bahngeschwindigkeitsprofil wirkt nicht zusammen mit Umdrehungsvorschub bei G95 sowie mit konstanter Schnittgeschwindigkeit bei G96/G961 und G97/G971.

Literatur

Weitere Informationen zu den programmierbaren Vorschubverläufen siehe Programmierhandbuch Arbeitsvorbereitung.

18.4.8 Vorschub für Fase/Rundung FRC, FRCM

Beim Übergang von Flächen zur Fase/Rundung können sich die Zerspanungsverhältnisse wesentlich ändern. Deshalb benötigen die Konturelemente Fase/Rundung zum Erreichen der gewünschten Oberflächengüte eigene, optimierte Vorschubwerte.

Funktion

Der Vorschub für Fase/Rundung ist über NC-Adressen programmierbar.

Programmierung

Syntax:

... FRC/FRCM=<Wert>

Bedeutung:

FRC: **satzweise** wirksamer Vorschub für Fase/Rundung

FRCM: **modal** wirksamer Vorschub für Fase/Rundung

<Wert>: Der Vorschubwert wird entsprechend dem aktiven Vorschubtyp interpretiert:

- G94, G961, G971: Vorschub in mm/min, inch/min oder °/min
- G95, G96, G97: Umdrehungsvorschub in mm/Umdr oder inch/Umdr

Hinweis

FRC ist nur wirksam, wenn im Satz eine Fase/Rundung programmiert ist bzw. RNDM aktiviert wurde.

FRC überschreibt im aktuellen Satz den F- bzw. FRCM-Wert.

Der unter FRC programmierte Vorschub muss größer Null sein.

FRCM=0 aktiviert für das Anfasen/Verrunden den unter F programmierten Vorschub.

Parametrierung

Zuordnung der Fase/Rundung zum Vorgänger- oder Nachfolgesatz

Der Vorschubtyp (G94, G95, G96, G961 ...) und damit die Konvertierung in das interne Format müssen im Satz einheitlich für F und FRC/FRCM sein. In diesem Zusammenhang muss das folgende Maschinendatum berücksichtigt werden:

MD20201 \$MC_CHFRND_MODE_MASK (Verhalten Fase/Rundung)

Bit	Wert	Bedeutung
0	0	Die Technologie der Fase/Rundung (Vorschub, Vorschubtyp, M-Befehle ...) wird vom Nachfolgesatz bestimmt (Standardeinstellung).
	1	Die Technologie der Fase/Rundung wird vom Vorgängersatz bestimmt (empfohlene Einstellung).

Maximale Anzahl Leersätze

Die Anzahl an Sätzen ohne Verfahreninformationen in der Korrektorebene (Leersätze), die bei aktiver Fase/Rundung zwischen zwei Sätzen mit Verfahreninformation stehen dürfen, ist begrenzt. Die maximale Anzahl wird festgelegt im Maschinendatum:

MD20200 \$MC_CHFRND_MAXNUM_DUMMY_BLOCKS (Leersätze bei Fase/Radien)

Randbedingungen**FLIN/FCUB**

Vorschubinterpolation FLIN und FCUB ist für Fase/Rundung **nicht** möglich.

G0

FRC/FRCM wirkt nicht, wenn eine Fase mit G0 verfahren wird. Die Programmierung ist entsprechend dem F-Wert ohne Fehlermeldung möglich.

Wechsel G94 ↔ G95

Ist FRCM programmiert, so muss äquivalent zu F der FRCM-Wert bei Wechsel G94 ↔ G95 usw. neu programmiert werden. Wird nur F neu programmiert und ist vor dem Wechsel des Vorschubtyps FRCM > 0, dann erfolgt eine Fehlermeldung.

Beispiel**Beispiel 1: MD20201 Bit 0 = 0; Vorschub vom Nachfolgesatz übernehmen (Standardeinstellung!)**

Programmcode	Kommentar
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Fase N20-N30 mit F=100 mm/min
N30 Y10 CHF=4	; Fase N30-N40 mit FRC=200 mm/min
N40 X20 CHF=3 FRC=200	; Fase N40-N60 mit FRCM=50 mm/min
N50 RNDM=2 FRCM=50	
N60 Y20	; Modale Rundung N60-N70 mit FRCM=50 mm/min
N70 X30	; Modale Rundung N70-N80 mit FRCM=50 mm/min
N80 Y30 CHF=3 FRC=100	; Fase N80-N90 mit FRC=100 mm/min
N90 X40	; Modale Rundung N90-N100 mit F=100 mm/min (Abwahl FRCM)
N100 Y40 FRCM=0	; Modale Rundung N100-N120 mit G95 FRC=1 mm/Umdr
N110 S1000 M3	
N120 X50 G95 F3 FRC=1	
...	
M02	

Beispiel 2: MD20201 Bit 0 = 1; Vorschub vom Vorgängersatz übernehmen (empfohlene Einstellung!)

Programmcode	Kommentar
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Fase N20-N30 mit F=100 mm/min
N30 Y10 CHF=4 FRC=120	; Fase N30-N40 mit FRC=120 mm/min
N40 X20 CHF=3 FRC=200	; Fase N40-N60 mit FRC=200 mm/min
N50 RNDM=2 FRCM=50	
N60 Y20	; Modale Rundung N60-N70 mit FRCM=50 mm/min
N70 X30	; Modale Rundung N70-N80 mit FRCM=50 mm/min
N80 Y30 CHF=3 FRC=100	; Fase N80-N90 mit FRC=100 mm/min
N90 X40	; Modale Rundung N90-N100 mit FRCM=50 mm/min
N100 Y40 FRCM=0	; Modale Rundung N100-N120 mit F=100 mm/min
N110 S1000 M3	
N120 X50 CHF=4 G95 F3 FRC=1	; Fase N120-N130 mit G95 FRC=1 mm/Umdr
N130 Y50	; Modale Rundung N130-N140 mit F=3 mm/Umdr
N140 X60	
...	
M02	

18.4.9 Satzweiser Vorschub FB**Funktion**

Mit der Funktion "Satzweiser Vorschub" kann für einen einzelnen Teileprogrammsatz ein separater Vorschub vorgegeben werden. Nach diesem Satz ist der zuvor wirksame modale Bahnvorschub wieder aktiv.

Programmierung**Syntax:**

... FB=<Wert>

Bedeutung:

FB: separater Vorschub für den aktuellen Satz

<Wert>: Der Vorschubwert wird entsprechend dem aktiven Vorschubtyp interpretiert:

- G94, G961, G971: Vorschub in mm/min, inch/min oder °/min
- G95, G96, G97: Umdrehungsvorschub in mm/Umdr oder inch/Umdr

Hinweis

Der unter FB programmierte Vorschub muss größer Null sein.

Ist im Satz keine Verfahrbewegung programmiert (z. B. Rechensatz), bleibt FB ohne Wirkung.

Ist kein expliziter Vorschub für Fase/Rundung programmiert, gilt der Wert von FB auch für ein in diesem Satz vorhandenes Konturelement Fase/Rundung.

Die gleichzeitige Programmierung von FB und FD (manuelles Verfahren per Handrad mit Vorschubüberlagerung) oder F (modaler Bahnvorschub) ist nicht möglich.

18.4.10 Beeinflussung der Einzelachsdynamik

Einzelachsen

Einzelachsen können im Teileprogramm, in Synchronaktionen und über die PLC programmiert werden:

- Teileprogramm:
 - POS[<Achse>]=...
 - POSA[<Achse>]=...
 - SPOS[<Achse>]=...
 - SPOSA[<Achse>]=...
 - OS[<Achse>]=...
 - OSCILL[<Achse>]=...
- Synchronaktionen:
 - EVERY ... DO
 - POS[<Achse>]=...
 - SPOS[<Spindel>]=...
 - MOV[<Achse>]=...
- PLC:
 - FC18

Dynamikverhalten

Das Dynamikverhalten einer Achse wird beeinflusst durch:

- MD32060 \$MA_POS_AX_VELO (Positionierachsgeschwindigkeit)
Die wirksame Positionierachsgeschwindigkeit kann verändert werden:
 - Teileprogramm/Synchronaktion: Axialer Vorschub FA oder prozentuale Vorschubkorrektur OVRA
 - PLC: Vorgabe von FRate bzw. Überschreiben des axialen Override
- MD32300 \$MA_MAX_AX_ACCEL (maximale Achsbeschleunigung)
Die wirksame maximale Achsbeschleunigung kann verändert werden:
 - Teileprogramm indirekt: Schreiben des Maschinendatums mit anschließendem "Maschinendaten wirksam setzen"
 - Teileprogramm direkt: Prozentuale Beschleunigungskorrektur ACC
 - Synchronaktionen indirekt: Schreiben des Maschinendatums und Auslösen eines ASUP für die Aktivierung der Funktion "Maschinendaten wirksam setzen"
 - Synchronaktionen direkt: Prozentuale Beschleunigungskorrektur ACC (nicht von der PLC aus vorgebbar).

Über PLC bestehen die gleichen Möglichkeiten wie in Synchronaktionen.

- Teileprogrammbefehle: BRISKA, SOFTA, DRIVEA, JERKA
Nicht in Synchronaktionen programmierbar (nur indirekt über ASUP).
Nicht von PLC vorgebbar (nur indirekt über ASUP).
- Aktiven Servo-Parametersatz
Der aktive Parametersatz kann verändert werden:
 - Teileprogramm/Synchronaktion: SCPARA
 - PLC: DB31, ... DBX9.0-2 (Regler-Parametersatz)

Ausführliche Informationen zu den Servo-Parametersätzen siehe "Parametersätze des Lagereglers (Seite 383)".

Hinweis

Dynamikänderungen

Im Teileprogramm vorgenommene Dynamikänderungen wirken nicht auf Kommando- bzw. PLC-Achsbewegungen. In Synchronaktionen vorgenommenen Dynamikänderungen wirken nicht auf die im Teileprogramm programmierten Verfahrbewegungen.

Vorsteuerung

Die Art der Vorsteuerung und welche Bahnachsen mit Vorsteuerung verfahren sollen, kann im Teileprogramm direkt mit FFWON/FFWOF programmiert werden. In Synchronaktionen und von PLC aus kann eine Programmierung nur indirekt über ein ASUP vorgenommen werden.

Prozentuale Beschleunigungskorrektur (ACC)

Im Teileprogramm oder Synchronaktion kann mit dem Befehl ACC die im Maschinendatum: MD32300 \$MA_MAX_AX_ACCEL (maximale Achsbeschleunigung) vorgegebene Beschleunigung in einem Bereich von 0% – 200% verändert werden.

Syntax:

ACC [<Achse>] = <Wert>

Bedeutung:

ACC: Schlüsselwort zur Programmierung der prozentualen Beschleunigungskorrektur
 <Achse>: Name der Kanalachse bzw. Spindel
 <Wert>: Beschleunigungsänderung in Prozent bezogen auf MD32300
 Wertebereich: 0 ... 200

Der aktuelle axiale Beschleunigungswert kann über die Systemvariable \$AA_ACC gelesen werden. Er bestimmt sich aus:

$$\$AA_ACC[<Achse>] = (MD32300 \$MA_MAX_AX_ACCEL[<Achse>]) * ACC[<Achse>] / 100$$

Über MD32320 \$MA_DYN_LIMIT_RESET_MASK kann die Löschestellung des mit ACC programmierten Werts bei Kanal-Reset bzw. Teileprogrammende M30 vorgegeben werden.

Hinweis

Die mit ACC programmierte Beschleunigungskorrektur kann über die Systemvariable \$AA_ACC gelesen werden. Das Lesen von \$AA_ACC im Teileprogramm findet jedoch zu einem anderen Zeitpunkt statt als das Lesen in einer Synchronaktion.

Die Systemvariablen \$AA_ACC enthält nur dann den im Teileprogramm mit ACC programmierten Wert, wenn die Beschleunigungskorrektur nicht zwischenzeitlich durch Programmieren von ACC in einer Synchronaktion verändert wurde. Gleiches gilt für den umgekehrten Fall.

Prozentuale Beschleunigungskorrektur und Hauptlauf-Achsen

Je nachdem, ob die Systemvariable \$AA_ACC im Teileprogramm oder Synchronaktion gelesen wird, wird der Wert für die mit ACC programmierte Beschleunigungskorrektur für NC-Achsen oder Hauptlaufachsen (Kommandoachsen, PLC-Achsen, asynchrone Pendelachsen etc.) ausgegeben.

Für korrekte Ergebnisse muss die Systemvariable \$AA_ACC daher immer an der gleichen Stelle (Teileprogramm oder Synchronaktion) gelesen werden, von der aus die Beschleunigungskorrektur mit ACC programmiert wurde.

Beispiele:

Schreiben von ACC im Teileprogramm:

```
N80 G01 POS[X]=100 FA[X]=1000 ACC[X]=90 IPOENDA[X]
```

Schreiben von ACC in Synchronaktion:

```
N100 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140 IPOENDA[X]
```

Schreiben von ACC und Lesen von \$AA_ACC im Teileprogramm:

```
ACC[X]=50 ; Schreiben
RO=$AA_ACC[X] ; Lesen

IF (RO <> $MA_MAX_AX_ACCEL[X] * 0,5) ; Überprüfen
    SETAL(61000)
ENDIF
```

Schreiben von ACC und Lesen von \$AA_ACC in Synchronaktion:

```
WHEN TRUE DO ACC[X]=25 R0=$AA_ACC[X] ; Schreiben und Lesen
G4 F1

IF (RO <> $MA_MAX_AX_ACCEL[X] * 0,25) ; Überprüfen
    SETAL(61001)
ENDIF
```

Bewegungsendekriterium bei Einzelachsen

Ähnlich dem Satzwechselkriterium bei Bahninterpolation (G601, G602, G603) kann auch das Bewegungsendekriterium bei Verfahrbewegungen von Einzelachsachsen in Teileprogrammen/Synchronaktionen programmiert werden:

Programmbefehl	Bewegungsendekriterium
FINEA[<Achse>]	"Genauhalt Fein"
COARSEA[<Achse>]	"Genauhalt Grob"
IPOENDA[<Achse>]	"Interpolator-Stopp" (IPO-Stopp)

Nach Programmende oder NC-Reset bleibt der zuletzt programmierte Wert erhalten.

Das wirksame Bewegungsendekriterium kann über die achsspezifische Systemvariable \$AA_MOTEND gelesen werden.

Hinweis

Abhängig davon, ob die Systemvariable \$AA_MOTEND im Teileprogramm oder Synchronaktion gelesen wird, enthält sie den Wert für NC-Achsen oder Hauptlauf-Achsen.

Beispiel:**Teileprogramm:**

```
N80 G01 POS[X]=100 FA[X]=1000 ACC[X]=90 COARSEA[X]
```


Synchronaktion:

```
N100 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140 IPOEN-
DA[X]
```

Literatur:

Weitere Informationen zum Satzwechsel und zu den Bewegungsendekriterien bei FINEA, COARSEA und IPOENDA siehe:

Funktionshandbuch Erweiterungsfunktionen; Positionierachsen (P2), Kapitel: Satzwechsel

Programmierbarer Servo-Parametersatz (SCPARA)

Im Teileprogramm/Synchronaktion kann mit SCPARA der Servo-Parametersatz vorgegeben werden.

Syntax

SCPARA [<Achse>] = <Parametersatz-Nummer>

Bedeutung

SCPARA:	Schlüsselwort zur Aktivierung des angegebenen Servo-Parametersatzes
<Achse>:	Name der Kanalachse
<Parametersatz-Nummer>:	Nummer des zu aktivierenden Servo-Parametersatzes

Hinweis

Die Aktivierung des mit SCPARA vorgegebenen Parametersatzes kann vom PLC-Anwenderprogramm aus unterbunden werden:

DB31, ... DBX9.3 = 1 (Parametersatzvorgabe durch SCPARA gesperrt)

Es wird dabei **keine** Meldung angezeigt.

Die Nummer des aktiven Parametersatzes kann über die Systemvariable \$AA_SCPAR gelesen werden.

Randbedingungen

Unterschiedliche Bewegungsendekriterien

Bei unterschiedlichen Bewegungsendekriterien werden Teileprogrammsätze unterschiedlich schnell beendet. Dies kann bei Technologiezyklen und PLC-Anwenderteilen zu Seiteneffekten führen.

Parametersatzwechsel

Falls der Servo-Parametersatz sowohl in einem Teileprogramm bzw. in einer Synchronaktion und der PLC gewechselt werden soll, muss das PLC-Anwenderprogramm erweitert werden.

Power On

Nach Power On sind folgende Grundstellungswerte eingestellt:

- Prozentuale Beschleunigungskorrektur für alle Einachsinterpolationen: 100%
- Bewegungsendekriterium für alle Einzelachsinterpolationen: FINEA
- Servo-Parametersatz: 1

Betriebsartenwechsel

Beim Betriebsartenwechsel von AUTOMATIK nach JOG behalten die programmierten Dynamikänderungen ihre Gültigkeit.

Reset

Mit Reset bleibt für die Teileprogrammvorgaben der letzte programmierte Wert bestehen. Die Vorgaben für die Hauptlauf-Interpolationen werden nicht geändert.

Satzsuchlauf

Jeweils das zuletzt programmierte Bewegungsendekriterium einer Achse wird aufgesammelt und in einem Aktionssatz ausgegeben. Der letzte im Satzsuchlauf behandelte Satz mit einem programmierten Bewegungsendekriterium dient als Behälter für alle programmierten Bewegungsendekriterien aller Achsen.

18.5 Randbedingungen

Maßeinheit

Die gültige Maßeinheit der Vorschübe ist abhängig vom eingestellten Maßsystem und der eingetragenen Achsart:

MD10240 \$MN_SCALING_SYSTEM_IS_METRIC (Grundsystem der Steuerung metrisch/ inch)

MD30300 \$MA_IS_ROT_AX (Rund- oder Linearachse)

Grundstellung für den Vorschubtyp

Die Grundstellung für den Vorschubtyp ist festgelegt im Maschinendatum:

MD20150 \$MC_GCODE_RESET_VALUES (Löschstellung der G-Gruppen)

Standardeinstellung ist G94.

Die Grundstellung des Vorschubtyps wird erst mit Start eines Teileprogramms angezeigt.

Wirksamkeit nach Reset

Ob die zuletzt programmierten F-, FA-, OVR-, OVRA- Werte auch nach Reset wirksam sind, ist abhängig von der Einstellung im Maschinendatum:

MD22410 \$MC_F_VALUES_ACTIVE_AFTER_RESET (F-Funktion über Reset hinaus wirksam)

Spindelpositionieren

Bei aktivem G95, G96, G961, G97, G971, G33, G34, G35 sollte kein Spindelpositionieren stattfinden, da nach erfolgter Spindelpositionierung der abgeleitete Bahnvorschub = 0 ist. Wenn dann die programmierte Achsposition noch nicht erreicht ist, kann der Satz nicht beendet werden.

18.6 Datenlisten

18.6.1 Maschinendaten

18.6.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
10651	IPO_PARAM_THREAD_NAME_TAB	Name der Interpolationsparameter für ballige Gewinde
10704	DRYRUN_MASK	Aktivierung des Probelaufvorschubs
10710	PROG_SD_RESET_SAVE_TAB	Zu aktualisierende Settingdaten
11410	SUPPRESS_ALARM_MASK	Maske zur Unterdrückung spezieller Alarmer
11550	STOP_MODE_MASK	Legt das Stopp-Verhalten fest
12000	OVR_AX_IS_GRAY_CODE	Achs-Vorschub-Override-Schalter Gray codiert
12010	OVR_FACTOR_AX_SPEED	Bewertung des Achs-Vorschub-Override-Schalters
12020	OVR_FEED_IS_GRAY_CODE	Bahnvorschub-Override-Schalter Gray-codiert
12030	OVR_FACTOR_FEEDRATE	Bewertung des Bahnvorschub-Override-Schalters
12040	OVR_RAPID_IS_GRAY_CODE	Eilgang-Override-Schalter Gray-codiert
12050	OVR_FACTOR_RAPID_TRA	Bewertung des Eilgang-Override-Schalters
12060	OVR_SPIND_IS_GRAY_CODE	Spindel-Override-Schalter Gray-codiert
12070	OVR_FACTOR_SPIND_SPEED	Bewertung des Spindel-Override-Schalters
12080	OVR_REFERENCE_IS_PROG_FEED	Override-Bezugsgeschwindigkeit
12082	OVR_REFERENCE_IS_MIN_FEED	Festlegung des Bezugs des Bahn-Override
12090	OVR_FUNCTION_MASK	Auswahl von Override-Spezifikationen
12100	OVR_FACTOR_LIMIT_BIN	Begrenzung bei binärcodiertem Override-Schalter
12200	RUN_OVERRIDE_0	Fahren bei Override 0
12202	PERMANENT_FEED	Festvorschübe für Linearachsen
12204	PERMANENT_ROT_AX_FEED	Festvorschübe für Rundachsen

18.6.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20100	DIAMETER_AX_DEF	Geometrieachsen mit Planachsfunktion
20150	GCODE_RESET_VALUES	Löschstellung der G-Gruppen
20172	COMPRESS_VELO_TOL	Max. erlaubte Abweichung der Bahnvorschubs bei Kompression
20200	CHFRND_MAXNUM_DUMMY_BLOCKS	Leersätze bei Phase/Radien
20201	CHFRND_MODE_MASK	Verhalten für Fase/Rundung
20750	ALLOW_GO_IN_G96	G0-Logik bei G96, G961
21200	LIFTFAST_DIST	Verfahrstrecke bei Schnellabheben von der Kontur
21220	MULTFEED_ASSIGN_FASTIN	Zuordnung der Eingangsbytes der NC-Peripherie für "Mehrere Vorschubwerte in einem Satz"
21230	MULTFEED_STORE_MASK	Speicherverhalten für die Funktion "Mehrere Vorschubwerte in einem Satz"
22240	AUXFU_F_SYNC_TYPE	Ausgabezeitpunkt der F-Funktionen
22410	F_VALUES_ACTIVE_AFTER_RESET	F-Funktion über Reset hinaus wirksam

18.6.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
30300	IS_ROT_AX	Rundachse / Spindel
32000	MAX_AX_VELO	Maximale Achsgeschwindigkeit
32060	POS_AX_VELO	Löschstellung für Positionierachsgeschwindigkeit
32300	MAX_AX_ACCEL	Achsbeschleunigung
32320	DYN_LIMIT_RESET_MASK	Resetverhalten von Dynamikbegrenzungen
34990	ENC_ACTIVATL_SMOOTH_TIME	Glättungszeitkonstante für Istwerte
35100	SPIND_VELO_LIMIT	Maximale Spindeldrehzahl
35130	GEAR_STEP_MAX_VELO_LIMIT	Maximaldrehzahl der Getriebestufe
35140	GEAR_STEP_MIN_VELO_LIMIT	Minimaldrehzahl der Getriebestufe
35160	SPIND_EXTERN_VELO_LIMIT	Spindeldrehzahlbegrenzung von PLC

18.6.2 Settingdaten

18.6.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42000	THREAD_START_ANGLE	Startwinkel bei Gewinde
42010	THREAD_RAMP_DISP	Beschleunigungsverhalten der Achse beim Gewinde-schneiden

Nummer	Bezeichner: \$SC_	Beschreibung
42100	DRY_RUN_FEED	Probelaufvorschub
42101	DRY_RUN_FEED_MODE	Mode für Probelaufvorschub
42110	DEFAULT_FEED	Defaultwert für Bahnvorschub
42600	JOG_FEED_PER_REV_SOURCE	Steuerung Umdrehungsvorschub im JOG
43300	ASSIGN_FEED_PER_RES_SOURCE	Umdrehungsvorschub für Positionsachsen/Spindel

18.6.2.2 Achs-/Spindel-spezifische Settingdaten

Nummer	Bezeichner: \$SA_	Beschreibung
43210	SPIND_MIN_VELO_G25	programmierte Spindeldrehzahlbegrenzung G25
43220	SPIND_MAX_VELO_G26	programmierte Spindeldrehzahlbegrenzung G26
43230	SPIND_MAX_VELO_LIMS	Spindeldrehzahlbegrenzung bei G96

18.6.3 Signale

18.6.3.1 Signale an Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Probelaufvorschub aktivieren	DB21,DBX0.6	DB320x.DBX0.6
Vorschub-Override	DB21,DBB4	DB320x.DBB4
Eilgangkorrektur	DB21,DBB5	DB320x.DBB5
Vorschubsperr	DB21,DBX6.0	DB320x.DBX6.0
Eilgangkorrektur wirksam	DB21,DBX6.6	DB320x.DBX6.6
Vorschub-Override wirksam	DB21,DBX6.7	DB320x.DBX6.7
Vorschub Halt, Geometrieachse 1	DB21,DBX12.3	DB320x.DBX1000.3
Vorschub Halt, Geometrieachse 2	DB21,DBX16.3	DB320x.DBX1004.3
Vorschub Halt, Geometrieachse 3	DB21,DBX20.3	DB320x.DBX1008.3

18.6.3.2 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Probelaufvorschub angewählt	DB21,DBX24.6	DB170x.DBX0.6
Vorschub-Override für Eilgang angewählt	DB21,DBX25.3	DB170x.DBX1.3
Festvorschub 1 aktivieren für Bahn-/Geometrieachsen	DB21,DBX29.0	DB320x.DBX13.0
Festvorschub 2 aktivieren für Bahn-/Geometrieachsen	DB21,DBX29.1	DB320x.DBX13.1
Festvorschub 3 aktivieren für Bahn-/Geometrieachsen	DB21,DBX29.2	DB320x.DBX13.2

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Festvorschub 4 aktivieren für Bahn-/Geometrieachsen	DB21,DBX29.3	DB320x.DBX13.3
Probelauf Vorschub aktiv	DB21,DBX318.6	DB330x.DBX4002.6

18.6.3.3 Signale an Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Vorschub-/Spindel-Override	DB31,DBB0	DB380x.DBB0
Korrektur wirksam	DB31,DBX1.7	DB380x.DBX1.7
Festvorschub 1 aktivieren für Maschinenachse	DB31,DBX3.2	DB380x.DBX3.2
Festvorschub 2 aktivieren für Maschinenachse	DB31,DBX3.3	DB380x.DBX3.3
Festvorschub 3 aktivieren für Maschinenachse	DB31,DBX3.4	DB380x.DBX3.4
Festvorschub 4 aktivieren für Maschinenachse	DB31,DBX3.5	DB380x.DBX3.5
Vorschub Halt / Spindel Halt	DB31,DBX4.3	DB380x.DBX4.3

18.6.3.4 Signale von Achse/Spindel

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Umdrehungsvorschub aktiv	DB31,DBX62.2	DB390x.DBX2.2
F-Funktion für Positionierachse	DB31,DBB81	-
Programmierte Drehzahl zu hoch	DB31,DBX83.1	DB390x.DBX2001.1

W1: Werkzeugkorrektur

19.1 Kurzbeschreibung

Verrechnung der Werkzeugkorrekturdaten

Die Steuerung SINUMERIK 840D sl ermöglicht eine Verrechnung folgender Werkzeugkorrekturdaten:

- Längenkorrektur
- Radiuskorrektur
- Ablage der Werkzeugdaten in flexiblem Werkzeugkorrekturspeicher:
 - Werkzeug-Kennzeichnung durch T-Nummern von 0 bis 32000
 - Definition eines Werkzeugs durch maximal 9 Schneiden
 - Schneide wird durch max. 25 Werkzeugparameter beschrieben
- Werkzeug-Anwahl wählbar: Sofort oder über wählbare M-Funktion
- Werkzeug-Radiuskorrektur:
 - An- und Abwahlstrategie wählbar: Normal oder konturbezogen
 - Korrektur wirkt für alle Interpolationsarten:
 - Linear
 - Kreis
 - Helikal
 - Spline
 - Polynom
 - Evolvente
 - Korrektur an Außenecken wählbar:
 - Übergangskreis / Ellipse (G450) oder Schnittpunkt der Äquidistanten (G451)
 - Parametergesteuerte Anpassung der Funktionen G450/G451 an die Kontur
 - Freifahren an Außenecken bei G450 mit Parameter DISC
 - Anzahl der Zwischensätze ohne Achsbewegung in der Korrekturebene wählbar
 - Kollisionsüberwachung wählbar:
 - Mögliche Konturverletzungen werden vorausschauend erkannt, wenn:
 - Bahnweg kürzer ist als Werkzeugradius
 - Breite einer Innenecke kleiner ist als Werkzeug-Durchmesser
 - Werkzeugradiuskorrektur konstant halten
 - Schnittpunktverfahren für Polynome

Orientierbare Werkzeugträger

Die Funktion ermöglicht die Bearbeitung schräger Flächen mit Berücksichtigung von Werkzeuglängenkorrektur, wenn die Kinematik des Werkzeugträgers statisch (ohne NC-Achsen) die Orientierung des Werkzeuges ermöglicht. Die aufwendigere 5-Achstransformation ist für diesen Fall nicht erforderlich.

Literatur:

Funktionshandbuch Sonderfunktionen; Mehrachstransformationen (F2)

Durch geeignete Wahl von Werkzeugdaten und Werkzeugträgerdaten wird für die Steuerung die Kinematik so beschrieben, dass diese die Werkzeuglängenkorrektur berücksichtigen kann. Einige der Beschreibungsdaten können vom aktuellen Frame direkt durch die Steuerung übernommen werden.

Hinweis

Weitere Erläuterungen zu Werkzeugen und Werkzeugkorrekturen sowie die vollständigen programmtechnischen Möglichkeiten der Werkzeugkorrektur (WLK und WRK) mit ihren Sonderfällen sind zu entnehmen:

Literatur:

Programmierhandbuch Grundlagen

Flache/eindeutige D-Nummern-Struktur

Korrekturanwahl über eindeutige D-Nummern mit Verwaltungsfunktion verfügbar.

Werkzeugkorrekt.-Sonderbehandlung

Die Bewertung der Vorzeichen lässt sich für Werkzeuglängen und Verschleiß steuern mit den Settingdaten:

SD42900 \$SC_MIRROR_TOOL_LENGTH (Vorzeichenwechsel Werkzeuglängen beim Spiegeln)

SD42960 \$SC_TOOL_TEMP_COMP (Temperaturkompens. bezogen auf Werkzeug)

Das gilt ebenfalls für das Verhalten der Verschleißkomponenten beim Spiegeln von Geometrieachsen oder beim Wechsel der Bearbeitungsebene über Settingdaten.

Literatur:

Programmierhandbuch Grundlagen; Werkzeugkorrekturen

G461/G462

Um in bestimmten Fällen beim Aktivieren und Deaktivieren der WRK auch das Ausräumen von Innenecken zu ermöglichen, werden die Befehle G461 und G462 eingeführt und damit die An-/Abfahrstrategie bei der Werkzeugradiuskorrektur erweitert.

- G461
Wenn kein Schnittpunkt des letzten WRK-Satzes mit einem Vorgängersatz möglich ist, berechnet die Steuerung einen Schnittpunkt, indem die Offsetkurve dieses Satzes mit einem Kreis verlängert wird, dessen Mittelpunkt im Endpunkt des nicht korrigierten Satzes liegt, und dessen Radius gleich dem Werkzeugradius ist.
- G462
Wenn kein Schnittpunkt des letzten WRK-Satzes mit einem Vorgängersatz möglich ist, berechnet die Steuerung einen Schnittpunkt, indem im Endpunkt des letzten Satzes mit Werkzeugradiuskorrektur eine Gerade eingefügt wird (der Satz wird durch seine Endtangente verlängert).

Wechsel von G40 nach G41/42

Der Wechsel von G40 nach G41/G42 und umgekehrt wird auch bei Werkzeugen mit relevanter Schneidenlage (Dreh- und Schleifwerkzeuge) nicht mehr als Werkzeugwechsel behandelt.

Werkzeugkorrektur-Umgebungen

Mit SW-Stand 7.1 werden Funktionen bereitgestellt, mit denen bezüglich den aktuellen Zuständen von Werkzeugdaten folgende Aktionen möglich sind:

- Abspeichern
- Löschen
- Lesen
- Modifizieren

Einige der Funktionen waren bisher innerhalb von Messzyklen realisiert. Sie stehen jetzt allgemein zur Verfügung.

Mit einer weiteren Funktion kann Information über die Zuordnung der Werkzeuglängen des aktiven Werkzeuges zu Abszisse, Ordinate und Applikate ermittelt werden.

19.2 Werkzeug

19.2.1 Allgemeine Informationen

WZ anwählen

Ein Werkzeug wird im Programm mit der T-Funktion angewählt.

Ob mit der T-Funktion sofort das neue Werkzeug eingewechselt wird, hängt ab von der Einstellung im Maschinendatum:

MD22550 \$MC_TOOL_CHANGE_MODE (neue Werkzeugkorrektur bei M-Funktion)

WZ-Wechsel sofort

MD22550 \$MC_TOOL_CHANGE_MODE = 0 (neue Werkzeugkorrektur bei M-Funktion)

Das neue Werkzeug wird mit der T-Funktion sofort eingewechselt.

Bei Drehmaschinen mit Werkzeugrevolver wird hauptsächlich diese Einstellung verwendet.

WZ-Wechsel mit M06

MD22550 \$MC_TOOL_CHANGE_MODE = 1 (neue Werkzeugkorrektur bei M-Funktion)

Das neue Werkzeug wird mit der T-Funktion zum Wechsel vorbereitet.

Bei Fräsmaschinen mit Werkzeugmagazin wird hauptsächlich diese Einstellung verwendet, um das neue Werkzeug hauptzeitparallel (die Bearbeitung wird nicht unterbrochen) auf die Werkzeugwechselposition zu bringen.

Das alte WZ wird aus der Spindel entfernt und das neue WZ in die Spindel eingewechselt mit der eingegebenen M-Funktion im Maschinendatum:

- MD22560 \$MC_TOOL_CHANGE_M_CODE (M-Funktion für Werkzeugwechsel)

Nach DIN 66025 soll dieser Werkzeugwechsel mit der M-Funktion M06 programmiert werden.

Das nächste Werkzeug wird vorgewählt mit dem Maschinendatum:

- MD20121 \$MC_TOOL_PRESEL_RESET_VALUE (Vorgewähltes Werkzeug bei RESET)

Dessen Werkzeuglängenkorrekturwerte sollen berücksichtigt werden bei RESET und Hochlauf entsprechend Maschinendatum:

- MD20110 \$MC_RESET_MODE_MASK (Feststellung der Steuerungs-Grundstellung nach RESET/TP-Ende)

Wertebereich der T-Funktion bzw. Werkzeugnummer

Die T-Funktion bzw. die Werkzeugnummer kann folgende ganzzahlige Werte annehmen:

- Minimalwert: T0 (kein Werkzeug)
- Maximalwert: T32000 (Werkzeug mit der Nummer 32000)

Werkzeugschneide D

Einem Werkzeug Tx können mehrere WZ-Schneiden D1 ... Dn zugeordnet werden.

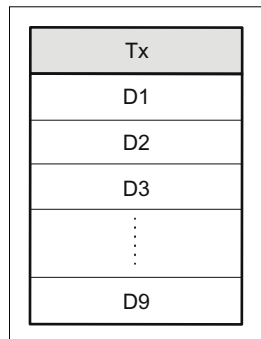


Bild 19-1 Beispiel für ein Werkzeug Tx mit 9 Schneiden (D1 bis D9)

Maximale Anzahl WZ-Schneiden pro Werkzeug

Die maximale Anzahl von WZ-Schneiden (D-Nummern) die pro WZ definiert werden können, wird über folgendes Maschinendatum festgelegt:

- MD18106 \$MN_MM_MAX_CUTTING_EDGE_PERTOOL

D-Funktion

Die Anwahl der zur Bearbeitung gewünschten Werkzeugschneide 1 ... n erfolgt durch Programmierung der entsprechenden D-Funktion z.B. im Teileprogramm mit D1 ... Dn. Die angewählte Werkzeugschneide bezieht sich immer auf das aktive Werkzeug. Eine Werkzeugschneide ohne aktives Werkzeug (aktuelle T-Nummer ist T0 ⇒ kein Werkzeug angewählt), ist unwirksam.

Mit D0 werden alle Werkzeugkorrekturen des aktiven Werkzeugs Tx abgewählt.

Anwahl der Schneide bei WZ-Wechsel

Nach der Anwahl eines neuen Werkzeugs mittels neuer T-Nummer und dem Einwechseln dieses WZ gibt es folgende Möglichkeiten zur Anwahl der Schneide:

- Die Schneidennummer Dx wird programmiert.
- Die Schneidennummer wird vorgegeben durch das Maschinendatum:
MD20270 \$MC_CUTTING_EDGE_DEFAULT = <Wert> (Grundstellung der Werkzeugschneide ohne Programmierung)

Wert	Bedeutung
= 0	Nach M06 erfolgt keine automatische Schneidenanwahl.
≠ 0	Nummer der Schneide, die nach M06 angewählt ist.
= -1	Die Schneiden-Nr. des alten Werkzeugs bleibt erhalten und wird nach M06 auch für das neue Werkzeug angewählt.

Aktivieren der WZK

Mit D_1 bis D_n wird die Werkzeugkorrektur einer WZ-Schneide für das aktive Werkzeug aktiviert. Die Werkzeuglängenkorrektur und die Werkzeugradiuskorrektur werden jedoch zu unterschiedlichen Zeitpunkten wirksam:

- Die **Werkzeuglängenkorrektur** (WLK) wird mit der ersten Verfahrbewegung der Achse, in der die WLK wirken soll, herausgefahren.
Diese Verfahrbewegung muss eine Linearinterpolation (G_0 , G_1 , POS , $POSA$) oder Polynominterpolation ($POLY$) sein. Ist die $POS/POSA$ -Achse eine der aktiven Geometrieachsen, wird die Werkzeuglängenkorrektur mit der ersten Verfahrbewegung der Achse, in der die WLK wirken soll, herausgefahren.
- Die **Werkzeugradiuskorrektur** (WRK) wird durch Programmierung von $G41/G42$ in der aktiven Ebene ($G17$, $G18$ oder $G19$) wirksam.
Im NC-Satz in dem die Werkzeugradiuskorrektur mit $G41/G42$ aktiviert wird, muss G_0 (Eilgang) oder G_1 (Linearinterpolation) aktiv sein und mindestens eine Geometrieachse der aktiven Arbeitsebene programmiert werden.
Wird beim Einschalten nur eine Geometrieachse angegeben, wird die letzte Position der zweiten Geometrieachse automatisch ergänzt und anschließend in beiden Achsen verfahren.
Sind die Achsen aktuell keine Geometrieachsen, müssen sie durch $GEOAX$ -Programmierung zu Geometrieachsen gemacht werden.
Die Anwahl der WRK mit $G41/G42$ darf nur in einem Programmsatz mit G_0 (Eilgang) oder G_1 (Linearinterpolation) erfolgen.

19.2.2 Korrekturspeicherstruktur

WZK-Speichergröße

Jeder Kanal kann einen eigenen WZK-Speicher (TO-Einheit) haben.

Welcher WZK-Speicher für den entsprechenden Kanal existiert, wird eingestellt mit dem Maschinendatum:

MD28085 \$MC_MM_LINK_TOA_UNIT (Zuordnung der TO-Einheit zu einem Kanal)

Die maximale Anzahl von WZ-Schneiden für alle von der NC verwalteten Werkzeuge wird eingestellt mit dem Maschinendatum:

MD18100 \$MN_MM_NUM_CUTTING_EDGES_IN_TOA (Anzahl der Werkzeugschneiden in der NC)

Werkzeuge

Der WZK-Speicher besteht aus Werkzeugen mit den Nummern T_1 bis T_{32000} .

Jedes Werkzeug kann über TOA-Dateien oder einzeln mit dem Softkey "Neues Werkzeug" eingerichtet werden. Nicht benötigte Korrekturen sind mit dem Wert Null zu belegen. Dies ist auch die Voreinstellung beim Anlegen der Korrekturspeicher. Die einzelnen Werte der

Korrekturspeicher (WZ-Parameter) sind über Systemvariablen vom Programm les- und schreibbar.

Hinweis

Die Werkzeuge (T1 bis T32000) müssen nicht in aufsteigender Reihenfolge, nicht lückenlos und nicht mit T1 beginnend in den WZK-Speicher eingebracht werden.

WZ-Schneiden

Jedes Werkzeug kann bis zu 9 WZ-Schneiden (D1 bis D9) besitzen. Wird ein neues Werkzeug in den WZK-Speicher eingebracht, wird automatisch die erste WZ-Schneide (D1) eingerichtet. Weitere WZ-Schneiden (max. 8) werden mit dem Softkey "Neue Schneide" in aufsteigender Reihenfolge lückenlos eingerichtet. Auf diese Weise kann jedem Werkzeug eine unterschiedliche Anzahl von WZ-Schneiden zugeordnet werden.

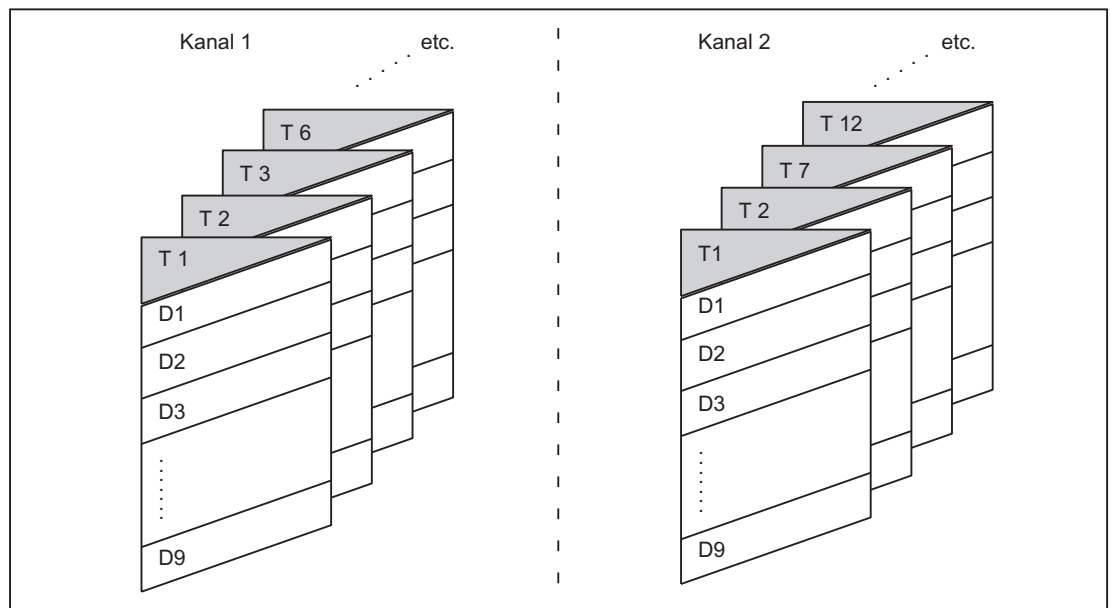


Bild 19-2 Beispiel einer WZK-Speicherstruktur für 2 Kanäle

19.2.3 Verrechnung der WZ-Korrektur

D-Nr.

Die **D-Nr.** reicht zur Ermittlung der WZ-Korrekturen aus (über MD einstellbar).

D-Nr.	DP1	DP2	DP3	DP25
D1						
D2						
:						
Dn						

Korrekturwert

Programmierung

Obiger Korrekturblock soll in der NC verrechnet werden.

Aufruf im Teileprogramm:

```

...
Dn

```

19.2.4 Adresserweiterung für die NC-Adressen T und M

MD20096

Ob auch bei **nicht** aktivierter Werkzeugverwaltung die Adresserweiterung von T und M als Spindelnummer interpretiert werden soll, kann festgelegt werden über das Maschinendatum:

MD20096 \$MC_T_M_ADDRESS_EXT_IS_SPINO (Spindelnummer als Adresserweiterung)

Für den Bezug der D-Nummer zur T-Nummer gelten dann dieselben Regeln wie bei aktiver Funktion "Werkzeugverwaltung".

Auswirkung auf die D-Nummer

Über die D-Nummer wird ein Korrekturdatensatz bestimmt.

Die D-Adresse kann nicht mit Adresserweiterung programmiert werden.

Die Auswertung der D-Adresse bezieht sich immer auf das momentan aktive Werkzeug.

Die programmierte D-Adresse bezieht sich auf das bzgl. der Masterspindel aktive Werkzeug (analog zur Funktion Werkzeugverwaltung) bei gesetztem Maschinendatum:

MD20096 \$MC_T_M_ADDRESS_EXT_IS_SPINO = TRUE (Spindelnummer als Adresserweiterung)

Auswirkung auf die T-Nummer

Bei aktiver Funktion "Werkzeugverwaltung" werden die bzgl. der Masterspindel (bzw. Master-WZ-Halter) programmierten Werte als programmierte/aktive T-Nummern angezeigt.

Ohne aktive Werkzeugverwaltung werden **alle** programmierten T-Werte als programmiert/aktiv angezeigt (unabhängig davon, welche Adresserweiterung dazu programmiert wurde).

Nur noch der bzgl. der Masterspindel programmierte T-Wert wird als programmiert/aktiv angezeigt, wenn:

MD20096 \$MC_T_M_ADDRESS_EXT_IS_SPINO = TRUE (Spindelnummer als Adresserweiterung)

Beispiel

Im folgenden Beispiel wird die Auswirkung von MD20096 gezeigt.

Es werden zwei Spindeln betrachtet. Die Spindel 1 ist Masterspindel. M6 wurde als WZ-Wechselsignal festgelegt.

T1 = 5	
M1= 6	
T2 = 50	
M2 = 6	
D4	

- Bei aktiver Werkzeugverwaltung bezieht sich D4 auf das Werkzeug "5". T2=50 bestimmt das Werkzeug für die Nebenspindel, deren Werkzeug nicht die Korrektur der Bahn beeinflusst. Die Bahn wird ausschließlich durch das für die Masterspindel programmierte Werkzeug bestimmt.
- D4 bezieht sich auf das Werkzeug "50" ohne aktive Werkzeugverwaltung und mit dem Maschinendatum:
MD20096 \$MC_T_M_ADDRESS_EXT_IS_SPINO = **FALSE** (Bedeutung der Adresserweiterung bei T, M Werkzeugwechsel)
Weder die Adresserweiterung von T noch von M werden im NC ausgewertet.
Jeder WZ-Wechselbefehl bestimmt eine neue Korrektur der Bahn.
- D4 bezieht sich auf das Werkzeug "5" (wie bei aktiver WZV) ohne aktive Werkzeugverwaltung und mit dem Maschinendatum:
MD20096 \$MC_T_M_ADDRESS_EXT_IS_SPINO = **TRUE**.
Die Adresserweiterung 1 (T1=..., M1=...) bezeichnet die Masterspindel.

Hinweis

Bisher führt bei nicht aktivierter Werkzeugverwaltung jede Programmierung eines Werkzeugwechsels (mit T bzw. M) zu einer Neueinrechnung der WZ-Korrektur in die Bahn. Die Adresserweiterung wird dabei nicht näher definiert. Die Bedeutung der Erweiterung wird vom Anwender festgelegt (über PLC-Anwenderprogramm).

19.2.5 Freie D-Nummernvergabe

"Relative" D-Nummern

IN der NC besteht die Möglichkeit, die D-Nummern für die WZ-Korrekturdatensätze als "relative" D-Nummern zu führen. Dabei werden jeder T-Nummer die entsprechenden D-Nummern zugeordnet.

Funktionen

Erweiterungen der Funktionen bei der Vergabe der D-Nummern:

- Die maximal erlaubte D-Nummer wird festgelegt über das Maschinendatum:
MD18105 \$MN_MM_MAX_CUTTING_EDGE_NO (Max. Wert der D-Nummern (DRAM))
Es ist der Wert 9 voreingestellt, so dass bestehende Anwendungen weiterhin lauffähig bleiben.
- Die Anzahl der Schneiden (bzw. Korrekturdatensätze) **je Werkzeug** kann festgelegt werden über das Maschinendatum:
MD18106 \$MN_MM_MAX_CUTTING_EDGE_PERTOOL (Max. Anzahl der D-Nummern pro Werkzeug (DRAM))
Damit kann die Anzahl der zu parametrierenden Schneiden pro Werkzeug an die Anzahl der real möglichen Schneiden angepasst und so überwacht werden.
- Es gibt außerdem die Möglichkeit, D-Nummern im NC umzubenennen und damit beliebige D-Nummern für die Schneiden zu vergeben.

Hinweis

Neben der relativen D-Nummernvergabe können die D-Nummern auch als "flache" bzw. "absolute" D-Nummern (1-32000) ohne Bezug zu einer T-Nummer vergeben werden (innerhalb der Funktion "Flache D-Nummernstruktur").

Schneidennummer CE

Beim Umbenennen der D-Nummern geht die Information über die im Werkzeugkatalog für diese Schneiden festgelegte Nummer verloren. Es kann somit nach dem Umbenennen nicht mehr erkannt werden, um welche Schneide des Kataloges es sich handelt.

Da diese Information bei Umrüstvorgängen benötigt wird, wurde für jede Schneide eine **Schneidennummer CE** eingeführt, die beim Umbenennen der D-Nummer erhalten bleibt.

Die D-Nummer bezeichnet im Teileprogramm die Schneidenkorrektur. Diese **Korrekturnummer D** wird getrennt von der **Schneidennummer CE** (die Nummer im WZ-Katalog) verwaltet und ist eine beliebige Zahl, mit der eine Korrektur im Teileprogramm und auf der Anzeige benannt wird.

Die CE-Nummer bezeichnet beim Umrüsten die konkrete physikalische Schneide. Die Schneidennummer CE wird vom NC bei der Korrekturanwahl beim Werkzeugwechsel nicht ausgewertet (nur über BTSS verfügbar).

Die Schneidennummer CE wird über die Systemvariable **\$TC_DPCE[t,d]** beschrieben:

- **t** steht für die interne T-Nummer
- **d** steht für die D-Nummer.

Schreibvorgänge werden auf Kollision überwacht; d.h. die Schneidennummern eines Werkzeugs müssen alle verschieden sein. Die Variable **\$TC_DPCE** ist Bestandteil des Schneidenparameterdatensatzes **\$TC_DP1,..., \$TC_DP25**.

Die Parametrierung von **\$TC_DPCE** ist nur sinnvoll, wenn die maximale Schneidennummer (MD18105) größer ist als die maximale Anzahl der Schneiden je Werkzeug (MD18106).

Dann ist die Vorbelegung der Schneidenummer gleich der Ordnungsnummer der Schneide. Die angelegten Korrekturen eines Werkzeugs werden mit 1 beginnend bis zur maximalen Anzahl der Schneiden je Werkzeug hochgezählt (MD18106).

Die Schneidenummer CE ist gleich der D-Nummer (in Kompatibilität zum bisherigen Verhalten) falls:

$MD18105 \leq MD18106$.

Eine Leseoperation liefert $CE=D$. Eine Schreiboperation wird ohne Alarmmeldung ignoriert.

Hinweis

Die Korrekturwerte \$TC_DP1,..., \$TC_DP25 der aktiven WZ-Korrektur können mit der Systemvariablen \$P_AD[n]; mit $n=1,...,25$ gelesen werden. Die Schneidenummer CE der aktiven Korrektur erhält man für $n=26$.

Befehle

Ist die maximale Schneidenummer **größer** als die maximale Anzahl der Schneiden pro Werkzeug, stehen die in der folgenden Tabelle aufgeführten Befehle zur Verfügung

- $(MD18105 \$MN_MM_MAX_CUTTING_EDGE_NO) > (MD18106 \$MN_MM_MAX_CUTTING_EDGE_PERTOOL)$

Befehl	Bedeutung
CHKDNO	Prüft die vorhandenen D-Nummern auf Eindeutigkeit; Die D-Nummern aller innerhalb einer TO-Einheit definierten Werkzeuge dürfen nur einmal auftreten. Ersatzwerkzeuge werden dabei nicht berücksichtigt.
GETDNO	Ermittelt zu einer Schneide eines Werkzeuges die D-Nummer. Existiert keine D-Nummer zu den eingegebenen Parametern, wird d=0 gesetzt. Ist die D-Nummer ungültig wird ein Wert größer 32000 zurückgegeben.
SETDNO	Setzt oder ändert die D-Nummer der Schneide CE eines Werkzeuges T. Existiert kein Datensatz zu den eingegebenen Parametern wird FALSE zurückgegeben. Syntaxfehler erzeugen einen Alarm. Die D-Nummer kann nicht explizit auf 0 gesetzt werden.
GETACTTD	Ermittelt zu einer absoluten D-Nummer die dazugehörige T-Nummer. Es erfolgt keine Prüfung auf Eindeutigkeit. Gibt es mehrere gleiche D-Nummern innerhalb einer TO-Einheit, wird die T-Nummer des ersten gefundenen Werkzeuges zurückgegeben. Bei Verwendung "flacher" D-Nummern ist die Verwendung des Befehls nicht sinnvoll, da hier immer der Wert 1 zurückgegeben wird (keine T-Nummer in der Dateneintragung).
DZERO	Kennzeichnet alle D-Nummern der TO-Einheit als ungültig; Der Befehl dient zur Unterstützung während dem Umrüsten. So gekennzeichnete Korrekturdatensätze werden nicht mehr vom Sprachbefehl CHKDNO geprüft. Um sie wieder zugänglich zu machen, muss die D-Nummer wieder mit SETDNO gesetzt werden.

Hinweis

Ist die maximale Schneidenummer **kleiner** als die maximale Anzahl der Schneiden pro Werkzeug, sind die in der Tabelle aufgeführten Befehle ohne Wirkung. Aus Kompatibilitätsgründen, ist dies die Voreinstellung im System.

Die einzelnen Befehle sind ausführlich beschrieben in:

Literatur:

Programmierhandbuch Grundlagen

Aktivierung

Um mit eindeutigen D-Nummern und damit mit den beschriebenen Sprachbefehlen arbeiten zu können, müssen die D-Nummern für die Werkzeuge frei benannt werden können.

Dazu müssen folgende Voraussetzungen erfüllt sein:

- MD18105 > MD18106
- Die Funktion "Flache D-Nummer" ist nicht aktiviert.
MD18102 \$MN_MM_TYPE_OF_CUTTING_EDGE (Art der D-Nummer Programmierung (SRAM))

Beispiele

MD18105 \$MN_MM_MAX_CUTTING_EDGE_NO = 1 (Max. Wert der D-Nummern)

Damit kann pro Werkzeug maximal eine Korrektur definiert werden (mit D-Nummer = 1).

Hinweis

Bei aktivierter Funktion "Flache D-Nummern" kann damit in der TO-Einheit nur genau eine D-Korrektur definiert werden.

MD18105 \$MN_MM_MAX_CUTTING_EDGE_NO = 9999

Damit können Werkzeuge mit eindeutigen D-Nummern versehen werden.

Beispiel: Zuordnung von D- (Schneiden) zu T-Nummern (Werkzeugen) und Aufzeigen von Prüfungsmöglichkeiten auf Eindeutigkeit:

- Zuordnung:
 - T-Nr. 1 ⇒ Zuordnung der D-Nummern 1, 2, 3
 - T-Nr. 2 ⇒ Zuordnung der D-Nummern 10, 20, 30, 40, 50
 - T-Nr. 3 ⇒ Zuordnung der D-Nummern 100, 200, 30 (**Fehler:** Statt 300 wurde 30 angegeben)
- Prüfung auf Eindeutigkeit:
 Die D-Nummern aller innerhalb einer TO-Einheit definierten Werkzeuge müssen eindeutig sein.
 Die Prüfung erfolgt mit dem Befehl `CHKDNO`. Ohne Angabe von Parametern werden alle D-Nummern aller Werkzeuge gegeneinander geprüft. Im vorliegenden Fall ist der Rückgabewert `== FALSE`, da D30 sowohl in Werkzeug T2 und T3 vorhanden ist.

Befehl	Prüfung	Prüfergebnis
<code>CHKDNO</code>	alle D-Nummern aller Werkzeuge gegen alle anderen Werkzeuge	Rückgabewert <code>== FALSE</code> , da D=30 doppelt vorhanden ist.
<code>CHKDNO (2, 3, 30)</code>	D-Nummer 30 in Werkzeug 2 und 3	Rückgabewert <code>== FALSE</code> , da D=30 doppelt vorhanden ist.
<code>CHKDNO (1, 3)</code>	alle D-Nummern der Werkzeuge 1 und 3	Rückgabewert <code>== TRUE</code> , obwohl D=30 vom dritten Werkzeug mit D=30 vom zweiten Werkzeug kollidiert.

Hinweis

Schneiden pro Werkzeug

Werden Werkzeuge mit jeweils **n** Schneiden eingesetzt, sollte der Wert des Maschinendatums auf **n** gesetzt werden. Dies verhindert die Definition unerlaubten Schneiden für ein Werkzeug.

MD18106 \$MN_MM_MAX_CUTTING_EDGE_PERTOOL = **n**

Programmierbeispiele

Umbenennen einer D-Nummer

Es soll die D-Nummer der Schneide mit CE = 3 von 2 auf den Wert 17 umbenannt werden. Dazu sollen folgende Vorgaben gelten:

- `$TC_DPx[<Werkzeug Tn>, <Schneide Dm>]`
- Interne T-Nummer `Tn = 1`

- D-Nummer Dm = 2
- Werkzeug mit einer Schneide mit:

Programmcode	Kommentar
\$TC_DP2[1, 2] = 120	Werkzeuglänge = 120
\$TC_DP3[1, 2] = 5.5	Werkzeugradius = 5.5
\$TC_DPCE[1, 2] = 3	Schneidenummer CE = 3

- MD18105 \$MN_MM_MAX_CUTTING_EDGE_NO = 20 (Max. Wert der D-Nummern)

Innerhalb des Teileprogramms wird diese Korrektur standardmäßig mit T1, . . . D2 programmiert.

Sie weisen die momentane D-Nummer der Schneide 3 einer Variablen (DNrAlt) zu und definieren für die neue D-Nummer die Variable DNrNeu:

Programmcode
def int DNrAlt, DNrNeu = 17
DNrAlt = GETDNO(1, 3)
SETDNO(1, 3, DNrNeu)

Damit wird der Schneide CE=3 der neue D-Wert 17 zugewiesen.

Jetzt werden die Daten dieser Schneide über die D-Nummer 17 angesprochen; sowohl über die Systemvariable, als auch in der Programmierung mit der NC-Adresse D.

Innerhalb des Teileprogramms wird diese Korrektur nun standardmäßig mit T1, . . . D17 programmiert und die Daten werden folgendermaßen angesprochen:

Programmcode	Kommentar
\$TC_DP2[1, 17] = 120	
\$TC_DP3[1, 17] = 5.5	
\$TC_DPCE[1, 17] = 3	; Schneidenummer CE

Hinweis

Sofern das Werkzeug noch eine weitere Schneide definiert hat, z.B.: \$TC_DPCE[1, 2] = 1 ; = CE, kann die D-Nummer 2 der Schneide 1 nicht gleich benannt werden wie die D-Nummer der Schneide 3; d.h.: SETDNO(1, 1, 17) liefert den Status = FALSE als Rückgabewert.

DZERO - D-Nummern ungültig setzen

Mit Aktivieren dieses Befehls werden alle D-Nummern der Werkzeuge der TO-Einheit ungültig. Es kann keine Korrektur mehr aktiviert werden, bis wieder gültige D-Nummern im NC verfügbar sind. Die D-Nummern werden über den Befehl SETDNO neu vergeben.

Es sind folgende Werkzeuge vorgegeben (alle mit der Schneidenummer1):

T1, D1	D-Nr. der Schneide CE=1
T2, D10	D-Nr. der Schneide CE=1
T3, D100	D-Nr. der Schneide CE=1

Nun wird programmiert:

Programmcode

DZERO

Wird jetzt eine der Korrekturen aktiviert (z.B. mit T3 D100), wird ein Alarm erzeugt, da D100 momentan nicht definiert ist.

Die D-Nummern werden neu definiert über:

Programmcode	Kommentar
SETDNO(1, 1, 100)	; T=1, Schneide 1 erhält die (neue) D-Nummer 100
SETDNO(2, 1, 10)	; T=2, Schneide 1 erhält die (alte) D-Nummer 10
SETDNO(3, 1, 1)	; T=3, Schneide 1 erhält die (neue) D-Nummer 1

Hinweis

Der Befehl DZERO kann bei Spannungsausfall den NC bzgl. der D-Nummern in einem undefinierten Zustand hinterlassen. Um dies zu beheben, wiederholen Sie nach dem Wiedereinschalten den Befehl DZERO.

Prinzipieller Ablauf eines Umrüstprogramms

Es soll sichergestellt werden, dass die benötigten Werkzeuge und Schneiden vorhanden sind. Das Magazin, der Werkzeugspeicher der NC, befindet sich in einem beliebigen Zustand. Die D-Nummern in den Teileprogrammen der neuen Bearbeitung stimmen meist nicht mit den D-Nummern der konkreten Schneiden überein. Das Umrüstprogramm kann wie folgt aussehen:

Programmcode	Kommentar
DZERO	; alle D-Nummern der TO-Einheit werden als ungültig gekennzeichnet
....	; Schleife oder mehrere Schleifen über die Plätze des Magazins bzw. der Magazine mit Prüfung der Werkzeuge und ihrer Schneidennummern.
	; Wird ein WZ gefunden, das noch nicht gesperrt ist (\$TC_TP8) und das die gewünschte Schneidennummer CE hat (GETDNO), dann wird die Schneide mit der neuen D-Nummer versehen (SETDNO).
....	; Entlade- und Beladevorgänge werden durchgeführt.
	; Es kann mit dem Status 'zu entladen' bzw. 'zu beladen' des Werkzeuges gearbeitet werden.
CHKDNO	; Das Ent-/ Beladen und der Vorgang zum Umbenennen der D-Nummern sind abgeschlossen.
	; Es können einzelne Werkzeuge und / oder D-Nummern geprüft und anhand des Rückgabewertes auftretende Kollisionen automatisch behandelt werden.

19.2.6 Korrektursatz bei Fehler im Werkzeugwechsel

MD22550

Wenn im Teileprogramm eine Werkzeugvorbereitung programmiert ist und der NC stellt dabei einen Fehler fest (z.B. der Datensatz zur programmierten T-Nummer ist im NC nicht vorhanden), hat der Bediener die Möglichkeit, die Fehlersituation zu beurteilen und entsprechende Handlungen vorzunehmen, um anschließend die Bearbeitung fortzusetzen.

Der Werkzeugwechsel kann unabhängig programmiert werden, in Abhängigkeit des Maschinendatums:

MD22550 \$MC_TOOL_CHANGE_MODE (Neue Werkzeugkorrektur bei M-Funktion)

MD22550 \$MC_TOOL_CHANGE_MODE = 0

```
T= 'T-Nr.'      ; WZ-Vorbereitung + Werkzeugwechsel in einem NC-Satz,
                ; d.h. mit Programmierung von T erfolgt im NC das
                ; Aktivwerden einer neuen D-Korrektur. Siehe dazu das Maschinenda-
                ; tum:
                ; MD20270 $MC_CUTTING_EDGE_DEFAULT (Grundstellung der Werkzeug-
                ; schneide ohne Programmierung)
```

MD22550 \$MC_TOOL_CHANGE_MODE = 1

```
T= 'T-Nr.'      ; Werkzeugvorbereitung
M06             ; Werkzeugwechsel
                ; (die Nummer des WZ-Wechsel-M-Codes ist einstellbar)
                ; d.h. mit Programmierung von M06 erfolgt in NC das
                ; Aktivwerden einer neuen D-Korrektur; (siehe dazu das
                ; Maschinendatum MD20270 $MC_CUTTING_EDGE_DEFAULT)
```

Bei nicht aktiver Werkzeugverwaltung können folgende Probleme erkannt werden:

- D-Korrekturdatensatz fehlt
- Fehler im Teileprogramm

Hinweis

Das Problem "Werkzeug ist nicht im Magazin" kann nicht erkannt werden, da bei der Werkzeugkorrektur dem NC keine Magazininformationen vorliegen.

D-Korrekturdatensatz fehlt

Die Programmabarbeitung bleibt auf dem Satz mit dem fehlerhaften D-Wert stehen (unabhängig vom Wert des Maschinendatums MD22550). Der Bediener muss entweder das Programm korrigieren, oder er muss den fehlenden Datensatz nachladen.

Dazu benötigt er bei flacher D-Nummernfunktion die D-Nummer und ansonsten zusätzlich die T-Nummer. Diese Parameter werden durch den ausgelösten Alarm mit übergeben.

Fehler im Teileprogramm

Die Bearbeitungsmöglichkeiten im Fehlerfall hängen davon ab, wie der Werkzeugwechsel programmiert wurde, festgelegt über das Maschinendatum:

MD22550 \$MC_TOOL_CHANGE_MODE (Neue Werkzeugkorrektur bei M-Funktion)

Werkzeugwechsel mit T-Programmierung (MD22550 = 0)

Für diesen Fall erfolgt die Bearbeitung über die im NC verfügbare Funktion "Korrektursatz". Das NC-Programm stoppt auf dem NC-Satz, in dem ein Fehler beim programmierten T-Wert erkannt wurde. Beim Fortsetzen des Programms wird der "Korrektursatz" erneut abgearbeitet.

Durch den Bediener können folgende Bearbeitungen vorgenommen werden:

- Das fehlerhafte Teileprogramm kann korrigiert werden.
- Fehlende Schneidenkorrektur-Daten können von HMI nachgeliefert werden.
- Fehlende Schneidenkorrektur-Daten können in "Überspeichern" im NC eingebracht werden.

Nach dem Bedienereingriff wird die `START`-Taste gedrückt und der beanstandete Satz wird wieder abgearbeitet. Wurde der Fehler richtig korrigiert, wird das Programm fortgesetzt. Ansonsten wird erneut ein Alarm erzeugt.

Werkzeugwechsel mit T und M06-Programmierung (MD22550 = 1)

In diesem Fall wird im NC-Satz der WZ-Vorbereitung (Programmierung von T) ein Fehler bemerkt, der aber zunächst ignoriert werden soll. Die Verarbeitung wird fortgesetzt, bis die WZ-Wechselanforderung (meist `M06`) im NC-Programm abgearbeitet wird. An dieser Stelle soll das Programm "halten".

Die programmierte T-Adresse kann beliebig viele Programmzeilen vor dem M06-Befehl stehen, oder die beiden Anweisungen können in unterschiedlichen (Unter-)Programmen stehen. Deshalb ist es im Allgemeinen nicht möglich, einen abgearbeiteten Satz über Korrektursatz zu ändern.

Der Bediener hat dieselben Bearbeitungsmöglichkeiten wie bei = 0.

Das Nachladen fehlender Daten ist möglich. Dann muss aber beim "Überspeichern" T programmiert werden.

Sofern ein Programmfehler vorliegt, kann nicht die Zeile mit dem Fehler korrigiert werden (Txx), sondern nur die Zeile, auf der die Programmabarbeitung gestoppt und den Alarm erzeugt hat. Nur wenn das Maschinendatum:

MD22562 \$MC_TOOL_CHANGE_ERROR_MODE Bit0 = 1 (Verhalten bei Fehlern im Werkzeugwechsel)

Es ergibt sich folgender Ablauf:

```
Txx          ; Fehler! Datensatz mit xx gibt es nicht
              ; Zustand merken; xx merken;
              ; weitermachen im Programm
.....
```



```

M06          ; Merker 'xx fehlt' erkennen → Alarm ausgegeben,
              ; Programm stoppen
              ; Satz korrigieren mit z.B. Tyy M06, starten,
              ; der Satz Tyy M06 wird interpretiert und ist o.k.
              ; Die Bearbeitung geht weiter.

```

Bei erneuter Abarbeitung der Programmstelle ergibt sich folgendes:

```

Txx          ; Fehler! Datensatz mit xx gibt es nicht,
              ; Zustand merken; xx merken;
              ; weitermachen im Programm
....
Tyy M06      ; Merker "xx fehlt" erkennen → ohne weitere Reaktion verwerfen,
              ; denn Tyy M06 ist korrekt → Programm stoppt nicht (richtig).

```

Falls erforderlich kann nach Programmende die ursprüngliche Stelle des T-Aufrufs korrigiert werden. Falls die WZ-Wechsellogik an der Maschine dies nicht verarbeiten kann, muss das Programm abgebrochen und die fehlerhafte Stelle korrigiert werden.

Sofern nur der Datensatz fehlt, wird dieser zum NC übertragen, in "Überspeichern" Txx programmiert und anschließend das Programm fortgesetzt.

Wie bei "fehlender D-Nummer" wird auch bei "fehlender T-Nummer" über den entsprechenden Alarm (17191) der erforderliche Parameter (T-Nummer) dem Bediener zur Verfügung gestellt.

Hinweis

Um die Programmkorrektur zu ermöglichen, wird sofort bei dem fehlerhaften Txx-Satz angehalten.

Bei Programmtestbetrieb wird auch angehalten beim Maschinendatum:

MD22562 \$MC_TOOL_CHANGE_ERROR_MODE Bit0=1 (Verhalten bei Fehlern im Werkzeugwechsel)

19.2.7 Definition der Wirkung der Werkzeugparameter

MD20360

Die Wirkung der Werkzeugparameter auf die Planachse kann im Zusammenhang mit Durchmesserprogrammierung gezielt gesteuert werden mit dem Maschinendatum:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

Details sind beim genannten MD beschrieben.

DRF manuelles Verfahren per Handrad mit halbem Weg

Beim DRF-Hanradverfahren kann eine Planachse nur mit dem halben Weg des vorgegebenen Inkrements wie folgt verfahren werden:

Festlegung der Wegvorgabe mit Handrad über das Maschinendatum:

MD11346 \$MN_HANDWHEEL_TRUE_DISTANCE = 1 (Handrad Weg- oder Geschwindigkeitsvorgabe)

Definition der DRF-Verschiebung in der Planachse als Durchmesserkorrekturwert einrechnen mit dem Maschinendatum:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK Bit 9 = 1 (Definition der WZ-Parameter)

Bei der axialen Abwahl DRF_{OF} einer DRF-Verschiebung wird eine existierende Werkzeugkorrektur (Handradüberlagerung in Werkzeugrichtung) auch gelöscht.

Hinweis

Weitere Informationen zu überlagerten Bewegungen mit Handrad entnehmen Sie bitte:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; manuelles Verfahren und manuelles Verfahren per Handrad (H1)

Programmierhandbuch Grundlagen

(Die Programmieranleitung beschreibt die vollständigen programmtechnischen Möglichkeiten, um die DRF-Verschiebung achsspezifisch abzuwählen.)

19.3 Flache D-Nummern-Struktur

19.3.1 Allgemeine Informationen

Für **Drehmaschinen** ist eine einfache Werkzeugverwaltung mittels flacher D-Nummern-Struktur () ausschließlich über D-Nummern möglich.

Daraus ergeben sich folgende Randbedingungen:

- nur bei **nicht** aktivierter allgemeiner Werkzeugverwaltung verfügbar
- keine Schwester-Werkzeuge definierbar
- keine Magazine definierbar
- keine Schleifwerkzeuge definierbar

Aktivierung

Welche Art der D-Nummernverwaltung wirksam ist, wird parametrisiert über:

MD18102 \$MN_MM_TYPE_OF_CUTTING_EDGE = <Wert>

Wert	Bedeutung
0	keine flache D-Nummernverwaltung aktiv
1	flache D-Nummern-Struktur mit absoluter direkter D-Programmierung aktiv

19.3.2 Neuanlegen einer D-Nummer (Korrektursatz)

Programmierung

Mit den Systemvariablen \$TC_DP1... \$TC_DP25 können Werkzeugkorrekturen programmiert werden. Inhaltlich ist die Bedeutung dieselbe wie bisher.

Die Schreibweise ändert sich: es wird keine T-Nummer mehr angegeben.

- Funktion "Flache D-Nummer" **aktiv**:
`$TC_DPx[d] = wert` ; mit x=Parameternr., d=D-Nummer
 D.h., Daten mit dieser Syntax können nur nach NC geladen werden, wenn die Funktion "Flache D-Nummer" aktiviert ist.
- Funktion 'Flache D-Nummer' **inaktiv**:
`$TC_DPx[t][d] = wert` ; mit t=T-Nummer, d=D-Nummer

Eine D-Nummer kann je Werkzeug nur einmal vergeben werden; d.h. jede D-Nummer steht genau für einen Korrekturdatensatz.

Ein neuer Datensatz wird im Speicher von NC dann angelegt, wenn zum ersten Mal eine nicht existierende D-Nummer programmiert wird.

Die max. Anzahl der D- bzw. Korrektur-Datensätze (max. 600) wird eingestellt über das Maschinendatum:

MD18100 \$MN_MM_NUM_CUTTING_EDGES_IN_TOA (Werkzeugkorrekturen im TO-Bereich)

Datensicherung

Die Datensicherung erfolgt im selben Format; d.h. eine Sicherungsdatei, die mit der Funktion "Flache D-Nummer" erstellt wird, kann nicht in den NC einer Steuerung gespielt werden, die die Funktion nicht aktiviert hat.

Das gilt auch für die umgekehrte Richtung einer Übertragung.

D-Nummernbereich

1 - 99 999 999

19.3.3 D-Nummer programmieren

MD18102 = 1

Wenn MD18102 \$MN_MM_TYPE_OF_CUTTING_EDGE = 1, dann wird die D-Korrektur ohne Bezug zu einem bestimmten Werkzeug aktiviert.

D0 behält weiterhin die bisherige Bedeutung "Abwahl der aktiven Korrektur in NC".

Adresserweiterung von D

Eine Adresserweiterung von D ist nicht möglich. Für die WZ-Bahn gibt es zu einem Zeitpunkt nur einen möglichen aktiven Korrekturdatensatz.

Programmierung

Im Teileprogramm wird wie bisher programmiert. Nur der Wertebereich der programmierten D-Nummer erhöht sich.

Beispiel 1:

MD-Parametrierung		Bedeutung
MD22550 \$MC_TOOL_CHANGE_MODE	= 0	Werkzeugwechsel mit der Programmierung von T.
MD18102 \$MN_MM_TYPE_OF_CUTTING_EDGE	= 1	D-Nummer-Programmierung ohne Bezug zu einem bestimmten Werkzeug.
MD20270 \$MC_CUTTING_EDGE_DEFAULT	= -1	D-Korrektur bleibt bei Werkzeug-Wechsel unverändert.

Programmcode	Kommentar
...	
D92	
X0	; Verfahre mit den Korrekturen von D92.
T17	; Gibt T=17 an PLC aus.
X1	; Verfahre mit den Korrekturen von D92.
D16	
X2	; Verfahre mit den Korrekturen von D16.
D32000	
X3	; Verfahre mit den Korrekturen von D32000.
T29000	; Gibt T=29000 an PLC aus.
X4	; Verfahre mit den Korrekturen von D32000.
D1	
X5	; Verfahre mit den Korrekturen von D1.
...	

Beispiel 2:

MD22550 = 0

Programmcode	Kommentar
T1	
T2	
T3	
D777	; Kein Warten, D777 wird aktiviert, T3= programmiertes und aktives WZ in der Anzeige, D777= programmierte und aktive Korrektur.

Hinweis

Der WZ-Wechsel und die Zuordnung einer D-Korrektur zu einem konkreten WZ liegt in der Verantwortung des NC-Programms und gegebenenfalls des PLC-Programms.

D-Nr. über Teileprogramm löschen

- **Mit flacher D-Nummer:**
\$TC_DP1[d] = 0
Korrekturdatensatz mit der Nummer D in der TO-Einheit wird gelöscht.
Der Speicher ist anschließend frei für die Definition einer anderen D-Nummer.
- **Ohne flache D-Nummer:**
\$TC_DP1[t][d] = 0
Schneide d des Werkzeugs t wird gelöscht.
- \$TC_DP1[0] = 0
Alle D-Korrekturen der TO-Einheit werden gelöscht.

Aktive Korrektur-Datensätze (D-Nummern) können nicht gelöscht werden. D. h. es kann eventuell erforderlich sein, vor dem Löschen ein D0 zu programmieren.

Werkzeug-MDs

Folgende Maschinendaten beeinflussen den Umgang mit Werkzeugen und Schneiden (D-Nummern) im NC:

Maschinendatum	Bedeutung
MD20270 \$MC_CUTTING_EDGE_DEFAULT	Standard-Werkzeugschneide nach Werkzeugwechsel
MD20130 \$MC_CUTTING_EDGE_RESET_VALUE	Werkzeugschneide - Längenkorrektur im Hochlauf (RESET / TP-Ende)
MD20120 \$MC_TOOL_RESET_VALUE	Werkzeug - Längenkorrektur im Hochlauf (RESET / TP-Ende)
MD20121 \$MC_TOOL_PRESEL_RESET_VALUE	Vorgewähltes Werkzeug bei RESET
MD22550 \$MC_TOOL_CHANGE_MODE	Werkzeugwechsel mit M-Funktion statt mit T-Funktion
MD22560 \$MC_TOOL_CHANGE_M_CODE	M-Funktion für Werkzeugwechsel

Maschinendatum	Bedeutung
MD20110 \$MC_RESET_MODE_MASK	Festlegung der Steuerungs-Grundstellung nach RESET / TP-Ende
MD20112 \$MC_START_MODE_MASK	Festlegung der Steuerungs-Grundstellung bei NC-START

19.3.4 T-Nummer programmieren

Bei aktiver Funktion "Flache D-Nummern-Struktur" wird die NC-Adresse T weiterhin ausgewertet, d.h. die programmierte wie die aktive T-Nr. werden angezeigt. Die D-Nr. ermittelt die NC aber unabhängig vom programmierten T-Wert.

Die NC kennt je Kanal 1 Masterspindel (über MD einstellbare Spindelnummer). Korrekturen und der Befehl M6 (WZ-Wechsel) werden nur in Bezug auf die Masterspindel berechnet.

Eine Adresserweiterung von T wird als Spindel-Nr. erkannt (z.B. T2 = 1; WZ 1 soll an Spindel 2 angewählt werden), ein WZ-Wechsel wird nur erkannt, wenn Spindel 2 die Masterspindel ist.

19.3.5 M6 programmieren

MD22550 und MD22560

Die NC kennt je Kanal 1 Masterspindel (über MD einstellbare Spindelnummer). Korrekturen und der Befehl M6 (WZ-Wechsel) werden nur in Bezug auf die Masterspindel berechnet.

Ob mit einer M-Funktion der WZ-Wechselbefehl erfolgt wird festgelegt über das Maschinendatum:

MD22550 \$MC_TOOL_CHANGE_MODE (Neue Werkzeugkorrektur bei M-Funktion)

T ist dabei der WZ-Vorbereitungsbefehl.

Der Name der M-Funktion für den WZ-Wechsel wird festgelegt über das Maschinendatum:

MD22560 \$MC_TOOL_CHANGE_M_CODE (M-Funktion für Werkzeugwechsel)

Standard ist M6. Eine Adresserweiterung von M6 wird als Spindel-Nr. erkannt.

Beispiel

Es sind zwei Spindeln definiert, Spindel 1 und Spindel 2, und es gilt:

```
MD20090 = 2      ; Spindel-Nr. 2 ist Masterspindel.
M6              ; WZ-Wechsel gewünscht, Befehl bezieht sich implizit auf Master-
                ; spindel
M1 = 6          ; kein WZ-Wechsel, da Spindel-Nr. 2 die Masterspindel ist
M2 = 6          ; WZ-Wechsel erfolgt, da Spindel-Nr. 2 die Masterspindel ist
```

19.3.6 Programmtest

MD20110

Dass das aktive WZ und die WZ-Korrektur wie folgt übernommen werden, kann eingestellt werden mit dem Maschinendatum:

MD20110 \$MC_RESET_MODE_MASK, Bit 3 (Festlegung der Steuerungs-Grundstellung nach RESET/TP-Ende)

Wert		Bedeutung
Bit 3	= 1	aus dem zuletzt beendeten Testprogramm im Testbetrieb
	= 0	aus dem zuletzt beendeten Programm vor Einschalten des Programmtests

Voraussetzung

Die Bits 0 und 6 müssen gesetzt sein vom Maschinendatum:

MD20110 \$MC_RESET_MODE_MASK, Bit 3 (Festlegung der Steuerungs-Grundstellung nach RESET/TP-Ende)

19.3.7 Werkzeugverwaltung oder "Flache D-Nummern-Struktur"

Merkmale

Werkzeugverwaltung

Die aktive Werkzeugverwaltung der NC arbeitet mit folgenden Annahmen:

1. Werkzeuge werden in Magazinen verwaltet.
2. Schneiden werden überwacht, d.h. erreichte Grenzwerte bewirken die Sperren des Werkzeugs.
3. Schwester-Werkzeuge: Werkzeuge werden für die Anwahl nur mit ihrem Namen programmiert. Die NC wählt das konkrete Werkzeug aus.

Hinweis

Diese Funktionalität steht bei SINUMERIK 828D nur als Option zur Verfügung!

Das bedeutet, dass der Einsatz der Werkzeugverwaltung nur Sinn ergibt, wenn konkrete Werkzeuge definiert sind, und diese von der NC verwaltet werden.

Flache D-Nummern-Struktur

Die Verwendung einer flachen D-Nummern-Struktur bedeutet, dass die Werkzeugverwaltung außerhalb der NC erfolgt (PLC) und kein Bezug zu den T-Nummern besteht.

Werkzeugverwaltung ODER flache D-Nummern-Struktur

Der gleichzeitige Einsatz von Werkzeugverwaltung (NC) und flacher D-Nummern-Struktur (PLC) macht keinen Sinn, da das Hauptargument für die Verwendung der Werkzeugverwaltung der Faktor **Zeit** ist. Das ist nur gegeben, wenn die Verwaltungsaufgaben von der NC durchgeführt werden. Das ist bei Verwendung der flache D-Nummern-Struktur aber nicht der Fall.

Hinweis

Werden Werkzeugverwaltung UND flache D-Nummern-Struktur aktiviert, besitzt die Werkzeugverwaltung die höhere Priorität.

19.4 Werkzeugschneide

19.4.1 Allgemeine Informationen

Werkzeugschneide

Eine WZ-Schneide ist eindeutig durch folgende Angaben beschrieben:

- Werkzeugtyp (Schafffräser, Bohrer, etc.)
- Geometrische Beschreibung
- Technologische Beschreibung

WZ-Parameter

Die geometrische Beschreibung, die technologische Beschreibung und der WZ-Typ werden für jede WZ-Schneide auf WZ-Parameter abgebildet.

Zu den jeweiligen Werkzeugtypen sind folgende WZ-Parameter vorhanden:

WZ-Parameter	Bedeutung	Bemerkung
1	Werkzeugtyp	
2	WZ-Schneidenlage	für Drehwerkzeuge oder für Fräs-/Schleifwerkzeuge bei 2D-WRK Kontur-WZ
Geometrie - Werkzeuglängen		
3	Länge 1	
4	Länge 2	
5	Länge 3	
Geometrie - Werkzeugform		
6	Radius 1/Länge 1	bei 3D Stirnfräsen
7	Länge 2	bei 3D Stirnfräsen

WZ-Parameter	Bedeutung	Bemerkung
8	Radius 1	bei 3D Stirnfräsen
9	Radius 2	bei 3D Stirnfräsen
10	Winkel 1 / Minimaler Grenzwinkel	bei 3D Stirnfräsen bei 2D-WRK Kontur-WZ
11	Winkel 2 / Maximaler Grenzwinkel	bei 3D Stirnfräsen bei 2D-WRK Kontur-WZ
Verschleiß - Werkzeuglängen		
12	Länge 1	
13	Länge 2	
14	Länge 3	
Verschleiß - Werkzeugform		
15	Radius 1/Länge 1	bei 3D Stirnfräsen
16	Länge 2	bei 3D Stirnfräsen
17	Radius 1	bei 3D Stirnfräsen
18	Radius 2	bei 3D Stirnfräsen
19	Winkel 1 / minimaler Grenzwinkel	bei 3D Stirnfräsen bei 2D-WRK Kontur-WZ.
20	Winkel 2 / maximaler Grenzwinkel	bei 3D Stirnfräsen bei 2D-WRK Kontur-WZ.
Basismaß / Adaptermaß		
21	Basis-Länge 1	
22	Basis-Länge 2	
23	Basis-Länge 3	
Technologie		
24	Freischneidwinkel	nur für Drehwerkzeuge
25	Reserviert*	

* "Reserviert" bedeutet, dass dieser WZ-Parameter nicht benutzt wird und für Erweiterungen reserviert ist.

3D-Stirnfräsen

Beim 3D-Stirnfräsen werden die Fräser typen 111, 120, 121, 130, 155, 156 und 157 speziell behandelt, indem gegebenenfalls die Werkzeugparameter (1-23) ausgewertet werden.

Literatur

Weitere Informationen zu verschiedenen Werkzeugtypen siehe:

- Funktionshandbuch Grundfunktionen; Werkzeugkorrektur (W1), Kapitel: "Werkzeugtyp (WZ-Parameter)"
- Programmierhandbuch Grundlagen; Kapitel: "Werkzeugkorrekturen" > "Liste der Werkzeugtypen"
- Funktionshandbuch Sonderfunktionen; 3D-Werkzeugradiuskorrektur (W5)

19.4.2 WZ-Parameter 1: Werkzeugtyp

Bedeutung

Der Werkzeugtyp (3-stellige Nummer) legt fest, um welches Werkzeug es sich handelt. Mit der Auswahl des Werkzeugtyps sind die anderen Komponenten wie Geometrie, Verschleiß oder Basismaß vorbestimmt.

Bedingungen

Für den Parameter "Werkzeugtyp" gilt:

- Der WZ-Typ muss für jede WZ-Schneide angegeben werden.
- Der WZ-Typ darf nur die festgelegten Werte annehmen.
- Der WZ-Typ "0" bedeutet, dass kein gültiges Werkzeug definiert ist.

Werkzeugtypen und Werkzeugparameter

In der folgenden Tabelle sind verschiedene Werkzeugtypen und die wichtigsten Werkzeugparameter aufgelistet. Die für einen bestimmten Werkzeugtyp verfügbaren Werkzeugparameter sind mit "x" gekennzeichnet:

Tabelle der wichtigsten verfügbaren Werkzeugparameter (notwendig ist nur der Werkzeugtyp)													
1	WZ-Parameter \$TC_DP	2	3	4	5	6	7	8/9	12/13/14	15/16	21/22/23	24	TPG 3
Werkzeugtyp	Verrechnung durch NCK	Schneidenlage	Geometrie – Länge 1	– Länge 2	– Länge 3	Geometrie – Radius 1	Nutbreite / – Radius 2	Überstand / – Länge 4/5	Verschleiß – Länge 1/2/3	Verschleiß – Radius 1/2	Adap/Basismaß – Länge 1/2/3	Freischneidwinkel	– Scheibenradius min
													TPG 4/5
													– Scheibenbreite min/akt.
													– Drehzahl max
													– Umfangsgeschwindigkeit max
													– Winkel schräge Scheibe
Fräswerkzeuge und Bohrwerkzeuge (alle übrigen)													
100	Fräswerkzeug nach CLDATA* ¹		x	x	x	x			x	x	x		
110	Kugelpkopfräser		x	x	x	x			x	x	x		
120	Schaftfräser		x	x	x	x			x	x	x		
121	Schaftfräser mit Eckenverrundung		x	x	x	x	x		x	x	x		
130	Winkelkopfräser		x	x	x	x			x	x	x		
131	Winkelkopfräser mit Eckenverrundung		x	x	x	x	x		x	x	x		
140	Planfräser		x	x	x	x			x	x	x		
145	Gewindefräser		x	x	x	x			x	x	x		
150	Scheibenfräser		x	x	x	x			x	x	x		
155	Kegelstumpfräser		x	x	x	x			x	x	x		
200	Spiralbohrer		x	x	x	x			x		x		
205	Vollbohrer		x	x	x	x			x		x		
210	Bohrstange		x	x	x	x			x		x		
220	Zentrierbohrer		x	x	x	x			x		x		
230	Spitzsenker		x	x	x	x			x		x		
231	Flachsenker		x	x	x	x			x		x		
240	Gewindebohrer Regelgewinde		x	x	x	x			x		x		
241	Gewindebohrer Feingewinde		x	x	x	x			x		x		
242	Gewindebohrer Whitworthgewinde		x	x	x	x			x		x		
250	Reibahle		x	x	x	x			x		x		
Schleifwerkzeuge und Drehwerkzeuge (400 – 599)													
400	Umfangsschleifscheibe	x	x	x	x	x			x	x	x		x
401	Umfangsschleifscheibe mit Überwachung	x	x	x	x	x			x	x	x	x	x
403	wie 401 aber ohne Basismaß für SUG* ²	x	x	x	x	x			x	x	x	x	x
410	Planscheibe	x	x	x	x	x			x	x	x		x
411	Planscheibe mit Überwachung	x	x	x	x	x			x	x	x	x	x
413	wie 411 aber ohne Basismaß für SUG* ²	x	x	x	x	x			x	x	x	x	x
490	Abrichter	x	x	x	x	x			x	x	x		
500	Schruppstahl	x	x	x	x	x			x	x	x	x	
510	Schlichtstahl	x	x	x	x	x			x	x	x	x	
520	Einstechstahl	x	x	x	x	x			x	x	x	x	
530	Abstechstahl	x	x	x	x	x			x	x	x	x	
540	Gewindestahl	x	x	x	x	x			x	x	x	x	
550	Pilzstahl / Formstahl (WZV)	x	x	x	x	x			x	x	x	x	
Sonderwerkzeuge (700)													
700	Nutsäge		x	x	x	x	x	x	x	x	x		
CLDATA* ¹ "cutter Location Data" — Werkzeugpositionsdaten nach DIN 66215, BI 1SUG* ² — Schleifenscheibenumfangsgeschwindigkeit													

Hinweis

Innerhalb der Gruppe Drehwerkzeuge hat der Werkzeugtyp keine Bedeutung.

Insbesondere bei den Schleifwerkzeugen (400-499) sind auch nicht aufgeführte Nummern zulässig.

Werkzeugkorrekturdaten

Die Werkzeugkorrekturdaten (TOA-Daten) werden in Systemvariablen abgelegt.

Beispiel: Werkzeugtyp Nutsäge (Typ 700)

	Geometrie	Verschleiß	Basis	Einheit
Längenkorrektur				
Länge 1	\$TC_DP3	\$TC_DP12	\$TC_DP21	mm
Länge 2	\$TC_DP4	\$TC_DP13	\$TC_DP22	mm
Länge 3	\$TC_DP5	\$TC_DP14	\$TC_DP23	mm
Sägeblattkorrektur				
Durchmesser d	\$TC_DP6	\$TC_DP15		mm
Nutbreite b	\$TC_DP7	\$TC_DP16		mm
Überstand k	\$TC_DP8	\$TC_DP17		mm

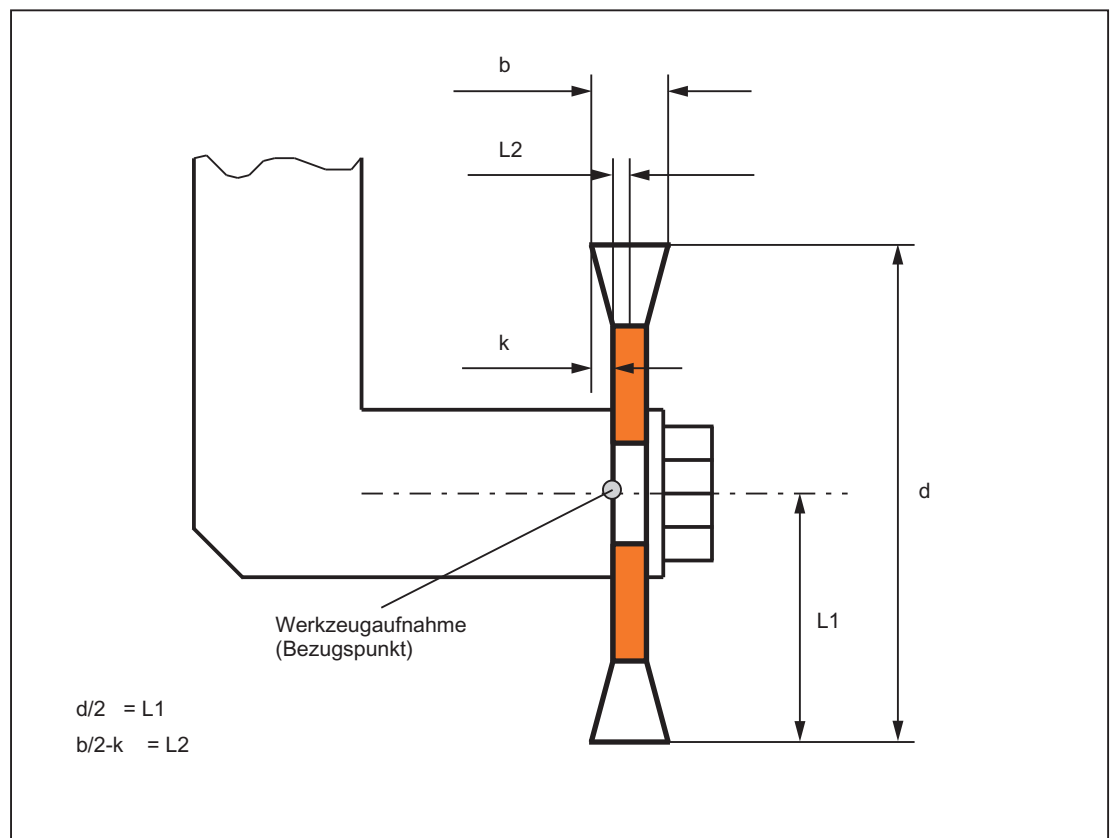


Bild 19-3 Geometrie der Nutsäge (analog zum Winkelkopfräser)

Die Breite des Sägeblattes wird bei der Werkzeugradiuskorrektur (G40 bis G42) wie folgt berücksichtigt:

Befehl	Bedeutung
G40	keine Sägeblattkorrektur
G41	Sägeblattkorrektur links
G42	Sägeblattkorrektur rechts

19.4.3 WZ-Parameter 2: Schneidenlage

Bedeutung

Die Schneidenlage beschreibt die Lage der Werkzeugspitze P in Bezug auf den Schneidenmittelpunkt S. Sie wird in den WZ-Parameter 2 eingetragen.

Die Schneidenlage wird zusammen mit dem Schneidenradius (WZ-Parameter 8) zur Berechnung der Werkzeugradiuskorrektur bei Drehwerkzeugen (Werkzeugtyp 5xx) benötigt.

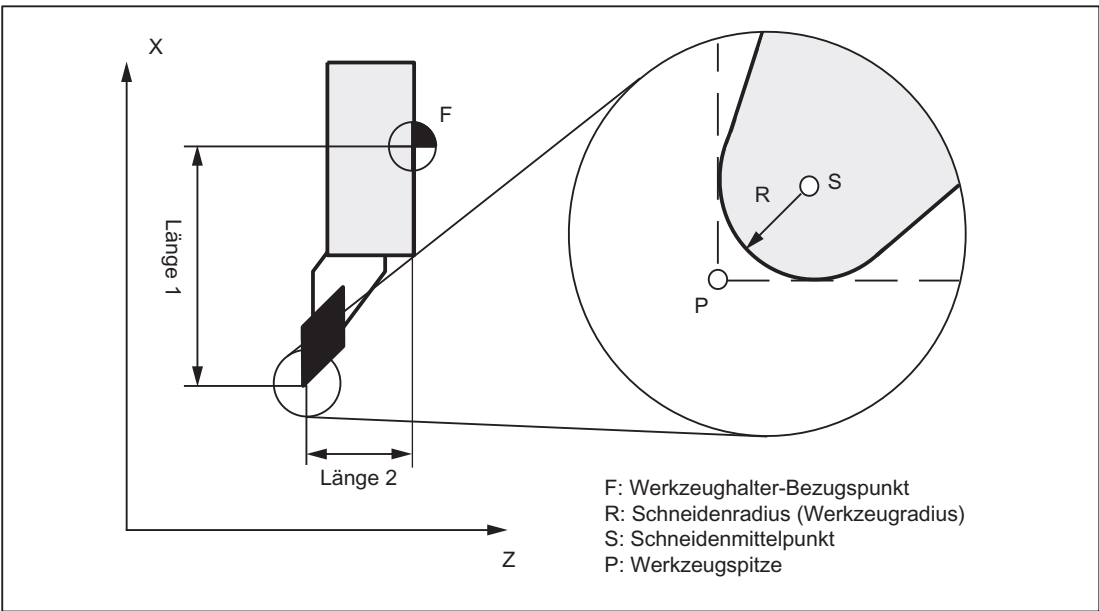


Bild 19-4 Abmessungen bei Drehwerkzeugen: Drehmeisel

Schneidenlagen-Parameterwerte

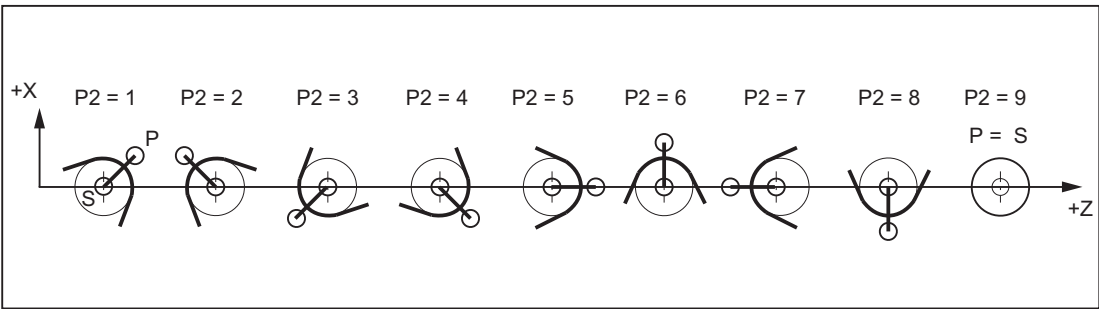


Bild 19-5 Werkzeugparameter 2 (P2): Bei Bearbeitung hinter der Drehmitte

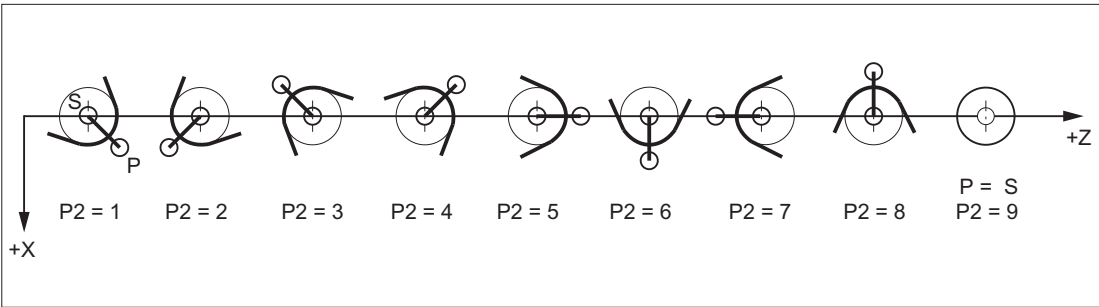


Bild 19-6 Werkzeugparameter 2 (P2): Bei Bearbeitung vor der Drehmitte

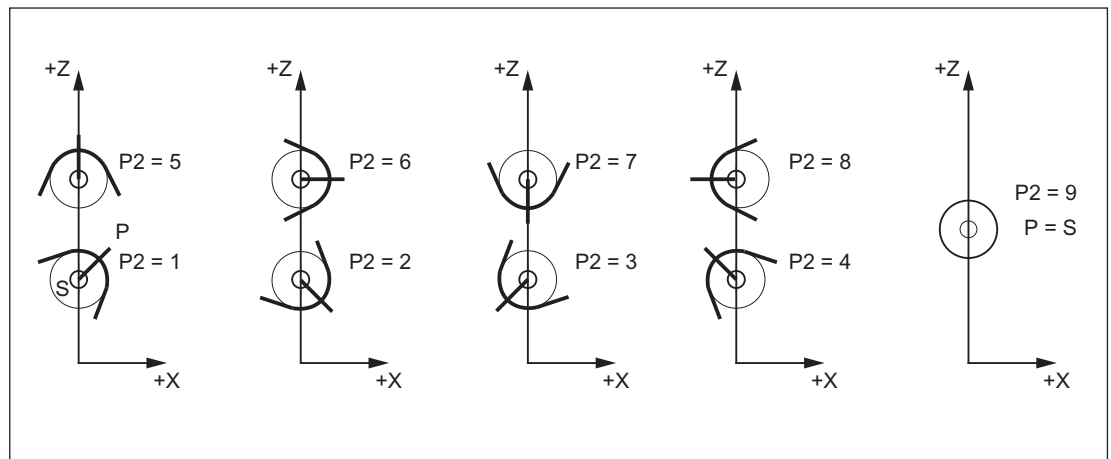


Bild 19-7 Werkzeugparameter 2 (P2): Schneidenlage bei Karusselldrehmaschinen

Besonderheiten

- Wird zur Ermittlung der WZ-Längenkorrektur als Bezugspunkt nicht der Punkt P, sondern der Schneidmittelpunkt S benützt, ist als WZ-Schneidenlage die Kennung 9 einzugeben.
- Die Kennung 0 (null) als WZ-Schneidenlage ist nicht zulässig.

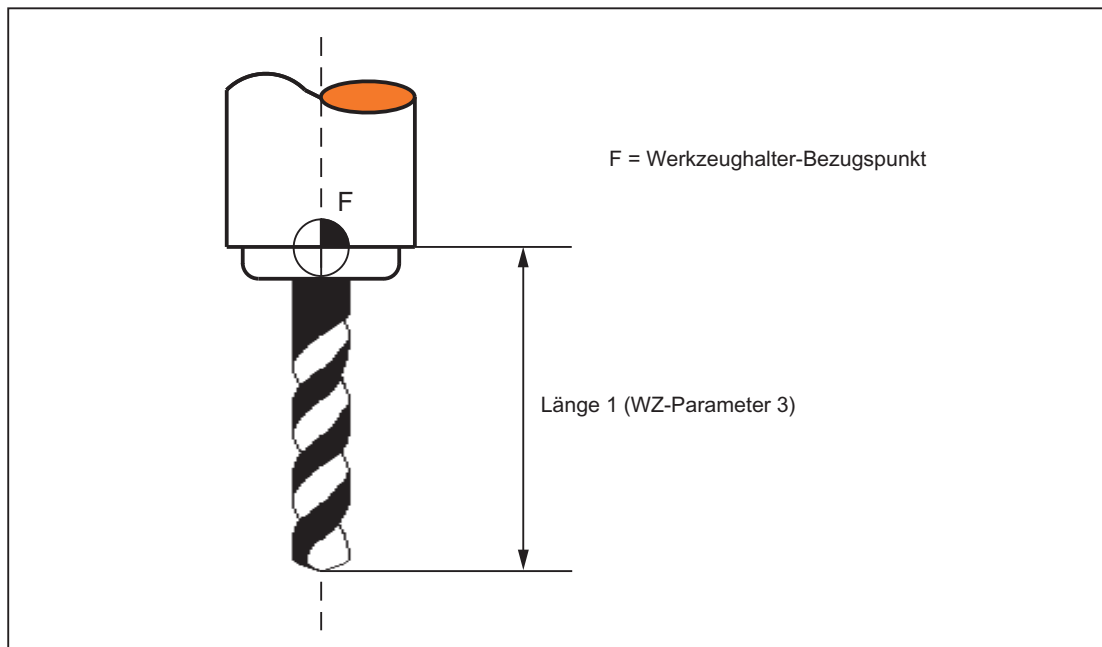
19.4.4 WZ-Parameter 3 - 5: Geometrie - Werkzeuglängen

Bedeutung

Die Längen der Werkzeuge werden für die Geometrie-Werkzeuglängenkorrektur benötigt. Sie werden als WZ-Länge 1 bis 3 in die WZ-Parameter 3 bis 5 eingegeben. Je nach WZ-Typ sind folgende Längenangaben mindestens erforderlich:

Werkzeugtyp	Erforderliche WZ-Längen
WZ-Typ 12x, 140, 145, 150:	WZ-Länge 1
WZ-Typ 13x:	WZ-Länge 1-3 (je nach Ebene G17-G19)
WZ-Typ 2xx:	WZ-Länge 1
WZ-Typ 5xx:	WZ-Länge 1-3

Beispiel: Spiralbohrer (WZ-Typ 200) mit WZ-Länge 1 (WZ-Parameter 3)



Hinweis

Unabhängig vom WZ-Typ werden immer alle drei WZ-Parameter 3 bis 5 (WZ-Länge 1 bis 3) in den drei Geometrieachsen verrechnet.

Wenn für einen WZ-Typ mehr WZ-Längen in die WZ-Parameter 3 bis 5 eingegeben werden als mindestens erforderlich sind, dann werden diese zusätzlichen WZ-Längen in den Geometrieachsen ohne Alarm verrechnet.

Besonderheit

Erst die Addition der Geometrie-Werkzeuglängenkorrektur (WZ-Parameter 3 bis 5) und der Verschleiß-Werkzeuglängenkorrektur (WZ-Parameter 12 bis 14) legt die aktive Größe des Werkzeugs fest. Zur Verrechnung der gesamten Werkzeuglängenkorrektur in den Geometrieachsen wird auch die Basismaß/Adaptermaß-Werkzeuglängenkorrektur addiert.

Literatur

Zur Eingabe der Werkzeuggröße (Längen) in die WZ-Parameter 3 bis 5 (WZ-Länge 1 bis 3) und die Verrechnung in den drei Geometrieachsen siehe → Bedienhandbuch.

19.4.5 WZ-Parameter 6 - 11: Geometrie - Werkzeugform

Bedeutung

Mit den WZ-Parametern 6 bis 11 wird die Form des Werkzeugs festgelegt. Die Angaben werden für die Geometrie-Werkzeugradiuskorrektur benötigt.

In den meisten Fällen wird nur der WZ-Parameter 8 (WZ-Radius 1) benutzt:

WZ-Parameter	Bedeutung	Verwendung	
6	WZ-Länge 1	nicht verwendet	
7	WZ-Länge 2	nicht verwendet	
8	WZ-Radius 1	Für folgende WZ-Typen muss der WZ-Radius angegeben werden:	
		WZ-Typ	Bedeutung
		1xx	Fräswerkzeuge
		5xx	Drehwerkzeuge Bei Drehwerkzeugen muss auch WZ-Parameter 2: Schneidenlage (Seite 1505) angegeben werden.
		Hinweis WZ-Typ 2xx: Für Bohrwerkzeuge muss kein WZ-Radius angegeben werden.	
9	WZ-Radius 2	nicht verwendet	
10	WZ-Winkel 1	nicht verwendet	
11	WZ-Winkel 2	nicht verwendet	

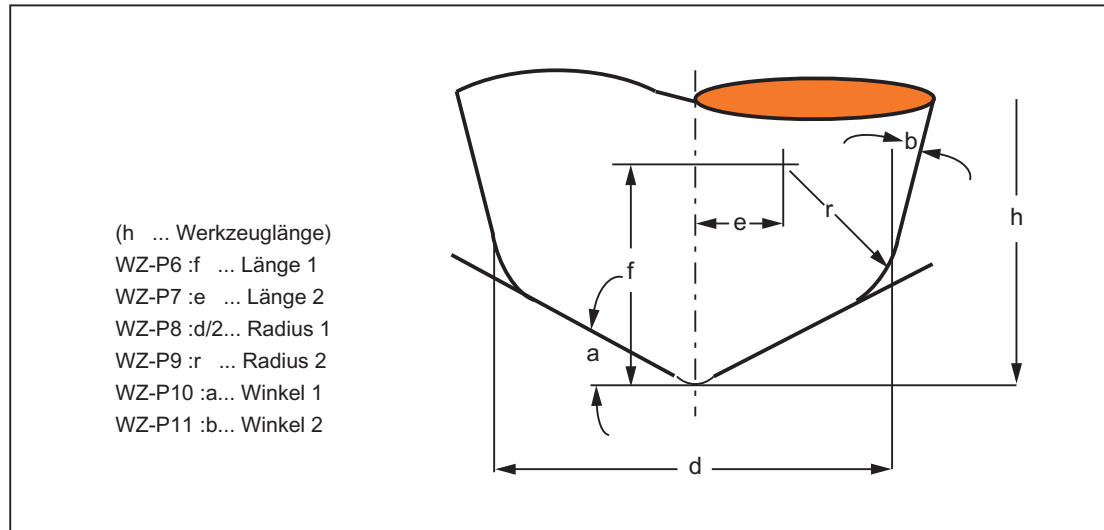
2D-WRK mit Konturwerkzeugen

Zur Definition von Konturwerkzeugen mit mehreren Werkzeugschneiden können der minimale und maximale Grenzwinkel angegeben werden. Die beiden Grenzwinkel beziehen sich jeweils auf den Vektor vom Schneidenmittelpunkt zum Schneidenbezugspunkt und werden im Gegenuhrzeigersinn gezählt.

WZ-Winkel 1	Minimaler Grenzwinkel je Werkzeugschneide
WZ-Winkel 2	Maximaler Grenzwinkel je Werkzeugschneide

3D-Stirnfräsen

Für die Werkzeugbeschreibung beim 3D-Stirnfräsen sind alle WZ-Parameter 6 bis 11 anzugeben:



Literatur

Zur Eingabe der Werkzeugform (Radius für die WRK) in die WZ-Parameter 6 bis 11 und die Verrechnung in den drei Geometrieachsen durch die Geometrie-Werkzeugradiuskorrektur siehe:

- Programmierhandbuch Grundlagen; Kapitel: "Werkzeugkorrekturen" > "2½ D-Werkzeugkorrektur"
- Funktionshandbuch Sonderfunktionen; 3D-Werkzeugradiuskorrektur (W5)

Bezüglich 3D-Stirnfräsen siehe:

- Programmierhandbuch Arbeitsvorbereitung; Kapitel: "Transformationen" > "Drei-, Vier- und Fünf-Achs-Transformation (TRAORI)"

19.4.6 WZ-Parameter 12 - 14: Verschleiß - Werkzeuglängen

Bedeutung

Während mit der Geometrie-Werkzeuglängenkorrektur (WZ-Parameter 3 bis 5) die Größe des Werkzeugs eingestellt wird, kann mit der Verschleiß-Werkzeuglängenkorrektur die Veränderung der aktiven Werkzeuggröße korrigiert werden.

Die aktive Werkzeuggröße kann sich ändern durch:

- Unterschiede zwischen der WZ-Aufnahme in der WZ-Messmaschine und der WZ-Aufnahme in der Werkzeugmaschine
- Verschleiß des Werkzeugs durch die Einsatzdauer bei der Bearbeitung
- Definition eines Schlichtaufmaßes

Aktive WZ-Größe

Die Geometrie-Werkzeuglängenkorrektur (WZ-Parameter 3 bis 5) und die Verschleiß-Werkzeuglängenkorrektur (WZ-Parameter 12 bis 14) werden addiert (Geometrie-WZ-Länge 1 mit Verschleiß-WZ-Länge 1, etc.) und stellen so die Größe des aktiven Werkzeugs dar.

19.4.7 WZ-Parameter 15 - 20: Verschleiß - Werkzeugform

Bedeutung

Während mit der Geometrie-Werkzeugradiuskorrektur (WZ-Parameter 6 bis 11) die Form des Werkzeugs eingestellt wird, kann mit der Verschleiß-Werkzeugradiuskorrektur die Veränderung der aktiven Werkzeugform korrigiert werden.

Die aktive Werkzeuggröße kann sich ändern durch:

- Verschleiß des Werkzeugs durch die Einsatzdauer bei der Bearbeitung
- Definition eines Schlichtaufmaßes

Aktive WZ-Form

Die Geometrie-Werkzeugradiuskorrektur (WZ-Parameter 6 bis 11) und die Verschleiß-Werkzeugradiuskorrektur (WZ-Parameter 15 bis 20) werden addiert (Geometrie-WZ-Radius 1 mit Verschleiß-WZ-Radius 1) und stellen so die Form des aktiven Werkzeugs dar.

19.4.8 WZ-Parameter 21 - 23: Basismaß / Adaptermaß

Bedeutung

Das Basismaß/Adaptermaß kann verwendet werden, wenn der Bezugspunkt des Werkzeughalters (WZ-Größe) nicht mit dem Bezugspunkt des Werkzeugträgers übereinstimmt.

Das ist der Fall, wenn:

- WZ und WZ-Adapter getrennt vermessen, aber schon zusammengebaut in die Maschine eingesetzt werden (WZ-Größe und Adaptergröße werden getrennt in eine WZ-Schneide eingegeben).
- das WZ in eine zweite Werkzeugaufnahme eingesetzt wird, die sich an einer anderen Position befindet (z. B: Vertikal - und Horizontalspindel).
- die Werkzeugaufnahmen eines Werkzeugrevolvers unterschiedliche Positionen besitzen.

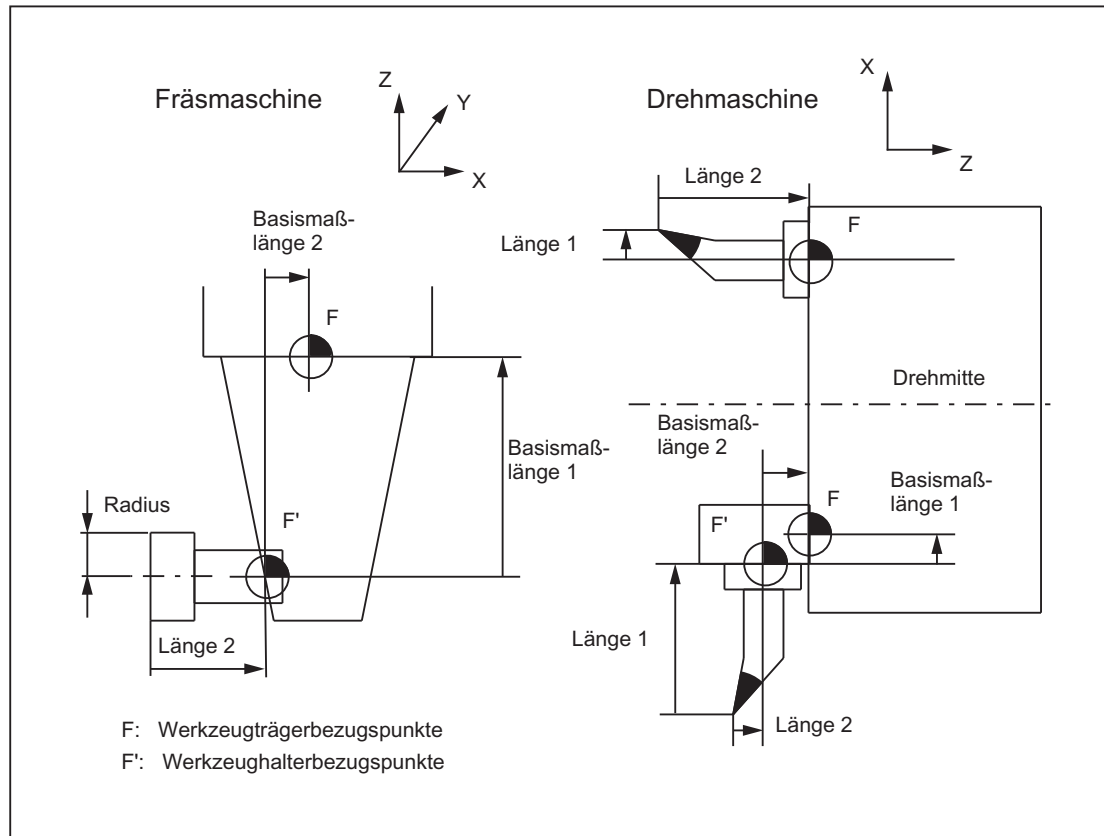


Bild 19-8 Anwendungsbeispiele für Basismaß / Adaptermaß-WLK

Basis-Länge 1 bis 3 (WZ-Parameter 21 bis 23)

Damit der Werkzeughalterbezugspunkt F zum Werkzeugträgerbezugspunkt F' in den drei Geometrieachsen (3-dimensional) korrigiert werden kann, wirken unabhängig von WZ-Typ immer alle 3 Basis-Längen. D. h. auch ein Spiralbohrer (WZ-Typ 200) mit einer Werkzeuglängenkorrektur (Länge 1) kann ein Basismaß/Adaptermaß in 3 Achsen haben.

Literatur

Zur genaueren Beschreibung der Basismaß/Adaptermaß-Werkzeuglängenkorrektur siehe:

- Programmierhandbuch Grundlagen

19.4.9 WZ-Parameter 24: Freischneidwinkel

Bedeutung

Bestimmte Drehzyklen, in denen Verfahrenbewegungen mit Hinterschneiden erzeugt werden, überwachen den Freischneidwinkel des aktiven Werkzeugs auf eine mögliche Konturverletzung.

Wertebereich

Der Winkel (0 bis 90° ohne Vorzeichen) wird als Freischneidwinkel in den WZ-Parameter 24 eingegeben.

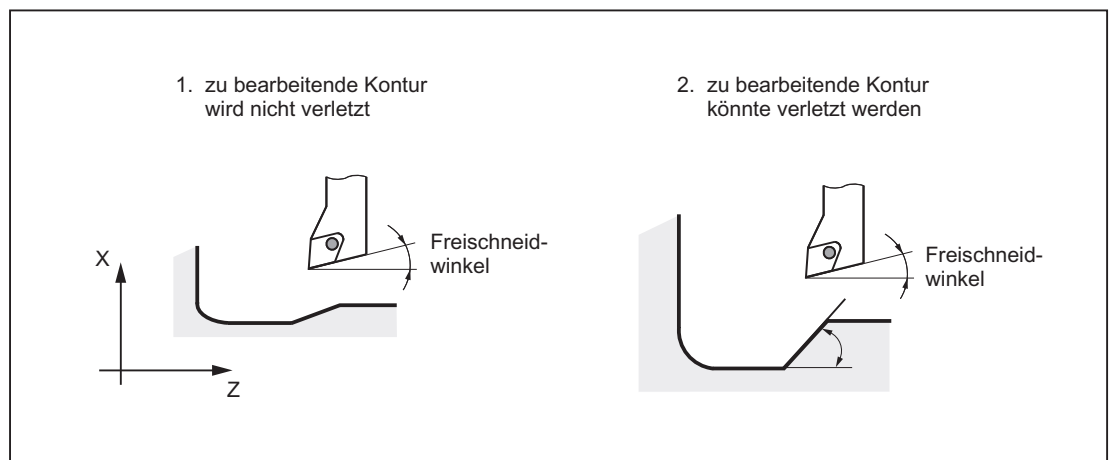


Bild 19-9 Freischneidwinkel des Drehwerkzeugs beim Hinterschneiden

Bearbeitungsart Längs oder Plan

Der Freischneidwinkel ist abhängig von der Bearbeitungsart Längs oder Plan unterschiedlich einzugeben. Soll ein Werkzeug für Längs- und Planbearbeitung eingesetzt werden, müssen bei unterschiedlichen Freischneidwinkeln zwei WZ-Schneiden definiert werden.

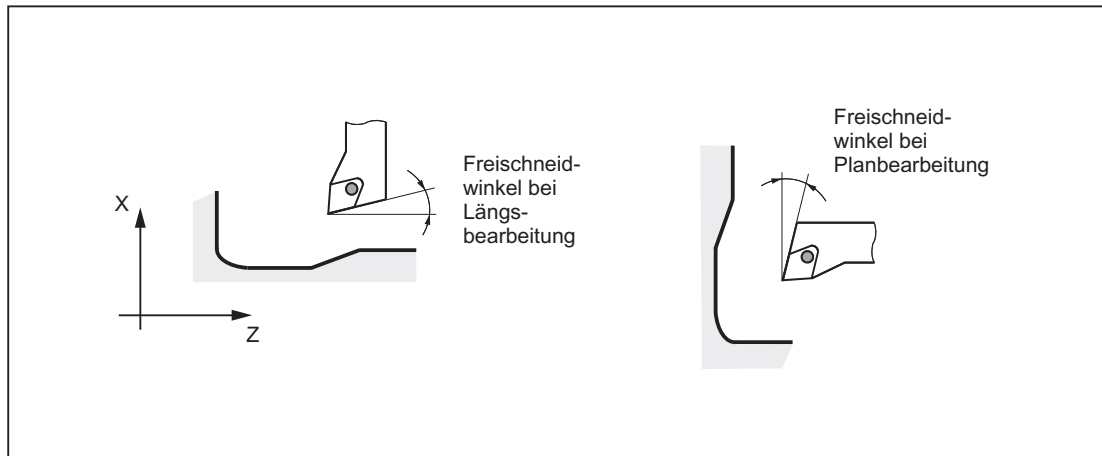


Bild 19-10 Freischneidwinkel für Längs- und Planbearbeitung

Hinweis

Wird der Freischneidwinkel (WZ-Parameter 24) mit Null eingegeben, erfolgt in den Drehzyklen keine Überwachung beim Hinterschneiden.

Literatur

Zur genaueren Beschreibung des Freischneidwinkels siehe:

- Programmierhandbuch Zyklen

19.4.10 Werkzeuge mit relevanter Schneidenlage

Für Werkzeuge mit relevanter Schneidenlage ist Folgendes zu beachten:

- Für die Berechnung von Schnittpunkten mit dem An- bzw. Abfahrsatz wird die Gerade zwischen den Schneidenmittelpunkten am Satzanfang und am Satzende verwendet. Die Differenz zwischen Schneidenbezugspunkt und Schneidenmittelpunkt wird dieser Bewegung überlagert.
Beim An- bzw. Abfahren mit `KONT` erfolgt die Überlagerung im linearen Teilsatz der An- bzw. Abfahrbewegung. Die geometrischen Verhältnisse sind deshalb bei Werkzeugen mit und ohne relevante Schneidenlage identisch.
- Der Wechsel eines Werkzeugs bei aktiver Werkzeugradiuskorrektur, bei dem sich der Abstand zwischen Schneidenmittelpunkt und Schneidenbezugspunkt ändert, ist in Kreissätzen und in Verfahrsätzen mit rationalen Polynomen mit einem Nennergrad > 4 verboten. Bei anderen Interpolationsarten ist ein Wechsel im Gegensatz zum bisherigen Zustand auch bei aktiver Transformation (z. B. `TRANSMIT`) zulässig.
- Bei der Werkzeugradiuskorrektur mit variabler Werkzeugorientierung ist die Transformation vom Schneidenbezugspunkt auf den Schneidenmittelpunkt nicht durch eine einfache Nullpunktverschiebung realisierbar. Werkzeuge mit relevanter Schneidenlage werden deshalb beim 3D-Umfangsfräsen verboten (Alarm).

Hinweis

Für das Stirnfräsen ist das Thema nicht relevant, da hier ohnehin nur definierte Werkzeugtypen ohne relevante Schneidenlage zugelassen sind. (Werkzeuge mit einem nicht ausdrücklich zugelassenen Werkzeugtyp werden als Kugelkopffräser mit dem angegebenen Radius behandelt. Die Angabe einer Schneidenlage wird ignoriert.)

19.5 2D-Werkzeugradiuskorrektur (2D-WRK)

19.5.1 Allgemeine Informationen

Hinweis

Zur Werkzeugradiuskorrektur (WRK) siehe:

Literatur:

Programmierhandbuch Grundlagen

Nur die Programmieranleitung beschreibt die vollständigen programmtechnischen Möglichkeiten der Werkzeugradiuskorrektur (WRK) mit ihren Sonderfällen.

Warum WRK?

Die im Teileprogramm programmierte Kontur (Geometrie) des Werkstücks soll unabhängig von den in der Fertigung eingesetzten Werkzeugen sein. Dazu ist es nötig, die Werte für Werkzeuglänge und Werkzeugradius einem aktuellen Korrekturspeicher zu entnehmen. Durch die WRK kann mit dem aktuellen Werkzeugradius die Äquidistante zur programmierten Kontur in der Ebene bestimmt werden.

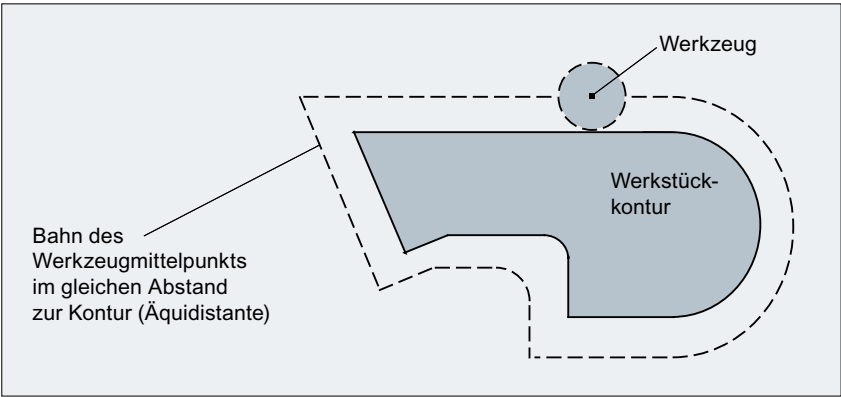


Bild 19-11 Werkstückkontur (Geometrie) mit Äquidistante

WRK in der Ebene

Die WRK wirkt in der aktuellen Ebene (G17 bis G19) bei folgenden Interpolationsarten:

• Linearinterpolation	...	G0, G1
• Kreisinterpolation	...	G2, G3, CIP
• Helikalinterpolation	...	G2, G3
• Splineinterpolation	...	ASPLINE, BSPLINE, CSPLINE
• Polynominterpolation	...	POLY

19.5.2 Anwahl der WRK (G41/G42)

Korrekturrichtung

Die WRK ermittelt eine äquidistante Bahn zur programmierten Kontur. Die Korrektur kann in Bewegungsrichtung links oder rechts von der programmierten Kontur erfolgen.

Befehl	Bedeutung
G41	WRK links von der Kontur in Bewegungsrichtung
G42	WRK rechts von der Kontur in Bewegungsrichtung
G40	Abwahl der WRK

Zwischensätze

Während der aktiven WRK werden in der Regel nur Programmsätze mit Positionen von Geometrieachsen in der aktuellen Ebene programmiert. Trotzdem können bei aktiver WRK auch einzelne Zwischensätze programmiert werden. Zwischensätze sind Programmsätze, die nicht mindestens die Position einer Geometrieachse in der aktuellen Ebene enthalten:

- Positionen in der Zustellachse
- Hilfsfunktionen
- etc.

Die max. Anzahl der Zwischensätze kann vorgegeben werden im Maschinendatum:

MD20250 \$MC_CUTCOM_MAXNUM_DUMMY_BLOCKS (Satzanzahl ohne Verfahrbewegung bei WRK)

Besonderheiten

- Die Anwahl der WRK kann nur in einem Programmsatz mit G0 (Eilgang) oder G1 (Linearinterpolation) erfolgen.
- Spätestens im Programmsatz mit der WRK-Anwahl muss ein Werkzeug eingewechselt (T-Funktion) und die WZ-Schneide/WZK (D1 bis D9) aktiviert werden.
- Bei einer WZ-Schneide/WZK von D0 erfolgt keine WRK-Anwahl.
- Wird bei der WRK-Anwahl nur eine Geometrieachse der Ebene programmiert, wird die zweite Achse der Ebene automatisch ergänzt (letzte progr. Position).
- Wird im Satz mit der WRK-Anwahl keine Geometrieachse der aktuellen Ebene programmiert, findet keine Anwahl statt.
- Wird im Satz nach der WRK-Anwahl die WRK abgewählt (G40), findet keine Anwahl statt.
- Bei Anwahl der WRK wird das Anfahrverhalten durch die Anweisungen NORM/KONT bestimmt.

19.5.3 An- und Abfahrverhalten (NORM/KONT/KONTC/KONTT)

NORM und KONT

Mit den Anweisungen NORM und KONT kann das Anfahrverhalten (Anwahl der WRK mit G41/ G42) und das Abfahrverhalten (Abwahl der WRK mit G40) gesteuert werden:

Befehl	Bedeutung
NORM	Normaleinstellung im Anfangspunkt/Endpunkt (Grundstellung)
KONT	Kontur umfahren im Anfangspunkt/Endpunkt
KONTC	krümmungsstetiges An-/Abfahren
KONTT	tangentenstetiges An-/Abfahren

Besonderheiten

- **KONT** unterscheidet sich von **NORM** nur bei einer WZ-Startposition, die hinter der Kontur liegt.

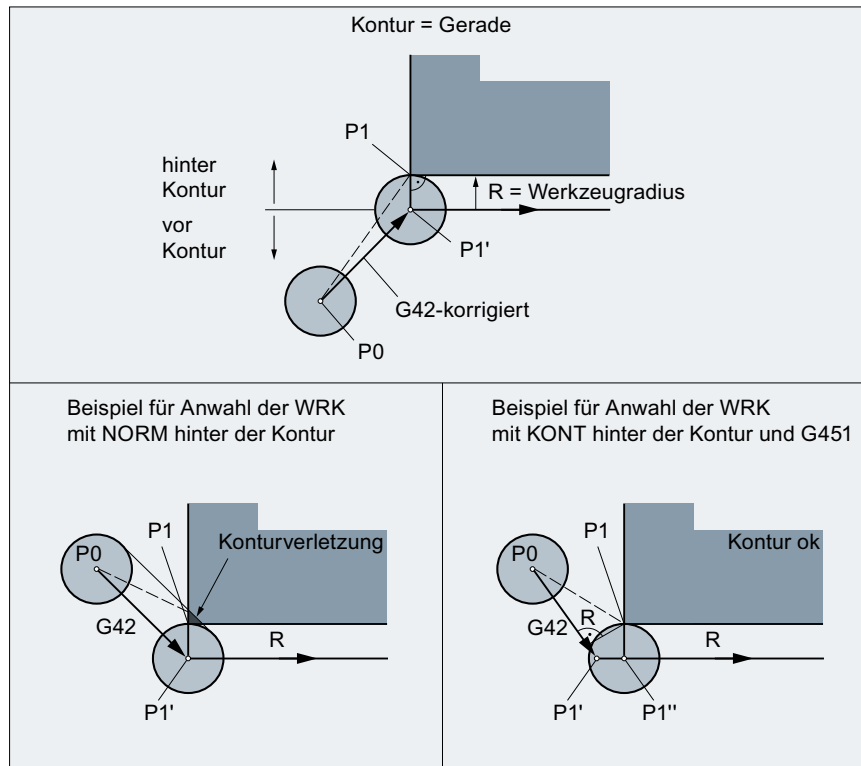


Bild 19-12 Beispiel für Anwahl der WRK mit **KONT** oder **NORM** vor und hinter der Kontur

- **KONT** und **G450/G451** (Eckenverhalten an Außenecken) wirkt gemeinsam und bestimmt so das An- Abfahrverhalten bei WRK.
- Bei Abwahl der WRK wird das Abfahrverhalten durch die Anweisungen **NORM/KONT** bestimmt.

Randbedingungen

Bei den beiden folgenden Varianten sind die An- bzw. Abfahrsätze Polynome. Sie stehen deshalb nur bei Steuerungsvarianten zur Verfügung, bei denen die Polynominterpolation freigegeben ist:

- **KONTT**
Mit **KONTT** wird tangentialstetig an die Kontur angefahren bzw. tangentialstetig von der Kontur abgefahren. Die Krümmungen am Satzübergang sind im Allgemeinen nicht stetig.
- **KONTC**
Mit **KONTC** ist beim Übergang vom Anfahr Satz bzw. zum Abfahr Satz nicht nur die Tangente stetig, sondern auch die Krümmung, so dass in diesem Fall bei der Aktivierung / Deaktivierung **kein Beschleunigungssprung** mehr auftreten kann.

Obwohl **KONTC** die Eigenschaft **KONTT** mit umfasst, ist die tangentialstetige Variante **KONTT** allein verfügbar, weil durch die geforderte Krümmungstetigkeit bei **KONTC** mitunter unerwünschte Konturen entstehen können.

Achsen

Die Stetigkeitsbedingung wird in allen **drei** Achsen eingehalten. Es ist somit zulässig, beim An-/Abfahren gleichzeitig eine Wegkomponente senkrecht zur Korrektorebene zu programmieren.

Für die originalen An- bzw. Abfahrssätze mit KONTT/KONTC sind nur **Linearsätze** zugelassen. Diese programmierten Linearsätze werden in der Steuerung durch die entsprechenden Polynomkurven ersetzt.

Ausschluss

KONTT und KONTC stehen bei den 3D-Varianten der Werkzeugradiuskorrektur (CUT3DC, CUT3DCC, CUT3DF) nicht zur Verfügung.

Werden sie dennoch programmiert, wird steuerungsintern ohne Fehlermeldung auf NORM umgeschaltet.

Beispiel KONTC

In den beiden folgenden Bildern ist eine typische Anwendung für das krümmungsstetige An- und Abfahren dargestellt:

In der Kreismitte beginnend wird an den Vollkreis angefahren. Dabei sind im Satzendpunkt des Anfahrssatzes dessen Richtung und dessen Krümmungsradius gleich den Werten des folgenden Kreises. In den beiden An-/Abfahrssätzen wird gleichzeitig in Z-Richtung zugestellt.

Das zugehörige NC-Programmsegment sieht folgendermaßen aus:

\$TC_DP1[1,1]=121	Fräser
\$TC_DP6[1,1]=10	Radius 10 mm
N10 G1 X0 Y0 Z60 G64 T1 D1 F10000	
N20 G41 KONTC X70 Y0 Z0	
N30 G2 I-70	Vollkreis
N40 G40 G1 X0 Y0 Z60	
N50 M30	

Erklärung:

In diesem Beispiel wird ein Vollkreis mit dem Radius 70 mm in der X-Y-Ebene bearbeitet. Da das Werkzeug einen Radius von 10 mm hat, ergibt sich als resultierende Werkzeugmittelpunktsbahn ein Kreis mit dem Radius 60 mm. Start-/ Endpunkt liegen bei X0 Y0 Z60, so dass gleichzeitig zur An-/Abfahrbewegung in der Korrektorebene eine Bewegung in Z-Richtung ausgeführt wird.

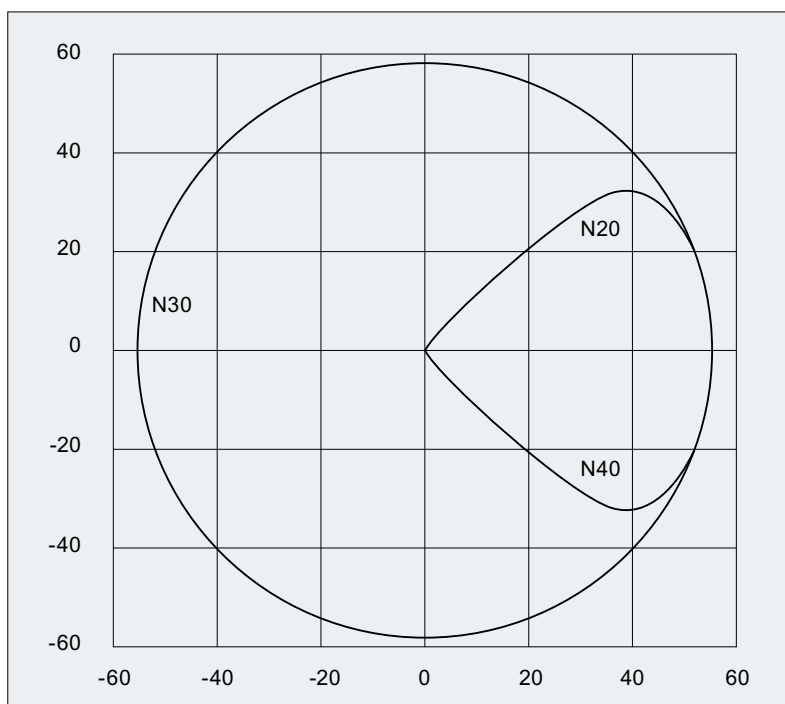


Bild 19-13 Krümmungsstetiges An- und Abfahren bei der Innenbearbeitung eines Vollkreises:
Projektion in die X-Y-Ebene

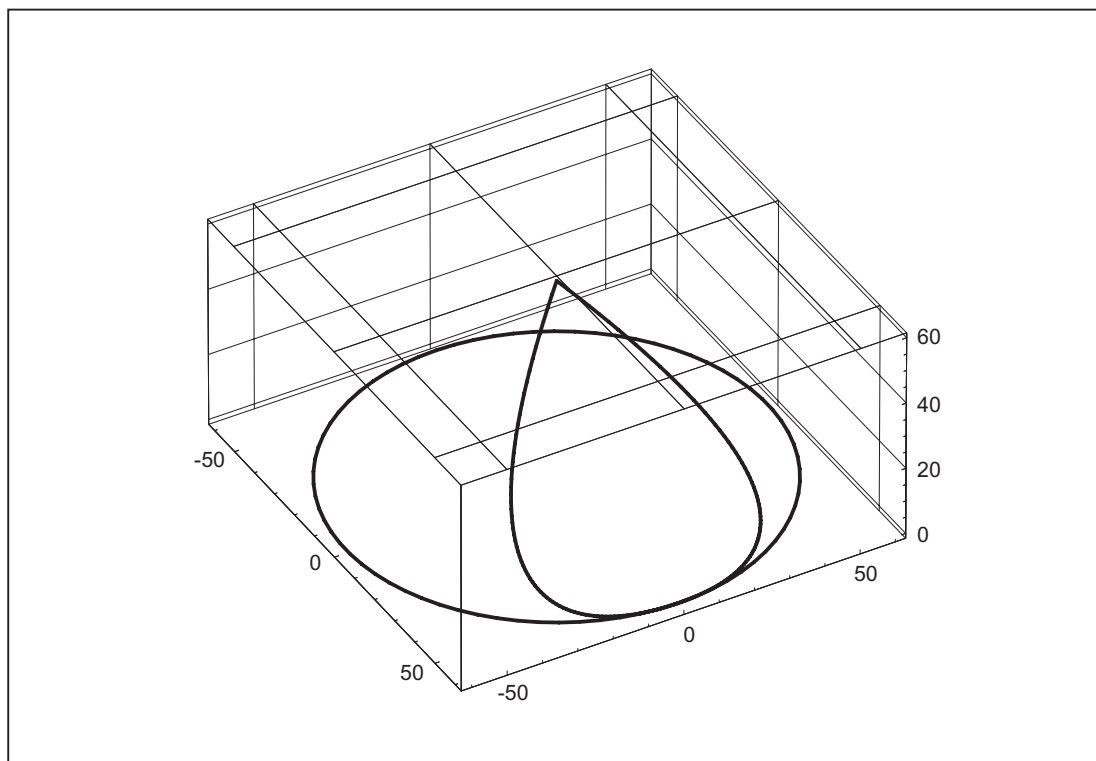


Bild 19-14 Krümmungsstetiges An- und Abfahren bei der Innenbearbeitung eines Vollkreises:
Räumliche Darstellung

KONTT und KONTC im Vergleich

Im folgenden Bild ist das unterschiedliche An-/Abfahrverhalten bei **KONTT** und **KONTC** dargestellt. Ein Kreis mit dem Radius 20 mm um den Mittelpunkt bei X0 Y-40 wird mit einem Werkzeug mit 20 mm Radius an der Außenseite korrigiert. Es ergibt sich deshalb eine kreisförmige Bewegung des Werkzeugmittelpunkts mit dem Radius 40 mm. Der Endpunkt des Abfahrsetzes liegt bei X40 Y30. Der Übergang zwischen dem Kreissatz und dem Abfahrsetz liegt im Nullpunkt. Wegen der verlangten Krümmungsstetigkeit bei **KONTC** führt der Abfahrsetz zunächst eine Bewegung mit negativer Y-Komponente aus. Dies wird häufig unerwünscht sein. Der Abfahrsetz mit **KONTT** zeigt dieses Verhalten nicht. Allerdings tritt in diesem Fall am Satzübergang ein Beschleunigungssprung auf.

Ist der **KONTT**- bzw. **KONTC**-Satz nicht der Ab- sondern der Anfahrsetz, ergibt sich exakt die gleiche Kontur, die lediglich in umgekehrter Richtung durchlaufen wird, d.h. **An- und Abfahrverhalten sind symmetrisch**.

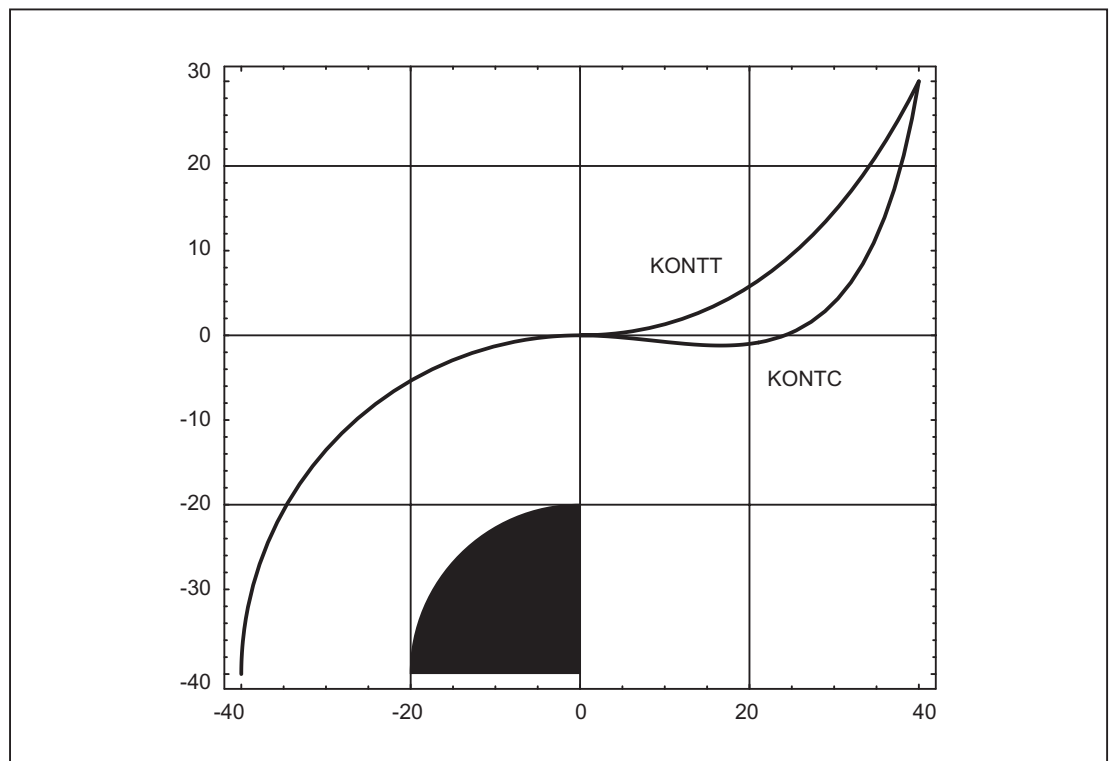


Bild 19-15 Unterschiede von **KONTT** und **KONTC**

Hinweis

Aus dem Bild ist ersichtlich, dass eine an den Viertelkreis der Kontur anschließende Gerade z. B. bis X20 Y-20 bei Ab-/Anfahren an X0, Y0 mit **KONTC** eine Konturverletzung zur Folge hätte.

19.5.4 Weiches An- und Abfahren

19.5.4.1 Funktion

Bedeutung

Die Funktion "Weiches An- und Abfahren (WAB)" dient dazu, im Startpunkt einer Kontur unabhängig von der Lage des Ausgangspunktes tangential anzufahren.

Das Anfahrverhalten kann dabei durch eine Reihe von ergänzenden Parametern variiert und speziellen Erfordernissen angepasst werden.

Die beiden Funktionen "Weiches Anfahren" und "Weiches Abfahren" sind weitgehend symmetrisch zueinander. Im Folgenden wird deshalb nur das Anfahren ausführlich beschrieben und auf Abweichungen beim Abfahren hingewiesen.

Teilbewegungen

Es gibt beim weichen An- und Abfahren maximal 4 Teilbewegungen mit folgenden Positionen:

- Startpunkt der Bewegung P_0
- Zwischenpunkte P_1 , P_2 , und P_3
- Endpunkt P_4

Die Punkte P_0 , P_3 und P_4 sind immer definiert. Die Zwischenpunkte P_1 und P_2 können je nach Parametrierung und den geometrischen Verhältnissen entfallen.

Beim Abfahren werden die Punkte in der umgekehrten Reihenfolge, d. h. beginnend mit P_4 und endend mit P_0 durchlaufen.

19.5.4.2 Parameter

Das Verhalten der Funktion weiches An- und Abfahren wird durch maximal 9 Parameter bestimmt.

Satzweise wirksamer G-Befehl zur Bestimmung der An- bzw. Abfahrkontur

Dieser G-Befehl kann nicht entfallen.

- G147: Anfahren mit einer Geraden
- G148: Abfahren mit einer Geraden
- G247: Anfahren mit einem Viertelkreis
- G248: Abfahren mit einem Viertelkreis

- G347: Anfahren mit einem Halbkreis
- G348: Abfahren mit einem Halbkreis

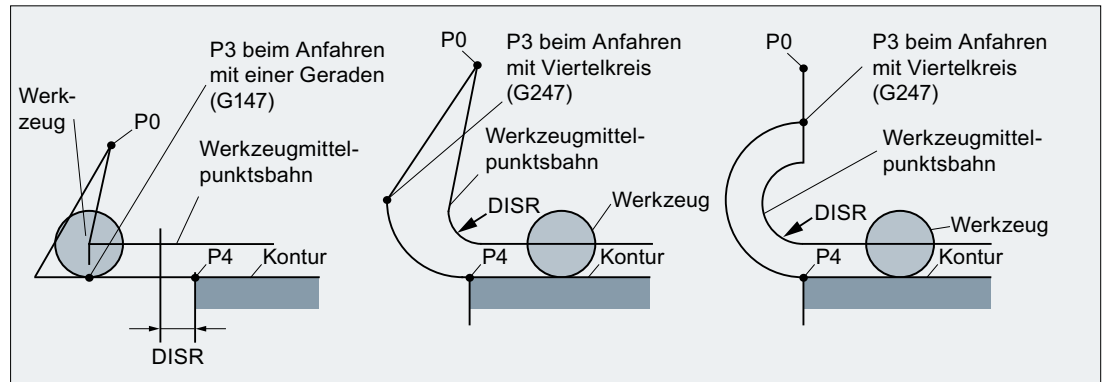


Bild 19-16 Anfahrverhalten in Abhängigkeit von G147 bis G347 und DISR (bei gleichzeitiger Aktivierung der Werkzeugradiuskorrektur)

Modaler G-Befehl zur Bestimmung der An- bzw. Abfahrrihtung

Dieser G-Befehl ist nur dann von Bedeutung, wenn die Anfahrkontur ein Viertel- oder ein Halbkreis ist. Die An- bzw. Abfahrrihtung kann wie folgt bestimmt werden:

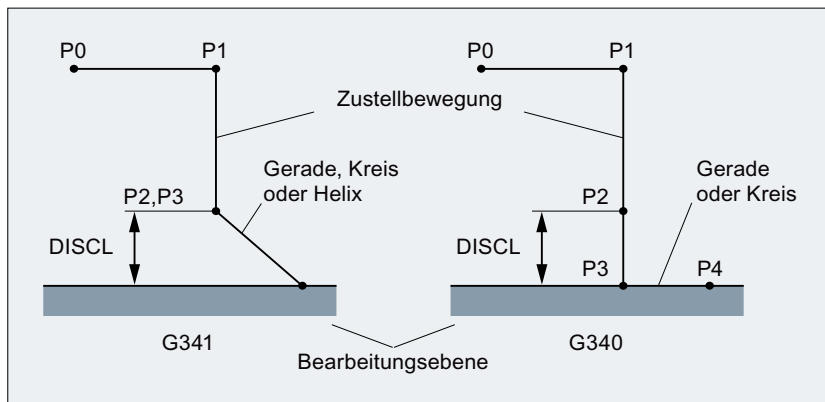
- G140:
Bestimmung der An- bzw. Abfahrrihtung mit Hilfe der aktiven Werkzeugradiuskorrektur. (G140 ist Grundstellungswert.)
Bei positivem Werkzeugradius:
 - G41 aktiv → anfahren von links
 - G42 aktiv → anfahren von rechts
Ist keine Werkzeugradiuskorrektur aktiv (G40), ist das Verhalten wie bei G143. Ein Alarm wird in diesem Fall nicht ausgegeben. Ist der Radius des aktiven Werkzeugs 0, so wird die Anfahr- bzw. Abfahrseite so bestimmt, als wäre der Werkzeugradius positiv.
- G141:
Anfahren von links an die Kontur bzw. nach links von der Kontur wegfahren.

- G142:
Anfahren von rechts an die Kontur bzw. nach rechts von der Kontur wegfahren.
- G143:
Automatische Bestimmung der Anfahrriechtung, d. h. es wird von der Seite aus an die Kontur angefahren, auf der der Startpunkt relativ zur Tangente im Startpunkt des Folgesatzes (P_4) liegt.

Hinweis

Beim **Abfahren** wird entsprechend die Tangente im Endpunkt des Vorgängersatzes verwendet. Ist beim Abfahren der Endpunkt nicht programmiert, d. h. er soll implizit bestimmt werden, so ist G143 beim Abfahren nicht zulässig, da zwischen Anfahrseite und Lage des Endpunktes eine wechselseitige Abhängigkeit besteht. Wird G143 in diesem Fall dennoch programmiert, wird ein Alarm ausgegeben. Gleiches gilt, wenn bei aktivem G140 wegen nicht aktiver Werkzeugradiuskorrektur automatisch auf das G143-Verhalten umgeschaltet wird.

Modaler G-Befehl (G340, G341), der die Aufteilung der Bewegung vom Start- zum Endpunkt auf die einzelnen Sätze definiert



G34 Das charakteristische Anfahren von P_0 bis P_4 ist im Bild dargestellt.

0: Ist G247 oder G347 aktiv (Viertel- oder Halbkreis) und der Startpunkt P_3 liegt nicht in der durch den Endpunkt P_4 definierten Bearbeitungsebene, wird statt eines Kreises eine Helix eingefügt. Der Punkt P_2 ist nicht definiert bzw. er fällt mit P_3 zusammen.

Die Kreisebene bzw. die Helixachse wird dabei durch die im WAB-Satz aktive Ebene (G17 - G19) bestimmt, d. h. vom Folgesatz wird nicht die Starttangente selbst zur Bestimmung des Kreises verwendet, sondern deren Projektion in die aktive Ebene.

Die Bewegung vom Punkt P_0 zum Punkt P_3 erfolgt in zwei Geraden mit der Geschwindigkeit, die vor dem WAB-Satz wirksam war.

G34 Das charakteristische Anfahren von P_0 bis P_4 ist im Bild dargestellt.

1: P_3 und P_4 liegen in der Bearbeitungsebene, so dass bei G247 bzw. G347 nie eine Helix, sondern immer ein Kreis eingefügt wird.

Bild 19-17 Ablauf der Anfahrbewegung abhängig von G340/G341

Hinweis

In allen Fällen, in denen die Lage der aktiven Ebene G17 - G19 eingeht (Kreisebene, Helixachse, Zustellbewegungen senkrecht zur aktiven Ebene), wird ein eventuell aktiver drehender Frame berücksichtigt.

DISR

DISR gibt die Länge der Anfahngeraden bzw. den Radius bei Anfahrkreisen an.

An-/Abfahren mit Geraden

Bei An-/Abfahren mit Geraden gibt DISR den Abstand der Fräserkante vom Startpunkt der Kontur an, d. h. die Länge der Geraden ergibt sich bei aktiver WRK als Summe von Werkzeugradius und programmierten Wert von DISR.

Ein Alarm wird ausgegeben:

- wenn DISR negativ ist und der Betrag größer ist als der Werkzeugradius (die Länge der resultierenden Anfahngeraden ist kleiner oder gleich null).

An-/Abfahren mit Kreisen

Bei An-/Abfahren mit Kreisen gibt DISR immer den Radius der Werkzeugmittelpunktsbahn an. Ist Werkzeugradiuskorrektur aktiviert, wird intern ein Kreis mit einem solchen Radius erzeugt, dass auch in diesem Fall die Werkzeugmittelpunktsbahn aus dem programmierten Radius resultiert.

Ein Alarm wird ausgegeben beim An- bzw. Abfahren mit Kreisen:

- wenn der Radius des intern erzeugten Kreises Null oder negativ ist.
- wenn DISR nicht programmiert ist.
- wenn der Radius einen Wert ≤ 0 hat.

DISCL

DISCL gibt den Abstand des Punktes P2 von der Bearbeitungsebene an.

Soll die Position des Punktes P₂ auf der Achse senkrecht zur Kreisebene absolut angegeben werden, ist der Wert in der Form DISCL = AC(...) zu programmieren.

Ist DISCL nicht programmiert, so sind die Punkte P₁, P₂ und P₃ bei G340 identisch und die Anfahrkontur wird von P₁ zu P₄ gebildet.

Es wird überwacht, dass der durch DISCL definierte Punkt zwischen P₁ und P₃ liegt, d. h. bei allen Bewegungen, die eine Komponente senkrecht zur Bearbeitungsebene haben (Zustellbewegungen, Anfahrbewegung von P₃ nach P₄), muss diese Komponente das gleiche Vorzeichen haben. Eine Richtungsumkehr ist nicht zugelassen. Gegebenenfalls wird ein Alarm ausgegeben.

Bei der Erkennung der Richtungsumkehr wird eine Toleranz zugelassen, definiert durch das Maschinendatum:

MD20204 \$MC_WAB_CLEARANCE_TOLERANCE (Richtungsumkehr bei WAB)

Programmcode	Kommentar
N10 G90 G0 X0 Y0 Z30 D1 T1	
N20 X10	
N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000	
N40 G1 X40 Y-10	
N50 G1 X50	
...	
...	

N30/N40 kann ersetzt werden durch:

N30 G41 G147 DISCL=3 DISR=13 X40 Y-10 Z0 F1000

oder:

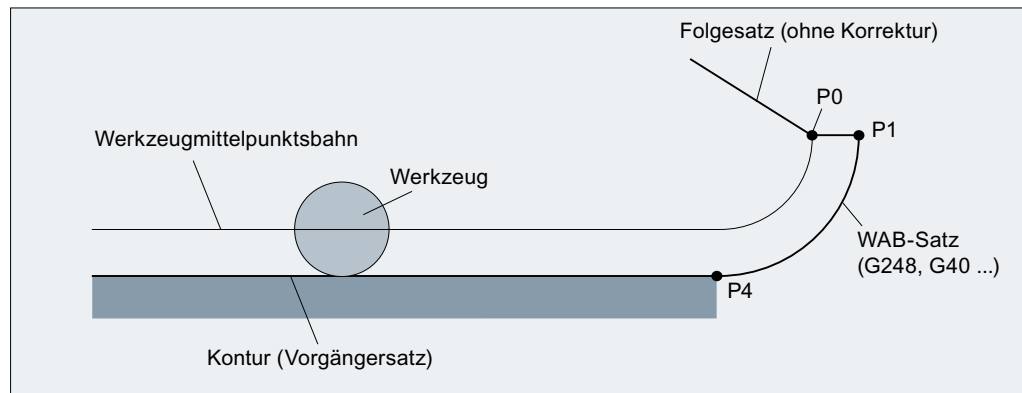
N30 G41 G147 DISCL=3 DISR=13 F1000

N40 G1 X40 Y-10 Z0

Mögliche Programmierung des Endpunktes P_0 beim Abfahren

Die Endposition wird immer aus dem WAB-Satz selbst entnommen, unabhängig wie viele Achsen programmiert wurden. Es sind folgende Fälle zu unterscheiden:

1. Im WAB-Satz ist keine Geometrieachse programmiert.
Die Kontur endet in diesem Fall im Punkt P_2 (bzw. im Punkt P_1 wobei P_1 und P_2 zusammenfallen). Die Position in den Achsen, die die Bearbeitungsebene bilden, ergibt sich aus der Wegfahrkontur (Endpunkt der Geraden bzw. des Kreises). Die Achskomponente senkrecht dazu wird durch DISCL definiert. Ist in diesem Fall $\text{DISCL} = 0$, verläuft die Bewegung deshalb vollständig in der Ebene.
2. Im WAB-Satz ist nur die Achse senkrecht zur Bearbeitungsebene programmiert.
Die Kontur endet in diesem Fall im Punkt P_1 . Die Position der übrigen beiden Achsen ergibt sich wie unter 1.



Abfahren mit WAB bei gleichzeitiger Deaktivierung der WRK

Ist der WAB-Abfahrsatz gleichzeitig der Deaktivierungssatz der Werkzeugradiuskorrektur, so wird in 1. und 2. ein zusätzlicher Weg von P_1 nach P_0 derart eingefügt, dass sich bei der Deaktivierung der Werkzeugradiuskorrektur am Ende der Wegfahrkontur keine Bewegung ergibt, d. h. dieser Punkt definiert dann nicht eine Position auf einer zu korrigierenden Kontur, sondern den Werkzeugmittelpunkt.

3. Es ist mindestens eine Achse der Bearbeitungsebene programmiert.
Die eventuell fehlende zweite Achse der Bearbeitungsebene wird aus ihrer letzten Position im Vorgängersatz modal ergänzt. Die Position der Achse senkrecht zur Bearbeitungsebene wird - abhängig davon, ob diese Achse programmiert ist oder nicht - wie unter 1. oder 2. gebildet. Die so gebildete Position definiert den Endpunkt P_0 .
Die Abwahl der Werkzeugradiuskorrektur muss nicht gesondert behandelt werden, da der programmierte Punkt P_0 bereits unmittelbar die Position des Werkzeugmittelpunktes am Ende der Gesamtkontur definiert.
Start- und Endpunkt der WAB-Kontur (P_0 bzw. P_4) dürfen sowohl beim An- als auch beim Abfahren zusammenfallen.

Geschwindigkeit des Vorgängersatzes (typisch G0)

Mit dieser Geschwindigkeit werden alle Bewegungen vom Punkt P_0 bis zum Punkt P_2 ausgeführt, d. h. die Bewegung parallel zur Bearbeitungsebene und der Teil der Zustellbewegung bis zum Sicherheitsabstand.

Programmierung der Vorschubgeschwindigkeit mit FAD

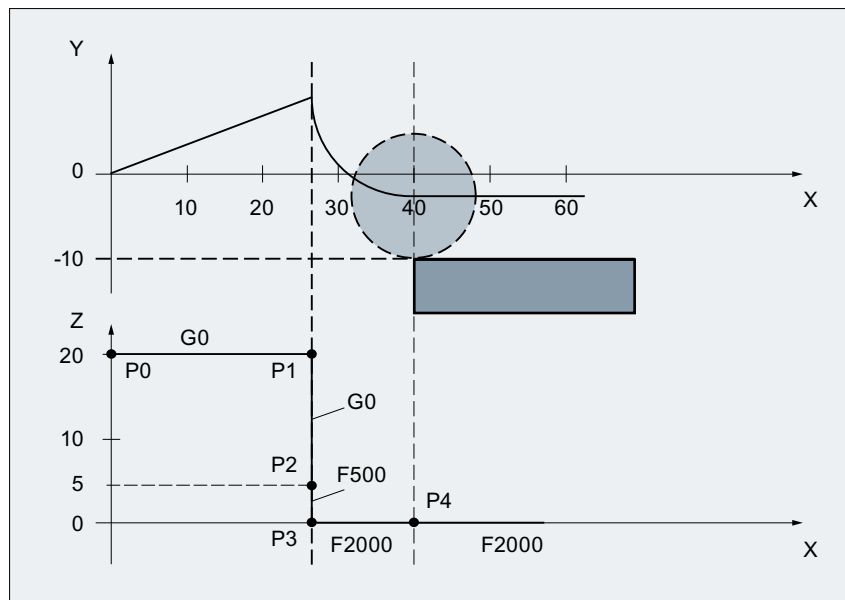
FAD programmiert bei ...	
G340	Vorschubgeschwindigkeit von P_2 bzw. P_3 zu P_4 .
G341	Vorschubgeschwindigkeit der Zustellbewegung senkrecht zur Bearbeitungsebene von P_2 nach P_3 .

Wird **FAD** nicht programmiert, wird dieser Teil der Kontur mit der modal wirksamen Geschwindigkeit des Vorgängersatzes verfahren, falls im WAB-Satz kein F-Wort programmiert ist, das die Geschwindigkeit festlegt.

Programmierverhalten:

- FAD=0 oder negativ → Alarmausgabe
- FAD=... → programmierter Wert wirkt entsprechend dem aktiven G-Befehl der Gruppe 15 (Vorschubtyp; G93, G94 usw.)
- FAD=PM (...) → programmierter Wert wird unabhängig vom aktiven G-Befehl der Gruppe 15 als Linearvorschub (wie G94) interpretiert
- FAD=PR (...) → programmierter Wert wird unabhängig vom aktiven G-Befehl der Gruppe 15 als Umdrehungsvorschub (wie G95) interpretiert

Beispiel:



Programmcode	Kommentar
\$TC_DP1[1,1]=120	; Fräserwerkzeug T1/D1
\$TC_DP6[1,1]=7	; Werkzeug mit 7mm Radius
N10 G90 G0 X0 Y0 Z20 D1 T1	
N20 G41 G341 G247 DISCL=AC(5) DISR=13FAD 500 X40 Y-10 Z=0 F2000	
N30 X50	
N40 X60	

Programmcode	Kommentar
...	

Programmierung Vorschub F

Dieser Vorschubwert ist ab dem Punkt P₃ (bzw. ab dem Punkt P₂, falls FAD nicht programmiert ist) wirksam. Wird im WAB-Satz kein F-Wort programmiert, so wirkt die Geschwindigkeit des Vorgängersatzes. Die durch FAD definierte Geschwindigkeit wird für Folgesätze nicht übernommen.

19.5.4.3 Geschwindigkeiten

Geschwindigkeiten beim Anfahren

In den beiden folgenden Darstellungen für das Anfahren wird davon ausgegangen, dass in dem auf den WAB-Satz folgenden Satz keine neue Geschwindigkeit programmiert wurde. Ist dies nicht der Fall, wird ab dem Punkt P₄ die neue Geschwindigkeit wirksam.

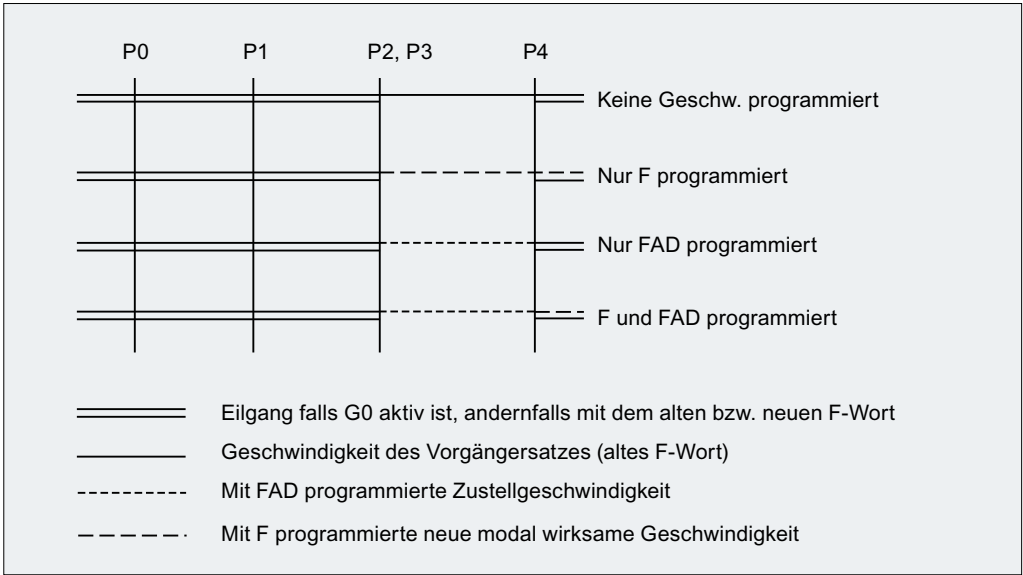


Bild 19-18 Geschwindigkeiten in den WAB-Teilsätzen beim Anfahren mit G340

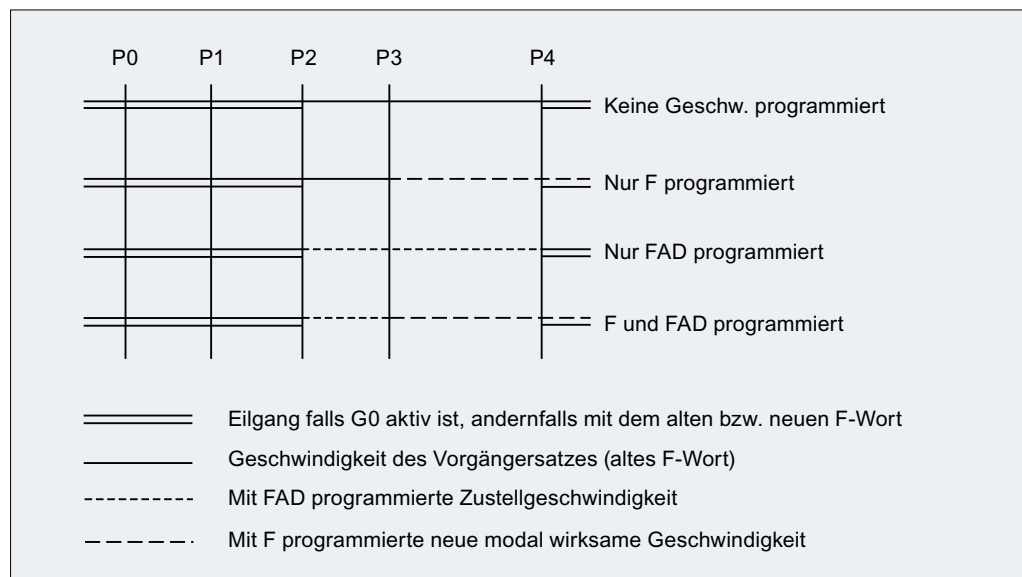


Bild 19-19 Geschwindigkeiten in den WAB-Teilsätzen beim Anfahren mit G341

Geschwindigkeiten beim Abfahren

Beim Abfahren sind die Rollen von modal wirksamem Vorschub aus dem Vorgängersatz und dem im WAB-Satz programmierten Vorschubwert vertauscht, d. h. die eigentliche Abfahrkontur (Gerade, Kreis, Helix) wird mit dem alten Vorschub verfahren, eine neu mit F-Wort programmierte Geschwindigkeit gilt entsprechend ab dem Punkt P_2 bis zum Punkt P_0 .

Ist ebenes Wegfahren aktiv und FAD ist programmiert, wird die Strecke P_3 nach P_2 mit FAD verfahren, andernfalls mit der alten Geschwindigkeit. Für die Strecke P_4 nach P_2 gilt immer das letzte in einem Vorgängersatz programmierte F-Wort. G0 wird in diesen Sätzen nicht wirksam.

Von P_2 nach P_0 wird mit dem im WAB-Satz programmierten F-Wort verfahren bzw., falls kein F-Wort programmiert wurde, mit dem modal wirksamen F-Wort aus einem Vorgängersatz. Dies gilt unter der Bedingung, dass G0 nicht aktiv ist.

Sollen beim Abfahren die Sätze von P_2 nach P_0 mit Eilgang verfahren werden, so muss G0 vor dem WAB-Satz oder im WAB-Satz selbst aktiviert werden. Ein eventuell zusätzlich programmiertes F-Wort hat dann in den WAB-Sätzen selbst keine Bedeutung mehr. Es bleibt jedoch modal für Folgesätze wirksam.

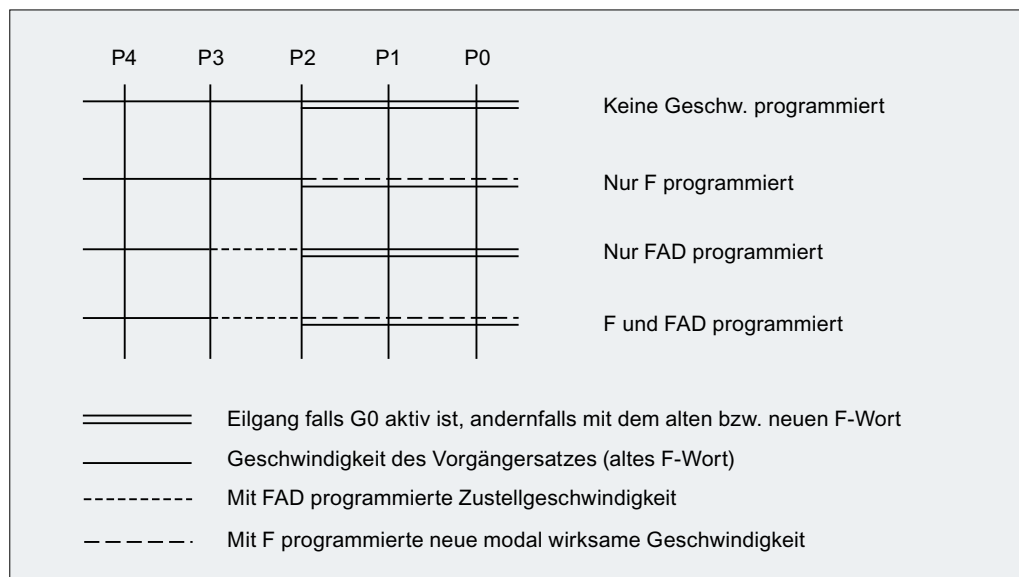


Bild 19-20 Geschwindigkeiten in den WAB-Teilsätzen beim Abfahren

19.5.4.4 Systemvariablen

Die Punkte P_3 und P_4 können beim Anfahren als Systemvariablen im WKS gelesen werden:

\$P_APR:	Lesen von P_3 (Aufstartpunkt) im WKS	
\$P_AEP:	Lesen von P_4 (Konturanfangspunkt) im WKS	
\$P_APDV	=1	Wenn der Inhalt von \$P_APR und \$P_AEP gültig ist, d. h. wenn diese die zum letzten programmierten WAB-Anfahrtsatz gehörenden Positionswerte enthalten.
	=0	Die Positionen eines älteren WAB-Anfahrtsatzes werden gelesen.

Eine Änderung des WKS zwischen WAB-Satz und Leseoperation hat keinen Einfluss auf die Positionswerte.

19.5.4.5 Randbedingungen

Randbedingungen

- In einem WAB-Satz können beliebige weitere NC-Befehle programmiert werden (z. B. Hilfsfunktionsausgaben, Synchronachsbewegungen, Positionierachsbewegungen usw.). Diese werden beim Anfahren im ersten und beim Abfahren im letzten Teilsatz ausgeführt.
- Wird der Endpunkt P_4 nicht aus dem WAB-Satz, sondern aus einem nachfolgenden Verfahrtsatz entnommen, so wird die eigentliche WAB-Kontur (Gerade, Viertel- oder Halbkreis) in diesem Satz abgefahren. Der letzte Teilsatz des ursprünglichen WAB-Satzes enthält dann keine Verfahrinformation für die Geometrieachsen. Er wird aber dennoch stets ausgegeben, da in diesem Satz eventuell weitere Aktionen (z. B. Einzelachsen) ausgeführt werden müssen.

- Es müssen immer mindestens zwei Sätze betrachtet werden:
 - der WAB-Satz selbst
 - der Satz, der die Anfahr- bzw. Abfahrriechtung definiert

Zwischen diesen beiden Sätzen können weitere Sätze programmiert werden.
Die Anzahl möglicher Zwischensätze wird begrenzt mit dem Maschinendatum:
MD20202 \$MC_WAB_MAXNUM_DUMMY_BLOCKS (Max. Satzanzahl ohne
Verfahrbewegung bei WAB)
- Wird in einem Anfahrstanz gleichzeitig die Werkzeugradiuskorrektur aktiviert, so ist der erste Geradensatz der WAB-Kontur der Aktivierungssatz.
Die gesamte Kontur, die von der WAB-Funktion erzeugt wurde, wird von der Werkzeugradiuskorrektur so behandelt, als ob sie explizit programmiert worden wäre (Kollisionsüberwachungen, Schnittpunktberechnungen, Anfahrverhalten *NORM / KONT*).
- Die Richtung der Zustellbewegung und die Lage der Kreisebene bzw. der Helixachse wird ausschließlich durch die aktive Ebene (G17 - G19) - gegebenenfalls gedreht mit einem aktiven Frame - bestimmt.
- Beim Anfahren darf zwischen dem WAB-Satz und dem Folgesatz, der die Tangentenrichtung definiert, kein Vorlaufstopp eingefügt werden.
Ein Vorlaufstopp in diesem Fall - gleichgültig ob explizit programmiert oder von der Steuerung automatisch eingefügt - führt zu einem Alarm.

Verhalten bei REPOS

Wird ein WAB-Zyklus unterbrochen und repositioniert, so wird bei *RMIBL* an der Unterbrechungsstelle wieder aufgesetzt. Der Aufsetzpunkt bei *RMEBL* ist der Endpunkt des letzten WAB-Satzes, bei *RMBBL* entsprechend der Startpunkt des ersten WAB-Satzes.

Wird *RMIBL* zusammen mit *DISPR* (Wiederanfahren im Abstand *DISPR* vor dem Unterbrechungspunkt) programmiert, kann der Wiederanfahrpunkt in einem Teilsatz des WAB-Zyklus vor dem Unterbrechungsteilsatz liegen.

19.5.4.6 Beispiele

Beispiel 1

Folgende Bedingungen sind gegeben:

- Weiches Anfahren wird in Satz *N20* aktiviert
- X=40 (Endpunkt); Y=0; Z=0
- Anfahrbewegung erfolgt mit Viertelkreis (G247)
- Anfahrriechtung nicht programmiert, es wirkt G140, d.h. da WRK aktiv (G42) und Korrekturwert positiv ist (10), wird Kontur von rechts angefahren
- Intern erzeugte Anfahrkreis (WAB-Kontur) hat Radius 20, so dass der Radius der Werkzeugmittelpunktsbahn gleich dem programmierten Wert *DISR=10* ist

- Die Anfahrbewegung erfolgt wegen G341 mit einem Kreis in der Ebene, so dass sich der Startpunkt (20, -20, 0) ergibt
- Der Punkt P2 hat wegen DISCL=5 die Position (20, -20, 5) und der Punkt P1 liegt wegen Z30 in N10 bei (20, -20, 30)

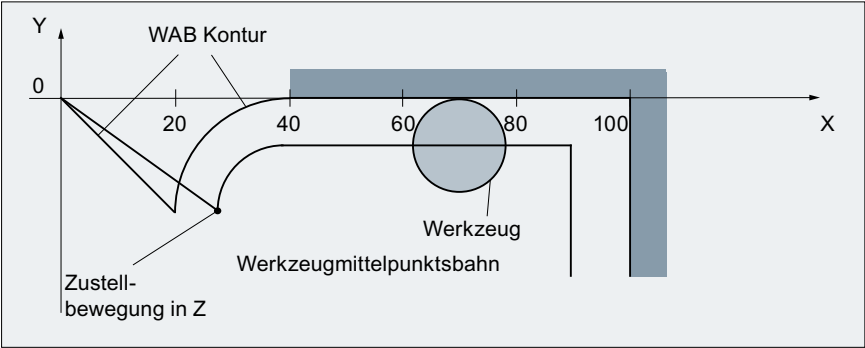


Bild 19-21 Kontur Beispiel 1

Teileprogramm:

Programmcode	Kommentar
\$TC_DP1[1,1]=120	; Werkzeugdefinition T1/D1
\$TC_DP6[1,1]=10	; Radius
N10 G0 X0 Y0 Z30	
N20 G247 G341 G42 NORM D1 T1 Z0 FAD=1000 F=2000 DISCL=5 DISR=10	
N30 X40	
N40 X100	
N50 Y-30	
...	

Beispiel 2

Folgende Bedingungen sind beim Anfahren gegeben:

- Weiches Anfahren wird in Satz N20 aktiviert
- Anfahrbewegung erfolgt mit Viertelkreis (G247)
- Anfahrrichtung nicht programmiert, es wirkt G140, d. h. da WRK aktiv (G41) ist, wird Kontur von links angefahren
- Konturoffset OFFN=5 (N10)
- aktueller Werkzeugradius=10, damit ist der effektive Korrekturradius für WRK=15; der Radius der WAB-Kontur ist damit gleich 25, so dass der Radius der Werkzeugmittelpunktsbahn gleich DISR=10 wird

- Endpunkt des Kreises ergibt sich aus N30, da in N20 nur Z-Position programmiert ist
- Zustellbewegung
 - von Z20 nach Z7 (DISCL=AC(7)) im Eilgang
 - anschließend nach Z0 mit FAD=200
 - Anfahrkreis in X-Y-Ebene und Folgesätze mit F1500
(Damit diese Geschwindigkeit in den Folgesätzen wirksam wird, muss der aktive G-Code G0 in N30 mit G1 überschrieben werden. Andernfalls würde die Kontur mit G0 weiter bearbeitet werden.)

Folgende Bedingungen sind beim Abfahren gegeben:

- Weiches Abfahren wird in Satz N60 aktiviert
- Abfahrbewegung erfolgt mit Viertelkreis (G248) und Helix (G340)
- FAD nicht programmiert, da bei G340 ohne Bedeutung
- Z=2 im Startpunkt; Z=8 im Endpunkt, da DISCL=6
- Bei DISR=5 ist Radius der WAB-Kontur=20, der der Werkzeugmittelpunktsbahn=5
- Nach dem Kreissatz folgen die Wegfahrbewegung von Z8 nach Z20 und die Bewegung parallel zur X-Y-Ebene zum Endpunkt bei X70 Y0

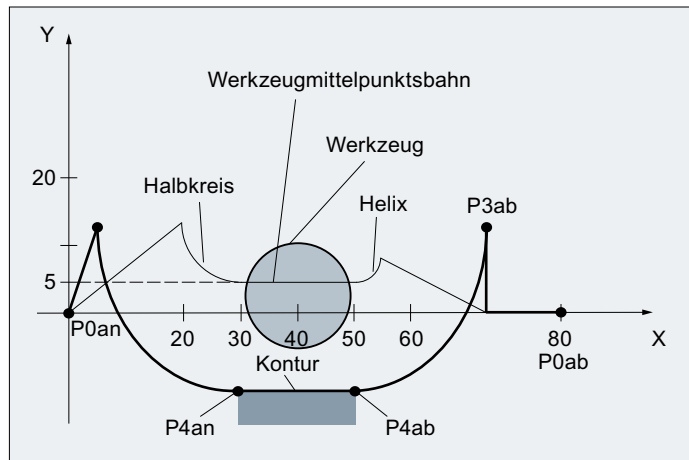


Bild 19-22 Kontur Beispiel 2

Teileprogramm:

Programmcode	Kommentar
\$TC_DP1[1,1]=120	; Werkzeugdefinition
\$TC_DP6[1,1]=10	T1/D1
	; Radius
N10 G0 X0 Y0 Z20 G64 D1 T1 OFFN = 5	; (P0an)
N20 G41 G247 G341 Z0 DISCL = AC(7) DISR = 10 F1500 FAD=200	; (P3an)
N30 G1 X30 Y-10	; (P4an)
N40 X40 Z2	

Programmcode	Kommentar
N50 X50	; (P4ab)
N60 G248 G340 X70 Y0 Z20 DISCL = 6 DISR = 5 G40 F10000	; (P3ab)
N70 X80 Y0	; (P0ab)
N80 M 30	

Hinweis

Die derart erzeugte Kontur wird durch die Werkzeugradiuskorrektur modifiziert, die im WAB-Anfahrsatz aktiviert und im WAB-Abfahrsatz deaktiviert wird.

Die Werkzeugradiuskorrektur berücksichtigt einen effektiven Radius von 15, der sich aus der Summe von Werkzeugradius (10) und Konturoffset (5) zusammensetzt. Der resultierende Radius der Werkzeugmittelpunktsbahn im Anfahrsatz ist deshalb 10, im Abfahrsatz 5.

19.5.5 Abwahl der WRK (G40)

Anweisung G40

Die Abwahl der WRK erfolgt mit der Anweisung G40.

Besonderheiten

- Die Abwahl der WRK kann nur in einem Programmsatz mit G0 (Eilgang) oder G1 (Linearinterpolation) erfolgen.
- Wird bei aktiver WRK D0 programmiert, dann erfolgt keine Abwahl, sondern die Fehlermeldung 10750.
- Wird im Satz mit der WRK-Abwahl eine Geometrieachse programmiert, findet eine Abwahl der WRK statt, auch wenn diese nicht in der aktuellen Ebene liegt.

19.5.6 Korrektur an Außenecken

G450/G451

Mit den G-Befehlen G450/G451 kann das Verhalten bei unstetigen Satzübergängen an Außenecken gesteuert werden:

Befehl	Bedeutung
G450	Unstetige Satzübergänge mit Übergangskreis
G451	Unstetige Satzübergänge mit Schnittpunkt der Äquidistanten

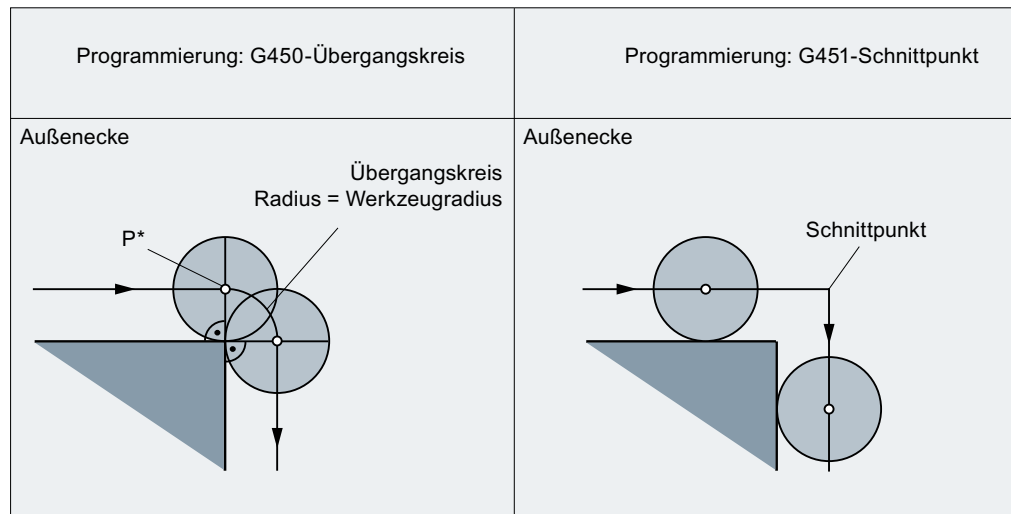


Bild 19-23 Beispiel für 90°-Außenecke mit G450 und G451

G450 (Übergangskreis)

Bei aktivem G-Befehl **G450** vollzieht der Mittelpunkt des Werkzeugs bei Außenecken eine Kreisbahn mit dem Werkzeugradius. Die Kreisbahn beginnt mit der Normalenstellung (senkrecht zur Bahntangente) im Endpunkt des vorherigen Bahnstückes (Programmsatzes) und endet in Normalenstellung im Anfangspunkt des neuen Bahnstückes (Programmsatzes).

Bei sehr flachen Außenecken nähert sich das Verfahren mit **G450** (Übergangskreis) und **G451** (Schnittpunkt) immer mehr an (siehe Abschnitt "Sehr flache Außenecken").

Sollen spitze Außenkonturecken entstehen, muss das Werkzeug von der Kontur abgehoben werden (siehe Abschnitt "DISC").

DISC

Mit **G450** - Übergangskreis können keine scharfen Außenkonturecken entstehen, weil durch den Übergangskreis die Werkzeugmittelpunktsbahn so geführt wird, dass die WZ-Schneide auf der Außenecke (programmierte Position) stillsteht. Sollen mit **G450** trotzdem scharfe Außenecken bearbeitet werden, kann mit der Anweisung **DISC** im Programm eine Überhöhung programmiert werden. Dadurch wird aus dem Übergangskreis ein Kegelschnitt und die WZ-Schneide hebt von der Außenecke ab.

Der Wertebereich der Anweisung **DISC** beträgt 0 bis 100 in Schritten von 1.

Wert	Bedeutung
DISC = 0	Überhöhung abgeschaltet, Übergangskreis wirksam
DISC = 100	Überhöhung so groß, dass sich theoretisch ein Verhalten wie bei Schnittpunkt (G451) ergibt.

Der programmierbare Maximalwert für **DISC** kann eingestellt werden über das Maschinendatum:

MD20220 \$MC_CUTCOM_MAX_DISC (max. Wert für DISC)

Sinnvolle Werte für `DISC` liegen in der Regel nicht über 50.

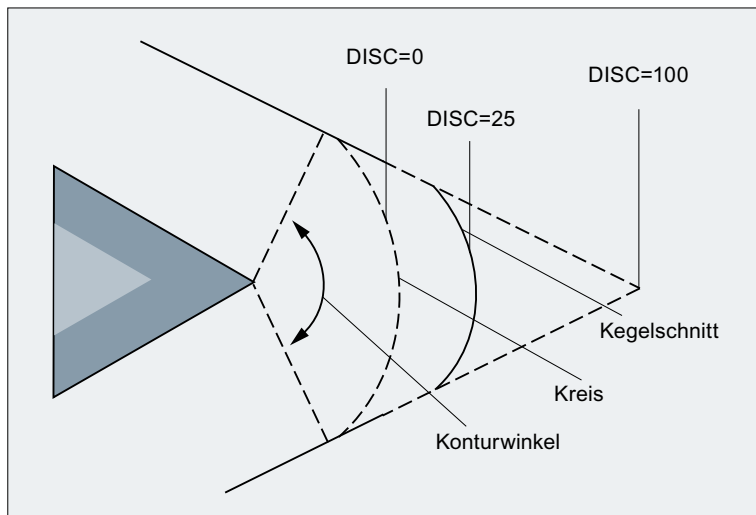


Bild 19-24 Beispiel: Überhöhung mit `DISC=25`

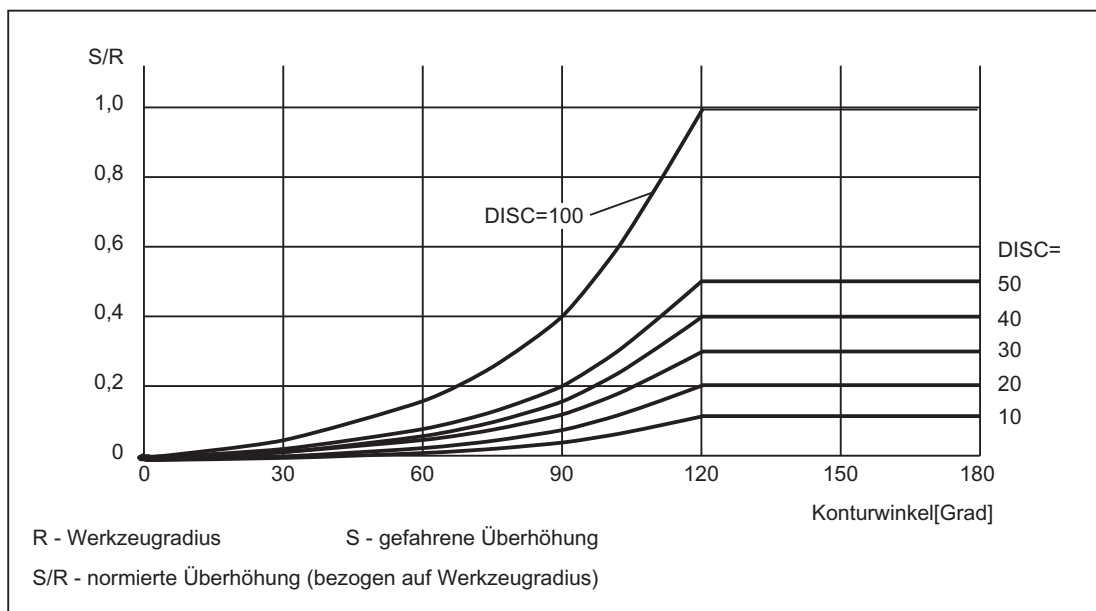


Bild 19-25 Überhöhung mit `DISC` abhängig von Konturwinkel

G451 (Schnittpunkt)

Bei aktivem G-Befehl `G451` wird die Position (der Schnittpunkt) angefahren, die sich aus den Bahnlinien (nur bei Gerade, Kreis oder Helix) ergibt, die sich im Abstand des Werkzeugradius zur programmierten Kontur befinden (Mittelpunktsbahn des Werkzeugs). Splines und Polynome werden grundsätzlich nicht verlängert.

Sehr spitze Außenecken

Bei sehr spitzen Außenecken kann es mit G451 zu langen Leerwegen kommen. Deshalb wird bei sehr spitzen Außenecken automatisch von G451 (Schnittpunkt) auf G450 (Übergangskreis, ggf. mit DISC) umgeschaltet.

Der Grenzwinkel (Konturwinkel) für diese automatische Umschaltung (Schnittpunkt → Übergangskreis) kann vorgegeben werden im Maschinendatum:

MD20210 \$MC_CUTCOM_CORNER_LIMIT (Maximalwinkel für Ausgleichssätze bei WRK)

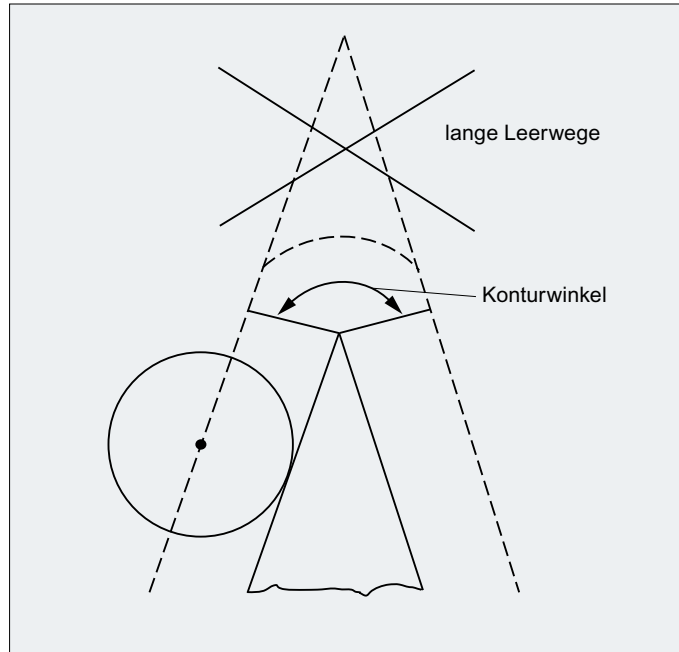


Bild 19-26 Beispiel für automatische Umschaltung auf Übergangskreis

Sehr flache Außenecken

Bei sehr flachen Außenecken nähern sich die Verfahren mit G450 (Übergangskreis) und G451 (Schnittpunkt) immer mehr an. In diesem Fall macht es keinen Sinn mehr, einen Übergangskreis einzufügen. Insbesondere bei der 5-Achsbearbeitung darf an diesen Außenecken kein Übergangskreis eingefügt werden, weil es sonst im Bahnsteuerbetrieb (G64) zu Geschwindigkeitseinbußen kommt. Deshalb wird bei sehr flachen Außenecken automatisch von G450 (Übergangskreis, ggf. mit DISC) auf G451 (Schnittpunkt) umgeschaltet.

Der Grenzwinkel (Konturwinkel) für diese automatische Umschaltung (Übergangskreis → Schnittpunkt) kann vorgegeben werden im Maschinendatum:

MD20230 \$MC_CUTCOM_CURVE_INSERT_LIMIT (Maximalwinkel für Schnittpunktberechnung bei WRK)

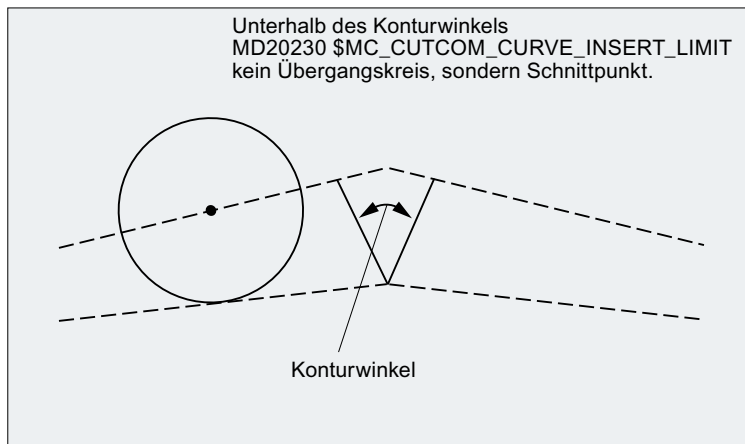


Bild 19-27 Beispiel für automatische Umschaltung auf Schnittpunkt

19.5.7 Korrektur an Innenecken

Schnittpunkt

Bilden zwei aufeinanderfolgende Sätze eine Innenecke, so wird versucht, einen Schnittpunkt der beiden Äquidistanten zu finden. Wird ein Schnittpunkt gefunden, wird die programmierte Kontur bis zum Schnittpunkt verkürzt (erster Satz am Ende verkürzt, zweiter Satz am Anfang verkürzt).

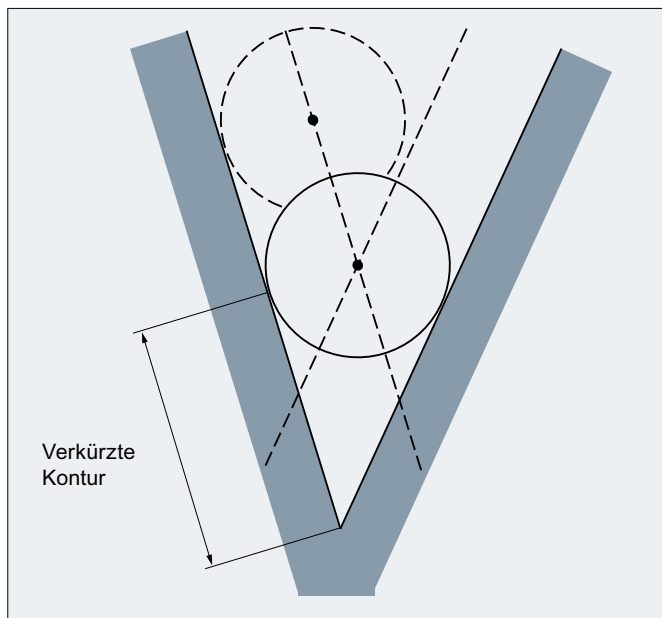


Bild 19-28 Beispiel für eine verkürzte Kontur

Kein Schnittpunkt

Bei Innenecken kann der Fall eintreten, dass zwischen zwei aufeinander folgenden Sätzen kein Schnittpunkt gefunden wird (siehe folgendes Bild).

Vorausschauende Konturberechnung

Wird zwischen zwei aufeinander folgenden Sätzen kein Schnittpunkt gefunden, betrachtet die Steuerung automatisch den nächsten Satz und versucht, mit der Äquidistanten dieses Satzes einen Schnittpunkt zu finden (siehe folgendes Bild: Schnittpunkt S). Diese automatische Betrachtung der nächsten Sätze (die vorausschauende Konturberechnung) wird immer so lange durchgeführt, bis eine per Maschinendatum einstellbare Satzanzahl erreicht ist:

MD20240 \$MC_CUTCOM_MAXNUM_CHECK_BLOCKS (Sätze für vorausschauende Konturberechnung bei WRK)

Wird innerhalb der vorausschauend betrachteten Sätze kein Schnittpunkt gefunden, bleibt die Programmbearbeitung stehen und ein Alarm wird gesetzt.

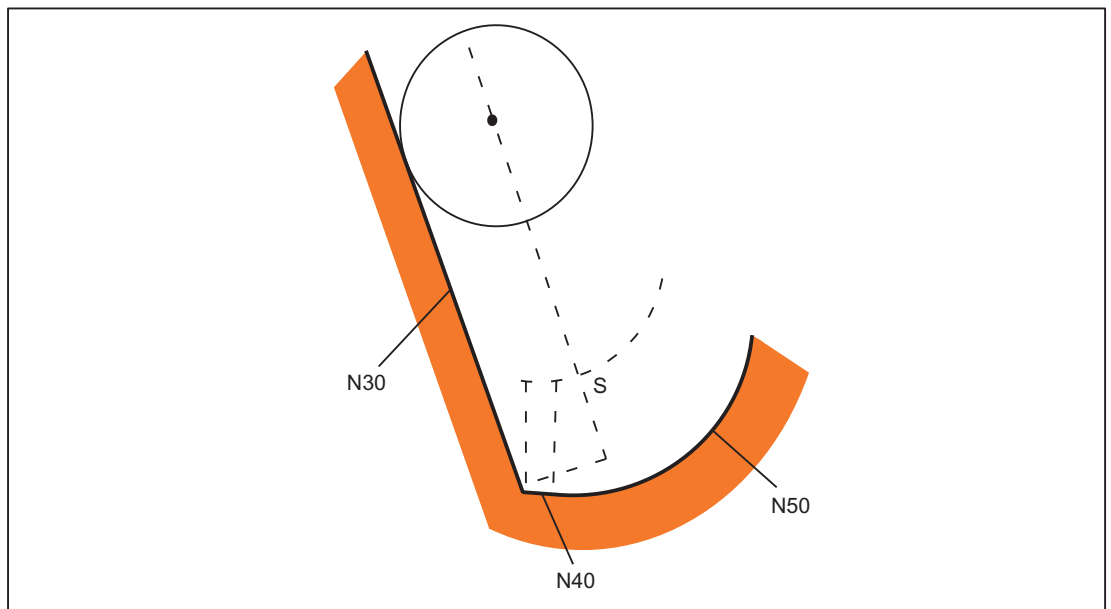


Bild 19-29 Gibt es keinen Schnittpunkt zwischen Satz N30 und Satz N40, dann wird der Schnittpunkt zwischen Satz N30 und Satz N50 berechnet.

Mehrere Schnittpunkte

Bei Innenecken kann der Fall eintreten, dass die vorausschauende Konturberechnung mehrere Schnittpunkte der Äquidistanten in mehreren aufeinander folgenden Sätzen findet. Dabei wird immer der letzte Schnittpunkt als gültiger Schnittpunkt festgelegt:

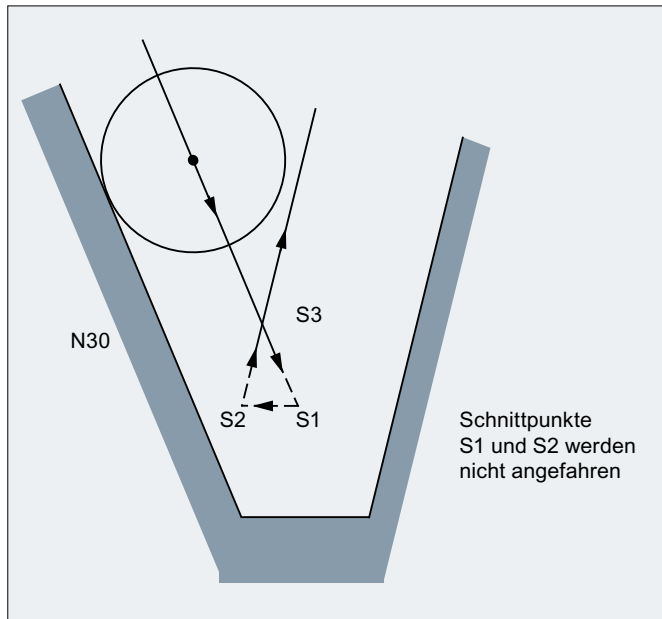


Bild 19-30 Beispiel: Innenecke mit WRK ohne Konturverletzung (3 Sätze vorausschauend)

Für weitere Informationen siehe auch Kapitel "Kollisionsüberwachung und Flaschenhalserkennung (Seite 1542)" Abschnitt "Kollisionsüberwachung".

Besonderheiten

Werden mehrere Schnittpunkte mit dem nächsten Satz gefunden, ist der Schnittpunkt gültig, der am nächsten beim Startpunkt des ersten Satzes liegt.

19.5.8 Kollisionsüberwachung und Flaschenhalserkennung

Funktion

Die Kollisionsüberwachung (Flaschenhalserkennung) überprüft, ob sich die Äquidistanten von nicht benachbarten Sätzen schneiden. Wird ein Schnittpunkt gefunden, ist das Verhalten wie bei Innenecken mit mehreren Schnittpunkten: Der letzte gefundene Schnittpunkt ist gültig.

Die maximale Anzahl von Sätzen, die vorausschauend betrachtet werden, wird eingestellt über:

MD20240 \$MC_CUTCOM_MAXNUM_CHECK_BLOCKS (Sätze für vorausschauende Konturberechnung bei WRK)

Syntax

CDON
CDOF
CDOF2

Bedeutung

CDON:	Kollisionsüberwachung EIN
CDOF:	Kollisionsüberwachung AUS
CDOF2:	Kollisionsüberwachung AUS, bei 3D-Umfangsfräsen. Bei allen anderen Bearbeitungsarten (z. B. 3D-Stirnfräsen) hat CDOF2 die gleiche Bedeutung wie CDOF.

Weitere Informationen

Bei CDOF wird ein Schnittpunkt zwischen zwei aufeinander folgenden Sätzen gesucht. Wird ein Schnittpunkt gefunden, werden keine weiteren Sätze betrachtet. Bei Außenecken wird zwischen zwei aufeinander folgenden Sätzen immer ein Schnittpunkt gefunden.

Die vorausschauende Betrachtung von mehr als zwei benachbarten Sätzen ist somit bei CDON und auch bei CDOF möglich.

Satz auslassen

Wird ein Schnittpunkt zwischen zwei nicht benachbarten Sätzen erkannt, werden die dazwischen liegenden Bewegungen in der Korrektorebene nicht ausgeführt. Alle anderen in den ausgelassenen Sätzen enthaltenen Bewegungen und ausführbaren Anweisungen (M-Befehle, Verfahren von Positionierachsen usw.), werden an der Position des Schnittpunktes in der Reihenfolge ausgeführt, in der sie im NC-Programm enthalten sind.

Warnung 10763: "Die Bahnkomponente des Satzes in der Korrektorebene wird Null."

Wurde ein Satz aufgrund der Kollisionsüberwachung oder der Flaschenhalserkennung ausgelassen, wird die Warnung 10763 angezeigt. Das NC-Programm wird nicht unterbrochen.

Die Anzeige der Warnung kann unterdrückt werden durch:

MD11410 \$MN_SUPPRES_ALARM_MASK, Bit 1 = 1 (Maske zur Unterstützung spezieller Alarmausgaben)

Flaschenhalserkennung und Außenecken

Bei der Schnittpunktüberprüfung nicht benachbarter Sätze werden nicht die programmierten Originalkonturen überprüft, sondern die zugehörigen berechneten Äquidistanten. Dies kann dazu führen, dass bei Außenecken ein "Flaschenhals" erkannt wird, obwohl das tatsächlich nicht der Fall ist. Die Ursache hierfür liegt darin, dass bei DISC > 0 die berechnete Werkzeugbahn nicht äquidistant zur programmierten Originalkontur verläuft.

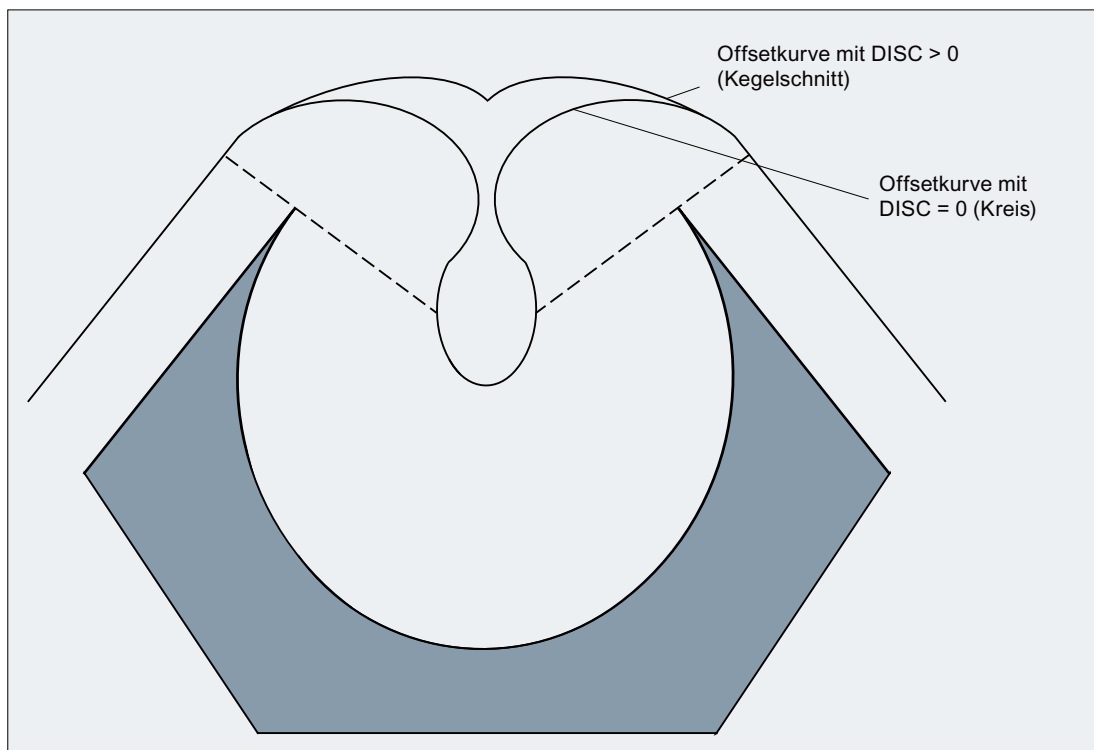


Bild 19-31 Flaschenhalserkennung und Außenecken

19.5.9 Schlitzformerkennung (Option) - nur 840D sl

Bei aktiver Werkzeugradiuskorrektur und eingeschalteter Kollisionsüberwachung (Seite 1542) wird ein programmierter Schlitz mit einer Breite kleiner als der Werkzeugdurchmesser als mögliche Konturverletzung erkannt und bei der Bearbeitung ausgelassen. In bestimmten technologieabhängigen Anwendungsfällen (z. B. bei der Herstellung von Biegelinien mit einer Laserschneidemaschine) ist es jedoch erforderlich, dass Schlitzte auch dann bearbeitet werden, wenn der Radius des Werkzeugs (Laserstrahl) etwas größer ist, als die programmierte Breite des Schlitzes zulassen würde. Um dem Anwender diese Möglichkeit zu bieten, wurde die Werkzeugradiuskorrektur um die Funktion "Schlitzformerkennung" erweitert.

Hinweis

Die Funktion "Schlitzformerkennung" ist eine lizenzpflichtige Option für SINUMERIK 840D sl.
 Artikelnummer: 6FC5800-0AS18-0YB0

Funktion

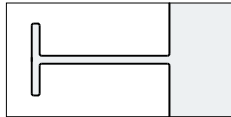
Bei aktiver Funktion "Schlitzformerkennung" werden Schlitzte, die eine vordefinierte Form haben, automatisch identifiziert und deren Bearbeitung sichergestellt.

Erkannt werden:

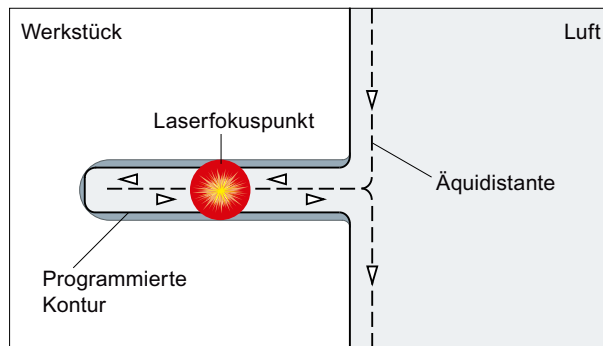
- I-Schlitz



- T-Schlitz



Ist die Schlitzbreite größer oder gleich dem Werkzeugdurchmesser, wird der Schlitz wie programmiert abgefahren. Ist die Schlitzbreite dagegen kleiner als der Werkzeugdurchmesser, wird die Äquidistante so berechnet, dass das Werkzeug in der Mitte des programmierten Schlitzes verfährt:



Folgende Bedingungen müssen erfüllt sein:

- Der Radius eines neuen Werkzeugs (R2) darf den Radius des Originalwerkzeugs (R1) nicht um den Faktor 2 überschreiten:

$$R2 < 2 * R1$$
- Der Schlitz muss eine ungerade Anzahl Sätze haben.
- Der mittlere Satz muss aus einer Geraden bestehen.
- Der Radius der Rundung am Anfang des Schlitzes bzw. die Länge des dem Schlitz vorausgehenden Satzes darf die Differenz $R2 - R1$ nicht unterschreiten.

Aktivierung

Zur Aktivierung der Funktion "Schlitzformerkennung" muss das folgende kanalspezifische Settingdatum auf "1" gesetzt werden:

SD42977 \$SC_SLOT_FORM_RECOGN = 1

Parametrierung

Anzahl Sätze für vorausschauende Merkmalerkennung

Die Schlitzformerkennung ist nur wirksam, wenn das folgende kanalspezifische Maschinendatum auf einen Wert größer null gesetzt ist:

MD28620 \$MC_MM_NUM_FEATURE_BLOCKS > 0

Mit MD28620 wird die minimale Anzahl an Sätzen festgelegt, die für die Erkennung der Schlitzform gleichzeitig betrachtet werden sollen.

Empfohlener Wert: 15 (Dies entspricht der Anzahl an Konturelementen bei einem T-Schlitz mit Kreisbögen zwischen den Geraden.)

Beispiel

Im folgenden Teileprogrammausschnitt ist eine Biegelinie programmiert, die mit einem Laserstrahl geschnitten werden soll.

Programmcode	Kommentar
N1 G01 G90 G71 F10000	
N2 \$TC_DP1[1,1] = 120	
N3 \$TC_DP3[1,1] = 0.0	
N4 \$TC_DP6[1,1] = 0.2	; Werkzeugradius = 0,2 mm
N6 G91 G41 G01 T=1 D=1	; WRK einschalten.
N7 X100.00	
N8 Y-24.500	
N9 G02 X-0.200 Y-0.200 I-0.200	; Rundung am Anfang der Biegelinienkontur.
N10 G01 X-49.700	
N11 G03 X-0.100 Y-0.100 J-0.100	
N12 G01 Y-0.100	
N13 G03 X0.100 Y-0.100 I0.100	
N14 G01 X49.700	
N15 G02 X0.200 Y-0.200 J-0.200	; Rundung am Ende der Biegelinienkontur.
N16 G01 Y-24.500	



Bei inaktiver Schlitzformerkennung (SD42977 \$SC_SLOT_FORM_RECOGN = 0) wird die Werkzeugradiuskorrektur die programmierte Biegelinie als unzulässige Kontur für das

verwendete Werkzeug (Laserstrahl) erkennen und auslassen, da die Schlitzbreite 0,3 mm beträgt, das Werkzeug aber den Radius 0,2 mm hat. Ist die Schlitzformerkennung dagegen aktiv (SD42977 \$SC_SLOT_FORM_RECOGN = 1), wird die Werkzeugradiuskorrektur die programmierte Biegelinie als I-Schlitz erkennen und der Laserstrahl wird die Biegelinie entlang der Mitte des programmierten Schlitzes schneiden.

Da in diesem Beispiel der Schlitz aus sieben Konturelementen besteht (4 Kreisbögen, 3 Geraden), muss in MD28620 \$MC_MM_NUM_FEATURE_BLOCK folgender Mindestwert eingestellt sein:

MD28620 \$MC_MM_NUM_FEATURE_BLOCK ≥ 7

19.5.10 Sätze mit veränderlichem Korrekturwert

Randbedingungen

Ein veränderlicher Korrekturwert ist bei allen Interpolationsarten zulässig (auch Kreis- und Spline-Interpolation).

Ebenso ist ein Wechsel des Vorzeichens (und damit ein Wechsel der Korrekturseite) zulässig.

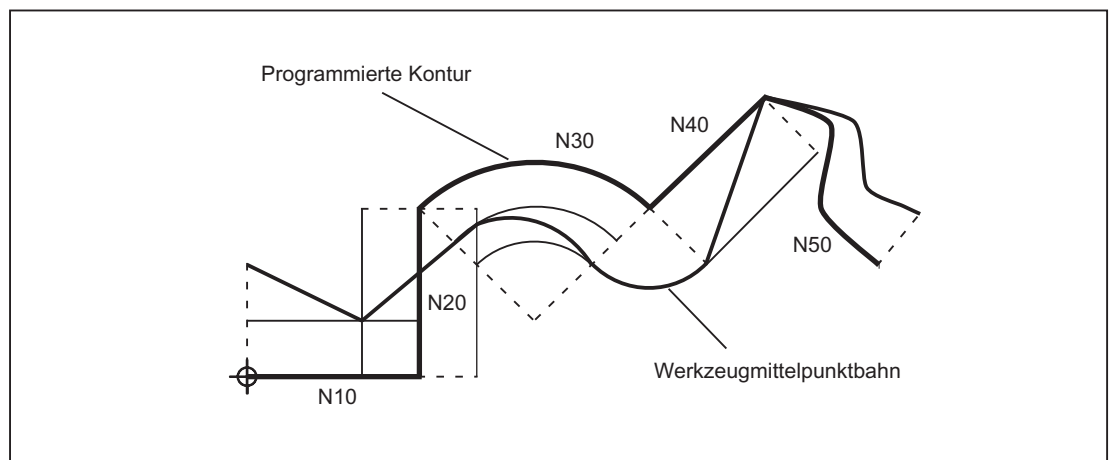


Bild 19-32 Werkzeugradiuskorrektur bei veränderlichem Korrekturwert

Schnittpunktberechnung

Bei der Schnittpunktberechnung in Sätzen mit veränderlichem Korrekturwert wird der Schnittpunkt der Offsetkurven (Werkzeugbahnen) immer mit der Annahme eines konstanten Korrekturwertes berechnet.

Ist der Satz mit dem veränderlichen Korrekturwert der erste der beiden zu betrachtenden Sätze in Fahrtrichtung, so wird der Korrekturwert am Satzende zur Berechnung herangezogen, andernfalls der Korrekturwert am Satzanfang.

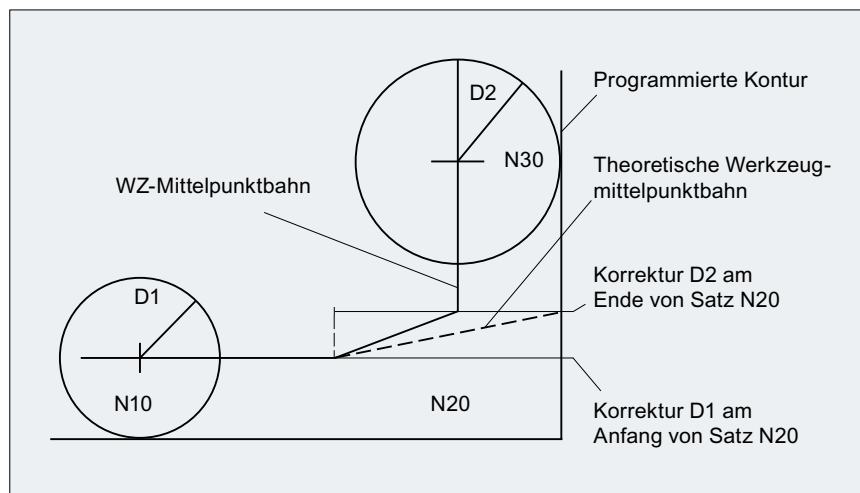


Bild 19-33 Schnittpunktberechnung bei veränderlichem Korrekturwert

Einschränkung

Wird bei der Bearbeitung an der Kreissinnenseite der Korrekturradius größer als der programmierte Kreisradius, so wird die Bearbeitung mit dem folgenden Alarm abgelehnt:

Alarm 10758 "Krümmungsradius mit veränderlichem Korrekturwert zu klein"

Geschlossene Kontur stabil erhalten

Wird ein Radius zweier Kreise geringfügig erhöht, kann dies einen dritten Satz erfordern, damit die geschlossene Kontur stabil erhalten bleibt. Dies ist dann der Fall, wenn zwei benachbarte Sätze, die zwei mögliche Schnittpunkte für eine geschlossene Kontur bilden, aufgrund der Korrektur übersprungen werden.

Eine stabil geschlossene Kontur kann dadurch erreicht werden, dass nicht der zweite mögliche Schnittpunkt gewählt wird, sondern der erste Schnittpunkt:

SD42496 \$SC_CUTCOM_CLSD_CONT \neq 0 (Verhalten der WRK bei geschlossener Kontur)

Damit wird im Falle des Überspringens dieses Satzes immer der zweite mögliche Schnittpunkt erreicht. Ein dritter Satz wird damit überflüssig.

19.5.11 Alarmverhalten

Alarm im Vorlauf

Tritt im Vorlauf ein Alarm der Werkzeugradiuskorrektur auf, wird die Bearbeitung im Hauptlauf am nächsten erreichbaren Satzende angehalten, d. h. in der Regel am Ende des aktuell interpolierten Satzes (bei aktivem Look-Ahead dann, wenn die Achsen zum Stillstand gekommen sind).

Alarmer bei Vorlaufstopp und aktiver Werkzeugradiuskorrektur

Die Werkzeugradiuskorrektur benötigt zur Bestimmung des Endpunktes eines Satzes in der Regel mindestens einen nachfolgenden Verfahrssatz (bei Flaschenhälsen auch mehr). Da bei aktivem Vorlaufstopp ein solcher Satz nicht zur Verfügung steht, wird deshalb auf den Offsetpunkt im letzten Satz verfahren. Entsprechend wird im ersten Satz nach einem Vorlaufstopp der Offsetpunkt im Startpunkt angefahren.

Die Kontur, die sich dabei ergibt, kann von der Kontur, die sich ohne Vorlaufstopp ergeben würde, erheblich abweichen. Insbesondere sind Konturverletzungen möglich. Es wurde deshalb folgendes Settingdatum eingeführt:

SD42480 \$MC_STOP_CUTCOM_STOPRE (Alarmreaktion bei WRK und Vorlaufstopp)

Abhängig von dessen Wert bleibt das Verhalten der Werkzeugradiuskorrektur gegenüber dem bisherigen Stand unverändert, bzw. wird bei einem Vorlaufstopp während aktiver Werkzeugradiuskorrektur ein Alarm ausgegeben und die Programmbearbeitung angehalten.

Der Bediener kann diesen Alarm quittieren und das NC-Programm mit NC-Start unverändert fortsetzen oder mit `RESET` abbrechen.

19.5.12 Schnittpunktverfahren für Polynome

Funktion

Bilden zwei Kurven bei aktiver Werkzeugradiuskorrektur eine Außenecke, wird abhängig vom G-Befehl der 18. Gruppe (Eckenverhalten bei Werkzeugkorrektur; `G450` / `G451`) und unabhängig vom Typ der beteiligten Kurven (Geraden, Kreise, Polynome):

- ein Kegelschnitt zum Umfahren der Ecke eingefügt
oder
- die beteiligten Kurven werden so extrapoliert, dass sie einen Schnittpunkt bilden

Wird bei aktivem `G451` kein Schnittpunkt gefunden oder ist der Winkel, den die beiden beteiligten Kurven bilden, zu spitz, wird automatisch auf das Einfügeverfahren umgeschaltet.

Das Schnittpunktverfahren für Polynome wird freigegeben mit dem Maschinendatum:

MD20256 \$MC_CUTCOM_INTERS_POLY_ENABLE (Schnittpunktverfahren für Polynome möglich)

Hinweis

Wird dieses Maschinendatum auf inaktiv gesetzt, dann wird stets ein (gegebenenfalls sehr kurzer) Satz eingefügt (auch wenn die Übergänge nahezu tangential sind). Diese kurzen Sätze führen im `G64`-Betrieb zwangsläufig zu unerwünschten Geschwindigkeitseinbrüchen.

19.5.13 G461/G462: Erweiterung An- / Abfahrstrategie

Funktion

In bestimmten geometrischen Sonderfällen werden gegenüber der bisherigen Realisierung erweiterte An- und Abfahrstrategien beim Aktivieren bzw. Deaktivieren der Werkzeugradiuskorrektur benötigt (siehe folgendes Bild).

Hinweis

Im Folgenden wird immer nur die Situation bei Deaktivieren der Werkzeugradiuskorrektur dargestellt. Das Verhalten beim Anfahren ist dazu völlig symmetrisch.

Beispiel

```
G42 D1 T1 ; Werkzeugradius 20mm
...
G1 X110 Y0
N10 X0
N20 Y10
N30 G40 X50 Y50
```

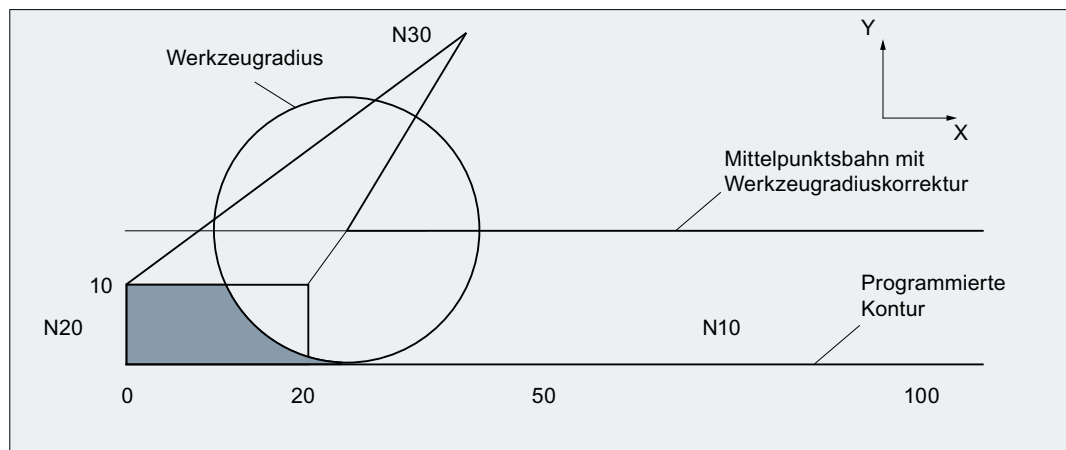


Bild 19-34 Abfahrverhalten bei G460

Der letzte Satz mit aktiver Werkzeugradiuskorrektur (N20) ist so kurz, dass beim aktuellen Werkzeugradius kein Schnittpunkt der Offsetkurve mit dem Vorgängersatz (oder einem weiter davor liegenden Satz) mehr existiert. Es wird deshalb ein Schnittpunkt zwischen den Offsetkurven von Folgesatz und Vorgängersatz gesucht, d. h. im Beispiel zwischen N10 und N30. Die für den Abfahrssatz verwendete Kurve ist dabei keine echte Offsetkurve, sondern eine Gerade vom Offsetpunkt im Endpunkt des Satzes N20 zum programmierten Endpunkt von N30. Wird ein Schnittpunkt gefunden, wird er angefahren. Der im Bild farbig dargestellte Bereich wird dann nicht bearbeitet, wie es mit dem verwendeten Werkzeug eigentlich möglich wäre.

G460

Bei G460 ist das An- bzw. Abfahrverhalten wie bisher.

G461

Wenn kein Schnittpunkt des letzten WRK-Satzes mit einem Vorgängersatz möglich ist, wird die Offsetkurve dieses Satzes mit einem Kreis verlängert, dessen Mittelpunkt im Endpunkt des nicht korrigierten Satzes liegt, und dessen Radius gleich dem Werkzeugradius ist.

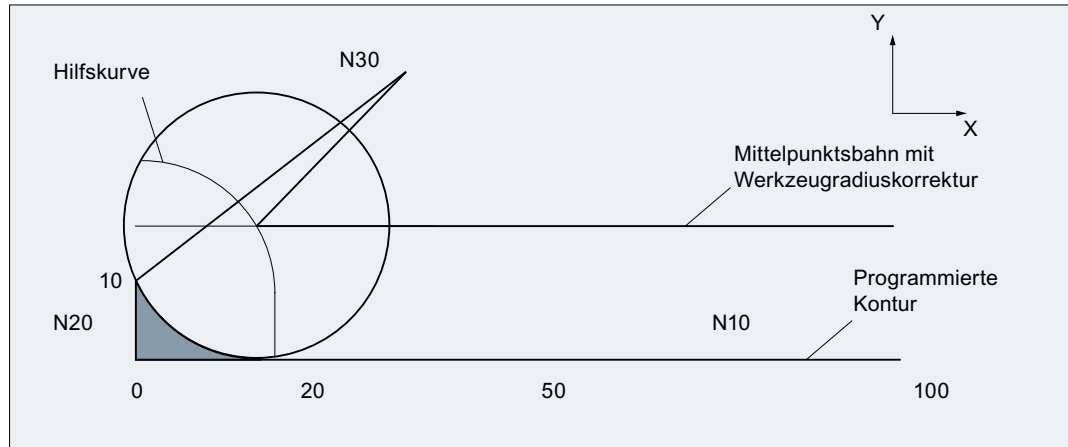


Bild 19-35 Abfahrverhalten bei G461

Die Steuerung versucht, diesen Kreis mit einem der Vorgängersätze zu schneiden. Dabei wird bei aktivem `CDOF` die Suche abgebrochen, wenn ein Schnittpunkt gefunden wurde, d. h. es wird nicht überprüft, ob auch noch Schnittpunkte mit weiter in der Vergangenheit liegenden Sätzen existieren.

Bei aktivem `CDON` wird auch dann, wenn bereits ein Schnittpunkt gefunden wurde, nach weiteren Schnittpunkten gesucht.

Ein so gefundener Schnittpunkt ist der neue Endpunkt eines Vorgängersatzes und der Startpunkt des Deaktivierungssatzes. Der eingefügte Kreis dient nur zur Schnittpunktberechnung und hat selbst keine Verfahrbewegung zur Folge.

Hinweis

Wird kein Schnittpunkt gefunden, wird folgender Alarm ausgegeben:

Alarm "10751 Kollisionsgefahr"

G462

Wenn kein Schnittpunkt des letzten WRK-Satzes mit einem Vorgängersatz möglich ist, wird beim Abfahren mit G462 (Grundstellung) im Endpunkt des letzten Satzes mit Werkzeugradiuskorrektur eine Gerade eingefügt (der Satz wird durch seine Endtangente verlängert):

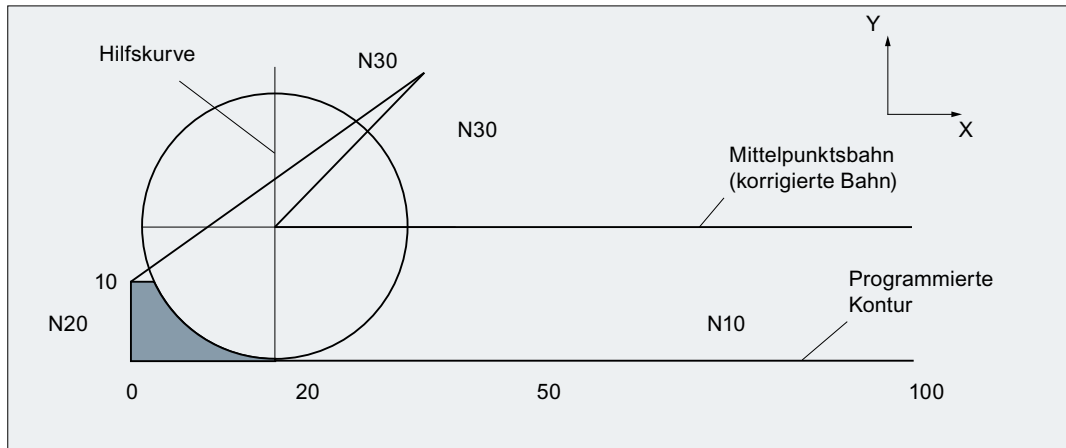


Bild 19-36 Abfahrverhalten bei G462

Die Schnittpunktsuche verläuft dann identisch zu der bei G461.

Bei G462 wird die im Beispielpogramm von N10 und N20 gebildete Ecke nicht soweit ausgeräumt, wie es mit dem verwendeten Werkzeug möglich wäre. Dieses Verhalten kann aber dennoch notwendig sein, wenn die Teilekontur (abweichend von der programmierten Kontur) im Beispiel links von N20 auch bei größeren Werten von y als 10 mm nicht verletzt werden darf.

Ist **KONT** aktiv (Kontur im Start- oder Endpunkt umfahren), wird unterschieden, ob der Endpunkt vor oder hinter der Kontur liegt.

Endpunkt vor der Kontur

Liegt der Endpunkt vor der Kontur, ist das Abfahrverhalten gleich wie bei **NORM**. Diese Eigenschaft ändert sich auch nicht, wenn der letzte Kontursatz bei G451 mit einer Geraden oder einem Kreis verlängert wird. Zusätzliche Umfahrungsstrategien, um eine Konturverletzung in der Nähe des Konturendpunktes zu vermeiden, sind deshalb nicht notwendig.

Endpunkt hinter der Kontur

Liegt der Endpunkt hinter der Kontur, wird immer abhängig von G450 / G451 ein Kreis bzw. eine Gerade eingefügt. G460 - G462 hat dann keine Bedeutung.

Hat der letzte Verfahrssatz in dieser Situation keinen Schnittpunkt mit einem Vorgängersatz, kann sich nun ein Schnittpunkt mit dem eingefügten Konturelement oder mit dem Geradenstück vom Endpunkt des Umfahrungskreises zum programmierten Endpunkt ergeben.

Ist das eingefügte Konturelement ein Kreis (G450) und bildet dieser mit dem Vorgängersatz einen Schnittpunkt, ist dieser gleich dem Schnittpunkt, der sich auch bei NORM und G461 ergeben würde. Im Allgemeinen bleibt jedoch ein zusätzliches Stück des Kreises zu verfahren. Für den linearen Teil des Abfahrtsatzes ist keine Schnittpunktberechnung mehr notwendig.

Im zweiten Fall (wenn kein Schnittpunkt des eingefügten Konturelements mit den Vorgängersätzen gefunden wird) wird auf den Schnittpunkt zwischen der Abfahrgeraden und einem Vorgängersatz verfahren.

Es kann sich somit bei aktivem G461 bzw. G462 nur dann ein gegenüber G460 verändertes Verhalten ergeben, wenn entweder NORM aktiv ist, oder das Verhalten bei KONT geometrisch bedingt identisch zu dem bei NORM ist.

Hinweis

Das Anfahrverhalten ist symmetrisch zum Abfahrverhalten.

Das An- bzw. Abfahrverhalten wird vom Zustand des G-Befehles im An- bzw. Abfahrtsatz bestimmt. Das Anfahrverhalten kann deshalb unabhängig vom Abfahrverhalten eingestellt werden.

Beispiel:

Programm für die Verwendung von G461 beim Anfahren:

```
N10 $TC_DP1[1,1]=120           ; Werkzeugtyp Fraeser
N20 $TC_DP6[1,1]=10           ; Radius
N30 X0 Y0 F10000 T1 D1
N40 Y20
N50 G42 X50 Y5 G461
N60 Y0 F600
N70 X30
N80 X20 Y-5
N90 X0 Y0 G40
N100 M30
```

19.6 Werkzeugradiuskorrektur konstant halten

Bedeutung

Die Funktion "Werkzeugradiuskorrektur konstant halten" dient dazu, die Werkzeugradiuskorrektur für eine Anzahl von Sätzen zu unterdrücken, wobei jedoch eine durch die Werkzeugradiuskorrektur in vorhergehenden Sätzen aufgebaute Differenz zwischen der programmierten und der tatsächlich abgefahrenen Bahn des Werkzeugmittelpunktes als Verschiebung beibehalten wird.

Sie kann z. B. dann vorteilhaft eingesetzt werden, wenn beim Zeilenfräsen in den Umkehrpunkten mehrere Verfahrssätze notwendig sind, die von der Werkzeugradiuskorrektur erzeugten Konturen (Umfahrungsstrategien) jedoch nicht erwünscht sind.

Aktivierung

Die Funktion "Werkzeugradiuskorrektur konstant halten" wird mit dem G-Befehl `CUTCONON` (CUTter compensation CONstant ON) aktiviert und mit dem G-Befehl `CUTCONOF` (CUTter compensation CONstant OFF) deaktiviert.

`CUTCONON` und `CUTCONOF` bilden eine modale G-Gruppe.

Grundstellungswert ist `CUTCONOF`.

Die Funktion ist unabhängig von der Art der Werkzeugradiuskorrektur ($2\frac{1}{2}$ D, 3D-Stirnfräsen, 3D-Umfangsfäsen) einsetzbar.

Normalfall

Im Normalfall ist vor der Aktivierung der Korrekturunterdrückung die Werkzeugradiuskorrektur bereits aktiv, und sie ist noch aktiv, wenn die Korrekturunterdrückung wieder deaktiviert wird.

Im letzten Verfahrssatz vor `CUTCONON` wird auf den Offsetpunkt im Satzendpunkt gefahren. Alle folgenden Sätze, in denen die Korrekturunterdrückung aktiv ist, werden ohne Korrektur verfahren. Sie werden dabei jedoch um den Vektor vom Endpunkt des letzten Korrektursatzes zu dessen Offsetpunkt verschoben. Der Interpolationstyp dieser Sätze (linear, zirkular, polynomial) ist beliebig.

Der Deaktivierungssatz der Korrekturunterdrückung, d. h. der Satz, der `CUTCONOF` enthält, wird normal korrigiert. Er beginnt im Offsetpunkt des Startpunktes. Zwischen dem Endpunkt des Vorgängersatzes, d. h. des letzten programmierten Verfahrssatzes mit aktivem `CUTCONON`, und diesem Punkt wird ein linearer Satz eingefügt.

Kreissätze, bei denen die Kreisebene senkrecht auf der Korrekturebene steht (vertikale Kreise), werden so behandelt, als ob in ihnen `CUTCONON` programmiert wäre. Diese implizite Aktivierung der Korrekturunterdrückung wird im ersten Verfahrssatz, der eine Verfahrbewegung in der Korrekturebene enthält und der kein derartiger Kreis ist, automatisch rückgängig gemacht. Vertikale Kreise in diesem Sinne können nur beim Umfangsfräsen auftreten.

Beispiel:

N10	; Definition des Werkzeugs d1
N20 \$TC_DP1[1,1]= 110	; Typ
N30 \$TC_DP6[1,1]=	; Radius
N40	
N50 X0 Y0 Z0 G1 G17 T1 D1 F10000	
N60	
N70 X20 G42 NORM	
N80 X30	
N90 Y20	
N100 X10 CUTCONON	; Einschalten der Korrekturunterdrückung

N110 Y30 KONT	; Beim Ausschalten der Konturunterdrückung gegebenenfalls Umfahungskreis einfügen
N120 X-10 CUTCONOF	
N130 Y20 NORM	; Kein Umfahungskreis beim Ausschalten der WRK
N140 X0 Y0 G40	
N150 M30	

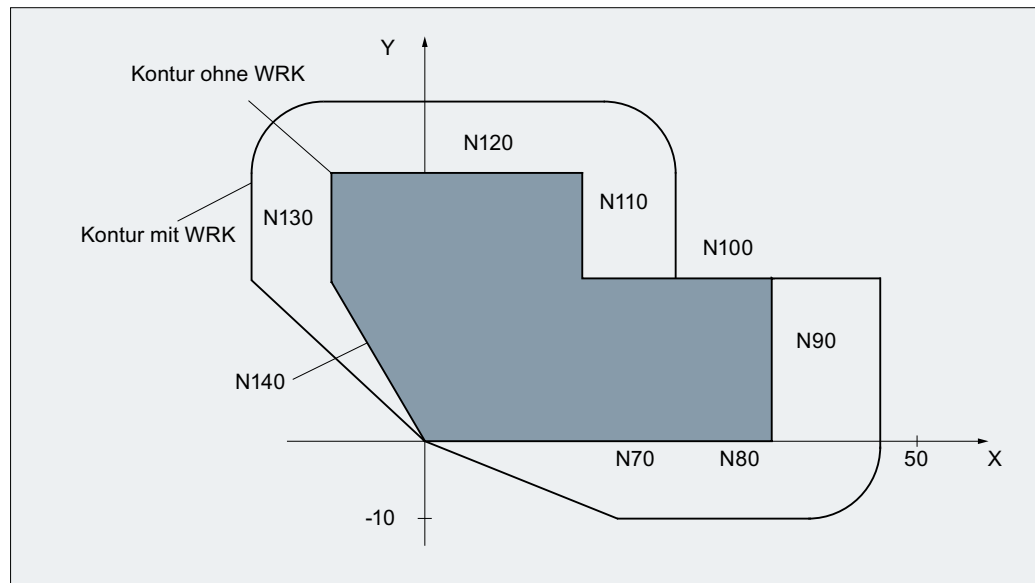


Bild 19-37 Beispielprogramm zur Konturunterdrückung

Sonderfälle

- Ist keine Werkzeugradiuskorrektur aktiv (G40), hat CUTCONON keine Wirkung. Es wird kein Alarm erzeugt. Der G-Befehl bleibt jedoch aktiv. Dies ist dann von Bedeutung, wenn in einem späteren Satz die Werkzeugradiuskorrektur mit G41 bzw. mit G42 eingeschaltet werden soll.
- Ein Wechsel des G-Befehls in der 7. G-Gruppe (Werkzeugradiuskorrektur; G40 / G41 / G42) bei aktivem CUTCONON ist zulässig. Ein Wechsel nach G40 wird sofort wirksam. Die Verschiebung, mit der die Vorgängersätze verfahren wurden, wird dabei herausgefahren.
- Wird CUTCONON oder CUTCONOF in einem Satz ohne Verfahrbewegung in der aktiven Korrektorebene programmiert, so wird die Wirksamkeit bis zum nächsten Satz mit einer solchen Verfahrbewegung verzögert.
- Wird CUTCONON bei aktiver Werkzeugradiuskorrektur programmiert und vor Programmende nicht mehr aufgehoben, so werden die betroffenen Verfahrsätze mit dem letzten gültigen Offset verfahren. Gleiches gilt bei erneuter Programmierung von G41 bzw. G42 im letzten Verfahr Satz eines Programms.

- Wird die Werkzeugradiuskorrektur mit G41 bzw. G42 eingeschaltet und ist gleichzeitig bereits CUTCONON aktiv, wird die Aktivierung der Korrektur bis zum ersten Verfahrssatz mit CUTCONOF verzögert.
- Beim Wiederaufahren an die Kontur mit CUTCONOF wird die 17. G-Gruppe (An- und Abfahrverhalten bei Werkzeugkorrektur; NORM / KONT) ausgewertet, d. h. bei KONT wird gegebenenfalls ein Umfahrungskreis eingefügt. Ein Umfahrungskreis wird unter den gleichen Bedingungen wie bei der Aktivierung der Werkzeugradiuskorrektur mit G41 bzw. G42 eingefügt.
- Die Zahl der Sätze mit unterdrückter Werkzeugradiuskorrektur ist begrenzt: MD20252 \$MC_CUTCOM_MAXNUM_SUPPR_BLOCKS (Maximale Satzzahl mit Korrekturunterdrückung)
Wird sie überschritten, wird die Bearbeitung mit einer Fehlermeldung abgebrochen. Die Begrenzung ist notwendig, weil beim Repositionieren die interne Satzbearbeitung im letzten Satz vor CUTCONON wieder aufgenommen werden muss.
- Das Verhalten bei erneuter Programmierung von G41 bzw. G42 bei bereits aktiver Werkzeugradiuskorrektur ist ähnlich wie bei der Korrekturunterdrückung. Es gelten folgende Abweichungen:
 - Es sind nur Linearsätze zugelassen
 - Ein einzelner Verfahrssatz, der G41 bzw. G42 enthält, wird so modifiziert, dass er im Versatzpunkt des Startpunktes des Folgesatzes endet. Dadurch entfällt das Einfügen eines Zwischensatzes. Gleiches gilt für den letzten einer Folge von Verfahrssätzen, von denen jeder G41 bzw. G42 enthält.
 - Das erneute Aufahren der Kontur erfolgt unabhängig vom G-Befehl der 17. Gruppe (An- und Abfahrverhalten bei Werkzeugkorrektur; NORM / KONT) immer mit NORM.
- Wird G41 / G42 mehrfach in aufeinander folgenden Verfahrssätzen programmiert, so werden alle Sätze bis auf den letzten wie bei CUTCONON bearbeitet.
- Die Art der Konturunterdrückung wird nur im ersten Verfahrssatz einer Folge von Sätzen ausgewertet.
Ist im ersten Satz sowohl CUTCONON als auch G41 bzw. G42 programmiert, wird das Verhalten beim Deaktivieren der Konturunterdrückung von CUTCONON bestimmt.
Ein Wechsel von G41 nach G42 bzw. umgekehrt kann in diesem Fall sinnvoll dazu verwendet werden, die Korrekturseite (links bzw. rechts von der Kontur) beim Wiederaufahren zu wechseln.
Ein Wechsel der Korrekturseite (G41 / G42) kann bei aktiver Konturunterdrückung auch in einem späteren Satz programmiert werden.
- Für alle Sätze mit aktiver Konturunterdrückung ist die Kollisionsüberwachung bzw. Flaschenhalserkennung ausgeschaltet.

19.7 Orientierbare Werkzeugträger

19.7.1 Allgemeine Informationen

Einführung

Bei einer Klasse von Werkzeugmaschinen ist die Orientierung des Werkzeugs (z. B. **durch Umrüsten**) veränderbar. Im Betrieb ist die einmal eingestellte Orientierung jedoch **fest** und kann insbesondere während des Verfahrens nicht verändert werden. Aus diesem Grund ist für derartige Maschinen eine kinematische Orientierungstransformation (3-, 4- oder 5-Achstransformationen, TRAORI) weder notwendig noch sinnvoll. Es besteht jedoch die Notwendigkeit, die durch eine Orientierungsänderung bedingten Änderungen der Werkzeuglängenkomponenten zu berücksichtigen, ohne den Anwender mit den dazu erforderlichen Berechnungen zu belasten. Die Steuerung übernimmt diese Berechnungen.

Verfügbarkeit

Die Funktion "Orientierbarer Werkzeugträger" steht bei SINUMERIK 828D nur für die Fräsvarianten zur Verfügung.

Datenvorgaben

Wenn die Steuerung Werkzeugkorrekturen bei orientierbaren Werkzeugträgern berücksichtigen soll, müssen vorliegen:

- Werkzeugdaten (Geometrie, Verschleiß ...)
- Werkzeugträgerdaten (Angaben zur Geometrie des orientierbaren Werkzeugträgers)

Werkzeugträgeranwahl

Für die Funktion "Orientierbarer Werkzeugträger" muss ein definierter Werkzeugträger der Steuerung angegeben werden. Dafür existiert der NC-Programmbefehl:

TCARR = m

m: Nummer des Werkzeugträgers

Mit dem Werkzeugträger ist ein Werkzeugträgerdatensatz verbunden, der dessen Geometrie beschreibt.

Die Aktivierung des Werkzeugträgers und seines Datensatzes wirkt sofort, d. h. ab dem nächsten Bewegungssatz.

Zuordnung Werkzeug / Werkzeugträger

Das bisher aktive Werkzeug wird dem neu angewählten Werkzeugträger zugeordnet.

Aus Sicht der Steuerung sind Werkzeugträgernummern m und Werkzeugnummern T frei kombinierbar. In der realen Anwendung können sich Kombinationen aus Bearbeitungsgründen

und mechanischen Gründen ausschließen. Die Steuerung prüft nicht auf sinnvolle Kombinationen.

Beschreibung der Kinematik des WZ-Trägers

Die Kinematik des orientierbaren Werkzeugträgers wird mit insgesamt 33 Parametersätzen beschrieben.

Die Daten des Datensatzes sind durch den Anwender bearbeitbar.

Orientierbare Werkzeugträger

Beispiel: Kardanischer Werkzeugträger mit zwei Achsen für die Werkzeugorientierung

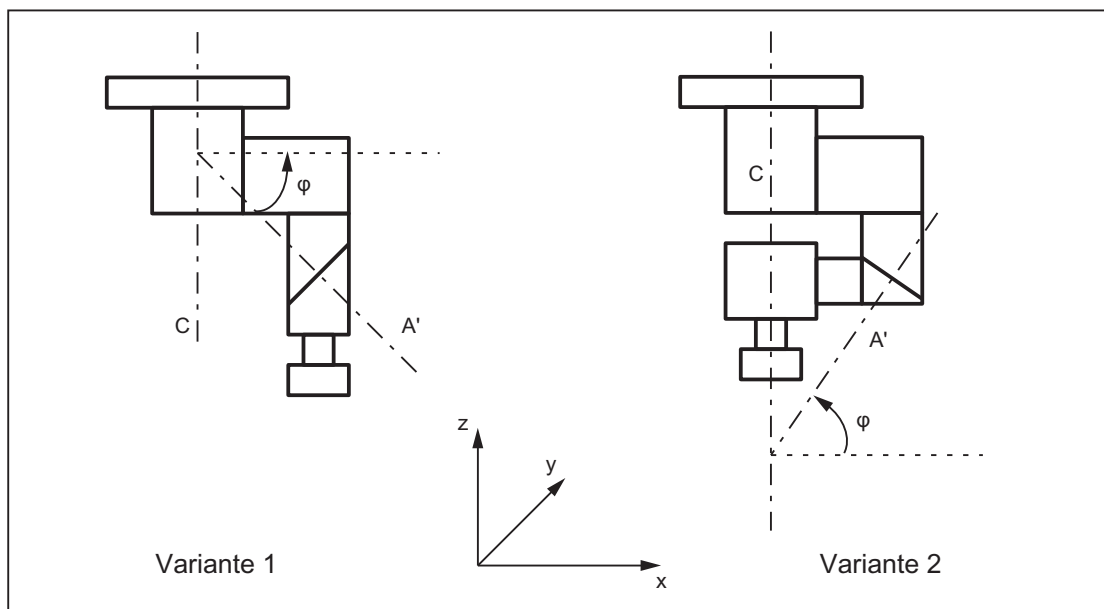


Bild 19-38 Kardanischer Werkzeugträger mit zwei Achsen

Bearbeitung der Werkzeuträger-Datensätze

Es werden zwei Möglichkeiten angeboten:

- Explizite Eintragung im Werkzeuträger-Datensatz aus dem Teileprogramm
- Automatische Übernahme einiger Werte (Winkel) aus einem Frame
 Voraussetzung hierfür ist, dass bei der Werkzeuträgeranwahl zusätzlich `TCOFr` (Tool Carrier Orientation FFrame) angegeben wird.
 Die für die Werkzeuglängenberechnung verwendete Werkzeugorientierung wird bei einem Werkzeuträgerwechsel neu aus dem zu diesem Zeitpunkt aktiven Frame ermittelt.

Orientierung in der Z-Richtung

Die G-Befehle `TOFRAME` definiert einen Frame so, dass die Z-Richtung in diesem Frame gleich der aktuellen Werkzeugorientierung ist.

Wenn kein Werkzeugträger oder ein Werkzeugträger ohne Orientierungsänderung aktiv ist, dann ist die Z-Richtung im neuen Frame:

- bei G17 gleich der alten Z-Richtung.
- bei G18 gleich der alten Y-Richtung.
- bei G19 gleich der alten X-Richtung.

TCOABS bei aktivem Frame

Die absolute Werkzeugträgerorientierung wird eingestellt durch:

TCOABS (Tool Carrier Orientation ABSolut)

Die für die Werkzeuglängenkorrektur berücksichtigte Orientierung ist **unabhängig** von der Orientierung des aktiven Frames.

Es kann jeweils nur eine der Anweisungen TCOABS oder TCOFR gültig sein.

Wechsel des Frames

Der Anwender kann nach der Anwahl des Werkzeuges den Frame wechseln. Das hat keinen Einfluss auf die Komponenten der Werkzeuglängenkorrektur.

Winkel in den Werkzeugträgerdaten:

Die in den Werkzeugträgerdaten abgelegten programmierten Drehwinkel werden durch die mit Frames festgelegten Drehwinkel nicht beeinflusst. Beim Wechsel von TCOFR nach TCOABS werden die ursprünglichen (programmierten) Drehwinkel in den Werkzeugträgerdaten wieder wirksam.

Werkzeugkorrektur-Typen

Die Werkzeugradienkorrektur WRK berücksichtigt die aktuelle Werkzeugorientierung, wenn CUT2D oder CUT3DFS aktiv ist.

Alle anderen Werkzeugkorrektur-Typen

Das sind alle Korrekturtypen der G-Gruppe 22 mit Ausnahme von CUT3DC und CUT3DF. Das Verhalten ist unverändert bezüglich der für die Korrektur verwendeten Ebene. Diese wird unabhängig von der Werkzeug-Orientierung aus dem aktiven Frame bestimmt.

Bei CUT2DF und CUT3DFF wird die für WRK verwendete Korrekturebene **unabhängig** von der aktuellen Werkzeugorientierung aus dem Frame ermittelt. Es wird die aktive Ebene (G17/G18/G19) berücksichtigt.

CUT3DC und CUT3DF

3D-Werkzeugkorrektur für das Umfangsfräsen

3D-Werkzeugkorrektur für das Stirnfräsen bei aktiver 5-Achs-Transformation sind von der Funktion "Orientierbare Werkzeugträger" nicht betroffen.

Die Orientierungsinformation wird durch die aktive kinematische 5-Achs-Transformation ermittelt.

Eingeschränkte Orientierung des Werkzeugträgers

Wird durch den Frame eine Orientierung festgelegt, die mit der definierten Werkzeugträgerkinematik nicht erreicht werden kann, wird ein Alarm ausgegeben.

Folgende Kinematiken können nicht jede Orientierung erreichen:

- wenn die beiden Drehachsen, die zur Definition der Kinematik notwendig sind, nicht aufeinander senkrecht stehen und wenn die Werkzeugachse, die die Werkzeugrichtung definiert, nicht senkrecht auf der zweiten Drehachse steht, oder
- wenn weniger als zwei Achsen definiert sind.

Nicht drehende Werkzeugträger

Die intern verwendete Werkzeugorientierung ist nur von der Grundorientierung des Werkzeugs und der aktiven Ebene (G17 - G19) abhängig.

Mehrdeutigkeiten

Eine bestimmte, durch Frame vorgegebene Werkzeugorientierung, kann mit zwei Achsen im Allgemeinen durch **zwei** verschiedene Drehwinkelpaare eingestellt werden. Die Steuerung wählt von diesen zwei möglichen Stellungen diejenige aus, bei der die Drehwinkel möglichst nahe an den programmierten Drehwinkeln liegen.

Ablage von Winkeln in den Werkzeugträgerdaten

In ungefähr allen Fällen, in denen eine Mehrdeutigkeit auftreten kann, ist es erforderlich, die aus dem Frame zu erwartenden Winkel in den Werkzeugträgerdaten abzulegen.

Parametersätze

Ein vollständiger Parametersatz für einen orientierbaren Werkzeugträger besteht aus 33 Werten.

Folgende Systemvariablen sind verfügbar:

- \$TC_CARR1 bis \$TC_CARR33
- \$TC_CARR34 bis \$TC_CARR65 zusätzlich zur freien Verfügung des Anwenders und für Feinverschiebungen.

Die Bedeutung der einzelnen Parameter wird im Folgenden unterschieden.

Kinematik der Maschine:

\$TC_CARR1 bis \$TC_CARR20 sowie \$TC_CARR23

Dabei definieren \$TC_CARR18 bis \$TC_CARR20 einen weiteren Vektor I_4 , der zur Beschreibung der Maschine bei aufgelösten Kinematiken (sowohl Werkzeug als auch Werkstück sind drehbar) benötigt wird.

\$TC_CARR21 und \$TC_CARR22 enthalten die Kanalachsennamen der Rundachsen, auf deren Position gegebenenfalls bei der Bestimmung der Orientierung des orientierbaren Werkzeugträgers zugegriffen wird.

Kinematiktyp:

\$TC_CARR23 mittels eines Buchstabens T, P oder M

Beim Kinematiktyp gibt es die folgenden drei Möglichkeiten, wobei Groß- und Kleinschreibung zulässig ist:

- T: Nur das Werkzeug (**T**ool) ist drehbar (Basiswert).
- P: Nur das Werkstück (**P**art) ist drehbar.
- M: Werkstück und Werkzeug sind drehbar (**M**ixed mode).

Jedes andere außer den genannten drei Zeichen führt bei dem Versuch, den orientierbaren Werkzeugträgern zu aktivieren, zu dem Alarm:

Alarm "14153 Kanal %1 Satz %2 Unbekannter Werkzeugtraegertyp: %3"

Parametrierung der Drehachsen:

\$TC_CARR24 bis \$TC_CARR33

Mit den Systemvariablen in \$TC_CARR24 bis \$TC_CARR33 können Offsets, Winkeloffsets, Hirth-Verzahnung sowie Achsgrenzen parametrierbar werden.

Hinweis

Die Systemvariablen sind sowohl mit als auch ohne aktive Werkzeugverwaltung verfügbar.

Komponenten und Vorbesetzung der Kette / Datensatz

Die Werte \$TC_CARR1 bis \$TC_CARR20 und \$TC_CARR24 - \$TC_CARR33 im Werkzeugträger-Datensatz sind vom NC-Sprachformat-Typ `REAL`.

Die Werte \$TC_CARR21 und \$TC_CARR22 für die Achsnamen der ersten Drehachse (v_1) und der zweiten Drehachse (v_2) sind vom NC-Sprachformat-Typ `AXIS`. Sie sind alle mit Null vorbesetzt.

Der Wert \$TC_CARR23 wird mit dem Großbuchstaben "T" (nur das Werkzeug ist drehbar) vorbesetzt.

\$TC_CARRn[m]

\$TC_CARR[0]= 0 hat Sonderbedeutung

Systemvariablen für orientierbare Werkzeugträger

\$TC_CARRn[m]

n: Parameter 1...33

m: Nummer des orientierbaren WZ-Trägers 1...Wert des Maschinendatums:

MD18088 \$MN_MM_NUM_TOOL_CARRIER (Maximale Anzahl definierbarer Werkzeugträger)

Beschreibung	NC-Variable	Sprach-Format	Vorbelegung
x-Komponente des Offsetvektors l_1	\$TC_CARR1	REAL	0
y-Komponente des Offsetvektors l_1	\$TC_CARR2	REAL	0
z-Komponente des Offsetvektors l_1	\$TC_CARR3	REAL	0
x-Komponente des Offsetvektors l_2	\$TC_CARR4	REAL	0
y-Komponente des Offsetvektors l_2	\$TC_CARR5	REAL	0
z-Komponente des Offsetvektors l_2	\$TC_CARR6	REAL	0
x-Komponente der Drehachse v_1	\$TC_CARR7	REAL	0
y-Komponente der Drehachse v_1	\$TC_CARR8	REAL	0
z-Komponente der Drehachse v_1	\$TC_CARR9	REAL	0
x-Komponente der Drehachse v_2	\$TC_CARR10	REAL	0
y-Komponente der Drehachse v_2	\$TC_CARR11	REAL	0
z-Komponente der Drehachse v_2	\$TC_CARR12	REAL	0
Drehwinkel α_1 (in Grad)	\$TC_CARR13	REAL	0
Drehwinkel α_2 (in Grad)	\$TC_CARR14	REAL	0
x-Komponente des Offsetvektors l_3	\$TC_CARR15	REAL	0
y-Komponente des Offsetvektors l_3	\$TC_CARR16	REAL	0
z-Komponente des Offsetvektors l_3	\$TC_CARR17	REAL	0
x-Komponente des Offsetvektors l_4	\$TC_CARR18	REAL	0
y-Komponente des Offsetvektors l_4	\$TC_CARR19	REAL	0
z-Komponente des Offsetvektors l_4	\$TC_CARR20	REAL	0
Achsname der Drehachse v_1	\$TC_CARR21	AXIS	0
Achsname der Drehachse v_2	\$TC_CARR22	AXIS	0
Kinematiktyp	\$TC_CARR23	CHAR	T
Offset der Drehachse v_1	\$TC_CARR24	REAL	0
Offset der Drehachse v_2	\$TC_CARR25	REAL	0
Winkeloffset der Drehachse v_1 (Hirthverz.)	\$TC_CARR26	REAL	0
Winkeloffset der Drehachse v_2 (Hirthverz.)	\$TC_CARR27	REAL	0
Winkelinkrement Drehachse v_1 (Hirthverz.)	\$TC_CARR28	REAL	0
Winkelinkrement Drehachse v_2 (Hirthverz.)	\$TC_CARR29	REAL	0
Minimalposition Drehachse v_1 (SW-Limit)	\$TC_CARR30	REAL	0
Minimalposition Drehachse v_2 (SW-Limit)	\$TC_CARR31	REAL	0
Maximalposition Drehachse v_1 (SW-Limit)	\$TC_CARR32	REAL	0
Maximalposition Drehachse v_2 (SW-Limit)	\$TC_CARR33	REAL	0

Systemvariablen für den Anwender und für Feinverschiebungen

- \$TC_CARR34 bis \$TC_CARR40
Enthalten Parameter, die dem Anwender zur freien Verfügung stehen.
- \$TC_CARR41 bis \$TC_CARR65
Enthalten Feinverschiebungsparameter, die zu den Werten in den Basisparametern addiert werden können. Der einem Basisparameter zugeordnete Feinverschiebungswert ergibt sich, wenn zur Parameternummer der Wert 40 addiert wird.
- \$TC_CARR47 bis \$TC_CARR54 sowie \$TC_CARR61 bis \$TC_CARR63
Sind nicht definiert und führen bei dem Versuch, lesend oder schreibend darauf zuzugreifen, zu einem Alarm.

Beschreibung	NC-Variable	Sprach-Format	Vorbelegung
Werkzeugträgername	\$TC_CARR34	String[32]	""
Achsname 1	\$TC_CARR35 *)	String[32]	""
Achsname 2	\$TC_CARR36 *)	String[32]	""
Kennung	\$TC_CARR37 *)	INT	0
Positionskomponente X	\$TC_CARR38 *)	REAL	0
Positionskomponente Y	\$TC_CARR39 *)	REAL	0
Positionskomponente Z	\$TC_CARR40 *)	REAL	0
x-Komp. Feinversch. des Offsetvektors l_1	\$TC_CARR41	REAL	0
y-Komp. Feinversch. des Offsetvektors l_1	\$TC_CARR42	REAL	0
z-Komp. Feinversch. des Offsetvektors l_1	\$TC_CARR43	REAL	0
x-Komp. Feinversch. des Offsetvektors l_2	\$TC_CARR44	REAL	0
y-Komp. Feinversch. des Offsetvektors l_2	\$TC_CARR45	REAL	0
z-Komp. Feinversch. des Offsetvektors l_2	\$TC_CARR46	REAL	0
x-Komp. Feinversch. des Offsetvektors l_3	\$TC_CARR55	REAL	0
y-Komp. Feinversch. des Offsetvektors l_3	\$TC_CARR56	REAL	0
z-Komp. Feinversch. des Offsetvektors l_3	\$TC_CARR57	REAL	0
x-Komp. Feinversch. des Offsetvektors l_4	\$TC_CARR58	REAL	0
y-Komp. Feinversch. des Offsetvektors l_4	\$TC_CARR59	REAL	0
z-Komp. Feinversch. des Offsetvektors l_4	\$TC_CARR60	REAL	0
Offset der Feinversch. der Drehachse v_1	\$TC_CARR64	REAL	0
Offset der Feinversch. der Drehachse v_2	\$TC_CARR65	REAL	0
Anmerkungen:			
*)	Die Systemvariablen \$TC_CARR35 bis \$TC_CARR40 werden in den Messzyklen sowie ShopMill und ShopTurn verwendet.		

19.7.2 Kinematische Zusammenhänge und Maschinenaufbau

Darstellung der kinematischen Kette

Zur Beschreibung der kinematischen Zusammenhänge zwischen einem Bezugspunkt und der Werkzeugspitze wird das Konzept der kinematischen Kette verwendet.

Die Kette gibt alle für den Werkzeugträger-Datensatz erforderlichen Angaben schematisch an. Für den konkreten Fall einer bestimmten Kinematik müssen die entsprechenden Komponenten der Kette mit den realen Vektoren, Längen und Winkeln besetzt werden. Die Kette stellt die maximal zulässige Konstellation dar. Einzelkomponenten können in einfacheren Fällen zu Null werden (z. B. Kinematiken, die nur eine oder überhaupt keine Drehachse haben).

An der Maschine müssen keine Achsen vorhanden sein, die Werkzeug und / oder Werkstück drehen. Die Funktion ist auch dann nutzbar, wenn die Orientierungen z. B. manuell durch Handräder oder Umbau eingestellt werden.

Der Maschinenaufbau wird durch folgende Angaben beschrieben:

- Zwei Drehachsen (v_1 und v_2) mit Angabe von jeweils einem Drehwinkel (α_1 bzw. α_2), der bei Drehung im Uhrzeigersinn und Blick in Richtung des Drehvektors positiv zählt.
- Bis zu vier Offsetvektoren (l_1 bis l_4) für relevante Maschinenabmessungen (Achsabstände, Abstände zu Maschinen- oder Werkzeugreferenzpunkten).

Nullvektoren

Die Vektoren v_1 bzw. v_2 können Null sein. Der jeweils zugehörige Drehwinkel (explizit programmiert oder aus dem aktiven Frame berechnet) muss dann, da die Richtung der Drehachse nicht definiert ist, ebenfalls Null sein. Ist diese Bedingung nicht erfüllt, wird bei der Aktivierung des Werkzeugträgers ein Alarm ausgegeben.

Weniger als zwei Drehachsen

Die Möglichkeit, eine Drehachse nicht zu definieren, ist dann sinnvoll, wenn der zu beschreibende Werkzeugträger das Werkzeug nur in einer Ebene drehen kann. Ein sinnvoller minimaler Datensatz kann deshalb aus einem einzigen von Null verschiedenen Eintrag in den Werkzeugträgerdaten bestehen, nämlich einem Wert in einer der Komponenten von v_1 bzw. v_2 zur Beschreibung einer achsparallelen Drehachse, wobei der zugehörige Drehwinkel α_1 bzw. α_2 aus einem Frame bestimmt wird.

Weitere Sonderfälle

Die beiden Vektoren v_1 bzw. v_2 können kollinear sein. Damit geht jedoch ein Freiheitsgrad der Orientierung verloren, d. h. eine derartige Kinematik ist gleichartig zu einer solchen, bei der nur eine Drehachse definiert ist. Alle möglichen Orientierungen liegen dann auf einem Kegelmantel. Der Kegelmantel entartet zu einer Geraden, wenn zusätzlich auch Werkzeugorientierung t und v_1 bzw. v_2 kollinear werden, d. h. in diesem Spezialfall ist deshalb eine Orientierungsänderung nicht mehr möglich. Der Kegelmantel entartet zu einer Kreisfläche (d. h. es sind alle Orientierungen in einer Ebene möglich), wenn Werkzeugorientierung t und v_1 bzw. v_2 aufeinander senkrecht stehen.

Es ist zulässig, dass beide Vektoren v_1 und v_2 Null werden. Eine Orientierungsänderung ist dann allerdings nicht mehr möglich. In diesem Spezialfall wirken die eventuell von Null verschiedenen Längen l_1 und l_2 wie zusätzliche Werkzeuglängenkorrekturen, deren Komponenten in den einzelnen Achsen durch einen Wechsel der Ebene (G17 - G19) nicht beeinflusst werden.

Erweiterungen der Kinematikdaten

- Möglichkeit, unmittelbar auf vorhandene Maschinenachsen zuzugreifen, um über die Rundachspalten die Einstellung der Werkzeugträger zu definieren.
- Erweiterung auf Kinematiken mit drehbarem Werkstück bzw. auf Kinematiken mit drehbarem Werkzeug und drehbarem Werkstück.
- Möglichkeit, für die Rundachspalten nur diskrete Werte in einem Raster zuzulassen (Hirth-Verzahnungen).

Die Erweiterungen sind kompatibel zu früheren Softwareständen und umfassen die Kinematikdatensätze von \$TC_CARR18 bis \$TC_CARR23.

Maschinen mit drehbarem Werkzeug

Bei Maschinen mit drehbarem Werkzeug ändert sich an der Definition der Kinematik gegenüber älteren Softwareständen nichts. Insbesondere hat der neu eingeführte Vektor l_4 keine Bedeutung. Ein eventuell von Null verschiedener Inhalt von l_4 wird nicht ausgewertet.

Der Begriff "Orientierbare Werkzeugträger" ist für die neuen Kinematiktypen, bei denen auch der Werkzeugschulter allein oder auch zusätzlich zum Werkzeug gedreht werden kann, eigentlich nicht mehr angemessen. Er wird jedoch aus Kompatibilitätsgründen beibehalten.

Die kinematischen Ketten zur Beschreibung der Maschine mit drehbarem Werkzeug (allgemeiner Fall) sind im folgenden Bild dargestellt:

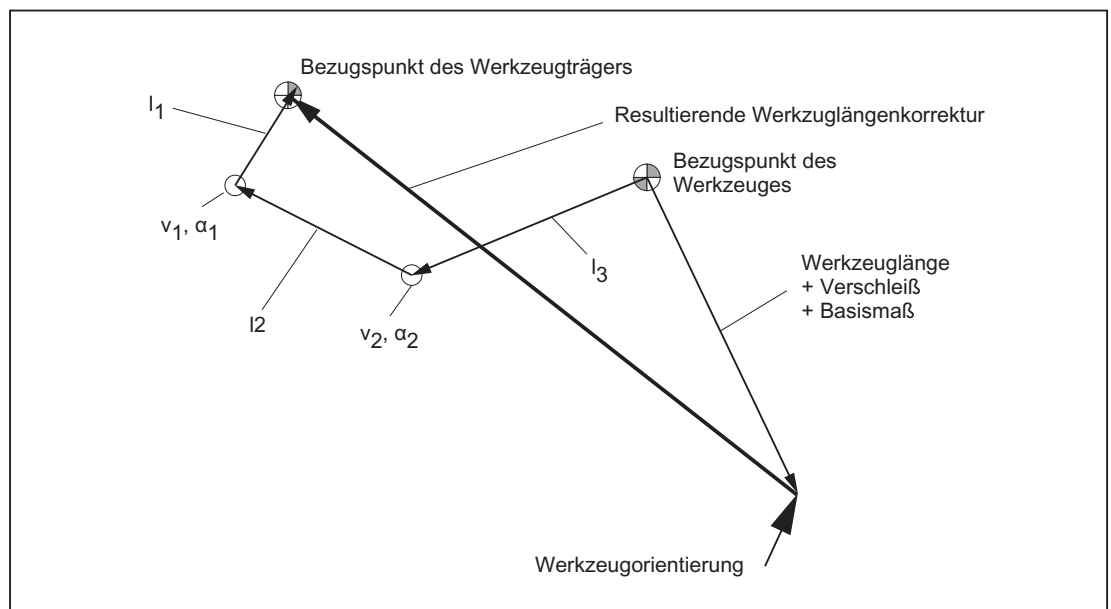


Bild 19-39 Kinematische Kette zur Beschreibung eines Werkzeuges mit Orientierung

Vektoren, die Offsets im drehbaren Kopf beschreiben, sind in Richtung von der Werkzeugspitze zum Bezugspunkt des Werkzeugträgers positiv definiert.

Für Maschinen mit drehbarem Werkzeug wird folgender Kinematiktyp definiert:

\$TC_CARR23 mittels des Buchstabens T

Maschinen mit drehbarem Werkstück

Bei Maschinen mit drehbarem Werkstück hat der Vektor l_1 keine Bedeutung. Ein eventuell von Null verschiedener Inhalt wird nicht ausgewertet.

Die zugehörige kinematische Kette der Maschine mit drehbarem Werkstück ist im folgenden Bild dargestellt.

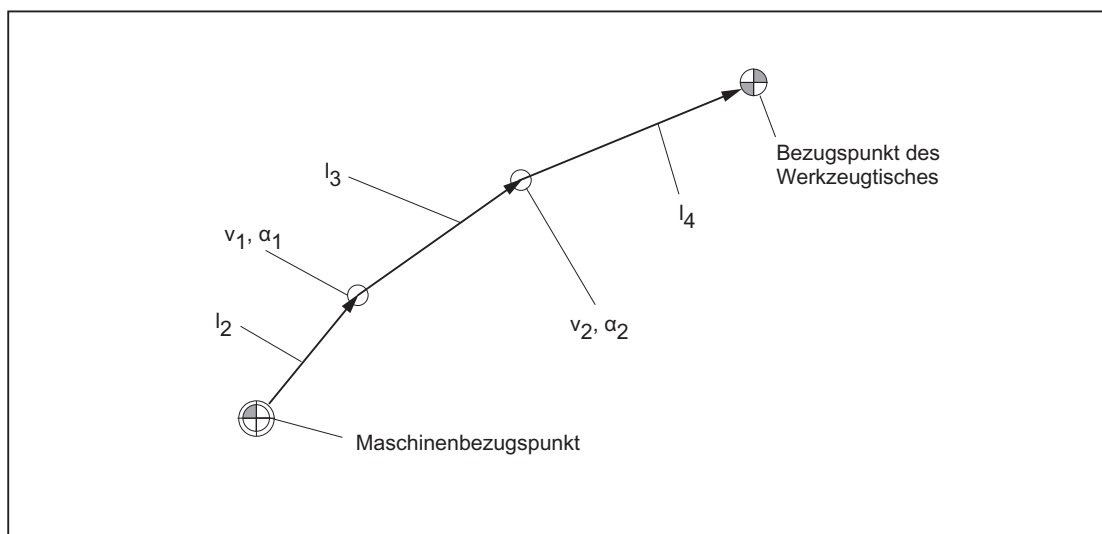


Bild 19-40 Kinematische Kette zur Beschreibung eines drehbaren Werkzeugs

Vektoren, die Offsets im drehbaren Tisch beschreiben, sind in Richtung vom Maschinenbezugspunkt zum Tisch positiv definiert.

Für Maschinen mit drehbarem Werkstück wird folgender Kinematiktyp definiert:

\$TC_CARR23 mittels des Buchstabens P

Hinweis

Bei Maschinen mit drehbarem Werkstück wird es in der Regel sinnvoll sein, den Maschinenbezugspunkt und den Bezugspunkt des Tisches identisch zu wählen. Eine derartige Wahl der Bezugspunkte hat den Vorteil, dass sich die Lage des Werkstücknullpunkts in Grundstellung (d.h. bei nicht gedrehten Rundachsen) beim Aktivieren des drehbaren Tisches nicht verändert. Die (offene) kinematische Kette (siehe Bild) wird dann geschlossen.

In diesem Sonderfall gilt deshalb: $l_2 = - (l_3 + l_4)$

Maschinen mit aufgelöster Kinematik

Bei Maschinen mit aufgelöster Kinematik (sowohl Werkzeug als auch Werkstück sind drehbar) kann jede der beiden Komponenten nur mit einer Achse gedreht werden.

Die Kinematik des drehbaren Werkzeugs wird mit der ersten Drehachse (v_1) und den beiden Vektoren l_1 und l_2 beschrieben, der drehbare Tisch mit der zweiten Drehachse (v_2) und den beiden Vektoren l_3 und l_4 . Die beiden kinematischen Teilketten der Maschine mit drehbarem Werkzeug und drehbarem Werkstück sind im folgenden Bild dargestellt.

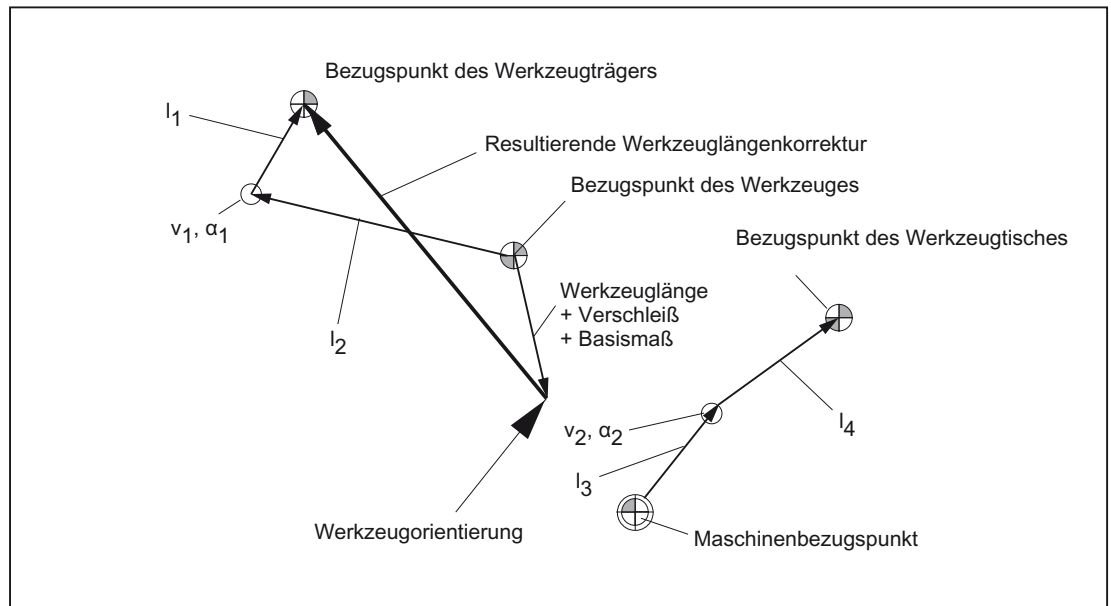


Bild 19-41 Kinematische Kette bei aufgelöster Kinematik

Für Maschinen mit drehbarem Werkzeug und drehbarem Werkstück wird folgender Kinematiktyp definiert:

\$TC_CARR23 mittels des Buchstabens M (aufgelöste Kinematik)

Hinweis

Bei Maschinenkinematiken mit aufgelöster Kinematik wird es aus dem gleichen Grund wie bei Maschinen, bei denen nur der Werkzeugschwenktisch drehbar ist, in der Regel sinnvoll sein, den Maschinenbezugspunkt und den Bezugspunkt des Tisches identisch zu wählen. Die (offene) Teilkette zur Beschreibung des Werkzeugschwenktisches (siehe Bild) wird dann geschlossen.

In diesem Sonderfall gilt: $l_3 = -l_4$

Werkzeug drehbar Typen T und M

Bei Maschinenkinematiken mit einem drehbaren Werkzeug (Typen T und M) wirkt der orientierbare Werkzeugträgeranteil, der den Werkzeug- oder Kopfanteil (im Gegensatz zum Tischanteil) beschreibt, zusammen mit dem aktiven Werkzeug wie ein neues Gesamtwerkzeug.

Feinverschiebung

Die Offsetvektoren I_1 bis I_4 sowie die Offsets der Drehachsen v_1 und v_2 können jeweils als Summe eines Basiswertes und einer Feinverschiebung dargestellt werden. Die den Basiswerten zugeordneten Feinverschiebungsparameter ergeben sich, indem **zum Index des Basiswertes der Wert 40 addiert** wird.

Beispiel:

Dem Parameter \$TC_CARR5 ist der Feinverschiebungswert \$TC_CARR45 zugeordnet.

Hinweis

Zur Bedeutung der für die Feinverschiebung verfügbaren Systemvariablen \$TC_CARR41 bis \$TC_CARR65 siehe:

Literatur:

Programmierhandbuch Arbeitsvorbereitung; Werkzeugkorrekturen

Aktivierung

Mit der folgenden Einstellung werden die Feinverschiebungswerte zu den Basiswerten addiert:

SD42974 \$SC_TOCARR_FINE_CORRECTION = 1 (Feinverschiebung TCARR ein/aus)

Randbedingungen

Der Betrag der zulässigen Feinverschiebung ist begrenzt.

Der maximal zulässige Wert wird festgelegt:

für:	mit dem Maschinendatum:
• die Komponenten der Vektoren I_1 bis I_4 :	MD20188 \$MC_TOCARR_FINE_LIM_LIN
• die Offsets der beiden Drehachsen v_1 und v_2 :	MD20190 \$MC_TOCARR_FINE_LIM_ROT

Ein unzulässiger Feinverschiebungswert wird erst erkannt, wenn:

- ein orientierbarer Werkzeugträger aktiviert wird, der solch einen Wert enthält und
- gleichzeitig folgendes Settingdatum gesetzt ist:
SD42974 \$SC_TOCARR_FINE_CORRECTION

Beschreibung einer Drehung

Ein Datensatz zur Beschreibung einer Drehung besteht aus jeweils einem Vektor v_1 bzw. v_2 zur Angabe der Richtung der Drehachse im Grundzustand und jeweils einem Winkel α_1 bzw. α_2 . Der Drehwinkel wird bei Drehung im Uhrzeigersinn und in Blickrichtung des Drehvektors positiv gezählt.

Die Bestimmung der beiden Werkzeugträgerwinkel α_1 und α_2 mittels eines Frames ist unabhängig von der aktuell angewählten aktiven Ebene (G17 - G19).

Die Werkzeugorientierung in Grundstellung (beide Winkel α_1 und α_2 gleich Null) ist (wie im Standardfall auch):

- bei G17 parallel zu Z.
- bei G18 parallel zu Y.
- bei G19 parallel zu X.

Werkzeugträger Daten zuordnen

Beispiel an einer Maschine mit drehbaren Werkzeugträger

An einer Maschine mit orientierbarem Werkzeugträger vom Kinematiktyp T ergeben sich am dargestellten Fräskopf die folgenden Einstellungen:

Komponente des Offsetvektors $l_1 =$	$(-200, 0, 0)$
Komponente des Offsetvektors $l_2 =$	$(0, 0, 0)$
Komponente des Offsetvektors $l_3 =$	$(-100, 0, 0)$
Komponente der Drehachse $v_1 =$	$(1, 0, 0)$
Komponente der Drehachse $v_2 =$	$(-1, 0, 1)$
Basismaß Bezugspunkt Werkzeug	$(0, 0, 250)$

Hinweis

Der Werkzeug-Bezugspunkt für das Basismaß wird durch den Referenzpunkt an der Maschine festgelegt.

Weitere Informationen zu den Bezugspunkten im Arbeitsraum siehe Kapitel "K2: Achsen, Koordinatensysteme, Frames (Seite 713)".

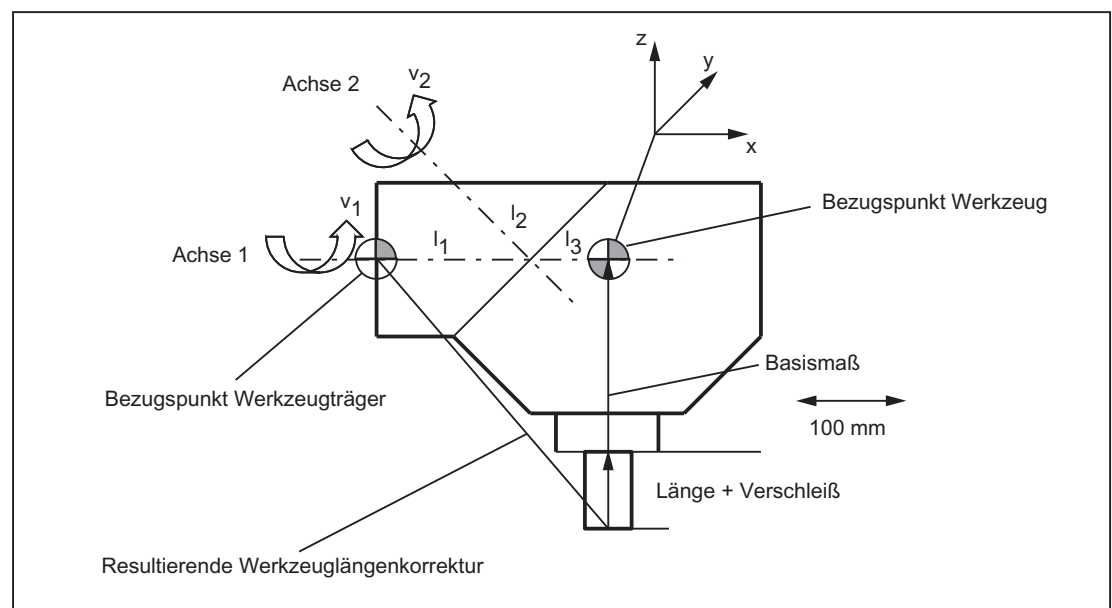


Bild 19-42 Zuordnung der Werkzeugträger-Daten

Für die folgenden Datensatzwerte wurden geeignete Annahmen getroffen:

- Die beiden Drehachsen schneiden sich in einem Punkt.
Deshalb sind alle Komponenten von l_2 Null.
- Die erste Drehachse liegt in der x-z-Ebene, die zweite Rundachse ist parallel zur x-Achse.
Durch diese Bedingungen sind die Richtungen von v_1 bzw. v_2 festgelegt (deren Längen sind - solange sie ungleich Null sind - ohne Bedeutung).
- Der Bezugspunkt des Werkzeugträgers liegt vom Schnittpunkt der beiden Rundachsen aus gesehen um 200 mm in negativer x-Richtung.
Durch diese Bedingung wird l_1 festgelegt.

Zugehörige Datensatzwerte angeben

Für den abgebildeten Werkzeugträger einer Maschine mit drehbarem Werkzeugträger werden folgende zugehörige Datensatzwerte angegeben:

Beschreibung	NC-Variable	Wert
x-Komponente des Offsetvektors l_1	\$TC_CARR1	- 200
y-Komponente des Offsetvektors l_1	\$TC_CARR2	0
z-Komponente des Offsetvektors l_1	\$TC_CARR3	0
x-Komponente des Offsetvektors l_2	\$TC_CARR4	0
y-Komponente des Offsetvektors l_2	\$TC_CARR5	0
z-Komponente des Offsetvektors l_2	\$TC_CARR6	0
x-Komponente der Drehachse v_1	\$TC_CARR7	1
y-Komponente der Drehachse v_1	\$TC_CARR8	0
z-Komponente der Drehachse v_1	\$TC_CARR9	0
x-Komponente der Drehachse v_2	\$TC_CARR10	-1
y-Komponente der Drehachse v_2	\$TC_CARR11	0
z-Komponente der Drehachse v_2	\$TC_CARR12	1
Drehwinkel α_1 (in Grad)	\$TC_CARR13	0
Drehwinkel α_2 (in Grad)	\$TC_CARR14	0
x-Komponente des Offsetvektors l_3	\$TC_CARR15	-100
y-Komponente des Offsetvektors l_3	\$TC_CARR16	0
z-Komponente des Offsetvektors l_3	\$TC_CARR17	0

Erklärungen

Bei der im Beispiel gewählten Werkzeugträgerkinematik bilden die beiden Drehachsen einen Winkel von 45° , so dass die Orientierung nicht jeden beliebigen Wert annehmen kann. Konkret sind in diesem Beispiel keine Orientierungen mit negativer X-Komponente darstellbar.

x-Komponente des Basismaßes:	0
y-Komponente des Basismaßes:	0
z-Komponente des Basismaßes:	250

Hinweis

Aus der Geometrie eines Werkzeugträgers sind die erforderlichen Daten nicht eindeutig zu bestimmen, d. h. der Anwender hat eine gewisse Wahlfreiheit, wo welche Daten abgelegt werden sollen. So besteht im Beispiel die Möglichkeit, für das Basismaß nur eine z-Komponente bis zur zweiten Achse anzugeben. In diesem Fall wäre dann l_2 nicht mehr Null, sondern enthielte die Komponenten des Abstandes dieses Punktes auf der zweiten Achse von einem weiteren Punkt auf der ersten Achse. Dieser Punkt auf der ersten Achse kann ebenfalls frei gewählt werden. Abhängig von der Wahl dieses Punktes muss dann l_1 so gewählt werden, dass der (ebenfalls frei wählbare) Bezugspunkt erreicht wird.

Allgemein gilt: Die Vektorkomponenten, die durch die Drehung einer Achse nicht verändert werden, können beliebig auf die Vektoren "vor" und "nach" der Drehung verteilt werden.

19.7.3 Schrägbearbeitung mit 3 + 2 Achsen

Funktionale Beschreibung

Schrägbearbeitung mit 3 + 2 Achsen beschreibt die Erweiterung des Konzepts der orientierbaren Werkzeugträger sowie die Übertragung dieses Konzepts auf Maschinen mit drehbarem Werkzeugschwenktisch auf solche Maschinen, bei denen die Orientierungen von Werkzeug und Tisch gleichzeitig veränderbar ist.

Die Funktion "Schrägbearbeitung mit 3 + 2 Achsen" dient zur Bearbeitung von Flächen im Raum, die gegenüber den Hauptebenen [X-Y-Ebene (G17), Z-X-Ebene (G18), Y-Z-Ebene (G19)] beliebig gedreht sind.

Eine entsprechende Orientierung des Werkzeugs relativ zum Werkstück kann erreicht werden, indem entweder das Werkzeug, das Werkstück oder sowohl Werkzeug als auch Werkstück gedreht wird.

Die Software berechnet dabei automatisch die sich aus den Werkzeuglängen, Hebelarmen und Drehachswinkeln erforderlichen Ausgleichsbewegungen. Dabei wird stets davon ausgegangen, dass zunächst die erforderliche Orientierung eingestellt und während eines Bearbeitungsganges, z. B. Fräsen einer Tasche auf einer schrägen Ebene, nicht mehr verändert wird.

Weiterhin werden folgende 3 Funktionen beschrieben, die bei der Schrägbearbeitung erforderlich sind:

- **Wegprogrammierung** in Richtung der Werkzeugorientierung unabhängig von einem aktiven Frame
- Definition einer **Framedrehung** durch Angabe der Raumwinkel
- Definition des **Rotationsanteils in Werkzeugrichtung** im programmierbaren Frame unter Erhaltung der übrigen Frameanteile

Abgrenzung zur 5-Achs-Transformation

Wird die Funktionalität benötigt, dass die Werkzeugspitze (TCP-Tool Center Point) bei einer Umorientierung in Bezug auf das Werkstück auch während der Interpolation unverändert bleibt, ist die 5-Achs-Software erforderlich.

Weitere Erläuterungen zur 5-Achs-Transformation siehe:

Literatur:

Funktionshandbuch Sonderfunktionen; 3- bis 5-Achs-Transformation (F2)

Festlegung der orientierbaren Werkzeugträger

Der orientierbare Werkzeugträger wird durch eine allgemeine 5-Achs-Kinematik nachgebildet, die durch einen Datensatz im Werkzeugkorrekturspeicher mit insgesamt 33 REAL-Werten beschrieben wird. Bei einem Werkzeugträger (z. B. einem Fräskopf), der über zwei rotatorische Achsen zur Orientierungseinstellung verfügt, sind 31 dieser Werte konstant.

In der aktuellen SW-Version wird ein Datensatz im Werkzeugkorrekturspeicher mit insgesamt 47-Real-Werte beschrieben. Beim Werkzeugträger, der über zwei rotatorische Achsen zur Orientierungseinstellung verfügt, sind 45 dieser Werte konstant.

Die restlichen beiden Werte sind variabel und dienen zur Festlegung der Orientierung. Die konstanten Werte beschreiben Offsets und die Richtungen und Einstellmöglichkeiten der Drehachsen, die variablen Werte beschreiben die Winkel der Drehachsen.

19.7.4 Maschine mit drehbarem Werkstück

Systemvariablen

Bei der Berechnung der aktiven Werkzeuglänge mit `TCOABS` wurde bisher immer auf die in `$TC_CARR13` bzw. `$TC_CARR14` abgelegten Winkel zugegriffen. Das gilt nach wie vor, wenn `$TC_CARR21` bzw. `$TC_CARR22` nicht auf Rundachsen verweisen. Enthält `$TC_CARR21` bzw. `$TC_CARR22` einen Verweis auf eine Rundachse im Kanal, wird als Winkel anstelle des Eintrags in `$TC_CARR13` bzw. `$TC_CARR14` die Achsposition der betreffenden Achse am Beginn des aktuellen Satzes verwendet.

Eine gemischte Betriebsart ist zulässig, d. h. die Winkel können für eine Achse aus dem Eintrag der Systemvariablen `$TC_CARR13` bzw. `$TC_CARR14` und für die andere Achse aus der Position einer Kanalachse ermittelt werden.

Damit ist es möglich, bei Maschinen, bei denen die Achsen innerhalb der NC bekannt sind, die zur Einstellung des orientierbaren Werkzeugträgers dienen, direkt auf deren Positionen zuzugreifen, während dazu bisher beispielsweise die Systemvariable `$AA_IM[Achse]` gelesen und das Ergebnis der Leseoperation in `$TC_CARR13/14` geschrieben werden musste. Damit entfällt insbesondere auch der implizite Vorlaufstopp beim Lesen der Achspositionen.

MD20180

Die Rundachsposition wird mit ihrem programmierten bzw. berechneten Wert verwendet wenn das Maschinendatum:

MD20180 \$MC_TOCARR_ROT_ANGLE_INCR[i] = 0 (Rundachseninkrement des orientierbaren Werkzeugträgers)

Ist dieses Maschinendatum dagegen ungleich Null, so wird die Position des nächstliegenden Rasterpunktes verwendet, die sich bei geeignet gewähltem ganzzahligem n aus der folgenden Gleichung ergibt:

$$\varphi = \$MC_TOCARR_ROT_ANGLE_OFFSET[i] + n * \$MC_TOCARR_ROT_ANGLE_INCR[i]$$

Diese Funktionalität wird dann benötigt, wenn die Rundachsen indexiert werden müssen und deshalb nicht beliebige Positionen einnehmen können (z. B. bei Hirth-Verzahnungen). In der Systemvariablen \$P_TCANG[i] wird der gerasterte Wert, in der Systemvariablen \$P_TCDIFF[i] die Differenz zwischen exaktem und gerastertem Wert geliefert.

Orientierung Frame TCOFR

Bei TCOFR (Bestimmung der Winkel aus der durch einen aktiven Frame definierten Orientierung) erfolgt die Rasterung nach der Bestimmung der Winkel aus der aktiven Framedrehung. Ist die verlangte Orientierung aufgrund der Maschinenkinematik nicht einstellbar, wird die Bearbeitung mit einem Alarm abgebrochen. Das gilt auch dann, wenn die Sollorientierung sehr nahe bei einer erreichbaren Orientierung liegt. Insbesondere kann in einer solchen Situation der Alarm durch die Winkelraasterung nicht verhindert werden.

TCARR Frameverschiebung

Eine Frameverschiebung als Folge eines Werkzeugträgerwechsels wird sofort bei der Anwahl mit TCARR= . . . wirksam. Dagegen wird eine Änderung der Werkzeuglänge nur dann sofort wirksam, wenn ein Werkzeug aktiv ist.

TCOFR / TCOABS Framedrehung

Eine Framedrehung wird mit der Aktivierung nicht bewirkt, bzw. eine bereits wirksame Drehung wird nicht verändert. Die Stellung der Rundachsen, die für die Berechnung verwendet wird, wird wie im Fall T (nur das Werkzeug ist drehbar) abhängig vom G-Befehl TCOFR / TCOABS aus dem Drehanteil eines aktiven Frames bzw. aus den Einträgen \$TC_CARRn ermittelt.

Durch die Aktivierung eines Frames verändert sich die Position im Werkstückkoordinatensystem entsprechend, ohne dass es zu einer Ausgleichsbewegung der Maschine kommt. Die Verhältnisse sind im folgenden Bild dargestellt:

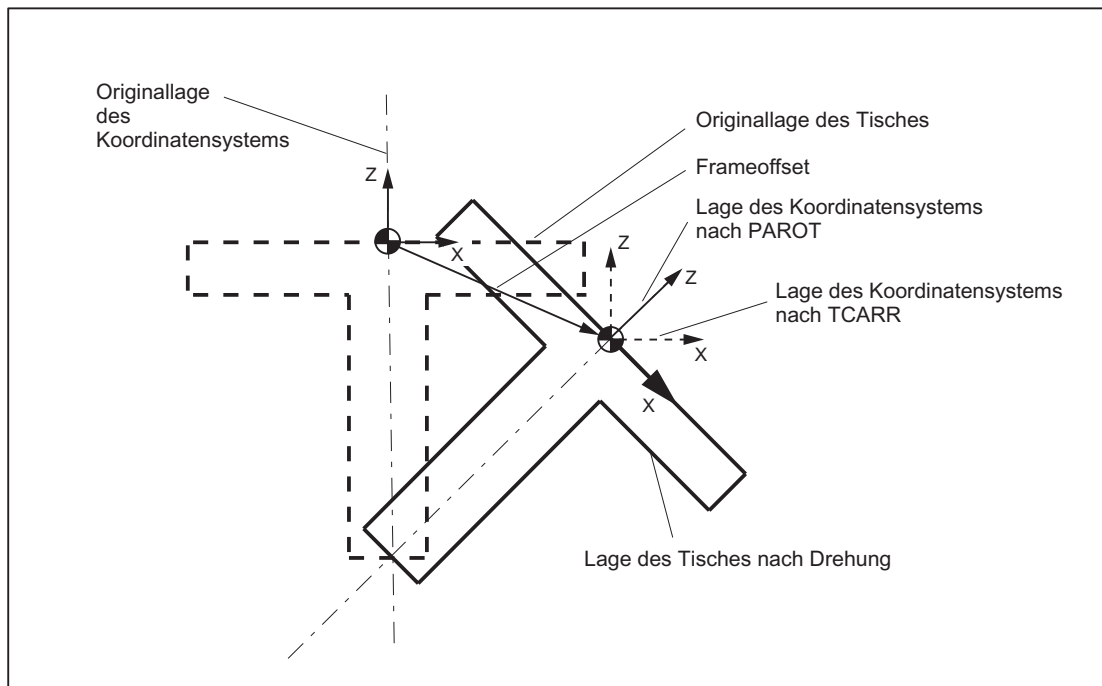


Bild 19-43 Nullpunktverschiebung bei Aktivierung eines drehbaren Werkzeugtisches mit TCARR

Beispiel

Bei der Maschine im Bild zeigt die Drehachse des Tisches in die positive Y-Richtung. Der Tisch ist um +45 Grad gedreht. Mit PAROT wird ein Frame definiert, der ebenfalls eine Drehung von 45 Grad um die Y-Achse beschreibt. Das gegenüber der Außenwelt nicht gedrehte Koordinatensystem (im Bild mit "Lage des Koordinatensystems nach TCARR" gekennzeichnet) ist gegenüber dem mitgeführten Koordinatensystem (Lage nach PAROT) aber um -45 Grad gedreht. Wird dieses Koordinatensystem z. B. mit ROT Y-45 definiert und anschließend der Werkzeugträger bei aktivem TCOFR angewählt, wird für die Drehachse des Werkzeugträgers ein Winkel von +45 Grad ermittelt.

Drehbarer Tisch

Bei drehbaren Tischen (Kinematiktypen P und M) hat die Aktivierung mit TCARR zunächst ebenfalls keine Drehung des Koordinatensystems zur Folge (siehe Bild), d.h. der Nullpunkt des Koordinatensystems verschiebt sich zwar in Bezug auf die Maschine und bleibt fest in Bezug auf den Nullpunkt des Werkstücks, die Orientierung bleibt jedoch unverändert raumfest.

Aktivierung Kinematiktyp P und M

Bei Kinematiken des Typs P und des Typs M wird bei der Anwahl eines Werkzeugträgers ein additiver Frame aktiviert (Tischoffset des orientierbaren Werkzeugträgers), der die Verschiebung des Nullpunktes als Folge der Drehung des Tisches berücksichtigt.

Die Nullpunktverschiebung kann in ein eigenes Systemframe \$P_PARTFR geschrieben werden. Dazu muss das Bit 2 gesetzt werden im Maschinendatum:

MD28082 \$MC_MM_SYSTEM_FRAME_MASK (Systemframes (SRAM))

Das durch das folgende Maschinendatum bezeichnete Basisframe wird dann für die Nullpunktverschiebung nicht mehr benötigt:

MD20184 \$MC_TOCARR_BASE_FRAME_NUMBER (Nummer des Basisframes für Aufnahme des Tischoffsets)

Aktivierung Kinematiktyp M

Bei Kinematiken des Typs M (Werkzeug und Tisch sind jeweils um eine Achse drehbar), bewirkt die Aktivierung eines Werkzeugträgers mit `TCARR` gleichzeitig eine entsprechende Änderung der effektiven Werkzeuglänge (falls ein Werkzeug aktiv ist) und der Nullpunktverschiebung.

Drehungen

Je nach Bearbeitungsaufgabe muss beim Einsatz drehbarer Werkzeugträger bzw. Tische nicht nur eine Nullpunktverschiebung (sei es als Frame oder als Werkzeuglänge), sondern auch eine Drehung berücksichtigt werden. Die Aktivierung eines orientierbaren Werkzeugträgers führt jedoch in keinem Fall unmittelbar zu einer Drehung des Koordinatensystems.

TOROT

Ist nur das Werkzeug drehbar, kann ein Frame mittels `TOFRAME` bzw. `TOROT` definiert werden, dessen Z-Achse in Werkzeugrichtung zeigt.

PAROT

Wird ein Koordinatensystem benötigt, das in Bezug auf das Werkstück fest ist, d. h. gegenüber der Originallage nicht nur verschoben sondern auch entsprechend der Tischdrehung gedreht ist, so kann analog zur Situation bei drehbarem Werkzeug mit `PAROT` eine entsprechende Drehung aktiviert werden.

Mit `PAROT` bleiben Translationen, Skalierungen und Spiegelungen im aktiven Frame erhalten, der Rotationsanteil wird jedoch durch den Rotationsanteil eines orientierbaren Werkzeugträgers, der dem Tisch entspricht, gedreht.

`PAROT` und `TOROT` berücksichtigen in den Fällen, in denen der Tisch bzw. das Werkzeug mit zwei Drehachsen orientiert werden, die gesamte Orientierungsänderung. Bei gemischten Kinematiken wird jeweils nur der entsprechende, durch eine Drehachse verursachte Anteil berücksichtigt. Damit ist es z. B. bei der Verwendung von `TOROT` möglich, ein Werkstück so zu drehen, dass eine schräge Ebene parallel zur raumfesten X-Y-Ebene liegt, bei der Bearbeitung aber eine Drehung des Werkzeugs zu berücksichtigen ist, wenn beispielsweise Bohrungen angebracht werden müssen, die nicht senkrecht zu dieser Ebene sind.

Der Sprachbefehl `PAROT` wird nicht abgelehnt, wenn kein orientierbarer Werkzeugträger aktiv ist. Er bewirkt keine Änderung im programmierten Frame.

Hinweis

Für weitere Erläuterungen zu den Funktionen `TCARR` und `TOROT` sowie `PAROT` bezüglich kanalspezifischer Systemframes siehe Kapitel "K2: Achsen, Koordinatensysteme, Frames (Seite 713)".

19.7.5 Vorgehen bei Benutzung der orientierbaren Werkzeugträger

Neuanlegen eines Werkzeugträgers

Die Anzahl der verfügbaren Werkzeugträgerdatensätze in der NC wird festgelegt mit dem Maschinendatum:

MD18088 \$MN_MM_NUM_TOOL_CARRIER (Maximale Anzahl definierbarer Werkzeugträger)

1. Der Wert wird wie folgt berechnet:
$$\text{MD18088} = \text{"Anzahl TO-Einheiten"} * \text{"Anzahl Werkzeugträgerdatensätze einer TO-Einheit"}$$

Jeder TO-Einheit werden $\text{MD18088} / \text{"Anzahl der TO-Einheiten"}$ fest zugeteilt.

Hinweis

Weitere Erläuterungen zur Definition und Zuordnung einer TO-Einheit durch das Maschinendatum:

MD28085 \$MC_MM_LINK_TOA_UNIT (Zuordnung einer TO-Einheit zu einem Kanal (SRAM))

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Speicherkonfigurationen (S7)

2. Nullsetzen der Werkzeugträgerdaten:
Zum Nullsetzen aller Werte aller Datensätze kann der Befehl `$TC_CARR1[0] = 0` verwendet werden.
Einzelne Werkzeugträgerdatensätze können selektiv mit dem Sprachbefehl `DELTC` bzw. über den PI-Dienst `_N_DELTCAR` gelöscht werden.

3. Zugriff auf die Daten eines Werkzeugträgers:

– Teileprogramm

→ $\$TC_CARRn[m] = \text{Wert}$

Damit wird der bisherige Wert der Systemvariablen n für den Werkzeugträger m mit dem neuen Wert "Wert" beschrieben.

→ $\text{Wert} = \$TC_CARRn[m]$

Mit "def real wert" - damit können die Parameter eines Werkzeugträger m gelesen werden, sofern sie bereits definiert wurden (z.B. MD18088 setzen). Andernfalls wird ein Alarm gemeldet.

– BTSS-Schnittstelle

Die Parameter eines orientierbaren WZ-Trägers können über die Variablendienste NC-HMI (BTSS) mit der Systemvariablen $\$P_TCANG[<n>]$ gelesen und geschrieben werden.

4. Datensicherung:

Die genannten Systemvariablen werden im Rahmen der allgemeinen NC-Datensicherung gesichert.

Werkzeugträger anwählen

Ein Werkzeugträger mit der Nummer m wird mit dem NC-Programmbefehl $TCARR = m$ ausgewählt ($TCARR$ Tool Carrier).

Mit $TCARR = 0$ wird ein aktiver Werkzeugträger ausgewählt.

Neues Werkzeug oder neuer Werkzeugträger

Ein neu aktiviertes Werkzeug wird so behandelt, als wäre es auf den aktiven Werkzeugträger montiert.

Ein neuer Werkzeugträger wird sofort bei seiner Programmierung wirksam, ein Wechsel oder eine Neuprogrammierung des aktiven Werkzeugs ist nicht notwendig. Werkzeugträger(-nummer) und Werkzeug(-nummer) sind voneinander unabhängig und können beliebig miteinander kombiniert werden.

Werkzeugträger aus G-Befehl der Gruppe 42

Werkzeugorientierung absolut $TCOABS$ (Tool Carrier Orientation **ABS**olute):

Die Werkzeugorientierung wird explizit bestimmt, wenn die entsprechenden Werte in die Systemvariablen $\$TC_CARR13$ bzw. $\$TC_CARR14$ eingetragen werden und in der G-Gruppe 42 der G-Befehl $TCOABS$ aktiviert wird.

Werkzeugorientierung Frame $TCOFR$ (Tool Carrier Orientation **FR**ame):

Ebenso kann die Werkzeugorientierung automatisch aus der aktuellen Orientierung eines aktiven Frames bei Werkzeuganwahl bestimmt werden, wenn bei der Werkzeugträgeranwahl in der G-Gruppe 42 einer der folgenden G-Befehle aktiv ist:

- `TCOFR` bzw. `TCOFRZ`
Der orientierbare Werkzeugträger wird so eingestellt, dass das Werkzeug in Z-Richtung zeigt.
- `TCOFRX`
Der orientierbare Werkzeugträger wird so eingestellt, dass das Werkzeug in X-Richtung zeigt.
- `TCOFRY`
Der orientierbare Werkzeugträger wird so eingestellt, dass das Werkzeug in Y-Richtung zeigt.

`TCOFR` bringt zum Ausdruck, dass bei Bearbeitung auf einer schrägen Fläche implizit die Werkzeugkorrekturen so berücksichtigt werden, als ob das Werkzeug auf dieser Fläche senkrecht stünde.

Hinweis

Die Werkzeugorientierung ist nicht starr an die Frameorientierung gebunden. Bei aktivem Frame und bei aktivem G-Befehl `TCOABS` kann ein Werkzeug angewählt werden, bei dem die Orientierung des Werkzeugs von der Orientierung des aktiven Frames unabhängig ist.

Nach der Werkzeuganwahl kann der Anwender den Frame wechseln, welcher die Komponenten der Werkzeuglängenkorrektur nicht beeinflusst. Es ist dann auch nicht mehr sichergestellt, dass das Werkzeug senkrecht auf der Bearbeitungsebene steht. Der Anwender sollte deshalb die Einhaltung der gewollten Werkzeugorientierung auf einer schrägen Fläche vorher kontrollieren.

Die für die Werkzeuglängenberechnung verwendete Werkzeugorientierung wird bei aktivem `TCOFR` usw. bei einem Werkzeugträgerwechsel immer neu aus dem zu diesem Zeitpunkt aktiven Frame ermittelt.

Werkzeugträger aus G-Befehl der Gruppe 53

Die Werkzeugorientierung kann mit Hilfe der G-Befehle der Gruppe 53 (`TOFRAME`, `TOROT` usw.) einen Frame so zu definieren, dass eine Achsrichtung (Z, Y oder X) in diesem Frame gleich der aktuellen Werkzeugorientierung ist.

Für die Werkzeugorientierung ist zum Zeitpunkt des Aufrufs von `TOFRAME` der aktive G-Befehl der Gruppe 6 (`G17` - `G19`) entscheidend.

Ist kein Werkzeugträger aktiv, oder ist ein Werkzeugträger aktiv, der keine Orientierungsänderung des Werkzeugs bewirkt, dann ist die Z-Richtung im neuen Frame:

- bei `G17` gleich der alten Z-Richtung.
- bei `G18` gleich der alten Y-Richtung.
- bei `G19` gleich der alten X-Richtung.

Bei drehenden Werkzeugträgern werden diese Richtungen entsprechend modifiziert. Entsprechendes gilt für die neuen X- bzw. Y-Richtungen.

Statt `TOFRAME` bzw. `TOROT` kann einer der G-Befehle `TOFRAMEX`, `TOFRAMEY`, `TOROTX` oder `TOROTY` verwendet werden. Dann sind die Bedeutungen der Achsen entsprechend vertauscht.

Gruppenwechsel

Ein Wechsel des G-Befehls aus der Gruppe 42 (`TCOABS`, `TCOFR` usw.) bewirkt eine Neuberechnung der relevanten Werkzeuglängenkomponenten.

Die in den Werkzeugträgerdaten abgelegten (programmierten) Drehwinkel werden nicht beeinflusst, so dass bei einem Wechsel von `TCOFR` nach `TCOABS` wieder die ursprünglich in den Werkzeugträgerdaten abgelegten Winkel wirksam werden.

Drehwinkel (α_1 bzw. α_2) lesen:

Die aktuell zur Berechnung der Orientierung verwendeten Winkel können über die Systemvariable `$P_TCANG[n]` mit $n = 1$ oder $n = 2$ gelesen werden.

Gibt es zu einer bestimmten Orientierung zwei zulässige Lösungen (ein zweites gültiges Winkelpaar), so kann auf die entsprechenden Werte mit `$P_TCANG[3]` bzw. `$P_TCANG[4]` zugegriffen werden. Die Anzahl der gültigen Lösungen 0 bis 2 kann mit `$P_TCSOL` gelesen werden.

Werkzeuggradientkorrektur mit `CUT2D` oder `CUT3DFS`:

Bei der Werkzeugradiuskorrektur wird die aktuelle Werkzeugorientierung berücksichtigt, wenn in der G-Gruppe 22 (Werkzeugkorrekturtyp) entweder `CUT2D` oder `CUT3DFS` aktiv ist.

Bei nicht drehenden Werkzeugträgern ist das Verhalten nur von der aktiven Ebene vom G-Befehl der Gruppe 6 (`G17` - `G19`) abhängig und somit identisch zum bisherigen Verhalten.

Alle anderen Werkzeugkorrektur Typen:

Das Verhalten bei allen anderen Werkzeugkorrekturtypen ist unverändert.

Insbesondere wird bei `CUT2DF` und bei `CUT3DEF` die für die Werkzeugradiuskorrektur verwendete Korrekturebene unabhängig von der aktuellen Werkzeugorientierung aus dem aktiven Frame ermittelt. Dabei wird die aktive Ebene (`G17` - `G19`) berücksichtigt, so dass das Verhalten identisch zum bisherigen ist.

Die übrigen beiden G-Befehle der Gruppe 22 `CUT3DC` und `CUT3DF` sind durch die Werkzeugträgerfunktionalität nicht betroffen, da die Werkzeugorientierungsinformation in diesen Fällen von der aktiven kinematischen Transformation bereitgestellt wird.

Zwei Drehachsen

Für zwei Drehachsen existieren allgemein zwei Lösungen. Die Auswahl dieser beiden Lösungspaare wird von der Steuerung selbst so getroffen, dass die Orientierungswinkel, die sich aus dem Frame ergeben, möglichst nahe an den vorgegebenen Winkeln liegen.

Für die Vorgabe der Winkel gibt es zwei Möglichkeiten:

1. Verweist `$TC_CARR21` bzw. `$TC_CARR22` auf eine Rundachse, so wird als Vorgabe die Position dieser Achse am Anfang des Satzes verwendet, in dem der Werkzeugträger aktiviert wird.
2. Verweist `$TC_CARR21` bzw. `$TC_CARR22` nicht auf eine Rundachse, werden die `$TC_CARR13` bzw. `$TC_CARR14` enthaltenen Werte verwendet.

Beispiel

Die Steuerung berechnet zunächst einen Winkel 10 Grad für eine Achse. Der vorgegebene Winkel ist 750 Grad. Zum Ausgangswinkel werden deshalb 720 Grad ($= 2 * 360 \text{ Grad}$) addiert, so dass der endgültige Winkel 730 Grad ist.

Rundachsenoffset

Mit Hilfe der beiden systemvariablen \$TC_CARR24 und \$TC_CARR25 kann ein Offset der Rundachsen vorgegeben werden. Ein Wert ungleich Null in einem dieser Parameter bedeutet, dass die zugehörige Rundachse in Grundstellung nicht die Position Null, sondern die durch den Parameter angegebene Position hat. Alle Winkelangaben beziehen sich dann auf das um diesen Wert verschobene Koordinatensystem.

Beim Wechsel der Bearbeitungsebene (G17 - G19) werden nur die Werkzeuglängenkomponenten des aktiven Werkzeugs vertauscht. Die des Werkzeugträgers werden nicht getauscht. Der resultierende Werkzeuglängenvektor wird dann entsprechend dem aktuellen Werkzeugträger gedreht und gegebenenfalls mit den zum Werkzeugträger gehörenden Offsets modifiziert.

Die Bestimmung der beiden Werkzeugträgerwinkel α_1 und α_2 mittels eines Frames ist unabhängig von der aktuell angewählten aktiven Ebene (G17 - G19).

Grenzwertlimits

Im Systemvariablensatz (\$TC_CARR30 bis \$TC_CARR33) zur Beschreibung des orientierbaren Werkzeugträgers können für jede Drehachse Grenzwinkel (Softwarelimits) angegeben werden. Sind sowohl der Minimal- als auch der Maximalwert Null, werden diese Grenzen nicht ausgewertet.

Ist mindestens einer der beiden Grenzwerte ungleich Null, wird überprüft, ob die zuvor ermittelte Lösung innerhalb der zulässigen Grenzen liegt. Ist das nicht der Fall, wird zunächst versucht, durch Addition oder Subtraktion von Vielfachen von 360 Grad zu der unzulässigen Achsposition eine gültige Einstellung zu erreichen. Ist dies nicht möglich und es existieren zwei verschiedene Lösungen, wird die erste Lösung verworfen und die zweite Lösung verwendet. Die zweite Lösung wird bezüglich der Achsgrenzen wie die erste Lösung behandelt.

Wird die erste Lösung verworfen und stattdessen die zweite Lösung verwendet, werden die Inhalte von \$P_TCANG[1/2] und \$P_TCANG[3/4] getauscht, so dass auch in diesem Fall die tatsächlich verwendete Lösung in \$P_TCANG[1/2] steht.

Die Achsgrenzen werden auch dann überwacht, wenn die Achswinkel nicht berechnet, sondern vorgegeben werden. Dies ist der Fall, wenn bei der Aktivierung eines orientierbaren Werkzeugträgers TCOABS aktiv ist.

19.7.6 Programmierung

Werkzeugträger-Anwahl

Ein Werkzeugträger wird mit **m** Nummer des Werkzeug-Trägers angewählt mit:

TCARR = m

Zugriff auf die Werkzeugträger Datensätze

Aus dem Teileprogramm heraus sind folgende Zugriffe möglich:

Der bisherige Wert des Parameters **n** für den Werkzeugträger **m** wird mit dem neuen "Wert" **beschrieben** mit:

`$TC_CARRn[m] = Wert`

Die Parameter eines Werkzeugträgers **m** können, sofern der Werkzeugträger-Datensatz bereits definiert wurde, **gelesen** werden mit:

`wert = $TC_CARRn[m]` (Wert muss eine REAL-Variable sein)

Die Werkzeugträger-Datensatz-Nr muss in dem Bereich liegen der definiert wird durch das Maschinendatum:

MD18088 \$MN_MM_NUM_TOOL_CARRIER (Gesamtzahl der definierbaren Werkzeugträger-Datensätze)

Für einen NC-Kanal können diese Anzahl geteilt durch die Anzahl der aktiven Kanäle Werkzeugträger-Datensätze definiert werden.

Ausnahme:

Wenn vom Standard abweichende Einstellungen gewählt werden über das Maschinendatum:

MD28085 \$MC_MM_LINK_TOA_UNIT (Zuordnung einer TO-Einheit zu einem Kanal)

Nullsetzen aller Werkzeugträger-Datensätze

Aus dem Teileprogramm heraus können alle Werte aller Werkzeugträger-Datensätze mit einem Kommando gelöscht werden.

`$TC_CARR1[0] = 0`

Vom Anwender nicht gesetzte Werte sind mit Wert=0 vorbelegt.

Aktivierung

Ein Werkzeugträger wird wirksam, wenn sowohl ein **Werkzeugträger** als auch ein **Werkzeug** aktiviert wurden. Die Anwahl eines Werkzeugträgers allein hat keine Wirkung. Die Wirkung der Anwahl eines Werkzeugträgers ist abhängig vom G-Befehl `TCOABS` / `TCOFR` (modale G-Gruppe Werkzeugträger).

Ein Wechsel des G-Befehls in der Gruppe `TCOABS` / `TCOFR` bewirkt eine Neuberechnung der Werkzeuglängenkomponenten, wenn ein Werkzeugträger aktiv ist. Bei `TCOABS` werden die in den Werkzeugträgerdaten abgelegten Werte für die beiden Drehwinkel α_1 und α_2 für die Ermittlung der Werkzeugorientierung verwendet.

Bei `TCOFR` werden die beiden Winkel aus dem aktuellen Frame ermittelt. Die in den Werkzeugträgerdaten abgelegten Werte bleiben jedoch unverändert erhalten. Diese werden darüber hinaus auch dazu verwendet, die Mehrdeutigkeit, die sich bei der Berechnung der Drehwinkel aus einem Frame ergeben kann, aufzulösen. Dabei werden unter mehreren

möglichen Winkeln die gewählt, die von den programmierten Winkeln am wenigsten abweichen.

Hinweis

Weitere Erläuterungen zur Programmierung von Werkzeugkorrekturen mit Werkzeugträgerkinematik und zu den Systemvariablen siehe:

Literatur:

Programmierhandbuch Arbeitsvorbereitung

19.7.7 Randbedingungen und Steuerungsverhalten für Orientierungen

Volle Orientierung

Mit einem gegebenen Datensatz, der eine bestimmte Kinematik beschreibt, sind nur dann alle denkbaren räumlichen Orientierungen darstellbar, wenn:

- die beiden Vektoren v_1 und v_2 , die Drehachsen beschreiben, auch definiert sind (d.h. beide Vektoren müssen ungleich Null sein).
- die beiden Vektoren v_1 und v_2 aufeinander senkrecht stehen.
- die Werkzeugorientierung senkrecht auf der zweiten Drehachse steht.

Nicht definierte Orientierung

Sind diese Bedingungen nicht erfüllt und wird mit `TCOFR` eine durch einen aktiven Frame bestimmte nicht erreichbare Orientierung angefordert, wird ein Alarm ausgegeben.

Abhängigkeiten Vektor / Drehwinkel

Ist ein die Richtung einer Drehachse beschreibender Vektor v_1 oder v_2 gleich Null, muss auch der zugehörige Drehwinkel α_1 bzw. α_2 gleich Null sein. Andernfalls wird ein Alarm ausgegeben. Der Alarm wird erst ausgegeben, wenn der Werkzeugträger aktiv wird, d.h. bei einem Werkzeugträgerwechsel.

Werkzeugfeinkorrektur kombiniert mit Orientierung

Die Kombination von Werkzeugfeinkorrektur und Werkzeugträgern ist **nicht** zulässig. Die Aktivierung der Werkzeugfeinkorrektur bei aktivem Werkzeugträger und umgekehrt die Aktivierung eines Werkzeugträgers bei aktiver Werkzeugfeinkorrektur führt zu einem Alarm.

Automatische Werkzeugträger Anwahl, RESET

Bei `RESET` oder bei Programmstart kann automatisch ein Werkzeugträger angewählt werden über das Maschinendatum:

MD20126 \$MC_TOOL_CARRIER_RESET_VALUE (Wirksamer Werkzeugträger bei RESET)

Er wird analog zu der gesteuerten Anwahl eines Werkzeugs behandelt über das Maschinendatum:

MD20120 \$MC_TOOL_RESET_VALUE (Werkzeuglängenkor. Hochlauf (RESET/TP-Ende))

Das Verhalten bei **RESET** oder Programmstart wird wie bei Werkzeuganwahl gesteuert über das gleiche Bit 6 in den Maschinendaten:

MD20110 \$MC_RESET_MODE_MASK (Festlegung der Steuerungsgrundstellung nach RESET/TP-Ende)

Oder:

MD20112 \$MC_START_MODE_MASK (Festlegung der Steuerungsgrundstellung bei NC-START)

Für weitere Informationen siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)".

Ab SW 6.3

War bei der letzten Anwahl vor Reset **TCOABS** aktiv, ist das Verhalten gegenüber den Vorgängerständen unverändert. Ein anderer aktiver G-Codes führt dazu, dass der orientierbare Werkzeugträger mit dem Frame aktiviert wird, der zum letzten Anwahlzeitpunkt vor Reset aktiv war. Dabei werden auch geänderte Werkzeugträgerdaten (\$TC_CARR...) berücksichtigt. Sind diese Daten unverändert, wird der Werkzeugträger wieder im exakt gleichen Zustand wie vor Reset aktiviert. Wurden nach der letzten Werkzeugträgeranwahl vor Reset Werkzeugträgerdaten verändert, ist eine Anwahl in Übereinstimmung mit dem alten Frame nicht immer möglich. In diesem Fall wird der orientierbare Werkzeugträger entsprechend den zu diesem Zeitpunkt gültigen Werten des G-Codes der Gruppe 42 und der aktiven Frames angewählt.

MD22530 Hilfsfunktionsausgabe an PLC

Dass bei Anwahl des Werkzeugträgers wahlweise ein konstanter oder ein M-Code, dessen Nummer von der Werkzeugträgennummer abgeleitet wird, ausgegeben wird kann eingestellt werden mit dem Maschinendatum:

MD22530 \$MC_TOCARR_CHANGE_M_CODE (M-Code bei Werkzeugträgerwechsel)

Für weitere Informationen siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)".

Werkzeugträgerkinematik

Für die Werkzeugträgerkinematik sind folgende Randbedingungen zu beachten:

- Die Werkzeugorientierung in Grundstellung, beide Winkel α_1 und α_2 sind Null, ist wie im Standardfall auch bei:
 - G17 parallel zu Z
 - G18 parallel zu Y
 - G19 parallel zu X
- Bezüglich der Achsgrenzwerte muss eine zulässige Stellung erreichbar sein.
- Nur wenn die beiden Rundachsen aufeinander senkrecht stehen sind auch beliebige Orientierungen einstellbar.
Zusätzlich muss bei Maschinen, bei denen beide Achsen den Tisch drehen, die Werkzeugorientierung senkrecht auf der ersten Drehachse stehen.
Bei Maschinen mit gemischter Kinematik muss die Werkzeugorientierung senkrecht auf der Achse stehen, die das Werkzeug dreht, d.h. ebenfalls auf der ersten Drehachse.

Für eine in einem **Frame** festgelegte Orientierung gilt:

- Die in einem Frame festgelegte Orientierung muss mit der definierten Werkzeugträgerkinematik erreichbar sein, sonst wird ein Alarm ausgegeben.
Dieser Fall kann immer dann auftreten, wenn die beiden Drehachsen, die zur Definition der Kinematik notwendig sind, nicht aufeinander senkrecht stehen.
Dies gilt, wenn weniger als zwei Drehachsen definiert sind und ist der Fall:
 - beim **Kinematiktyp T mit drehbarem Werkzeug**, wenn die Werkzeugachse, die die Werkzeugrichtung definiert, nicht senkrecht auf der **zweiten** steht.
 - beim **Kinematiktyp M und P mit drehbarem Werkstück**, wenn die Werkzeugachse, die die Werkzeugrichtung definiert, nicht senkrecht auf der **ersten** steht.
- Drehachsen, die zum Erreichen einer bestimmten Position einen Frame mit vorgegebener Werkzeugorientierung erfordern, sind nur im Falle einer Drehachse eindeutig bestimmt.
Für zwei Drehachsen existieren allgemein zwei Lösungen.
- In allen Fällen, in denen eine Mehrdeutigkeit zu berücksichtigen ist, ist es besonders wichtig, dass in den Werkzeugdaten die ungefähr aus den Framedaten zu erwartenden Winkel abgelegt werden, bzw. dass die Drehachsen in der Nähe der zu erwartenden Positionen stehen.

Verhalten bei ASUP, REPOS

Der Werkzeugträger kann in einem asynchronen Unterprogramm ASUP gewechselt werden. Bei Fortsetzung des unterbrochenen Programms mit **REPOS** wird bei der Anfahrbewegung der neue Werkzeugträger berücksichtigt und das Programm mit diesem fortgesetzt. Auch hier ist die Behandlung analog zu der des Werkzeugwechsels in einem ASUP.

Für weitere Informationen siehe Kapitel "K1: BAG, Kanal, Programmbetrieb, Reset-Verhalten (Seite 477)".

19.7.8 Beispiele

19.7.8.1 Beispiel: Orientierbare Werkzeugträger

Anforderung

Im nachfolgenden Beispiel wird ein Werkzeugträger verwendet, der durch eine Drehung um die Y-Achse vollständig beschrieben wird. Es reicht deshalb aus, mit dem Eintrag eines Wertes die Drehachse zu definieren (Satz N20).

In den Sätzen N50 bis N70 wird ein Schaftfräser mit der Länge 20 mm und dem Radius 5 mm beschrieben.

Im Satz N90 wird eine Drehung von 37 Grad um die Y-Achse definiert.

Im Satz N120 wird die Werkzeugradiuskorrektur aktiviert, und es werden alle Einstellungen vorgenommen um die in den nachfolgenden Sätzen beschriebene Korrektur unter einer Drehung von 37 Grad um die y-Achse bearbeiten zu können.

```

N10                                ; Definition des Werkzeugtraegers 1
N20 $TC_CARR8[1] = 1              ; Komponente der ersten Drehachse in Y-
                                   Richtung
N30
N40                                ; Definition des Werkzeugkorrekturspei-
                                   chers T1/D1
N50 $TC_DP1[1,1] = 120            ; Schaftfraeser
N60 $TC_DP3[1,1] = 20             ; Laenge 1
N70 $TC_DP6[1,1] = 5              ; Radius
N80
N90 ROT Y37                        ; 37 Grad-Drehung um y-Achse
N100
N110 X0 Y0 Z0 F10000
N120 G42 CUT2DF TCOFR TCARR = 1 T1 D1 X10
N130 X40
N140 Y40
N150 X0
N160 Y0
N170 M30

```

19.7.8.2 Beispiel Orientierbare Werkzeugträger mit drehbarem Tisch

Verwendung des MOV-Befehls

Für die Verwendung des MOV-Befehls wird angenommen, dass das Programm auf einer 5-Achs-Maschine abläuft, bei der eine Drehung der B-Achse das Werkzeug um die Y-Achse dreht:

```

N10 TRAORI()

```

N20 X0 X0 Z0 B45 F2000	; Einstellen der Werkzeugorientierung
N30 MOVT=-10	; Zustellbewegung 10mm in Werkzeugrichtung
	; (unter 45 Grad in der Y-Z-Ebene)
N40 MOVT=AC(20)	; Rueckzug in Werkzeugrichtung auf Abstand
	; 20mm vom Nullpunkt

Maschine mit drehbarem Tisch

Vollständige Definition für die Verwendung eines orientierbaren Werkzeugträgers mit drehbarem Tisch:

N10 \$TC_DP1[1,1]= 120	
N20 \$TC_DP3[1,1]= 13	; Werkzeuglänge 13mm
; Definition des Werkzeugträgers 1:	
N30 \$TC_CARR1[1] = 0	; X-Komponente des 1. Offsetvektors
N40 \$TC_CARR2[1] = 0	; Y-Komponente des 1. Offsetvektors
N50 \$TC_CARR3[1] = 0	; Z-Komponente des 1. Offsetvektors
N60 \$TC_CARR4[1] = 0	; X-Komponente des 2. Offsetvektors
N70 \$TC_CARR5[1] = 0	; Y-Komponente des 2. Offsetvektors
N80 \$TC_CARR6[1] = -15	; Z-Komponente des 2. Offsetvektors
N90 \$TC_CARR7[1] = 1	; X-Komponente der 1. Achse
N100 \$TC_CARR8[1] = 0	; Y-Komponente der 1. Achse
N110 \$TC_CARR9[1] = 0	; Z-Komponente der 1. Achse
N120 \$TC_CARR10[1] = 0	; X-Komponente der 2. Achse
N130 \$TC_CARR11[1] = 1	; Y-Komponente der 2. Achse
N140 \$TC_CARR12[1] = 0	; Z-Komponente der 2. Achse
N150 \$TC_CARR13[1] = 30	; Drehwinkel der 1. Achse
N160 \$TC_CARR14[1] = -30	; Drehwinkel der 2. Achse
N170 \$TC_CARR15[1] = 0	; X-Komponente des 3. Offsetvektors
N180 \$TC_CARR16[1] = 0	; Y-Komponente des 3. Offsetvektors
N190 \$TC_CARR17[1] = 0	; Z-Komponente des 3. Offsetvektors
N200 \$TC_CARR18[1] = 0	; X-Komponente des 4. Offsetvektors
N210 \$TC_CARR19[1] = 0	; Y-Komponente des 4. Offsetvektors
N220 \$TC_CARR20[1] = 15	; Z-Komponente des 4. Offsetvektors
N230 \$TC_CARR21[1] = A	; Bezug fuer 1. Achse

```

N240 $TC_CARR22[1] = B                ; Bezug fuer 2. Achse
N250 $TC_CARR23[1] = "P"              ; Typ des Werkzeugtraegers

N260 X0 Y0 Z0 A0 B45 F2000
N270 TCARR=1 X0 Y10 Z0 T1 TCOABS
N280 PAROT
N290 X0 Y0 Z0
N300 G18 MOV T=AC(20)
N310 G17 X10 Y0 Z0
N320 MOV T=-10
N330 PAROTOF
N340 TCOFR
N350 X10 Y10 Z-13 A0 B0
N360 ROT S X-45 Y45
N370 X20 Y0 Z0 D0
N380 Y20
N390 X0 Y0 Z20
N400 M30

```

Die Definition des orientierbaren Werkzeugträgers ist vollständig angegeben. Die Komponenten, die den Wert 0 enthalten, müssten nicht angegeben werden, da sie ohnehin mit Null vorbesetzt sind.

In N270 wird der Werkzeugträger aktiviert.

Da in *\$TC_CARR21* bzw. in *\$TC_CARR22* auf die Maschinenachsen A und B verwiesen wird und TCOABS aktiv ist, werden die Einträge in *\$TC_CARR13* und *\$TC_CARR14* ignoriert, d.h. für die Drehung werden die Achsposition A0 B45 verwendet.

Die Drehung des 4. Offsetvektors (Länge 15 mm in Z-Richtung) um die B-Achse bewirkt eine Verschiebung des Nullpunktes um X10.607 [= 15 * sin(45)] und Z-4.393 [= -15 * (1. - cos(45))]. Diese Nullpunktverschiebung wird durch einen automatisch beschriebenen Basis- bzw. Systemframe berücksichtigt, so dass die Position X10.607 Y10.000 Z8.607 angefahren wird. In Z-Richtung ergibt sich dabei durch die Werkzeuganwahl eine zusätzliche Verschiebung von 13 mm, die Y-Komponente wird durch die Tischdrehung nicht beeinflusst.

In N280 wird eine Rotation entsprechend der Tischdrehung des orientierbaren Werkzeugträgers definiert. Die neue X-Richtung zeigt deshalb in Richtung der Winkelhalbierenden im 4. Quadranten, die neue Z-Achse in Richtung der Winkelhalbierenden im 1. Quadranten.

In N290 wird der Nullpunkt angefahren, d.h. die Maschinenposition X10.607 Y0 Z-4.393, da die Lage des Nullpunkts durch die Drehung nicht verändert wird.

In N300 wird in Y auf die Position Y33.000 verfahren, da G18 aktiv ist und die Y-Komponenten durch den aktiven Frame nicht beeinflusst werden. X- und Z-Position bleiben unverändert.

In N310 wird die Position X17.678 Y0 Z1.536 angefahren.

In N320 verändert sich als Folge des MOV T-Befehls nur die Z-Position auf den Wert -8.464. Da nur der Tisch drehbar ist, bleibt die Werkzeugorientierung unverändert parallel zur

Maschinen-Z-Richtung, auch wenn die Z-Richtung des aktiven Frames um 45 Grad gedreht ist.

N330 löscht den Basis- bzw. Systemframe, die Framedefinition aus N280 wird damit wieder rückgängig gemacht.

In N340 wird mit TCOFR angegeben, dass der orientierbare Werkzeugträger entsprechend dem aktiven Frame ausgerichtet werden soll. Da wegen des PAROTOF-Befehls in N330 keine Drehung mehr aktiv ist, ergibt sich als Resultat die Grundstellung. Die Frameverschiebung wird 0.

In N350 wird deshalb die Position X10 X10 Z0 (= Z-13 + Werkzeuglänge) angefahren. Achtung: Durch die gleichzeitige Programmierung der beiden Rundachsen A und B wird die tatsächliche Lage des orientierbaren Werkzeugträgers mit der in N340 verwendeten in Übereinstimmung gebracht. Die Position, die von den drei Linearachsen angefahren wird ist davon jedoch unabhängig.

In N360 wird mit Hilfe von Raumwinkeln eine Ebene definiert deren Schnittgeraden in der X-Z- und in der Y-Z-Ebene jeweils einen Winkel von +45 Grad bzw. -45 Grad mit der X- bzw. der Y-Achse bilden. Die so definierte Ebene hat damit folgende Lage: die Flächennormale zeigt in Richtung der Raumdiagonalen.

In N370 wird im neuen Koordinatensystem auf die Position X20 Y0 Z0 verfahren. Da gleichzeitig das Werkzeug mit D0 abgewählt wird, gibt es in Z keinen zusätzlichen Offset mehr. Da die neue X-Achse in der alten X-Z-Ebene liegt, wird in diesem Satz die Maschinenposition X14.142 Y0 Z-14.142 erreicht.

In N380 verfährt im gedrehten Koordinatensystem nur die Y-Achse. Das führt zu einer Bewegung aller drei Maschinenachsen. Die Maschinenposition ist X5.977 Y16.330 Z-22.307.

In N390 wird ein Punkt auf der neuen Z-Achse angefahren. Bezogen auf die Maschinenachsen liegt er deshalb auf der Raumdiagonalen. Alle drei Achsen erreichen deshalb die Position 11.547.

19.7.8.3 Korrekturen einsatzort- und werkstückspezifisch einrechnen

Werkzeug mit Adapter

Im nachfolgenden Programmbeispiel ist ein Werkzeug mit Adapter und orientierbarem Werkzeugträger definiert. Um die Verhältnisse überschaubar zu halten, ist beim Werkzeug selbst, bei den Summen- und Einsatzkorrekturen sowie beim Adapter jeweils nur die Länge L1 von Null verschieden. Die Offsetvektoren des orientierbaren Werkzeugträgers sind alle Null.

N10	\$TC_TP2[1] = "MillingTool"	; Bezeichner
N20	\$TC_TP7[1] = 9	; Platztype
N30	\$TC_TP8[1] = 2	; Status : Freigeben und nicht gesperrt
; D corr. D=1		
N40	\$TC_DP1[1,1]=120	; WZ-Typ - Fräsen
N50	\$TC_DP3[1,1]=; Längenkorrekturvektor	
N60	\$TC_DP12[1,1]=; Verschleiß	

N70	\$TC_SCP13[1,1]=0.1	; Summenkoorektur DL=1
N80	\$TC_ECP13[1,1]=0.01	; Einsatzkorrektur DL=1
N90	\$TC_ADPTT[1]=5	; Adaptertransformation
N100	\$TC_ADPT1[1]=0.001	; Adaptermaß
		; Magazinedaten
N110	\$TC_MAP1[1]=3	; Art des Magazins: Revolver
N120	\$TC_MAP2[1]="Revolver"	; Name eines Magazins
N130	\$TC_MAP3[1]=17	; Zustand des Magazins
N140	\$TC_MAP6[1]=1	; Dimension - Zeile
N150	\$TC_MAP7[1]=2	; Dimension - Spalte -> 2 Plätze
N160	\$TC_MPP1[1,1]=1	; Platzart
N170	\$TC_MPP2[1,1]=9	; Platztype
N180	\$TC_MPP4[1,1]=2	; Platzzustand
N190	\$TC_MPP7[1,1]=1	; Adapter auf Platz bringen
N200	\$TC_MPP6[1,1]=1	; T-Nummer "MillingTool"
N210	\$TC_MAP1[9999]= 7	; Art des Magazins: Zw.speicher
N220	\$TC_MAP2[9999]="Zw.speicher"	; Name eines Magazins
N230	\$TC_MAP3[9999]=17	; Zustand des Magazins
N240	\$TC_MAP6[9999]=1	; Dimension - Zeile
N250	\$TC_MAP7[9999]=1	; Dimension - Spalte -> 1 Platz
N260	\$TC_MPP1[9999,1]=2	; Platzart
N270	\$TC_MPP2[9999,1]=9	; Platztype
N280	\$TC_MPP4[9999,1]=2	; Platzzustand
N290	\$TC_MPP5[9999,1]=1	; Spindelnr. 1
N300	\$TC_MDP2[1,1]=0	; Distanz Spindel zu Mag.1
		; Definition des Werkzeugträgers 1
N310	\$TC_CARR10[1] = 1	; Komponente 2. Drehachse in X-Richtung
N320	\$TC_CARR14[1] = 45	; Drehwinkel der 2. Achse
N330	\$TC_CARR23[1] = "T"	; Tool Mode
N340	Stopre	
N350	\$SC_WEAR_TRANSFORM = 'B101'	
N360	T0 D0 DL=0	
N370	ROT X30	
N380	G90 G1 G17 F10000 X0 Y0 Z0	
N390	T="MillingTool" X0 Y0 Z0 TOWSTD	; X 0.000 Y11.110 Z 0.001
N400	T="MillingTool" X0 Y0 Z0 TOWMCS	; X 0.000 Y10.100 Z 1.011
N410	T="MillingTool" X0 Y0 Z0 TOWWCS	; X 0.000 Y 9.595 Z 0.876
N420	TCARR=1 X0 Y0 Z0	; X 0.000 Y 6.636 Z 8.017
N430	G18 X0 Y0 Z0	; X10.100 Y-0.504 Z 0.876
N440	m30	

Erläuterungen zum Beispiel oben

Beginnend mit Satz N390 wird in unterschiedlichen Varianten jeweils die Position $X_0 \ Y_0 \ Z_0$ angefahren. Die erreichten Maschinenpositionen sind in den Sätzen jeweils im Kommentar angegeben. Im Anschluss an das Programm wird erläutert, wie die Positionen zu Stande kommen.

N390: Die Adaptertransformation 5 (Satz N90) transformiert die Länge L_1 in die Länge L_2 . Dieser Transformation ist lediglich das Adaptermaß selbst nicht unterworfen. Damit ergibt sich der Y-Wert (L_2 bei G17) aus der Summe aus Werkzeuglänge (10), Werkzeugverschleiß (1), Summenkorrektur (0.1) und Einsatzkorrektur (0.01). Das Adaptermaß (0.001) steht in Z (L_1).

N400: In Satz N350 wurden die Bits 0 und 2 gesetzt im Settingdatum:

SD42935 \$SC_WEAR_TRANSFORM (Transformationen für Werkzeugkomponenten)

Dies bedeutet, dass wegen TOWMCS in Satz N400 der Werkzeugverschleiß und die Einsatzkorrektur der Adaptertransformation nicht unterworfen werden. Die Summe dieser beiden Korrekturen ist 1.01. Um dieses Maß erhöht sich deshalb die Z-Position und vermindert sich entsprechend die Y-Position gegenüber Satz N390.

In N410 ist TOWMCS aktiv. Die Summe aus Werkzeugverschleiß und Einsatzkorrektur wirkt deshalb im aktiven Werkstückkoordinatensystem. In Satz N370 wurde eine Drehung um die X-Achse um 30 Grad aktiviert. Der ursprüngliche Korrekturwert von 1.01 in Z-Richtung ergibt deshalb eine neue Z-Komponente von $0.875 (= 1.01 * \cos(30))$ und eine neue Y-Komponente von $-0.505 (= 1.01 * \sin(30))$. Zu der, wie in Satz N390, gebildeten Summe aus Werkzeuglänge, Summenkorrektur und Adaptermaß addiert, ergibt sich das im Programm-Kommentar angegebene Maß.

In N420 wird zusätzlich ein orientierbarer Werkzeugträger aktiviert. Dieser führt eine Drehung um 45 Grad um die X-Achse aus (s. N310 - N330). Da alle Offsetvektoren des Werkzeugträgers Null sind, ergibt sich keine zusätzliche Nullpunktverschiebung. Der orientierbare Werkzeugträger wirkt auf die die Summe aus Werkzeuglänge, Summenkorrektur und Adaptermaß. Der resultierende Teilvektor ist $X_0 \ Y_{7.141} \ Z_{7.142}$. Darauf wird wie in Satz N410 die im WKS bewertete Summe aus Werkzeugverschleiß und Einsatzkorrektur addiert.

In N430 wird G18 aktiviert. Die Komponenten der Summe aus Werkzeuglänge, Summenkorrektur und Adaptermaß werden entsprechend getauscht. Auf diesen neuen Vektor wirkt der orientierbare Werkzeugträger unverändert (Drehung um X-Achse um 45 Grad). Dabei ergibt sich der resultierende Teilvektor $X_{10.100} \ Y_{0.0071} \ Z_{0.0071}$. Der aus Werkzeugverschleiß und Einsatzkorrektur gebildete Vektor ($X_0 \ Y_{-0.505} \ Z_{0.875}$) wird durch den Wechsel der Ebene nicht beeinflusst. Die Summe beider Vektoren ergibt das im Kommentar von N430 angegebene Maß.

19.8 Schneidendaten-Modifikation bei drehbaren Werkzeugen

19.8.1 Funktion

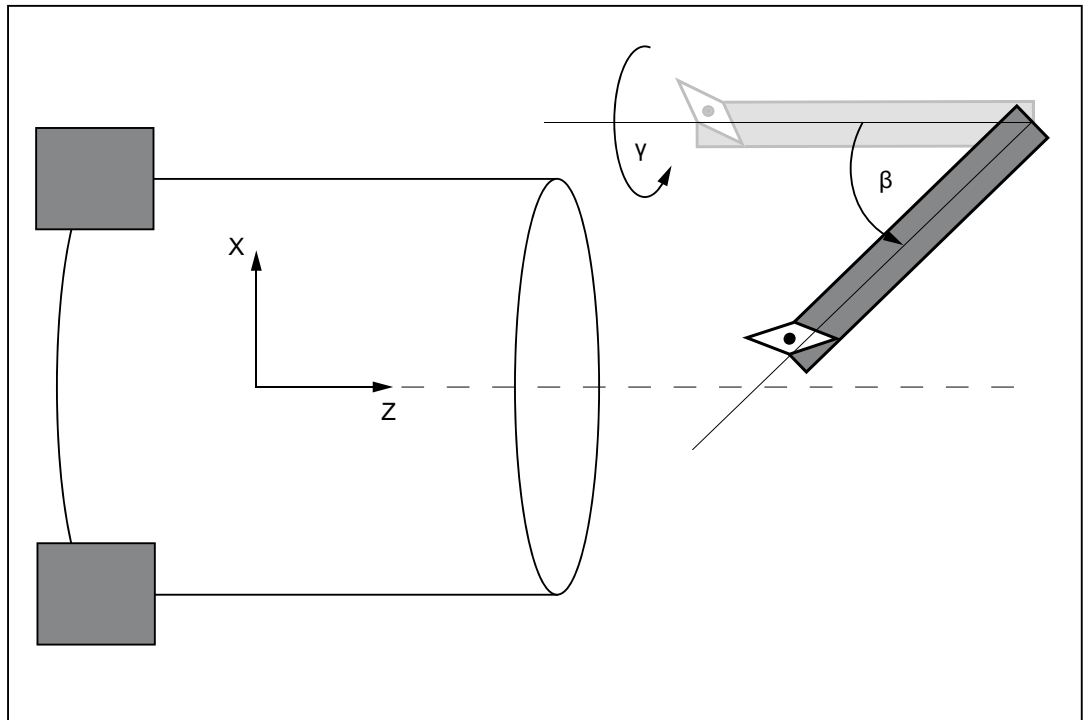
Mit der Funktion "Schneidendaten-Modifikation bei drehbaren Werkzeugen" können die veränderten geometrischen Verhältnisse, die sich bei der Drehung von Werkzeugen (vorwiegend Drehwerkzeuge, aber auch Bohr- und Fräswerkzeuge) relativ zum bearbeiten Werkstück ergeben, berücksichtigt werden.

19.8.2 Ermittlung des Drehwinkels

Die aktuelle Drehung des Werkzeugs wird immer aus einem aktuell aktiven orientierbaren Werkzeugträger ermittelt (siehe Kapitel "Orientierbare Werkzeugträger (Seite 1557)").

Die Definition der Drehwinkel des orientierbaren Werkzeugträgers erfolgt dabei in der Regel (aber nicht notwendigerweise) mit dem Befehl `TCOFR` aus einem aktiven Frame. Durch diese Vorgehensweise ist es möglich, die Werkzeugorientierung unabhängig von der tatsächlichen Kinematik, mit der das Werkzeug gedreht wird, immer auf gleiche Art mit Hilfe zweier Winkel zu definieren.

Zur Definition der Werkzeugdrehung werden die beiden maschinenunabhängigen Ausrichtwinkel β (Beta) und γ (Gamma) verwendet. β ist dabei der Drehwinkel um die Applikate (typischerweise eine B-Achse bei `G18`) und γ eine Drehung um die Abszisse (typischerweise eine C-Achse bei `G18`). Zuerst wird die Drehung um γ ausgeführt, anschließend die um β , d. h. die γ -Achse wird von der β -Achse gedreht:



19.8.3 Schneidenlage, Schnittrichtung und Winkel bei Drehwerkzeugen

Drehwerkzeuge

Unter Drehwerkzeugen werden im Folgenden Werkzeuge verstanden, deren Werkzeugtyp (\$TC_DP1) Werte im Bereich 500 bis 599 hat. Schleifwerkzeuge (Werkzeugtypen 400 bis 499) sind den Drehwerkzeugen gleichgestellt.

Werkzeuge werden unabhängig vom Werkzeugtyp wie Drehwerkzeuge behandelt, wenn:

SD42950 \$SC_TOOL_LENGTH_TYPE = 2

Schneidenlage und Schnittrichtung

Drehwerkzeuge sind durch ihre Haupt- und Nebenschneiden begrenzt. Der Werkzeug-Parameter "Schneidenlage" wird durch die Lage dieser beiden Schneiden relativ zu den Koordinatenachsen definiert. Die Verhältnisse sind in dem folgenden Bild schematisch dargestellt:

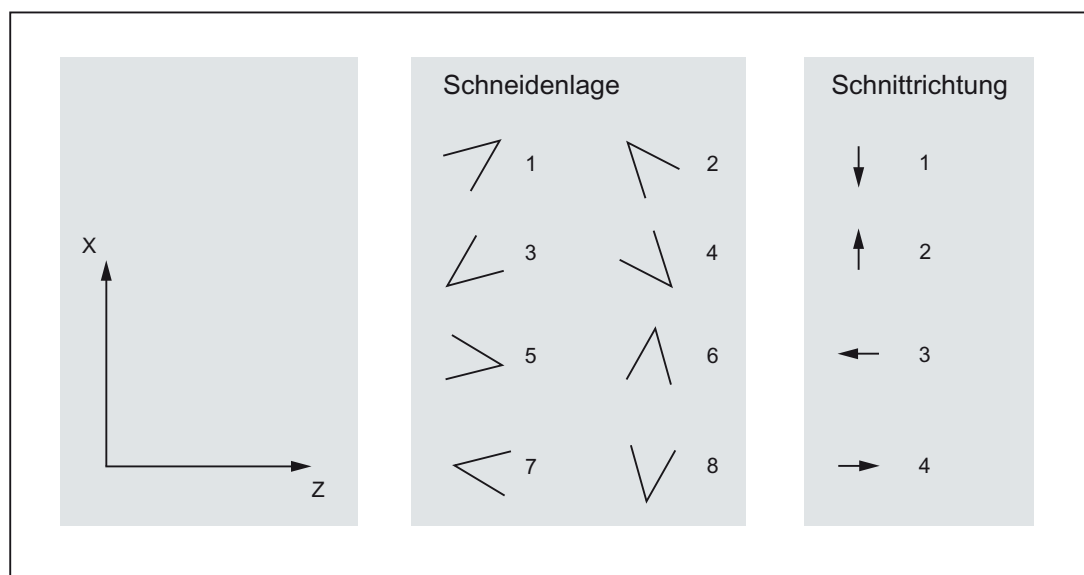


Bild 19-44 Schneidenlage und Schnittrichtung bei Drehwerkzeugen

Die Werte von 1 bis 4 kennzeichnen die Fälle, in denen beide Schneiden im gleichen Quadranten liegen, bei den Werten 5 bis 8 liegen beide Schneiden in benachbarten Quadranten bzw. es liegt eine Koordinatenachse zwischen beiden Schneiden. Die Schneidenlage ist im Werkzeug-Parameter \$TC_DP2 abgelegt.

Für jedes Drehwerkzeug kann eine Schnittrichtung definiert werden. Diese ist im Werkzeug-Parameter \$TC_DP11 abgelegt. Sie hat Werte zwischen 1 und 4 und bezeichnet jeweils eine positive oder negative Richtung der Koordinatenachsen:

Wert:	Bedeutung:
1	Ordinate -
2	Ordinate +

- 3 Abszisse -
4 Abszisse +

Jeder Schneidenlage können zwei verschiedene Schnitttrichtungen zugeordnet werden:

Schneidenlage:	1	2	3	4	5	6	7	8
Schnitttrichtung:	2, 4	2, 3	1, 3	1, 4	1, 2	3, 4	1, 2	3, 4

Halterwinkel und Freiwinkel

Im folgenden Bild sind für ein Drehwerkzeug mit Schneidenlage 3 die beiden Winkel (Halterwinkel und Freiwinkel) dargestellt, die zur Beschreibung der Geometrie der Werkzeugschneide notwendig sind. Die Schnitttrichtung in diesem Beispiel ist 3, d. h. sie bezeichnet die negative Z-Richtung (= Abszissenrichtung bei G18).

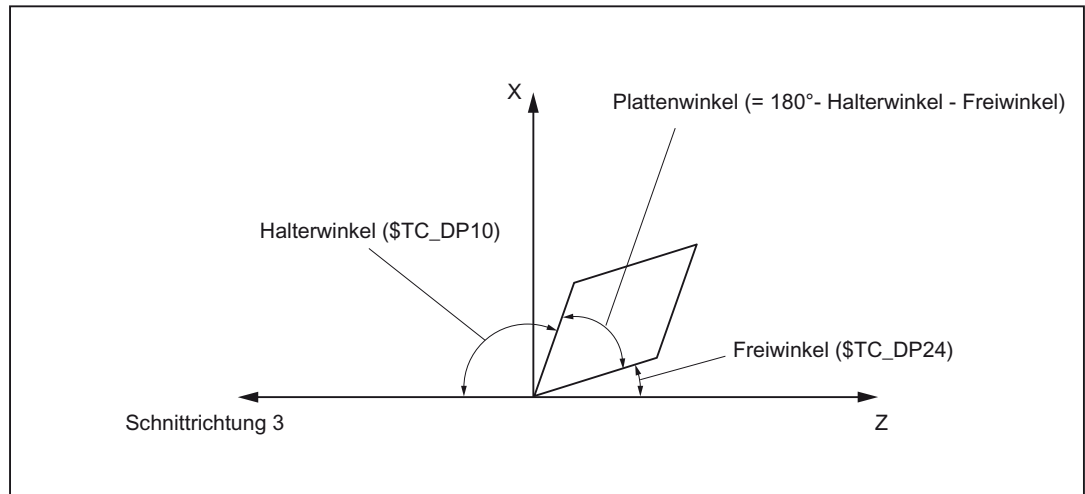


Bild 19-45 Winkel und Schnitttrichtung bei einem Drehwerkzeug mit Schneidenlage 3

Die Schnitttrichtung gibt die Bezugsrichtung für den Halterwinkel an. Der Freiwinkel ist der zwischen der inversen Schnitttrichtung und der benachbarten Schneide (positiv) gemessene Winkel. Halterwinkel und Freiwinkel sind in den Werkzeug-Parametern \$TC_DP10 bzw. \$TC_DP24 abgelegt.

Hinweis

Schnitttrichtung und Werkzeugwinkel sind nur bei den Schneidenlagen 1 bis 8 von Bedeutung.

19.8.4 Modifikationen bei der Drehung von Drehwerkzeugen

Werkzeugorientierung

Drehwerkzeuge sind im Gegensatz zu Fräswerkzeugen nicht rotationssymmetrisch. Das bedeutet, dass zur Beschreibung der Werkzeugorientierung in der Regel 3 Freiheitsgrade bzw. drei rotatorische Achsen notwendig sind. Die Unabhängigkeit von der konkreten Kinematik an der Maschine ist deshalb nur insoweit gegeben, wie die gewünschte Orientierung überhaupt einstellbar ist. Gegebenenfalls kann der dritte Freiheitsgrad durch eine Drehung des Werkstückkoordinatensystems ersetzt werden.

Hinweis

Die Aufteilung der Orientierung in einen Anteil, der durch den orientierbaren Werkzeugträger erzeugt wird, und einen weiteren Anteil, der durch eine Drehung des Koordinatensystems erreicht wird, liegt in der Verantwortung der Applikation. Die Steuerung stellt dafür keine weitere Funktionalität zur Verfügung.

Schneidenform

Wird ein Drehwerkzeug gegenüber der Bearbeitungsebene (d. h. um eine Achse in der Bearbeitungsebene, typischerweise eine C-Achse) um einen Winkel gedreht, der nicht ein Vielfaches von 180° ist, wird die Projektion der (kreisförmigen) Werkzeugschneide in die Bearbeitungsebene zu einer Ellipse. Es wird davon ausgegangen, dass die durch derartige Drehungen entstehenden Abweichungen von der Kreisform so gering sind, dass sie vernachlässigt werden können (Kippwinkel $< 5^\circ$), d. h. die Steuerung geht unabhängig von der Werkzeugorientierung immer von einer kreisförmigen Schneide aus.

Das bedeutet letztlich, dass die Steuerung bezüglich der aktiven Ebene als einzige von der Grundstellung abweichende Einstellung eine Drehung um 180° akzeptiert. Diese Einschränkung gilt nur für die Schneidenform. Die Werkzeuglängen werden bei beliebigen räumlichen Drehungen immer korrekt berücksichtigt.

Schneidenlage, Schnitttrichtung und Winkel

Eine Drehung von 180° um eine Achse in der Bearbeitungsebene bedeutet, dass beim Einsatz des Werkzeugs am gleichen Einsatzort die Spindeldrehrichtung gegenüber dem Einsatz des nicht gedrehten Werkzeugs invertiert werden muss.

Schnitttrichtung und Schneidenlage werden ebenso wie der Schneidenbezugspunkt (siehe unten) nicht modifiziert, wenn das Werkzeug um $\pm 90^\circ$ (mit einer Toleranz von etwa 1°) aus der Ebene gedreht ist, weil dann die Projektion der Schneide in die aktuelle Ebene nicht definiert ist.

Wird ein Werkzeug in der Ebene gedreht (Drehung um eine Achse senkrecht zur Bearbeitungsebene bzw. um die Y-Achse bei G18), wird die Schneidenlage aus den resultierenden Winkeln ermittelt, die sich für Frei- und Halterwinkel ergeben. Sind diese beiden Winkel für das Werkzeug nicht angegeben (d. h. \$TC_DP10 und \$TC_DP24 sind beide Null), wird die neue Schneidenlage alleine aus dem Drehwinkel bestimmt. Dabei gilt die Besonderheit, dass sich die Schneidenlage nur in 90° -Schritten ändert, d. h. die Schneidenlage bleibt abhängig vom Grundzustand immer entweder im Wertebereich 1 bis 4 oder im

Wertebereich 5 bis 8. Die neue Schneidenlage wird auch dann ausschließlich aus dem Drehwinkel ermittelt, wenn die vorgegebenen Werte für Halterwinkel und Freiwinkel unzulässig sind (negative Werte, resultierender Plattenwinkel negativ oder größer als 90°). In allen diesen Fällen werden Freiwinkel und Halterwinkel nicht modifiziert.

Die Schnittrichtung wird abhängig von der Drehung so modifiziert, dass der resultierende Freiwinkel kleiner 90° bleibt. Passen originale Schnittrichtung und originale Schneidenlage nicht zusammen wird die Schnittrichtung bei Drehung des Werkzeugs nicht modifiziert (siehe Kapitel "Schneidenlage, Schnittrichtung und Winkel bei Drehwerkzeugen (Seite 1592)").

Der Drehwinkel in der Ebene, wie er aus dem orientierbaren Werkzeugträger ermittelt wurde, steht in den Systemvariablen $\$P_CUTMOD_ANG$ bzw. $\$AC_CUTMOD_ANG$ zur Verfügung. Dieser Winkel ist der Originalwinkel ohne eventuell anschließende Rundungen auf Vielfache von 45° oder 90° .

Grenzfälle

Haben bei einem Drehwerkzeug Schneidenlage, Schnittrichtung, Frei- und Halterwinkel gültige Werte, so dass sich durch geeignete Drehungen in der Ebene alle Schneidenlagen (1 bis 8) ergeben können, werden in den Fällen, in denen eine der beiden Schneiden (Haupt- bzw. Nebenschneide) um weniger als ein halbes Eingabe-Inkrement ($0,0005^\circ$ bei einer Eingabe-Feinheit von 3 Nachkommastellen) von einer Koordinatenachse entfernt ist, die Schneidenlagen 1 bis 4 gegenüber den Schneidenlagen 5 bis 8 bevorzugt.

In allen anderen Fällen (Fräswerkzeuge bzw. Drehwerkzeuge ohne gültige Schneidenparameter), in denen Drehungen nur in 90° -Schritten möglich sind, gilt: Ist der Betrag des Drehwinkels kleiner als $45^\circ + 0,5$ Eingabe-Inkremente (entspricht $45,0005^\circ$ bei einer Eingabe-Feinheit von 3 Nachkommastellen), werden Schneidenlage und Schnittrichtung nicht verändert, d. h. diese Fälle werden wie Drehungen behandelt, die kleiner 45° sind. Analog dazu werden Drehungen, deren Betrag um weniger als $45^\circ + 0,5$ Eingabe-Inkremente von 180° abweicht, wie Drehungen im Bereich zwischen 135° und 225° behandelt.

Schneidenbezugspunkt

Bei Drehwerkzeugen sind einerseits der Schneidenmittelpunkt und andererseits der Schneidenbezugspunkt definiert. Die Lage dieser beiden Punkte zueinander wird dabei durch die Schneidenlage bestimmt.

Der Abstand der beiden Punkte ist für die Schneidenlagen 1 bis 4 gleich dem $\sqrt{2}$ -fachen des Schneidenradius, für die Schneidenlagen 5 bis 8 gleich dem 1-fachen Schneidenradius. Im ersten Fall liegt der Schneidenbezugspunkt relativ zum Schneidenmittelpunkt in der Bearbeitungsebene auf einer Winkelhalbierenden, im zweiten Fall auf einer Koordinatenachse.

Dreht man das Werkzeug um einen beliebigen Winkel um eine Achse senkrecht zur Bearbeitungsebene, würde sich der Schneidenbezugspunkt mitdrehen, wenn er relativ zum Werkzeug eine feste Position hätte. Die oben genannte Bedingung (Lage auf einer Achse bzw. einer Winkelhalbierenden) wäre dann nicht mehr erfüllt. Dies ist nicht erwünscht. Vielmehr soll der Schneidenbezugspunkt immer so modifiziert werden, dass der Abstandsvektor zwischen Schneidenbezugspunkt und Schneidenmittelpunkt eine der 8 genannten Richtungen hat. Die Schneidenlage muss dazu gegebenenfalls modifiziert werden.

Die Verhältnisse sind beispielhaft im folgenden Bild dargestellt:

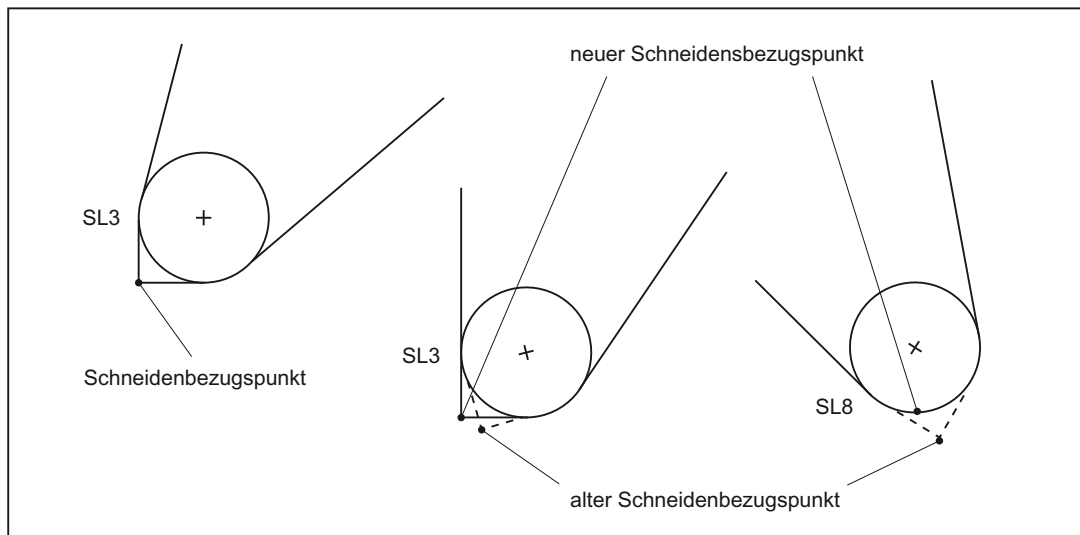


Bild 19-46 Schneidenbezugschwerpunkt und Schneidenlage (SL) bei Werkzeugdrehung

Ein Werkzeug mit der Schneidenlage 3, dem Freiwinkel $22,5^\circ$ und dem Halterwinkel $112,5^\circ$ wird gedreht. Bei Drehungen bis $22,5^\circ$ bleibt die Schneidenlage erhalten, die Lage des Schneidenbezugschwerpunkts relativ zum Werkzeug wird jedoch so korrigiert, dass die relative Lage beider Punkte in der Bearbeitungsebene erhalten bleibt. Bei größeren Drehungen (bis $67,5^\circ$) ändert sich die Schneidenlage auf den Wert 8.

Hinweis

Da der Schneidenbezugschwerpunkt durch den Werkzeuglängenvektor definiert ist, ändert sich durch die Modifikation des Schneidenbezugschwerpunkts die effektive Werkzeuglänge.

19.8.5 Schneidenlage bei Fräs- und Bohrwerkzeugen

Fräs- und Bohrwerkzeuge

Unter Fräs- und Bohrwerkzeugen werden im Folgenden Werkzeuge verstanden, deren Werkzeugtyp (\$TC_DP1) Werte im Bereich 100 bis 299 hat.

Werkzeuge werden unabhängig vom Werkzeugtyp wie Fräs- und Bohrwerkzeuge behandelt, wenn:

SD42950 \$SC_TOOL_LENGTH_TYPE = 1

Schneidenlage

Für die so definierten Fräs- und Bohrwerkzeuge wird ebenfalls eine Schneidenlage eingeführt, die bei Drehungen entsprechend der nachfolgenden Beschreibung modifiziert wird.

Eine eventuell angegebene Schneidenlage bei Werkzeugen, die keine Fräs- und Bohrwerkzeuge bzw. keine Drehwerkzeuge entsprechend den genannten Definitionen sind, wird nicht ausgewertet.

Die Schneidenlage der Bohr- und Fräswerkzeuge ist wie bei Drehwerkzeugen im Werkzeugparameter \$TC_DP2 hinterlegt. In Anlehnung an die Definition der Schneidenlage bei Drehwerkzeugen kann dieser Parameter die Werte 5 bis 8 annehmen. Die Schneidenlage gibt dabei die Orientierung (die Richtung der Rotationsachse) des Werkzeugs an:

Schneidenlage	Richtung der Rotationsachse des Werkzeugs
5	Abszisse +
6	Ordinate +
7	Abszisse -
8	Ordinate -

Beispiel:

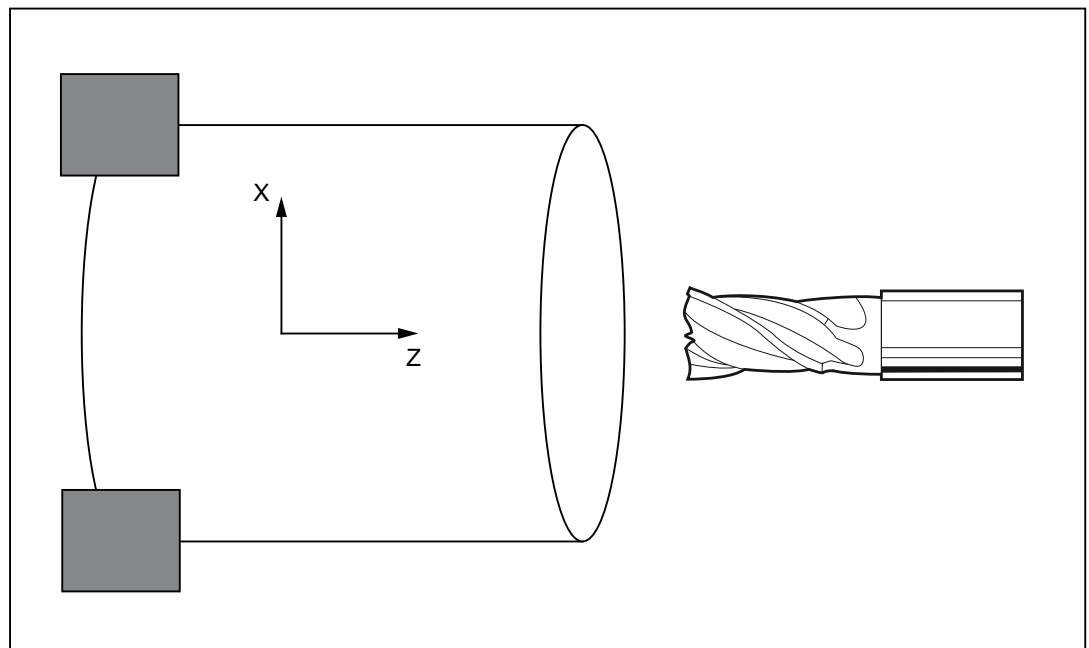


Bild 19-47 Fräswerkzeug mit Schneidenlage 7

19.8.6 Modifikationen bei der Drehung von Fräs- und Bohrwerkzeugen

Bei einer Drehung eines Fräs- oder Bohrwerkzeugs wird die Schneidenlage entsprechend umgerechnet. Schnittrichtung und Werkzeugwinkel (Freiwinkel bzw. Halterwinkel) sind für Fräs- und Bohrwerkzeuge nicht definiert, so dass die Veränderung der Schneidenlage ausschließlich aus der Drehung abgeleitet wird. Daraus folgt weiter, dass sich bei Fräs- und Bohrwerkzeugen die Schneidenlage immer dann ändert, wenn der Betrag der Drehung gegenüber der Nullstellung größer als 45° wird.

19.8.7 Parametrierung

Verhalten im Fehlerfall

Bei Wirksamwerden der Funktion "Schneidendaten-Modifikation bei drehbaren Werkzeugen" (durch expliziten Aufruf mit `CUTMOD` oder durch eine Werkzeuganwahl) können verschiedene Fehlerzustände auftreten.

Für jeden dieser möglichen Fehlerzustände kann eingestellt werden, ob der Fehler zu einer Alarmausgabe führen soll, und falls ja, ob ein solcher Alarm nur angezeigt werden soll (Warnhinweis), oder ob zusätzlich die Interpretation des Teileprogramms abgebrochen werden soll. Die Einstellungen erfolgen über das Maschinendatum:

MD20125 \$MC_CUTMOD_ERR

Jedem Fehlerzustand sind zwei Bit des Maschinendatums zugeordnet:

Fehlerzustand	Bit	Bedeutung
Für das aktive Werkzeug ist keine gültige Schnitttrichtung definiert.	0	Alarmausgabe bei Fehler "Ungültige Schnitttrichtung"
	1	Programmstopp bei Fehler "Ungültige Schnitttrichtung"
Die Schneidenwinkel (Freiwinkel und Halterwinkel) des aktiven Werkzeugs sind beide Null.	2	Alarmausgabe bei Fehler "Nicht definierte Schneidenwinkel"
	3	Programmstopp bei Fehler "Nicht definierte Schneidenwinkel"
Der Freiwinkel des aktiven Werkzeugs hat einen unzulässigen Wert ($< 0^\circ$ oder $> 180^\circ$).	4	Alarmausgabe bei Fehler "Ungültiger Freiwinkel"
	5	Programmstopp bei Fehler "Ungültiger Freiwinkel"
Der Halterwinkel des aktiven Werkzeugs hat einen unzulässigen Wert ($< 0^\circ$ oder $> 90^\circ$).	6	Alarmausgabe bei Fehler "Ungültiger Halterwinkel"
	7	Programmstopp bei Fehler "Ungültiger Halterwinkel"
Der Plattenwinkel des aktiven Werkzeugs hat einen unzulässigen Wert ($< 0^\circ$ oder $> 90^\circ$).	8	Alarmausgabe bei Fehler "Ungültiger Plattenwinkel"
	9	Programmstopp bei Fehler "Ungültiger Plattenwinkel"
Die Kombination Schneidenlage - Halterwinkel des aktiven Werkzeugs ist unzulässig (bei den Schneidenlagen 1 bis 4 muss der Halterwinkel $\leq 90^\circ$ sein, bei den Schneidenlagen 5 bis 8 muss er $\geq 90^\circ$ sein).	10	Alarmausgabe bei Fehler "Ungültige Kombination Schneidenlage - Halterwinkel"
	11	Programmstopp bei Fehler "Ungültige Kombination Schneidenlage - Halterwinkel"
Unzulässige Drehung des aktiven Werkzeugs (das Werkzeug wurde um $\pm 90^\circ$ (mit einer Toleranz von etwa 1°) aus der aktiven Bearbeitungsebene herausgedreht). Dadurch ist die Schneidenlage in der Bearbeitungsebene nicht mehr definiert.	12	Alarmausgabe bei Fehler "Ungültige Drehung"
	13	Programmstopp bei Fehler "Ungültige Drehung"

Verhalten bei POWER ON

Die Funktion "Schneidendaten-Modifikation bei drehbaren Werkzeugen" (CUTMOD) wird bei POWER ON automatisch mit dem Wert initialisiert, der hinterlegt ist im Maschinendatum:

MD20127 \$MC_CUTMOD_INIT

Wenn der Wert dieses Maschinendatums "-2" beträgt, dann wird CUTMOD auf den Wert gesetzt, der eingestellt ist im Maschinendatum:

MD20126 \$MC_TOOL_CARRIER_RESET_VALUE (Wirksamer Werkzeugträger bei RESET)

19.8.8 Programmierung

Die Funktion "Schneidendaten-Modifikation bei drehbaren Werkzeugen" wird mit dem Befehl CUTMOD aktiviert.

Syntax

CUTMOD=<Wert>

Bedeutung

CUTMOD	Befehl zum Einschalten der Funktion "Schneidendaten-Modifikation bei drehbaren Werkzeugen"
<Wert>	Dem CUTMOD-Befehl können folgende Werte zugewiesen werden:
0	Die Funktion ist deaktiviert. Die von den Systemvariablen \$P_AD... gelieferten Werte sind gleich den korrespondierenden Werkzeugparametern.
> 0	Die Funktion wird aktiviert, falls ein orientierbarer Werkzeugträger mit der angegebenen Nummer aktiv ist, d. h. die Aktivierung ist an einen bestimmten orientierbaren Werkzeugträger gebunden. Die von den Systemvariablen \$P_AD... gelieferten Werte sind gegenüber den korrespondierenden Werkzeugparametern abhängig von der aktiven Drehung gegebenenfalls modifiziert. Die Deaktivierung des bezeichneten orientierbaren Werkzeugträgers deaktiviert die Funktion temporär, die Aktivierung eines anderen orientierbaren Werkzeugträgers deaktiviert sie permanent. Im ersten Fall wird die Funktion deshalb bei erneuter Anwahl des gleichen orientierbaren Werkzeugträgers wieder aktiviert, im zweiten Fall ist eine erneute Anwahl notwendig, auch dann wenn zu einem späteren Zeitpunkt der orientierbare Werkzeugträger mit der angegebenen Nummer erneut aktiviert wird. Die Funktion wird durch Reset nicht beeinflusst.
-1	Die Funktion wird immer aktiviert, falls ein orientierbarer Werkzeugträger aktiv ist. Beim Wechsel des Werkzeugträgers oder bei dessen Abwahl und einer späteren erneuten Anwahl muss CUTMOD nicht erneut gesetzt werden.
-2	Die Funktion wird immer aktiviert, falls ein orientierbarer Werkzeugträger aktiv ist, dessen Nummer gleich der des aktuell aktiven orientierbaren Werkzeugträgers ist. Ist kein orientierbarer Werkzeugträger aktiv, ist das gleichbedeutend mit CUTMOD=0. Ist ein orientierbarer Werkzeugträger aktiv, ist das gleichbedeutend mit der unmittelbaren Angabe der aktuellen Werkzeugträgernummer.
< -2	Werte kleiner 2 werden ignoriert, d. h. dieser Fall wird so behandelt, als wäre CUTMOD nicht programmiert. Hinweis: Dieser Wertebereich sollte nicht verwendet werden, weil er für eventuelle spätere Erweiterungen reserviert ist.

Hinweis

SD42984 \$SC_CUTDIRMOD

Die über den Befehl CUTMOD aktivierbare Funktion ersetzt die über das Settingdatum SD42984 \$SC_CUTDIRMOD aktivierbare Funktion. Diese Funktion steht jedoch weiterhin unverändert zur Verfügung. Da es aber nicht sinnvoll ist, beide Funktionen parallel zu nutzen, kann sie nur aktiviert werden, wenn CUTMOD gleich Null ist.

Wirksamkeit der modifizierten Schneidendaten

Die modifizierte Schneidenlage und der modifizierte Schneidenbezugspunkt werden bei Programmierung auch für ein bereits aktives Werkzeug sofort wirksam. Eine Werkzeugneuanwahl ist dazu nicht notwendig.

Einfluss der aktiven Arbeitsebene

Für die Bestimmung von modifizierter Schneidenlage, Schnitttrichtung und Halter- bzw. Freiwinkel ist die Betrachtung der Schneide in der jeweils aktiven Ebene (G17 - G19) maßgebend.

Enthält jedoch das Settingdatum SD42940 \$SC_TOOL_LENGTH_CONST (Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel) einen gültigen Wert ungleich Null (plus oder minus 17, 18 oder 19), so bestimmt dessen Inhalt die Ebene, in der die relevanten Größen betrachtet werden.

Systemvariablen

Folgende Systemvariablen stehen zur Verfügung:

Systemvariablen	Bedeutung
\$P_CUTMOD_ANG / \$AC_CUTMOD_ANG	Liefert den (nicht gerundeten) Winkel in der aktiven Bearbeitungsebene, der für die Modifikation der Schneidendaten (Schneidenlage, Schnitttrichtung, Freiwinkel und Halterwinkel) bei den mit CUTMOD bzw. \$SC_CUTDIRMOD aktivierten Funktionen zugrunde gelegt wurde. \$P_CUTMOD_ANG bezieht sich auf den aktuellen Zustand im Vorlauf, \$AC_CUTMOD_ANG auf den aktuellen Hauptlaufsatz.
\$P_CUTMOD / \$AC_CUTMOD	Liest den aktuell gültigen Wert, der zuletzt mit dem Befehl CUTMOD programmiert wurde (Nummer des Werkzeugträgers, für den die Schneidendaten-Modifikation aktiviert werden soll). War der letzte programmierte CUTMOD-Wert = -2 (Aktivierung mit dem aktuell aktiven orientierbaren Werkzeugträger), dann wird in \$P_CUTMOD nicht der Wert -2, sondern die Nummer des zum Zeitpunkt der Programmierung aktiven orientierbaren Werkzeugträgers zurückgeliefert. \$P_CUTMOD bezieht sich auf den aktuellen Zustand im Vorlauf, \$AC_CUTMOD auf den aktuellen Hauptlaufsatz.
\$P_CUT_INV / \$AC_CUT_INV	Liefert den Wert TRUE, wenn das Werkzeug so gedreht ist, dass die Spindeldrehrichtung invertiert werden muss. Dazu müssen in dem Satz, auf den sich die jeweilige Leseoperation bezieht, die folgenden vier Bedingungen erfüllt sein: <ol style="list-style-type: none"> 1. Es ist ein Dreh- oder Schleifwerkzeug aktiv (Werkzeugtypen 400 bis 599 und / oder SD42950 \$SC_TOOL_LENGTH_TYPE = 2). 2. Die Schneidenbeeinflussung wurde mit dem Sprachbefehl CUTMOD aktiviert. 3. Es ist ein orientierbarer Werkzeugträger aktiv, der durch den numerischen Wert von CUTMOD bezeichnet wurde. 4. Der orientierbare Werkzeugträger dreht das Werkzeug um eine Achse in der Bearbeitungsebene (typischerweise die C-Achse) so, dass die resultierende Normale der Werkzeugschneide gegenüber der Ausgangslage um mehr als 90° (typischerweise 180°) gedreht ist. <p>Ist mindestens eine der genannten vier Bedingungen nicht erfüllt, ist der Inhalt der Variablen FALSE. Für Werkzeuge, deren Schneidenlage nicht definiert ist, ist der Wert der Variablen immer FALSE.</p> <p>\$P_CUT_INV bezieht sich auf den aktuellen Zustand im Vorlauf und \$AC_CUT_INV auf den aktuellen Hauptlaufsatz.</p>

Alle Hauptlaufvariablen (\$AC_CUTMOD_ANG, \$AC_CUTMOD und \$AC_CUT_INV) können in Synchronaktionen gelesen werden. Ein Lesezugriff aus dem Vorlauf generiert einen Vorlaufstopp.

Modifizierte Schneidendaten

Falls eine Werkzeugdrehung aktiv ist, werden die modifizierten Daten in den folgenden Systemvariablen zur Verfügung gestellt:

Systemvariable	Bedeutung
\$P_AD[2]	Schneidenlage
\$P_AD[10]	Halterwinkel
\$P_AD[11]	Schnitttrichtung
\$P_AD[24]	Freiwinkel

Hinweis

Die Daten sind gegenüber den korrespondierenden Werkzeugparametern (\$TC_DP2[...], ...] usw.) immer dann modifiziert, wenn die Funktion "Schneidendaten-Modifikation bei drehbaren Werkzeugen" mit dem Befehl CUTMOD aktiviert wurde und ein orientierbarer Werkzeugträger aktiv ist, der eine Werkzeugdrehung bewirkt.

19.8.9 Beispiel

Das folgende Beispiel bezieht sich auf ein Werkzeug mit der Schneidenlage 3 und einem orientierbaren Werkzeugträger, der das Werkzeug um die B-Achse drehen kann.

Die Zahlenwerte in den Kommentaren geben jeweils die Satzendpositionen in Maschinenkoordinaten (MKS) in der Reihenfolge X, Y, Z an.

Programmcode	Kommentar			
N10 \$TC_DP1[1,1]=500				
N20 \$TC_DP2[1,1]=3	; Schneidenlage			
N30 \$TC_DP3[1,1]=12				
N40 \$TC_DP4[1,1]=1				
N50 \$TC_DP6[1,1]=6				
N60 \$TC_DP10[1,1]=110	; Halterwinkel			
N70 \$TC_DP11[1,1]=3	; Schnitttrichtung			
N80 \$TC_DP24[1,1]=25	; Freiwinkel			
N90 \$TC_CARR7[2]=0 \$TC_CARR8[2]=1 \$TC_CARR9[2]=0	; B-Achse			
N100 \$TC_CARR10[2]=0 \$TC_CARR11[2]=0 \$TC_CARR12[2]=1	; C-Achse			
N110 \$TC_CARR13[2]=0				
N120 \$TC_CARR14[2]=0				
N130 \$TC_CARR21[2]=X				
N140 \$TC_CARR22[2]=X				
N150 \$TC_CARR23[2]="M"				
N160 TCOABS CUTMOD=0				
N170 G18 T1 D1 TCARR=2	X	Y	Z	
N180 X0 Y0 Z0 F10000	; 12.000	0.000	1.000	

Programmcode	Kommentar		
N190 \$TC_CARR13[2]=30			
N200 TCARR=2			
N210 X0 Y0 Z0	; 10.892	0.000	-5.134
N220 G42 Z-10	; 8.696	0.000	-17.330
N230 Z-20	; 8.696	0.000	-21.330
N240 X10	; 12.696	0.000	-21.330
N250 G40 X20 Z0	; 30.892	0.000	-5.134
N260 CUTMOD=2 X0 Y0 Z0	; 8.696	0.000	-7.330
N270 G42 Z-10	; 8.696	0.000	-17.330
N280 Z-20	; 8.696	0.000	-21.330
N290 X10	; 12.696	0.000	-21.330
N300 G40 X20 Z0	; 28.696	0.000	-7.330
N310 M30			

Erläuterungen:

In Satz N180 wird zunächst das Werkzeug bei CUTMOD=0 und nicht gedrehtem orientierbaren Werkzeugträger angewählt. Da alle Offsetvektoren des orientierbaren Werkzeugträgers 0 sind, wird die Position angefahren, die den in \$TC_DP3[1,1] und \$TC_DP4[1,1] angegebenen Werkzeuglängen entspricht.

In Satz N200 wird der orientierbare Werkzeugträger mit einer Drehung von 30° um die B-Achse aktiviert. Da die Schneidenlage wegen CUTMOD=0 nicht modifiziert wird, ist nach wie vor der alte Schneidenbezugspunkt maßgebend. Deshalb wird in Satz N210 die Position angefahren, die den alten Schneidenbezugspunkt im Nullpunkt beibehält (d. h. der Vektor (1, 12) wird in der Z/X-Ebene um 30° gedreht).

In Satz N260 ist im Unterschied zu Satz N200 CUTMOD=2 wirksam. Aufgrund der Drehung des orientierbaren Werkzeugträgers wird die modifizierte Schneidenlage 8. Daraus folgen auch abweichende Achspositionen.

In den Sätzen N220 bzw. N270 wird jeweils die Werkzeugradiuskorrektur (WRK) aktiviert. Die unterschiedliche Schneidenlage in beiden Programmstücken hat auf die Endpositionen der Sätze, in denen die WRK aktiv ist, keinen Einfluss, die entsprechenden Positionen sind deshalb identisch. Erst in den Abwabsätzen N260 bzw. N300 wirken sich die unterschiedlichen Schneidenlagen wieder aus.

19.9 Inkrementell programmierte Korrekturwerte

19.9.1 G91 Erweiterung

Voraussetzungen

Kettenmaßprogrammierung mit G91 ist so definiert, dass bei Anwahl einer Werkzeugkorrektur der Korrekturwert additiv zum inkrementell programmierten Wert gefahren wird.

Anwendungen

Für Anwendungen wie Ankratzen ist es erforderlich, im Kettenmaß nur den programmierten Weg zu fahren. Die aktivierte Werkzeugkorrektur wird nicht gefahren.

Ablauf

Anwahl einer Werkzeugkorrektur mit Kettenmaßprogrammierung

- Mit Werkzeugspitze an Werkstück ankratzen.
- Die aktuelle Istposition reduziert um Werkzeugkorrektur in Basisframe übernehmen (Istwertsetzen).
- Ab der Nullposition inkrementell fahren.

Aktivierung

Ob bei `FRAME` und inkrementeller Programmierung einer Achse eine geänderte Werkzeuglänge herausgefahren wird oder nur der programmierte Weg gefahren wird ist einstellbar mit dem Settingdatum:

SD42442 \$SC_TOOL_OFFSET_INCR_PROG (Werkzeuglängenkorrekturen)

Nullpunktverschiebung / Frames G91

Ob beim Wert = 1 standardmäßig Nullpunktverschiebung bei `FRAME` und inkrementeller Programmierung einer Achse herausgefahren wird oder beim Wert = 0 nur der programmierte Wert gefahren wird ist einstellbar mit dem Settingdatum:

SD42440 \$SC_FRAME_OFFSET_INCR_PROG (Nullpunktverschiebung in Frames)

Für weitere Informationen siehe Kapitel "K2: Achsen, Koordinatensysteme, Frames (Seite 713)".

Randbedingung

Ist das Verhalten so eingestellt, dass die Verschiebung über Programmende und `RESET` hinweg aktiv bleibt

MD20110 \$MC_RESET_MODE_MASK, Bit6=1 (Festlegung der Steuerungs-Grundstellung nach Reset/TP-Ende)

und wird im 1. Teileprogrammsatz ein inkrementeller Weg programmiert, so wird die Korrektur additiv zum programmierten Weg immer herausgefahren.

Hinweis

Bei dieser Konfiguration sollten Teileprogramme immer mit Absolut-Programmierung beginnen.

19.9.2 Verfahren in Richtung der Werkzeugorientierung (MOVT)

Typische Anwendung)

An Maschinen mit orientierbarem Werkzeugträger kann in Werkzeugrichtung verfahren werden (typischerweise beim Bohren), ohne dass ein Frame aktiviert wird (z.B. mittels `TOFRAME` oder `TOROT`), bei dem eine der Achsen in Richtung des Werkzeugs zeigt.

Ebenso an Maschinen, bei denen bei einer Schrägbearbeitung ein Frame aktiv ist, der die schräge Ebene definiert, das Werkzeug aber nicht exakt senkrecht eingestellt werden kann, weil wegen eines indexierten Werkzeugträgers (Hirth-Verzahnung) die Werkzeugorientierung nicht beliebig einstellbar ist.

In diesen Fällen muss dann - abweichend von der eigentlich verlangten Bewegung senkrecht zur Ebene - in Werkzeugrichtung gebohrt werden, da andernfalls der Bohrer nicht in Richtung seiner Längsachse geführt würde, was im Allgemeinen den Bruch des Bohrers zur Folge haben würde.

MOVT

Der Endpunkt einer solchen Bewegung wird mit `MOVT=...` programmiert. Der programmierte Wert wirkt standardmäßig inkrementell in Werkzeugrichtung. Die positive Richtung ist dabei von der Werkzeugspitze zur Werkzeugaufnahme definiert. Der Inhalt von `MOVT` ist bei der Zustellbewegung (beim Bohren) deshalb in der Regel negativ, bei der Rückzugsbewegung dagegen positiv. Das entspricht den Verhältnissen bei normaler achsparalleler Bearbeitung z.B. mit `G91Z...`

Wird die Bewegung in der Form `MOVT=AC(...)` programmiert, wirkt `MOVT` absolut. Dazu wird eine Ebene definiert, die durch den aktuellen Nullpunkt verläuft, und deren Flächennormalenvektor parallel zur Werkzeugorientierung ist. `MOVT` gibt dann die Lage bezüglich dieser Ebene an:

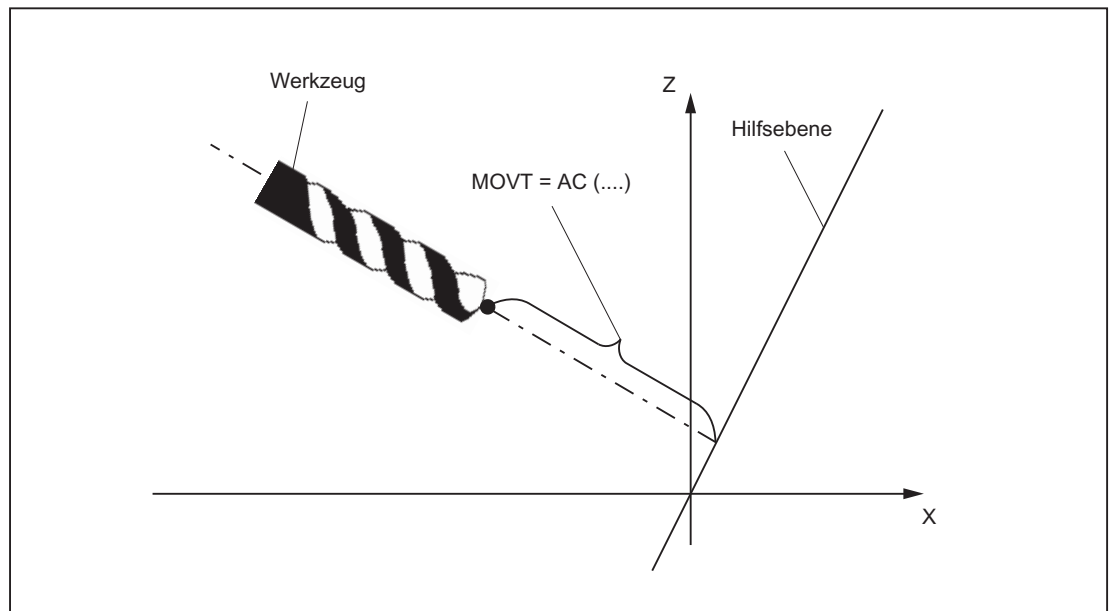


Bild 19-48 Definition der Position bei Absolutprogrammierung einer Bewegung in Werkzeugrichtung

Der Bezug zu dieser Hilfsebene dient nur zur Berechnung der Endposition. Aktive Frames werden durch diese interne Berechnung nicht beeinflusst.

Anstelle von `MOV T = . . .` kann auch `MOV T = IC (. . .)` geschrieben werden, wenn deutlich sichtbar zum Ausdruck gebracht werden soll, dass `MOV T` inkrementell wirkt. Ein funktioneller Unterschied zwischen beiden Schreibweisen besteht nicht.

Randbedingungen

Für die Programmierung mit `MOV T` gibt es folgende Randbedingungen:

- Sie ist unabhängig vom Vorhandensein eines orientierbaren Werkzeugträgers. Die Richtung der Bewegung ist von der aktiven Ebene abhängig. Sie verläuft in Richtung der Applikate, d.h. bei `G17` in Z-Richtung, bei `G18` in Y-Richtung und bei `G19` in X-Richtung. Dies gilt sowohl für den Fall, dass keine orientierbarer Werkzeugträger aktiv ist als auch für den Fall eines orientierbaren Werkzeugträgers ohne drehbares Werkzeug oder mit drehbarem Werkzeug in Grundstellung.
- `MOV T` wirkt bei aktiver Orientierungstransformation (3-4-5-Achs-Transformation) in gleicher Weise.
- Wird in einem Satz mit `MOV T` gleichzeitig die Werkzeugorientierung verändert (z.B. bei aktiver 5-Achstransformation durch gleichzeitige Interpolation der Rundachsen), ist die Orientierung am Satzanfang für die Bewegungsrichtung von `MOV T` maßgebend. Die Bahn der Werkzeugspitze (TCP-Tool Center Point) wird durch die Orientierungsänderung nicht beeinflusst.
- Linear- oder Spline-Interpolation (`G0`, `G1`, `ASPLINE`, `BSPLINE`, `CSPLINE`) muss aktiv sein. Andernfalls wird ein Alarm ausgegeben. Ist eine Spline-Interpolationsart aktiv, ist die resultierende Bahn im Allgemeinen keine Gerade, da der von `MOV T` ermittelte Endpunkt so behandelt wird, als sei er explizit mit X, Y, Z programmiert worden.
- In einem Satz mit `MOV T` dürfen keine Geometrieachsen programmiert sein (Alarm 14157).

19.10 Werkzeuggrundorientierung

19.10.1 Funktion

Standardmäßig hängt die dem Werkzeug selbst zugeordnete Orientierung ausschließlich von der aktiven Bearbeitungsebene ab. So ist z. B. die Werkzeuggrundorientierung bei G17 parallel zu Z, bei G18 parallel zu Y und bei G19 parallel zu X.

Mit Hilfe der Funktion "Werkzeugschneiden Orientierung zuordnen" kann jeder Werkzeugschneide eine vom Standard abweichende Grundorientierung zugeordnet werden. Dafür stellt die Funktion folgende Systemvariablen zur Verfügung:

Systemvariable	Bedeutung	Datentyp	Vorbelegung
\$TC_DPV[<t>, <d>]	Werkzeugschneidenorientierung	INT	0
\$TC_DPV3[<t>, <d>]	L1-Komponente der Werkzeugschneidenorientierung	REAL	0
\$TC_DPV4[<t>, <d>]	L2-Komponente der Werkzeugschneidenorientierung	REAL	0
\$TC_DPV5[<t>, <d>]	L3-Komponente der Werkzeugschneidenorientierung	REAL	0

<t>: T-Nummer der Schneide

<d>: D-Nummer der Schneide

Hinweis

Die Bezeichnungen \$TC_DPV3 bis \$TC_DPV5 sind in Analogie zu den Bezeichnern \$TC_DP3 bis \$TC_DP5 der Werkzeuglängenkomponenten gewählt.

Die Systemvariablen stehen nur dann zur Verfügung, wenn das Maschinendatum zur Aktivierung der Funktion (\$MN_MM_ENABLE_TOOL_ORIENTATION, siehe "Parametrierung (Seite 1610)") ungleich Null ist. Hat dieses Maschinendatum den Wert 1, steht nur die Systemvariable \$TC_DPV1[<t>, <d>] zur Verfügung, hat es den Wert 2, sind alle vier Systemvariablen vorhanden.

Ist der Inhalt aller vier Systemvariablen gleich Null, wird die Werkzeuggrundorientierung durch die aktive Bearbeitungsebene definiert.

Ist die Systemvariable \$TC_DPV[<t>, <d>] gleich Null, definieren die übrigen drei Systemvariablen, falls vorhanden, einen Richtungsvektor. Der Betrag des Vektors ist dabei unerheblich.

Beispiel:

\$TC_DPV[1, 1] = 0

\$TC_DPV3[1, 1] = 1.0

\$TC_DPV4[1, 1] = 0.0

\$TC_DPV5[1, 1] = 1.0

Die Werkzeuggrundorientierung zeigt in diesem Beispiel in Richtung der Winkelhalbierenden in der L1-L3-Ebene, d. h. bei einem Fräswerkzeug und aktiver Ebene G17 in Richtung der Winkelhalbierenden der Z-X-Ebene.

\$TC_DPV[<t>, <d>]

Die Systemvariable \$TC_DPV[<t>, <d>] dient dazu, bestimmte, häufig benötigte Werkzeuggrundorientierungen (parallel zu Koordinatenachsen) auf vereinfachte Art und Weise angeben zu können. Die zulässigen Werte sind in der folgenden Tabelle dargestellt. Dabei sind die Angaben in der ersten Spalte und die in der zweiten bzw. dritten Spalte jeweils äquivalent.

\$TC_DPV[<t>, <d>]	(\$TC_DPV2[<t>, <d>], \$TC_DPV3[<t>, <d>], \$TC_DPV4[<t>, <d>]) bei \$TC_DPV[<t>, <d>] = 0	
	Fräswerkzeuge	Drehwerkzeuge ¹⁾
≤ 0 oder > 6	(\$TC_DPV5[<t>, <d>], \$TC_DPV4[<t>, <d>], \$TC_DPV3[<t>, <d>]) ²⁾	(\$TC_DPV3[<t>, <d>], \$TC_DPV5[<t>, <d>], \$TC_DPV4[<t>, <d>]) ²⁾
1	(0, 0, V)	(0, V, 0)
2	(0, V, 0)	(0, 0, V)
3	(V, 0, 0)	(V, 0, 0)
4	(0, 0, -V)	(0, -V, 0)
5	(0, -V, 0)	(0, 0, -V)
6	(-V, 0, 0)	(-V, 0, 0)

¹⁾ Als Drehwerkzeuge werden hier alle Werkzeuge verstanden, deren Werkzeugtyp (\$TC_DP1[<t>, <d>]) zwischen 400 und 599 liegt. Alle anderen Werkzeugtypen bezeichnen Fräswerkzeuge.

²⁾ Sind alle drei Werte \$TC_DPV3[<t>, <d>], \$TC_DPV4[<t>, <d>], \$TC_DPV5[<t>, <d>] gleich Null, wird die Werkzeuggrundorientierung durch die aktive Bearbeitungsebene bestimmt (Standardeinstellung).

V: Steht für einen positiven Wert in der entsprechenden Systemvariablen.

Beispiele:

- Fräswerkzeuge

\$TC_DPV[<t>, <d>] = 3 ist gleichbedeutend mit (\$TC_DPV3[<t>, <d>] = 0,
\$TC_DPV4[<t>, <d>] = 0,
\$TC_DPV5[<t>, <d>] = V)

- Drehwerkzeuge

\$TC_DPV[<t>, <d>] = 2 ist gleichbedeutend mit (\$TC_DPV3[<t>, <d>] = 0,
\$TC_DPV4[<t>, <d>] = V,
\$TC_DPV5[<t>, <d>] = 0)

19.10.2 Parametrierung

Maschinendaten

Aktivierung

Die Systemvariablen zur Beschreibung der Werkzeuggrundorientierung stehen nur dann zur Verfügung, wenn das folgende Maschinendatum ungleich Null ist:

MD18114 \$MN_MM_ENABLE_TOOL_ORIENT (Werkzeugschneiden Orientierung zuordnen)

Wert	Bedeutung
0	Die Funktion ist nicht aktiv.
1	Jeder Werkzeugschneide <d> des Werkzeugs <t> wird der Systemparameter \$TC_DPV[<t>, <d>] zugeordnet, mit dessen Hilfe eine von 6 möglichen Werkzeuggrundorientierungen in positive bzw. negative Koordinatenrichtung definiert werden kann.
2	Jeder Werkzeugschneide <d> des Werkzeugs <t> werden zusätzlich zum Systemparameter \$TC_DPV[<t>, <m>] die weiteren drei Systemparameter \$TC_DPV3[<t>, <m>], \$TC_DPV4[<t>, <m>] und \$TC_DPV5[<t>, <m>] zugeordnet, mit deren Hilfe eine beliebige räumliche Werkzeuggrundorientierung definiert werden kann.

Settingdaten

Die nachfolgend genannten Settingdaten sind nur dann wirksam, wenn die Werkzeuggrundorientierung durch einen Eintrag in mindestens einer der Systemvariablen \$TC_DPVx[<t>, <d>] definiert ist. Diese Settingdaten haben keine Wirkung, wenn die Werkzeuggrundorientierung nur durch die Ebenenanwahl G17 - G19 bestimmt wird.

Ebenenzuordnung der Grundorientierung unabhängig vom Werkzeugtyp

Die Grundorientierung von Dreh- und Schleifwerkzeugen wird bei G18 eingenommen, die der Fräswerkzeuge bei G17. In diesen Fällen ist die wirksame Werkzeugorientierung also unverändert so, wie sie in \$TC_DPVx[<t>, <d>] angegeben wurde. Die Grundorientierung ist dabei immer die Richtung, die senkrecht auf der Ebene steht, in der gegebenenfalls eine Werkzeugradiuskorrektur ausgeführt wird. Bei Drehwerkzeugen fällt die Werkzeugorientierung in der Regel nicht mit der Werkzeuglängsachse zusammen.

Mit dem folgenden Settingdatum kann festgelegt werden, ob eine Werkzeugschneide bei der Ebenenzuordnung der Grundorientierung unabhängig vom Eintrag in \$TC_DP1 wie ein Fräs- bzw. wie ein Dreh- oder Schleifwerkzeug behandelt werden soll:

SD42950 \$SC_TOOL_LENGTH_TYPE

Wert	Bedeutung
0	Die Zuordnung erfolgt standardmäßig. Es wird zwischen Dreh- und Schleifwerkzeugen (Werkzeugtypen 400 bis 599) und anderen Werkzeugen (Fräswerkzeugen) unterschieden.
1	Die Zuordnung erfolgt unabhängig vom tatsächlichen Werkzeugtyp immer wie bei Fräswerkzeugen.
2	Die Zuordnung erfolgt unabhängig vom tatsächlichen Werkzeugtyp immer wie bei Dreh- und Schleifwerkzeugen.
3	Die Zuordnung erfolgt getrennt für Dreh- und Schleifwerkzeuge (Werkzeugtypen 400 bis 599) einerseits und alle anderen Werkzeuge (Fräswerkzeuge) andererseits.

Die Ebenenzuordnung der Grundorientierung wird dabei vom folgenden Settingdatum bestimmt:

- Fräswerkzeuge:
SD42940 \$SC_TOOL_LENGTH_CONST
- Dreh- und Schleifwerkzeuge:
SD42942 \$SC_TOOL_LENGTH_CONST_T

Siehe "Verhalten bei Ebenenwechsel"!

Verhalten bei Ebenenwechsel

Ein Ebenenwechsel führt zu einer Orientierungsänderung.

Dabei werden folgende Drehungen ausgeführt:

Ebenenwechsel	Drehung
G17 → G18	Drehung von -90 Grad um die Z-Achse und anschließende Drehung von -90 Grad um die X-Achse.
G18 → G19	
G19 → G17	
G17 → G19	Drehung von 90 Grad um die X-Achse und anschließende Drehung von 90 Grad um die Z-Achse.
G18 → G17	
G19 → G18	

Diese Drehungen sind die gleichen, die auch ausgeführt werden müssen, um bei Ebenenwechsel die Komponenten des Werkzeuglängenvektors zu tauschen.

Die Werkzeuggrundorientierung wird bei einer aktiven Adaptertransformation mitgedreht.

Ist das folgende Settingdatum ungleich Null, wird die Werkzeuggrundorientierung bei einem Ebenenwechsel nicht gedreht:

- Fräswerkzeuge:
SD42940 \$SC_TOOL_LENGTH_CONST
- Dreh- und Schleifwerkzeuge:
SD42942 \$SC_TOOL_LENGTH_CONST_T

19.10.3 Beispiel

Es wird ein Fräswerkzeug mit der Länge L1=10 definiert, dessen Werkzeuggrundorientierung jedoch in der Winkelhalbierenden der X-Z-Ebene liegt.

Programmcode	Kommentar
N10 \$TC_DP1[1,1]=120	; Werkzeugtyp: Fräswerkzeug
N20 \$TC_DP3[1,1]=10	; Längenkorrekturvektor L1=10
N30 \$TC_DPV[1,1]=0	; N30 bis N60: Programmierung der Werkzeuggrundorientierung.
N40 \$TC_DPV3[1,1]=1	
N50 \$TC_DPV4[1,1]=0	
N60 \$TC_DPV5[1,1]=1	

Programmcode	Kommentar
N70 G17 F1000 X0 Y0 Z0 T1 D1	; Werkzeug aktivieren und Nullposition anfahren. ; Wegen der Werkzeuglänge ergeben sich die Maschinenpositionen X0 Y0 Z10.
N80 MOVT=10	; Inkrementelle Verfahrbewegung in Werkzeugrichtung. ; Die resultierenden Achspositionen sind X7.071 Y0 Z17.071.
N80 M30	

19.11 Werkzeugkorrektur-Sonderbehandlungen

19.11.1 Relevante Settingdaten

SD42900- 42960

Über die Settingdaten SD42900 - SD42940 lassen sich bezüglich der Werkzeugkorrektur folgende Einstellungen vornehmen:

- Vorzeichen der Werkzeuglänge
- Vorzeichen des Verschleiß
- Verhalten der Verschleißkomponenten beim Spiegeln von Geometrieachsen
- Verhalten der Verschleißkomponenten beim Wechsel der Bearbeitungsebene über Settingdaten
- Zuordnung der Werkzeuglängenkomponenten unabhängig vom tatsächlichen Werkzeugtyp
- Transformation der Verschleißkomponenten in ein geeignetes Koordinatensystem zur Beeinflussung der wirksamen Werkzeuglänge

Hinweis

In der nachfolgenden Beschreibung ist unter Verschleiß die Summe der Werte folgender Komponenten zu verstehen:

- Verschleißwerte: \$TC_DP12 bis \$TC_DP20
 - Summenkorrektur, bestehend aus:
 - Verschleißwerte: \$SCPX3 bis \$SCPX11
 - Einrichtungswerte: \$ECPX3 bis \$ECPX11
-

Ausführliche Informationen zu Summen- und Werkzeugkorrekturen finden sich in:

Literatur:

Funktionshandbuch Werkzeugverwaltung

Programmierhandbuch Grundlagen; Werkzeugkorrekturen

Erforderliche Settingdaten

- SD42900 \$SC_MIRROR_TOOL_LENGTH (Spiegeln von Werkzeuglängenkomponenten und Komponenten des Basismaßes)
- SD42910 \$SC_MIRROR_TOOL_WEAR (Spiegeln von Verschleißwerten der Werkzeuglängenkomponenten)
- SD42920 \$SC_WEAR_SIGN_CUTPOS (Vorzeichenbewertung der Verschleißkomponenten)
- SD42930 \$SC_WEAR_SIGN (Invertiert die Vorzeichen der Verschleißmaße)
- SD42940 \$SC_TOOL_LENGTH_CONST (Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen)
- SD42950 \$SC_TOOL_LENGTH_TYPE (Zuordnung der Werkzeuglängenkomponenten unabhängig vom Werkzeugtyp)
- SD42935 \$SC_WEAR_TRANSFORM (Transformation der Verschleißwerte)
- SD42960 \$SC_TOOL_TEMP_COMP (Werkzeuglängenoffsets)

19.11.2 Werkzeuglängen spiegeln (SD42900 \$SC_MIRROR_TOOL_LENGTH)

Aktivierung

Die Aktivierung der Werkzeuglängen-Spiegelung erfolgt über das Settingdatum:

SD42900 \$SC_MIRROR_TOOL_LENGTH <> 0 (TRUE) (Vorzeichenwechsel Werkzeuglänge beim Spiegeln)

Funktion

Durch Vorzeicheninvertierung werden folgende Komponenten gespiegelt:

- Werkzeuglängen: \$TC_DP3, \$TC_DP4, \$TC_DP5
- Basismaße: \$TC_DP21, \$TC_DP22, \$TC_DP23

Die Spiegelung erfolgt für alle Basismaße, deren zugehörige Achsen gespiegelt sind. Verschleißwerte werden **nicht** mitgespiegelt.

Verschleißwerte spiegeln

Zum Spiegeln der Verschleißwerte ist folgendes Settingdatum zu setzen:

SD42910 \$SC_MIRROR_TOOL_WEAR <> 0 (Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln)

Durch Vorzeicheninvertierung werden die Verschleißwerte der Werkzeuglängenkomponenten, deren zugehörige Achsen gespiegelt sind, gespiegelt.

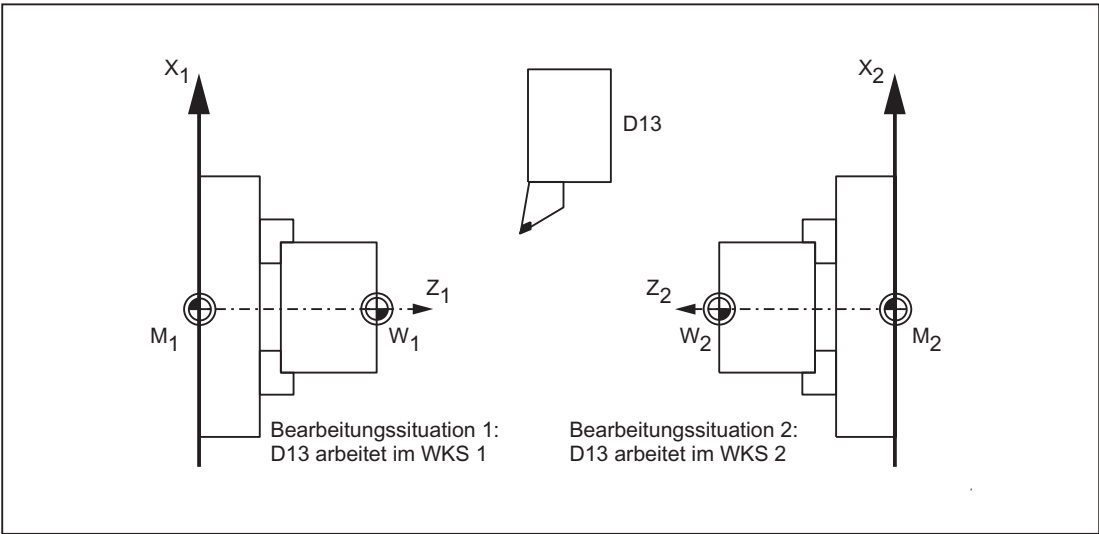


Bild 19-49 Anwendungsbeispiel: Doppelspindel-Drehmaschine

19.11.3 Verschleißlängen spiegeln (SD42920 \$SC_WEAR_SIGN_CUTPOS)

Aktivierung

Die Aktivierung der Verschleißlängen-Spiegelung erfolgt über:
SD42920 \$SC_WEAR_SIGN_CUTPOS <> 0 (TRUE) (Vorzeichen des Verschleißes bei Werkzeugen mit Schneidenlage)

Funktion

Schneidenlage	Länge 1	Länge 2
1	---	---
2	---	invertiert
3	invertiert	invertiert
4	invertiert	---
5	---	---
6	---	---
7	---	invertiert
8	invertiert	---
9	---	---

Bei Werkzeugtypen ohne relevanter Schneidenlage wird keine Verschleißlängen-Spiegelung durchgeführt.

Hinweis

Die Spiegelung (Vorzeicheninvertierung) in einer oder mehreren Komponenten kann sich aufheben durch gleichzeitige Aktivierung der Funktionen:

Werkzeuglängen-Spiegelung (SD42900 <> 0)

Und:

Verschleißlängen-Spiegelung (SD42920 <> 0)

SD42930 \$SC_WEAR_SIGN

Settingdatum ungleich Null:

Das Vorzeichen aller Verschleißmaße wird invertiert. Es wirkt sowohl auf die Werkzeuglängen als auch auf die übrigen Größen wie Werkzeugradius, Verrundungsradius usw.

Wird ein positives Verschleißmaß eingegeben, wird somit das Werkzeug "kürzer" und "dünner".

Wirksamwerden der veränderten Settingdaten

Die Neubewertung von Werkzeugkomponenten bei einer Änderung der beschriebenen Settingdaten wird erst wirksam, wenn das nächste Mal eine Werkzeugschneide angewählt wird. Ist ein Werkzeug bereits aktiv und die Bewertung der Daten dieses Werkzeugs soll verändert wirksam werden, muss dieses Werkzeug erneut angewählt werden.

Beispiel:

```
N10 $SC_WEAR_SIGN = 0           ; Keine Vorzeichenumkehr der Verschleißwerte
N20 $TC_DP1[1,1] = 120         ; Schaftfräser
N30 $TC_DP6[1,1] = 100         ; Werkzeugradius 100 mm
N40 $TC_DP15[1,1] = 1          ; Verschleißmaß des Werkzeugradius 1 mm, resul-
                                tierender Werkzeugradius 101 mm

N100 T1 D1 G41 X150 Y20
....
N150 G40 X300N10
....
N200 $SC_WEAR_SIGN = 1         ; Vorzeichenumkehr aller Verschleißwerte, durch
                                Neuanwahl (D1) wird der neue Radius von 99 mm
                                wirksam. Ohne D1 wäre der wirksame Radius wei-
                                terhin 101 mm.

N300 D1 G41 X350 Y-20
N310 ....
```

Entsprechendes gilt für den Fall, dass sich die resultierende Werkzeuglänge ändert, weil der Spiegelungszustand einer Achse geändert wurde. Das Werkzeug muss nach dem

Spiegelbefehl erneut angewählt werden, damit die geänderten Werkzeuglängenkomponenten wirksam werden.

19.11.4 Werkzeuglänge und Ebenenwechsel (SD42940 \$SC_TOOL_LENGTH_CONST)

Ebenenwechsel

Die Zuordnung der Werkzeuglängenkomponenten (Länge, Verschleiß und Basismaß) zu den Geometrieachsen beim Wechsel der Bearbeitungsebene (G17–G19) wird nicht verändert.

Zuordnung der Werkzeuge

Die Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen für Dreh- und Schleifwerkzeuge (WZ-Typ 400 bis 599) ergibt sich gemäß folgender Tabelle aus dem Wert des Settingdatums:

SD42940 \$SC_TOOL_LENGTH_CONST (Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel)

Ebene	Länge 1	Länge 2	Länge 3
17	Y	X	Z
*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X
*) Jeder Wert ungleich 0, der nicht gleich einem der sechs aufgeführten Werte ist, wird wie der Wert 18 bewertet.			

Folgende Tabelle zeigt die Zuordnung der Werkzeuglängenkomponenten zu den Geometrieachsen für **alle anderen Werkzeuge** (WZ-Typ < 400 bzw. > 599):

Ebene	Länge 1	Länge 2	Länge 3
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z
*) Jeder Wert ungleich 0, der nicht gleich einem der sechs aufgeführten Werte ist, wird wie der Wert 17 bewertet.			

Hinweis

Bei der Darstellung in den Tabellen wird davon ausgegangen, dass die Geometrieachsen 1 bis 3 mit X, Y, Z bezeichnet werden. Für die Zuordnung einer Korrektur zu einer Achse ist nicht der Achsname, sondern die Achsreihenfolge maßgebend.

Drei WZ-Längenkomponenten können auf obige 6 verschiedene Arten angeordnet werden.

19.11.5 Werkzeugtyp (SD42950 \$SC_TOOL_LENGTH_TYPE)**Merkmale**

Die Zuordnung der Werkzeuglängenkomponenten (Länge, Verschleiß und Basismaß) unabhängig zu den Geometrieachsen vom Werkzeugtyp festlegen.

Settingdatum **ungleich** Null: (Festlegung erfolgt standardmäßig)

Es wird zwischen Dreh- und Schleifwerkzeugen (Werkzeugtypen 400 bis 599) und anderen Werkzeugen (Fräswerkzeugen) unterschieden.

Die Werte 0 bis 2 können angenommen werden. Jeder andere Wert wird wie der Wert 0 behandelt.

Die Zuordnung der Werkzeuglängenkomponenten erfolgt unabhängig vom tatsächlichen Werkzeugtyp:

- Beim Wert 1: immer wie bei Fräswerkzeugen
- Beim Wert 2: immer wie bei Drehwerkzeugen

Orientierbare Werkzeugträger**Settingdaten SD42900 - SD42950**

Die Settingdaten SD42900 - SD42950 wirken nicht auf die Komponenten eines eventuell aktiven orientierbaren Werkzeugträgers. Ein Werkzeug geht immer mit seiner gesamten resultierenden Länge (Werkzeuglänge + Verschleiß + Basismaß) in die Berechnung mit einem

orientierbaren Werkzeugträger ein. Bei der Berechnung der resultierenden Gesamtlänge werden alle Änderungen, die durch die Settingdaten verursacht wurden, berücksichtigt.

Hinweis

Häufig werden beim Einsatz orientierbarer Werkzeugträger alle Werkzeuge für ein nicht gespiegeltes Grundsystem definiert, auch diejenigen, die nur bei Spiegelbearbeitung verwendet werden. Bei Bearbeitung mit gespiegelten Achsen wird dann der Werkzeugträger so gedreht, dass die tatsächliche Lage des Werkzeugs richtig beschrieben wird. Alle Werkzeuglängenkomponenten wirken dann automatisch in der richtigen Richtung, so dass sich eine Steuerung der Bewertung einzelner Komponenten über Settingdaten abhängig vom Spiegelungszustand einzelner Achsen erübrigt.

Der Einsatz orientierbarer Werkzeugträger ist auch dann sinnvoll, wenn der Maschinentyp es physikalisch nicht zulässt, Werkzeuge die mit verschiedenen Orientierungen fest installiert sind, auch zu drehen. Die Werkzeugvermessung kann dann einheitlich in einer Grundorientierung vorgenommen werden, und die für die Bearbeitung relevanten Maße ergeben sich durch Drehungen eines virtuellen Werkzeugträgers.

19.11.6 Werkzeuglängen im WKS unter Berücksichtigung der Orientierung

Wechsel Werkzeug oder Arbeitsebene

Die angezeigten Werte für das Werkzeug entsprechen der Ausdehnung im WKS. Soll ein Werkzeugträger mit schräger Werkzeugaufhängung verwendet werden, so ist darauf zu achten, dass die verwendete Transformation diesen unterstützt. Ist dies nicht der Fall, werden falsche Werkzeugabmessungen angezeigt. Beim Wechseln der Arbeitsebene von G17 auf G18 oder G19 sollte ebenfalls sichergestellt werden, dass die Transformation auch für diese Arbeitsebenen verwendet werden kann. Handelt es sich um eine Transformation, die nur für G17-Bearbeitung vorgesehen ist, so werden beim Ebenenwechsel auch weiterhin die Abmessungen für ein Werkzeug in Z-Richtung angezeigt.

Bei abgeschalteter Transformation wird das Basiswerkzeug in X-, Y- oder Z-Richtung, je nach Arbeitsebene, angezeigt. Ein programmierter Werkzeugträger wird berücksichtigt. Diese Abmessungen des Werkzeugs werden beim Verfahren ohne Transformation nicht verändert.

19.11.7 Werkzeuglängenoffsets in Werkzeugrichtung

Temperaturkompensation in Echtzeit

Bei 5-Achs-Maschinen mit beweglichem Werkzeug können in den Bearbeitungsköpfen Temperaturveränderungen auftreten, die wiederum Ausdehnungsänderungen hervorrufen und in Form von Längenausdehnung auf die Werkzeugspindel übertragen. Ein typischer Fall ist bei 5-Achs-Köpfen z.B. die Wärmeausdehnung in Richtung der Spindellängsachse.

Diese Wärmeausdehnung kann auch bei vorhandener Werkzeugorientierung kompensiert werden, indem die Temperaturkompensationswerte nicht mehr den Maschinenachsen,

sondern dem Werkzeug zugeordnet werden. Damit sind Längenänderungen bei Änderung der Werkzeugorientierung kompensierbar.

Mit Hilfe der Orientierungstransformation, deren Richtung von der aktuellen Werkzeugorientierung abhängt, können Bewegungen in Echtzeit überlagert und gleichzeitig mitgedreht werden. Dabei werden die Kompensationswerte ständig im Werkzeugkoordinatensystem mitgeführt.

Die Temperaturkompensation wird nur wirksam, wenn die zu kompensierende Achse auch wirklich referenziert ist.

Aktivierung

Die Temperaturkompensation in Werkzeugrichtung wird aktiviert, indem man das folgende Maschinendatum auf einen Wert **ungleich Null** setzt:

MD20390 \$MC_TOOL_TEMP_COMP_ON (Aktivierung der Temperaturkompensation für Werkzeuglänge)

Zusätzlich muss das Bit 2 gesetzt sein für jede betroffene Kanalachse im Maschinendatum:

MD32750 \$MA_TEMP_COMP_TYPE[<Achsisindex>] (Temperaturkompensationstyp)

Das können mehr als drei Achsen sein, wenn als Folge von Geoachstausch oder Transformationsumschaltungen sukzessive mehr als drei Kanalachsen zeitweise zu Geometrieachsen werden können. Ist dieses Bit für eine bestimmte Kanalachse nicht gesetzt, kann in dieser Achse kein Kompensationswert wirksam werden. Andere Achsen bleiben davon jedoch unbeeinflusst. Ein Alarm wird in diesem Fall nicht ausgegeben.

Wirkungsbereiche

Die Temperaturkompensation in Werkzeugrichtung wirkt nur bei den generischen 5-Achstransformationen mit:

- Trafotyp 24
Zwei Achsen drehen das Werkzeug.
- Trafotyp 56
Eine Achse dreht das Werkzeug, die andere Achse dreht das Werkstück ohne Temperaturkompensation.

Bei der generischen 5-Achstransformation mit:

- Trafotyp 40
Beim drehbaren Werkstück ist die Werkzeugorientierung konstant, so dass in diesem Fall eine Bewegung der Rundachsen der Maschine keinen Einfluss auf die Richtung der Temperaturkompensation hat.

Die Temperaturkompensation in Werkzeugrichtung wirkt außerdem bei den Orientierungstransformationen (sind keine generischen 5-Achs-Trafos) mit:

- Trafotyp 64 bis 69
Drehbare Linearachse.

Hinweis

Bei allen anderen Transformationstypen ist die Temperaturkompensation aktivierbar. Sie wird aber nicht durch eine Änderung der Werkzeugorientierung beeinflusst. Die Achsen verfahren so, als ob keine Orientierungstransformation mit einer Temperaturkompensation aktiv wäre.

Grenzwerte

Die Kompensationswerte werden auf Maximalwerte begrenzt durch das Maschinendatum:

MD20392 \$MC_TOOL_TEMP_COMP_LIMIT[0...2] (Maximale Temperaturkompensation für Werkzeuglänge)

Die Vorbelegung des Grenzwertes ist 1 mm. Wird ein Temperaturkompensationswert vorgegeben, der größer als dieser Grenzwert ist, wird dieser ohne Alarm begrenzt.

SD42960

Die drei Temperaturkompensationswerte bilden zusammen einen Kompensationsvektor und sind enthalten im Settingdatum:

SD42960 \$SC_TOOL_TEMP_COMP[0...2] (Temperaturkompensation bezogen auf Werkzeuge)

Die Settingdaten werden z.B. mittels Synchronaktionen oder von der PLC aus vom Anwender beschrieben. Die Kompensationswerte können deshalb auch für andere Kompensationszwecke verwendet werden.

Alle drei Kompensationswerte wirken in Grundstellung bzw. bei ausgeschalteter Orientierungstransformation in Richtung der drei Geometrieachsen (in der typischen Reihenfolge X, Y, Z). Die Zuordnung der Komponenten zu den Geometrieachsen ist unabhängig vom Werkzeugtyp (Dreh- / Fräswerkzeuge bzw. Schleifwerkzeuge) und der angewählten Bearbeitungsebene G17 bis G19. Werteveränderungen in den Settingdaten werden sofort wirksam.

Orientierbarer Werkzeugträger

Ist ein orientierbarer Werkzeugträger aktiv, so wird der Vektor der Temperaturkompensation bei einer Orientierungsänderung mitgedreht. Das gilt unabhängig von einer eventuell aktiven Orientierungstransformation.

Ist ein orientierbarer Werkzeugträger zusammen mit einer generischen 5-Achs-Transformation oder einer Transformation mit drehbarer Linearachse aktiv, so wird der Vektor der Temperaturkompensation beiden Drehungen unterworfen.

Hinweis

Die Transformationen mit drehbaren Linearachsen berücksichtigen zwar die Änderung des Werkzeugvektors (Länge), jedoch **nicht** dessen Orientierungsänderung, die durch einen orientierbaren Werkzeugträger bewirkt wird.

Die Temperaturkompensationswerte folgen unmittelbar jeder wirksam werdenden Orientierungsänderung. Dies gilt insbesondere auch dann, wenn eine Orientierungstransformation ein- oder ausgeschaltet wird.

Gleiches gilt bei einer Änderung der Geometrieachsenzuordnung zu den Kanalachsen. Der Temperaturkompensationswert für eine Achse, die z.B. nach einem Transformationswechsel nicht mehr Geometrieachse ist, wird (interpolatorisch) auf Null reduziert. Umgekehrt baut sich in einer Achse, die nach einer Umschaltung zur Geometrieachse wird, ein eventuell vorhandener Temperaturkompensationswert sofort auf.

Beispiele

Temperaturkompensation in Werkzeugrichtung

Es ist eine 5-Achs-Maschine mit drehbarem Werkzeug gegeben, bei der das Werkzeug um die C- und um die B-Achse gedreht werden kann.

In Grundstellung ist das Werkzeug parallel zur Z-Achse. Wird die B-Achse um 90 Grad gedreht, zeigt das Werkzeug in X-Richtung.

Deshalb wirkt ein Temperaturkompensationswert im folgenden Settingdatum bei aktiver Transformation ebenfalls in Richtung der Maschinen X-Achse:

SD42960 \$SC_TOOL_TEMP_COMP[2] (Temperaturkompensation bezogen auf Werkzeuge)

Wird in dieser Stellung die Transformation ausgeschaltet, ist die Werkzeugorientierung definitionsgemäß wieder parallel zur Z-Achse und weicht damit von der tatsächlichen Orientierung ab. Der Temperaturoffset in Richtung der X-Achse wird deshalb abgebaut und gleichzeitig in Z-Richtung aufgebaut.

Es sei eine 5-Achs-Maschine mit drehbarem Werkzeug (Trafotyp 24) gegeben. Die wesentlichen Maschinendaten sind im Folgenden dargestellt:

- Die erste Rundachse dreht um Z C-Achse.
- Die zweite Rundachse dreht um Y B-Achse.

Die wesentlichen Maschinendaten sind in der folgenden Tabelle dargestellt:

Maschinendatum	Wert	Anmerkung
MD20390 \$MC_TOOL_TEMP_COMP_ON	= TRUE	Temperaturkompensation aktiv
MD32750 \$MA_TEMP_COMP_TYPE[AX1]	= 4	Korrektur in Werkzeugrichtung
MD32750 \$MA_TEMP_COMP_TYPE[AX2]	= 4	Korrektur in Werkzeugrichtung
MD32750 \$MA_TEMP_COMP_TYPE[AX3]	= 4	Korrektur in Werkzeugrichtung

Maschinendatum	Wert	Anmerkung
		Zuordnung des Trafotyps 24:
MD24100 \$MC_TRAFO_TYPE_1	= 24	Trafotyp 24 im 1. Kanal
MD24110 \$MC_TRAFO_AXES_IN_1[0]	= 1	1. Achse der Transformation
MD24110 \$MC_TRAFO_AXES_IN_1[1]	= 2	2. Achse der Transformation
MD24110 \$MC_TRAFO_AXES_IN_1[2]	= 3	3. Achse der Transformation
MD24110 \$MC_TRAFO_AXES_IN_1[3]	= 5	5. Achse der Transformation
MD24110 \$MC_TRAFO_AXES_IN_1[4]	= 4	4. Achse der Transformation
MD24120 \$MC_TRAFO_GEOAX_AS-SIGN_TAB_1[0]	= 1	Geoachse zu Kanalachse 1
MD24120 \$MC_TRAFO_GEOAX_AS-SIGN_TAB_1[1]	= 2	Geoachse zu Kanalachse 2
MD24120 \$MC_TRAFO_GEOAX_AS-SIGN_TAB_1[2]	= 3	Geoachse zu Kanalachse 3
MD24570 \$MC_TRAFO5_AXIS1_1[0]	= 0.0	
MD24570 \$MC_TRAFO5_AXIS1_1[1]	= 0.0	Richtung
MD24570 \$MC_TRAFO5_AXIS1_1[2]	= 1.0	1. Rundachse ist parallel Z
MD24572 \$MC_TRAFO5_AXIS1_2[0]	= 0.0	Richtung
MD24572 \$MC_TRAFO5_AXIS1_2[1]	= 1.0	2. Rundachse ist parallel Y
MD24572 \$MC_TRAFO5_AXIS1_2[2]	= 0.0	
MD25574 \$MC_TRAFO5_BASE_ORIENT_1[0]	= 0.0	
MD25574 \$MC_TRAFO5_BASE_ORIENT_1[1]	= 0.0	Werkzeuggrundorientierung
MD25574 \$MC_TRAFO5_BASE_ORIENT_1[2]	= 1.0	in Z-Richtung

Temperaturkompensationswerte im NC-Programm

Die den beiden Achsen X und Z zugeordneten Kompensationswerte sind ungleich Null und werden bei der Temperaturkompensation bezüglich der Werkzeuglängen berücksichtigt. Die jeweils erreichten Maschinenachsenpositionen sind als Kommentare in den Programmzeilen angegeben.

Programmcode	Kommentar
\$SC_TOOL_TEMP_COMP[0] = -0.3	; Kompensationswert in X
\$SC_TOOL_TEMP_COMP[1] = 0.0	
\$SC_TOOL_TEMP_COMP[2] = -1.0	; Kompensationswert in Z
	; Sollpositionen der Maschinenachsen
N10 G74 X0 Y0 Z0 A0 B0	; X Y Z
N20 X20 Y20 Z20 F10000	; 20.30 20.00 21.00
N30 TRAORI()	; 20.30 20.00 21.00
N40 X10 Y10 Z10 B90	; 11.00 10.00 9.70
N50 TRAFOOF	; 10.30 10.00 11.00
N60 X0 Y0 Z0 B0 C0	; 0.30 0.00 1.00

Programmcode	Kommentar
N70 M30	

Mit Ausnahme des Satzes N40 wirkt die Temperaturkompensation immer in den Originalrichtungen, da das Werkzeug in Richtung der Grundorientierung zeigt. Das gilt insbesondere auch im Satz N50. Das Werkzeug zeigt zwar tatsächlich noch in Richtung der X-Achse, weil die B-Achse noch auf 90 Grad steht. Die zu berücksichtigende Orientierung ist aber wegen der bereits ausgeschalteten Transformation wieder parallel zu Z.

Maschinendatum	Wert	Anmerkung
MD20390 \$MC_TOOL_TEMP_COMP_ON	= TRUE	Temperaturkompensation aktiv
MD32750 \$MA_TEMP_COMP_TYPE[AX1]	= 4	Korrektur in Werkzeugrichtung
MD32750 \$MA_TEMP_COMP_TYPE[AX2]	= 4	Korrektur in Werkzeugrichtung
MD32750 \$MA_TEMP_COMP_TYPE[AX3]	= 4	Korrektur in Werkzeugrichtung

Weitere Literatur

Weitere Erläuterungen zur "Temperaturkompensation" siehe:

Literatur:

Funktionshandbuch Erweiterungsfunktionen; Kompensationen (K3)

Informationen zur "Generischen 5-Achstransformation" siehe:

Literatur:

Funktionshandbuch Sonderfunktionen; 3- bis 5-Achs-Transformation (F2)

19.12 Summen- und Einrichtekorrekturen

19.12.1 Allgemeine Informationen

Summenkorrekturen

Summenkorrekturen können als in der Bearbeitung **programmierbare Prozesskorrekturen** betrachtet werden und setzen sich aus all den Fehlergrößen zusammen (einschließlich dem Verschleiß), die die Abweichung des Werkstücks vom Sollmaß ausmachen.

Summenkorrekturen sind eine verallgemeinerte Form von Verschleiß. Sie sind Bestandteil der Werkzeugschneidendaten. Die Parameter der Summenkorrektur beziehen sich auf die geometrischen Daten einer Schneide.

Die Korrekturdaten einer Summenkorrektur werden über eine **DL-Nummer** angesprochen (**DL**: Location dependent; Korrekturen bezüglich des jeweiligen Einsatzortes).

Die Verschleißwerte einer D-Nummer beschreiben dagegen den physikalischen Verschleiß der Schneide, d.h. im speziellen Fall kann die Summenkorrektur dem Verschleiß der Schneide entsprechen.

Summenkorrekturen können allgemein eingesetzt werden; d.h. mit aktiver oder inaktiver Werkzeugverwaltung sowie auch mit flacher D-Nummernfunktion.

Die Summenkorrekturen werden durch Setzen entsprechender Maschinendaten aufgeteilt in:

- Summenkorrektur fein
- Summenkorrektur grob (Einrichtekorrektur)

Einrichtekorrektur

Die Einrichtekorrektur ist die vom Einrichter vor der Bearbeitung einzustellende Korrektur. Diese Werte belegen im NC eigenen Speicher. Dem Bediener wird dann nur die "Summenkorrekturen fein" über HMI zugänglich gemacht.

"Summenkorrektur fein" und "Summenkorrektur grob" werden NC-intern addiert. Dieser Wert soll im Folgenden als Summenkorrektur bezeichnet werden.

Hinweis

Die Funktion wird eingeschaltet über die Maschinendatum-Einstellung:

MD18080 \$MN_MM_TOOL_MANAGEMENT_MASK, Bit 8=1 (Stufenweise Speicher-Reservierung für die Werkzeugverw.)

Bei aktiven kinematischen Transformationen (z.B. 5-Achs-Transformation) wird die Werkzeuglänge zunächst unter Berücksichtigung der verschiedenen Verschleißkomponenten wie beschrieben berechnet. Diese gesamte Werkzeuglänge geht dann in die Transformation ein. Damit werden auch die Verschleißwerte unabhängig vom G-Befehl der Gruppe 56 - anders als beim orientierbaren Werkzeugträger - immer mittransformiert.

19.12.2 Funktionale Beschreibung

Summenkorrekturen

Pro D-Nummer können mehrere Summenkorrekturen (DL-Nummern) definiert werden. Damit lassen sich z.B. **werkstückortsabhängige** Korrekturwerte ermitteln und einer Schneide zuweisen. Summenkorrekturen wirken wie Verschleiß (d.h. additiv auf die Korrekturwerte der D-Nummer). Die Daten sind fest einer D-Nummer zugeordnet.

Einstellungen

Über Maschinendaten können Sie folgende Einstellungen vornehmen:

- Summenkorrektur aktivieren
- Anzahl der maximal im NC anzulegenden DL-Datensätze festlegen
- Anzahl der maximal einer D-Nummer zuzuordnenden DL-Nummern festlegen

- Festlegen, ob die Summenkorrekturen (fein / grob) bei Datensicherung mitgesichert werden sollen
- Festlegen, welche Summenkorrektur aktiviert werden soll, wenn:
 - eine neue Schneidenkorrektur aktiviert wird
 - ein Bedientafelfront-RESET erfolgt
 - ein Bedientafelfront-START erfolgt
 - das Programmende erreicht wurde

Dabei orientiert sich der Name an der Logik der entsprechenden Maschinendaten für Werkzeuge und Schneiden.

"Einrichtekorrektur" und "Summenkorrektur fein" können über Systemvariablen und entsprechende BTSS-Dienste gelesen und geschrieben werden.

Hinweis

Bei aktiver WZV kann über ein Maschinendatum festgelegt werden, ob die Summenkorrektur von einem Werkzeug, das während eines programmierten Werkzeugwechsels aktiviert wird, unverändert bleibt oder auf Null gesetzt werden soll.

Zusammenfassung der Korrekturparameter \$TC_DPx

Bisher wurden folgende allgemeine Systemvariablen zur Beschreibung einer Schneide definiert:

\$TC_DP1	Werkzeugtyp
\$TC_DP2	Schneidenlage

Parameter für Geometrie und Verschleiß

Die Systemvariablen \$TC_DP3 bis \$TC_DP11 sind die Korrekturen der Werkzeuggeometrie zugeordnet. Die Systemvariablen \$TC_DP12 bis \$TC_DP20 erlauben es, für jeden dieser Parameter einen Verschleiß zu benennen.

Geometrie	Verschleiß	Längenkorrekturen
\$TC_DP3	\$TC_DP12	Länge 1
\$TC_DP4	\$TC_DP13	Länge 2
\$TC_DP5	\$TC_DP14	Länge 3
Geometrie	Verschleiß	Radiuskorrektur
\$TC_DP6	\$TC_DP15	Radius
\$TC_DP7	\$TC_DP16	Eckenradius (WZ-Typ 700; Nutsäge)
Geometrie	Verschleiß	weitere Korrekturen
\$TC_DP8	\$TC_DP17	Länge 4 (WZ-Typ 700; Nutsäge)
\$TC_DP9	\$TC_DP18	Länge 5
\$TC_DP10	\$TC_DP19	Winkel 1 (Winkel zwischen Stirnfläche des WZs und Torusfläche)
\$TC_DP11	\$TC_DP20	Winkel 2 (Winkel zwischen WZ-Längsachse und oberem Ende der Torusfläche)

Basismaß bzw. Adaptermaß

\$TC_DP21	Adapter - Länge 1
\$TC_DP22	Adapter - Länge 2
\$TC_DP23	Adapter - Länge 3

Technologie

Systemvariable	Freiwinkel
\$TC_DP24	<ul style="list-style-type: none"> Für ManualTurn wird hier der Freiwinkel abgelegt; WZ-Typ 5xx. Gleiche Bedeutung wie in Standardzyklen für Drehwerkzeuge. Für ShopMill wird hier der Spitzenwinkel des Bohrers abgelegt; WZ-Typ 2xx. Verwendung in Standardzyklen für Drehwerkzeuge; WZ-Typ 5xx. Bei diesen Werkzeugen ist das der Winkel an der Nebenschneide.
\$TC_DP25	<ul style="list-style-type: none"> Für ManualTurn wird hier der Wert für die Schnittgeschwindigkeit abgelegt. Für ShopMill wird hier ein bitcodierter Wert für verschiedene Zustände von Werkzeugen des Typs 1xx und 2xx abgelegt.

Parameter der Summen und Einrichtekorrekturen (\$TC_SCPxy-, \$TC_ECPxy)

Die Nummerierung der Parameter orientiert sich an der Nummerierung der Systemvariablen \$TC_DP3 bis \$TC_DP11.

Die Wirkung der Parameter ist dem Verschleiß analog (additiv zur WZ-Geometrie). Pro Schneidenparameter sind maximal sechs Summen-/Einrichteparameter definierbar.

WZ-Geometrie-Parameter, auf den die Korrektur addiert wird.	Summen- / Einrichteparameter - Längenkorrkturen	WZ-Verschleiß-Parameter
\$TC_DP3	Länge 1 \$TC_SCP13, \$TC_SCP23,\$TC_SCP33, \$TC_SCP43,\$TC_SCP53,\$TC_SCP63 \$TC_ECP13, \$TC_ECP23,\$TC_ECP33, \$TC_ECP43,\$TC_ECP53,\$TC_ECP63 Die fettgedruckten Ziffern 1, 2, ... 6 bezeichnen die Parameter der maximal sechs (ortsabhängigen o.ä.) mit DL =1, ... 6 programmierbaren Korrekturen zum in Spalte eins genannten Parameter.	\$TC_DP12
\$TC_DP4	Länge 2 \$TC_SCP14, \$TC_SCP24,\$TC_SCP34, \$TC_SCP44,\$TC_SCP54,\$TC_SCP64 \$TC_ECP14, \$TC_ECP24,\$TC_ECP34, \$TC_ECP44,\$TC_ECP54,\$TC_ECP64	\$TC_DP13
\$TC_DP5	Länge 3 etc. ...	\$TC_DP14
	Radiuskorrektur	

WZ-Geometrie-Parameter, auf den die Korrektur addiert wird.	Summen- / Einrichteparameter - Längenkorrekturen	WZ-Verschleiß-Parameter
\$TC_DP6	Radius	\$TC_DP15
\$TC_DP7	Eckenradius	\$TC_DP16
	weitere Korrekturen	
\$TC_DP8	Länge 4	\$TC_DP17
\$TC_DP9	Länge 5	\$TC_DP18
\$TC_DP10	Winkel 1 ...etc.	\$TC_DP19
\$TC_DP11	Winkel 2 \$TC_SCP21, \$TC_SCP31,\$TC_SCP41, \$TC_SCP51,\$TC_SCP61,\$TC_SCP71 \$TC_ECP21, \$TC_ECP31,\$TC_ECP41, \$TC_ECP51,\$TC_ECP61,\$TC_ECP71 Die fettgedruckten Ziffern 2, 3, ... 7 bezeichnen die Parameter der maximal sechs (ortsabhängigen o.ä.) mit DL =1, ... 6 programmierbaren Korrekturen zum in Spalte eins genannten Parameter.	\$TC_DP20

Randbedingungen

Die maximale Anzahl der DL-Datensätze einer Schneide und die Gesamtanzahl der Summenkorrekturen im NC werden über Maschinendaten festgelegt. Standardmäßig ist der Wert gleich Null, d.h. es können keine Summenkorrekturen programmiert werden.

Mit aktivierter "Überwachungsfunktion" ist es möglich, ein Werkzeug auf Verschleiß bzw. auch auf "Summenkorrektur" zu überwachen.

Die zusätzlichen Summen-/Einrichtedatensätze belegen zusätzlichen gepufferten Speicher. Pro Parameter werden 8 Byte benötigt.

Ein Summenkorrekturdatensatz benötigt: 8 Byte * 9 Parameter = 72 Bytes

Ein Einrichtekorrekturdatensatz benötigt ebensoviel. Zusätzlich werden einige Bytes an internen Verwaltungsdaten benötigt.

19.12.3 Aktivierung

Funktion

Die Funktion muss aktiviert werden über das Maschinendatum:

MD18108 \$MN_MM_NUM_SUMCORR (Summenkorrekturen im TO-Bereich)

Über die Systemvariable \$TC_ECPx und \$TC_SCPx bzw. über die BTSS-Schnittstelle definierte Einrichte- und Summenkorrekturen ("fein") können im Teileprogramm aktiviert werden.

Dies erfolgt durch Programmierung des Sprachbefehls `DL="nr"`.

Mit dem Aktivieren einer neuen D-Nummer ist entweder eine neue DL-Nummer zu programmieren, oder es wird die über folgendes Maschinendatum festgelegte DL-Nummer wirksam:

MD20272 \$MC_SUMCORR_DEFAULT (Grundstellung Summenkorrektur ohne Programm)

DL-Programmierung

Die Programmierung der Summenkorrektur ist immer relativ zur aktiven D-Nummer und erfolgt mit dem Befehl:

DL = "n"

Die Summenkorrektur "n" wird zum Verschleiß der aktiven D-Nummer addiert.

Hinweis

Wird mit "Einrichtekorrektur" und "Summenkorrektur fein" gearbeitet, werden beide Korrekturen zusammengefasst und dann zum Verschleiß des Werkzeuges addiert.

Die Abwahl der Summenkorrektur erfolgt mit dem Befehl:

DL = 0

Hinweis

DL0 ist nicht erlaubt. Mit der Abwahl der Korrektur (D0 und T0) wird die Summenkorrektur ebenfalls unwirksam.

Programmieren einer nicht vorhandenen Summenkorrektur löst einen Alarm aus, analog der Programmierung einer nicht vorhandenen D-Korrektur.

Damit ist nur noch der definierte Verschleiß Bestandteil der Korrektur (in den Systemvariablen \$TC_DP12 bis \$TC_DP20 definiert).

Wird bei aktiver D-Korrektur eine Summenkorrektur programmiert (auch bei Abwahl), wirkt sich das auf die Bahn genauso aus wie die Programmierung von D. Damit verliert z.B. eine gerade aktive Radiuskorrektur ihren Bezug zu Nachbarsätzen.

Konfiguration

MD18112 \$MN_MM_KIND_OF_SUMCORR, Bit 4=0: (Eigenschaften der Summenkorrekturen im TO-Bereich) Standardeinstellung:

Es ist nur **ein** Datensatz Summenkorrekturen pro DL-Nummer vorhanden.

Es wird allgemein von Summenkorrektur gesprochen.

Es sind damit die durch \$TC_SCPx dargestellten Daten gemeint.

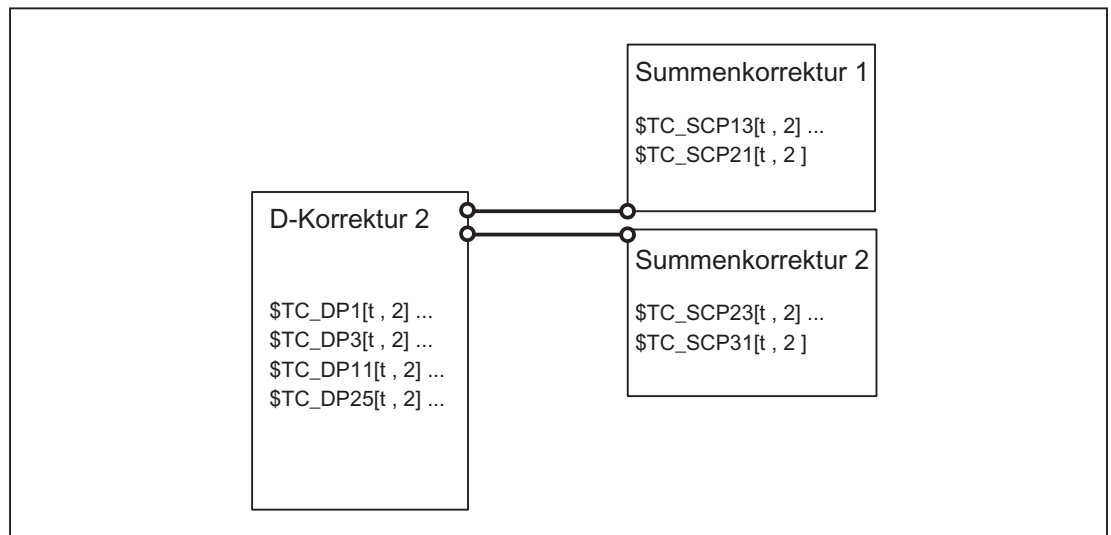


Bild 19-50 MD18112 \$MN_MM_KIND_OF_SUMCORR, Bit 4 = 0

Das Werkzeug mit T = t ist aktiv. Mit den Daten im Bild wird programmiert:

D2	;	Schneiden Korrekturen; d.h. \$TC_DP3,... \$TC_DP11 + Verschleiß (\$TC_DP12,...\$TC_DP20) + Adaptermaß
...		
DL=1	;	zusätzlich zu den bisherigen Korrekturen von D2 wird die Summenkorrektur 1 addiert d.h. \$TC_SCP13,...\$TC_SCP21
...		
DL=2	;	zur Korrektur D2 wird nicht mehr Summenkorrektur 1 addiert, sondern Summenkorrektur 2 d.h. \$TC_SCP23,...\$TC_SCP31
...		
DL=0	;	Abwahl der Summenkorrektur; nur noch die Daten von D2 sind wirksam

MD18112 \$MN_MM_KIND_OF_SUMCORR, Bit 4=1: Einrichtekorrekturen stehen zur Verfügung

Die Summenkorrektur setzt sich nun zusammen aus der "Summenkorrektur fein" (dargestellt durch \$TC_SCPx) und der Einrichtekorrektur (dargestellt durch \$TC_ECPx). Damit gibt es zu einer DL-Nummer zwei Datensätze. Die Summenkorrektur ergibt sich durch Addition der entsprechenden Komponenten (\$TC_ECPx + \$TC_SCPx).

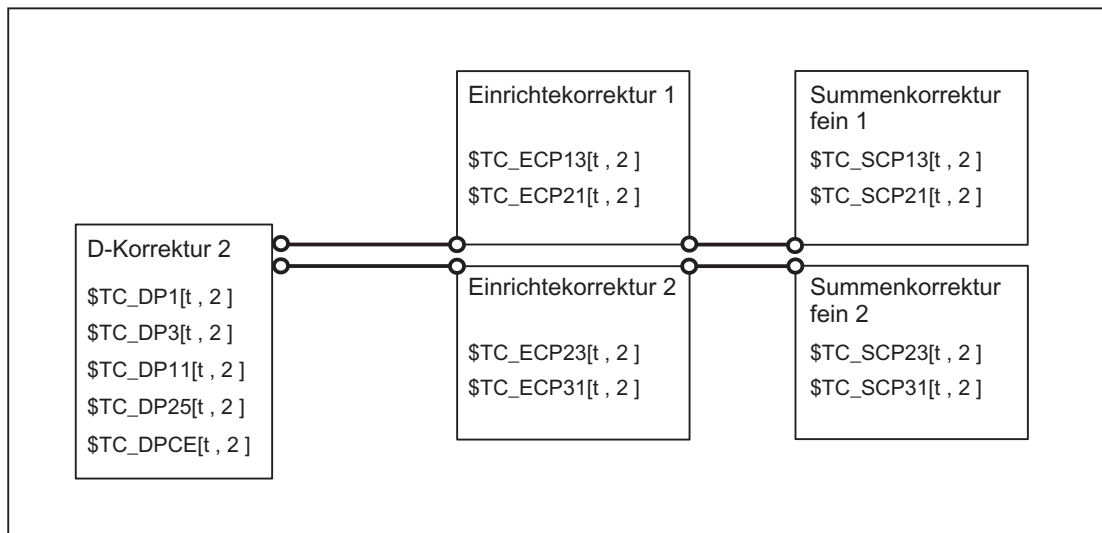


Bild 19-51 MD18112 \$MN_MM_KIND_OF_SUMCORR, Bit 4 = 1 "Einrichtekorrekturen" + "Summenkorrekturen fein"

Das Werkzeug mit T = t ist aktiv. Mit den Daten im Bild wird programmiert:

D2	;	Schneiden Korrekturen; d.h. \$TC_DP3,... \$TC_DP11 + Verschleiß (\$TC_DP12,...\$TC_DP20) + Adaptermaß
...		
DL=1	;	zusätzlich zu den bisherigen Korrekturen von D2 wird die Summenkorrektur 1 addiert; d.h. \$TC_ECP13 + \$TC_SCP13,...\$TC_ECP21 + \$TC_SCP21
...		
DL=2	;	zur Korrektur D2 wird nicht mehr Summenkorrektur 1 addiert, sondern Summenkorrektur 2; d.h. \$TC_ECP23 + \$TC_SCP23,...\$TC_ECP31 + \$TC_SCP31
...		
DL=0	;	Abwahl der Summenkorrektur. Nur noch die Daten von D2 sind wirksam.

Lesen / Schreiben im Teileprogramm

Die einzelnen Summenkorrekturparametersätze werden durch Nummernbereiche der Systemvariablen \$TC_SCP unterschieden.

Die Bedeutung der einzelnen Variablen ist analog den Geometrie-Variablen \$TC_DP3 bis \$TC_DP11. Für die Grundfunktionalität sind nur Länge 1, Länge 2, Länge 3 gesetzt (Variablen \$TC_SCP13 - \$TC_SCP15 für die erste Summenkorrektur der Schneide).

R5 = \$TC_SCP13[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der ersten Komponente der Summenkorrektur 1 der Schneide (d) des Werkzeuges (t).
R6 = \$TC_SCP21[t, d]	;	Setzt den Wert des R-Parameters auf den Wert der letzten Komponente der Summenkorrektur 1 der Schneide (d) des Werkzeuges (t).

$R5 = \$TC_SCP23[t, d]$;	Setzt den Wert des R-Parameters auf den Wert der ersten Komponente der Summenkorrektur 2 der Schneide (d) des Werkzeuges (t).
$\$TC_SCP43[t, d] = 1.234$;	Setzt den Wert der ersten Komponente der Summenkorrektur 4 der Schneide (d) des Werkzeuges (t) auf den Wert 1.234.

Die Aussagen für die Einrichtekorrekturen (falls NCK damit konfiguriert ist) sind analog, d.h.:

$R5 = \$TC_ECP13[t, d]$;	Setzt den Wert des R-Parameters auf den Wert der ersten Komponente der Einrichtekorrektur 1 der Schneide (d) des Werkzeuges (t).
$R6 = \$TC_ECP21[t, d]$;	Setzt den Wert des R-Parameters auf den Wert der letzten Komponente der Einrichtekorrektur 1 der Schneide (d) des Werkzeuges (t).
etc.		

Beim Arbeiten mit Einrichtekorrekturen werden die "Summenkorrekturen fein" mit den Systemvariablen $\$TC_SCP_x$ beschrieben.

Neuanlegen - Summenkorrektur

Falls der Korrekturdatensatz (x) noch nicht existiert, wird er mit der ersten Schreiboperation auf einen seiner Parameter (y) angelegt.

$\$TC_SCP_{xy}[t, d] = r.r$;	Der Parameter y der Summenkorrektur x erhält den Wert "r.r". Die anderen Parameter von x haben den Wert Null.
------------------------------	---	---

Beim Arbeiten mit Einrichtekorrekturen werden die "Summenkorrekturen fein" mit den Systemvariablen $\$TC_SCP_x$ beschrieben.

Hinweis

Beim Arbeiten mit Einrichtekorrekturen wird mit dem Anlegen eines Datensatzes für "Summenkorrektur fein" der zugehörige Datensatz für die Einrichtekorrektur mit angelegt, falls bis dahin zu [t, d] noch kein Datensatz existierte.

Neuanlegen - Einrichtekorrektur

Falls der Korrekturdatensatz (x) noch nicht existiert, wird er mit der ersten Schreiboperation auf einen seiner Parameter (y) angelegt.

$\$TC_ECP_{xy}[t, d] = r.r$;	Der Parameter y der Einrichtekorrektur x erhält den Wert "r.r". Die anderen Parameter von x haben den Wert Null.
------------------------------	---	--

Hinweis

Beim Arbeiten mit Einrichtekorrekturen wird mit dem Anlegen eines Datensatzes für Einrichtekorrekturen der zugehörige Datensatz für "Summenkorrektur fein" mit angelegt, falls bis dahin zu [t, d] noch kein Datensatz existierte.

DELDL - Löschen Summenkorrektur

Summenkorrekturen sind meist nur während der Bearbeitung mit einer Schneide zu einem bestimmten Zeitpunkt an einem bestimmten Ort des Werkstückes von Bedeutung. Mit dem NC-Sprachbefehl **DELDL** können die Summenkorrekturen aus Schneiden gelöscht werden (Freigeben von benötigtem Speicher).

status = DELDL(t, d)	;	löscht alle Summenkorrekturen der Schneide d des Werkzeuges t
		t, d sind optionale Parameter

Ist d nicht angegeben, werden alle Summenkorrekturen aller Schneiden des Werkzeuges t gelöscht.

Sind d und t nicht angegeben, werden alle Summenkorrekturen der Schneiden aller Werkzeuge der TO-Einheit gelöscht (für den Kanal, in dem der Befehl programmiert wird).

Falls mit Einrichtekorrekturen gearbeitet wird, werden mit dem **DELDL**-Befehl sowohl die Einrichtekorrektur, als auch die "Summenkorrektur fein" der genannten Schneide(n) gelöscht.

Hinweis

Beim "Löschen" wird der Speicher für die Datensätze freigegeben.

Anschließend können die gelöschten Summenkorrekturen nicht mehr aktiviert oder programmiert werden.

Summenkorrekturen, Einrichtekorrekturen aktiver Werkzeuge können nicht gelöscht werden (analog zum Löschverhalten von D-Korrekturen bzw. den Werkzeugdaten).

Der Rückgabewert "status" zeigt das Ergebnis des Löschbefehls an:

0:	Löschen erfolgreich durchgeführt
-1:	Löschen nicht (eine Schneide) oder nicht vollständig (mehrere Schneiden) durchgeführt

Datensicherung

Die Daten werden im Rahmen der allgemeinen Werkzeugdatensicherung (als Bestandteil der D-Nummerndatensätze) gesichert.

Eine Datensicherung der Summenkorrekturen ist sinnvoll, da es möglich sein muss, bei akuten Problemen den momentanen Zustand zu sichern. Über Maschinendaten können die

Summenkorrekturen von der Datensicherung ausgeschlossen werden (für "Einrichtekorrekturen" und "Summenkorrekturen fein" getrennt einstellbar).

Hinweis

Summenkorrekturen verhalten sich bzgl. Satzsuchlauf und Repos analog zu D-Korrekturen. Das Verhalten bei Reset und PowerOn kann durch Maschinendaten festgelegt werden.

Falls über das folgende Maschinedatum eingestellt sein, dass nach PowerOn die letzte aktive WZ-Korrekturnummer (D) aktiviert werden soll, ist die zuletzt aktive DL-Nummer jedoch nicht mehr aktiv:

MD20110 \$MC_RESET_MODE_MASK (Festlegung der Steuerungs-Grundstellung nach Reset / TP-Ende)

19.12.4 Beispiele

Beispiel 1

Dass keine Korrektur und keine Summenkorrektur wirksam werden, soll beim Werkzeugwechsel festgelegt werden durch die Maschinendaten:

- MD20270 \$MC_CUTTING_EDGE_DEFAULT=0 (Grundstellung der Werkzeugschneide ohne Programmierung)
- MD20272 \$MC_SUMCORR_DEFAULT=0 (Grundstellung Summenkorrektur ohne Programm)

```
T5 M06           ; WZ mit Nummer 5 wird eingewechselt - keine Korrektur aktiv
D1 DL=3          ; Korrektur D1 + Summenkorrektur 3 von D1 werden aktiv
X10
DL=2             ; Korrektur D1 + Summenkorrektur 2 werden aktiv
X20
DL=0             ; Summenkorrekturabwahl, nur noch Korrektur D1 ist aktiv
D2              ; Korrektur D2 wird aktiv - kein Beitrag der Summenkorrektur zur
                ; Korrektur
X1
DL=1             ; Korrektur D2 + Summenkorrektur 1 werden aktiv
X2
D0              ; Korrekturabwahl
X3
DL=2             ; ohne Auswirkung - DL2 von D0 ist Null (analog zur Programmie-
                ; rung T0 D2)
```

Beispiel 2

Beim Werkzeugwechsel soll festgelegt werden, dass die Korrektur D2 und die Summenkorrektur DL=1 wirksam werden, durch die Maschinendaten:

MD20270 \$MC_CUTTING_EDGE_DEFAULT=2 (Grundstellung der Werkzeugschneide ohne Programmierung)

MD20272 \$MC_SUMCORR_DEFAULT=1 (Grundstellung Summenkorrektur ohne Programm)

T5 M06	; WZ mit Nummer 5 wird eingewechselt - D2 + DL=1 sind aktiv (=Werte der Maschinendaten)
D1 DL=3	; Korrektur D1 + Summenkorrektur 3 von D1 werden aktiv
X10	
DL=2	; Korrektur D1 + Summenkorrektur 2 werden aktiv
X20	
DL=0	; Summenkorrekturabwahl, nur noch Korrektur D1 ist aktiv
D2	; Korrektur D2 wird aktiv - Summenkorrektur DL=1 wird aktiviert
X1	
DL=2	; Korrektur D2 + Summenkorrektur 2 werden aktiv
D1	; Korrektur D1 + Summenkorrektur 1 werden aktiv

19.12.5 Erweiterungen der Werkzeuglängenbestimmung

19.12.5.1 Korrekturen einsatzort- und werkstückspezifisch einrechnen

Zusammensetzung der wirksamen Werkzeuglänge

Bei einer Werkzeugkorrektur ohne aktive kinematische Transformation setzt sich die wirksame Werkzeuglänge aus maximal 8 Vektoren zusammen:

- | | |
|---|-----------------------------|
| • Werkzeuglänge (Geometrie) | (\$TC_DP3 - \$TC_DP5) |
| • Verschleiß | (\$TC_DP12 - \$TC_DP14) |
| • Basismaß (siehe Hinweis) | (\$TC_DP21 - \$TC_DP23) |
| • Adaptermaß (siehe Hinweis) | (\$TC_ADPT1 - \$TC_ADPT3) |
| • Summenkorrekturen fein | (\$TC_SCPx3 - \$TC_SCPx5) |
| • Summenkorrekturen grob bzw. Einrichtkorrekturen | (\$TC_ECPx3 - \$TC_ECPx5) |
| • Offsetvektor I_1 des orientierbaren Werkzeugträgers | (\$TC_CARR1 - \$TC_CARR3) |
| • Offsetvektor I_2 des orientierbaren Werkzeugträgers | (\$TC_CARR4 - \$TC_CARR6) |
| • Offsetvektor I_3 des orientierbaren Werkzeugträgers | (\$TC_CARR15 - \$TC_CARR17) |

Hinweis

Basismaß und Adaptermaß können nur alternativ wirksam sein.

Wirkungsweise der einzelnen Vektoren

Die Wirkungsweise der einzelnen Vektoren oder Gruppen von Vektoren hängt dabei von folgenden weiteren Größen ab:

Einflussgröße	Wirkungsweise
G-Befehle	aktive Bearbeitungsebene
Werkzeugtyp	Fräswerkzeuge bzw. Dreh- oder Schleifwerkzeuge
Maschinendaten	Werkzeugverwaltung aktiv / nicht aktiv, orientierbarer WZ-Träger vorhanden / nicht vorh.
Settingdaten	Verhalten von Werkzeuglängenkompontenten beim Spiegeln oder bei Ebenenwechsel
Orientierbare Werkzeugträger	Einstellwerte des orientierbaren Werkzeugträgers
Adaptertransformationen	transformierte Werkzeug-Korrekturwerte

Verteilung auf die Geometrieachskomponenten

Wie die drei Vektorkomponenten von Teilsummen der beteiligten Vektoren auf die drei Geometrieachskomponenten verteilt werden, bestimmen die folgenden Größen:

Einflussgröße	Abhängigkeiten
aktive Bearbeitungsebene: G17 X/Y-Richtung G18 Z/X-Richtung G19 Y/Z-Richtung	Zustellebene: Z Y X
Werkzeugtyp: Fräswerkzeuge, Bohrwerkzeuge, Schleifwerkzeuge, Drehwerkzeuge	Siehe Kapitel "WZ-Parameter 1: Werkzeugtyp (Seite 1502)", Tabelle "Minimal notwendige WZ-Parameter"
SD42900 \$SC_MIRROR_TOOL_LENGTH SD42910 \$SC_MIRROR_TOOL_WEAR SD42920 \$SC_WEAR_SIGN_CUTPOS SD42930 \$SC_WEAR_SIGN SD42940 \$SC_TOOL_LENGTH_CONST SD42950 \$SC_TOOL_LENGTH_TYPE	Siehe Kapitel "Werkzeugkorrektur-Sonderbehandlungen (Seite 1612)" und Kapitel "Settingdaten (Seite 1673)".
Adaptertransformationen	Literatur: Funktionshandbuch Werkzeugverwaltung

Die resultierende Werkzeugorientierung bleibt dabei immer parallel zu einer der drei Achsrichtungen X, Y oder Z und hängt ausschließlich von der aktiven Bearbeitungsebene G17–G19 ab, da dem Werkzeug bisher keine Orientierung zugeordnet werden konnte.

Stufenlose Veränderung der Werkzeugorientierung

Der orientierbare Werkzeugträger eröffnet zusätzlich zu weiteren Verschiebungen oder Längenänderungen mit Hilfe der Offsetvektoren I_1 - I_3 die Möglichkeit, die Werkzeugorientierung stufenlos zu verändern.

Für weitere Erläuterungen siehe Kapitel "Orientierbare Werkzeugträger (Seite 1557)".

Kleinere Bediener-Korrekturen

Kleinere Korrekturen müssen hingegen auch im normalen Produktionsbetrieb verändert werden können.

Gründe dafür sind z. B.:

- Werkzeugverschleiß
- Aufspannfehler
- Temperaturgang der Maschine

Diese Korrekturen werden dabei wie folgt definiert:

Definition	Verschleißkomponenten
Verschleiß	\$TC_DP12 - \$TC_DP14,
Summenkorrekturen fein	\$TC_SCPx3 - \$TC_SCPx5,
Summenkorrekturen grob bzw. Einrichtkorrekturen	\$TC_ECPx3 - \$TC_ECPx5

Insbesondere sind Korrekturen, welche die Werkzeuglängenberechnung betreffen, in den Koordinaten einzutragen, in denen sie auch gemessen wurden.

Diese werkstückspezifischen Korrekturen können einfacher vorgenommen werden, mit Hilfe der G-Gruppe 56, mit den drei Werten `TOWSTD`, `TOWMCS` und `TOWWCS`, und des Settingdatums:

SD42935 \$SC_WEAR_TRANSFORM (Transformation der Werkzeugkomponenten)

SD42935

Welche der Verschleißkomponenten:

- Verschleiß (\$TC_DP12 - \$TC_DP14):
- Einrichtkorrekturen bzw. Summenkorrekturen grob (\$TC_ECPx3 - \$TC_ECPx5)
- Summenkorrekturen fein (\$TC_SCPx3 - \$TC_SCPx5)

bei den Transformationen:

- Adaptertransformation
- Orientierbarer Werkzeugträger

transformiert bzw. nicht transformiert werden sollen, kann festgelegt werden über das Settingdatum:

SD42935 \$SC_WEAR_TRANSFORM (Transformation der Verschleißwerte)

In der Grundstellung des Settingdatums werden alle Verschleißwerte transformiert.

Das Settingdatum wird bei folgenden Funktionen berücksichtigt:

- Verschleißwerte im Maschinenkoordinatensystem
Teileprogrammanweisung: TOWMCS
- Verschleißwerte im Werkstückkoordinatensystem
Teileprogrammanweisung: TOWWCS

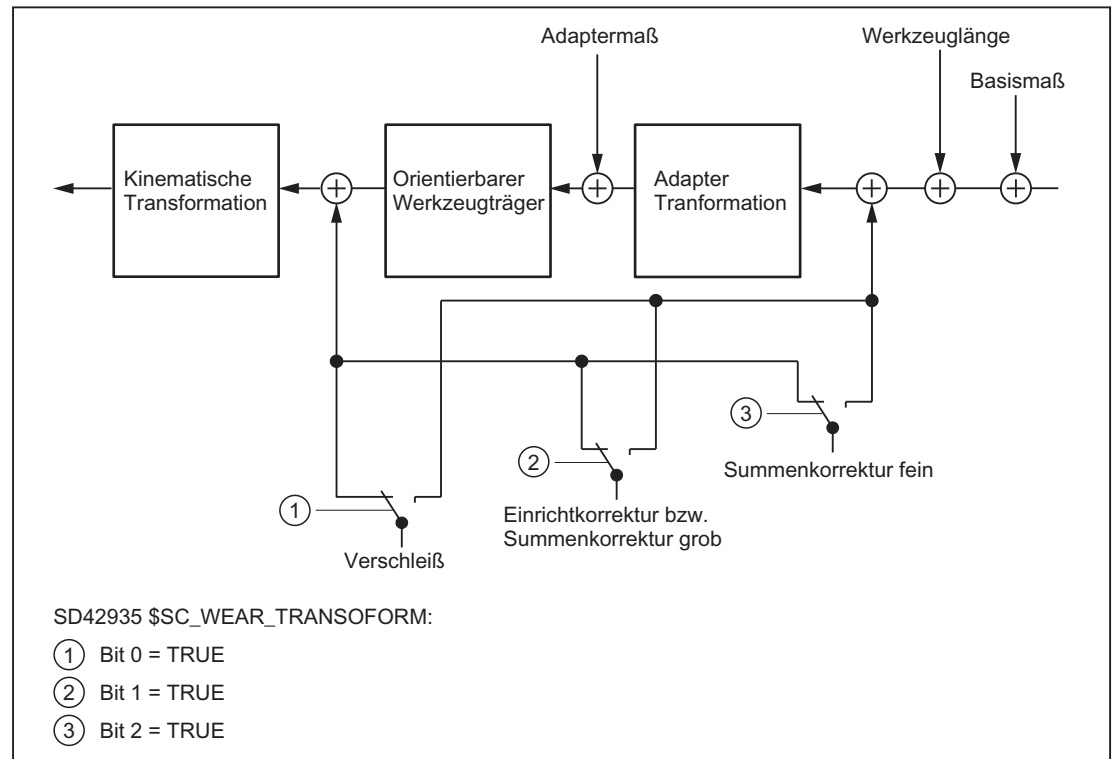


Bild 19-52 Transformation der Verschleißdaten abhängig von SD42935

Programmierung

Mit der G-Gruppe 56 können folgende Werte festgelegt werden:

Syntax	Korrekturen
TOWSTD	Grundstellungswert für Korrekturen in der Werkzeuglänge
TOWMCS	Verschleißwerte im Maschinenkoordinatensystem (MKS)
TOWWCS	Verschleißwerte im Werkstückkoordinatensystem (WKS)
TOWBCS	Verschleißwerte im Basiskoordinatensystem (BKS)
TOWTCS	Verschleißwerte im Werkzeugkoordinatensystem (TCS) an der Werkzeughalteraufnahme (T Werkzeugträgerbezugspunkt)
TOWKCS	Verschleißwerte im Werkzeugkoordinatensystem der kinematischen Transformation (KCS) des Werkzeugkopfes

Koordinatensysteme für Offsets der Werkzeuglänge

Mit den G-Befehlen `TOWMCS`, `TOWWCS`, `TOWBCS`, `TOWTCS` und `TOWKCS` kann z. B. die Werkzeuglängenkomponente Verschleiß in fünf verschiedenen Koordinatensystemen gemessen werden.

- | | |
|---|-----|
| 1. Maschinenkoordinatensystem | MKS |
| 1. Basiskoordinatensystem | BKS |
| 1. Werkstückkoordinatensystem | WKS |
| 1. Werkzeugkoordinatensystem der kinematischen Transformation | KCS |
| 1. Werkzeugkoordinatensystem | TCS |

Die berechnete Werkzeuglänge oder auch eine Werkzeuglängenkomponente kann mit der vordefinierten Funktion `GETTCOR` (Seite 1647) in einem dieser Koordinatensysteme dargestellt und ausgelesen werden.

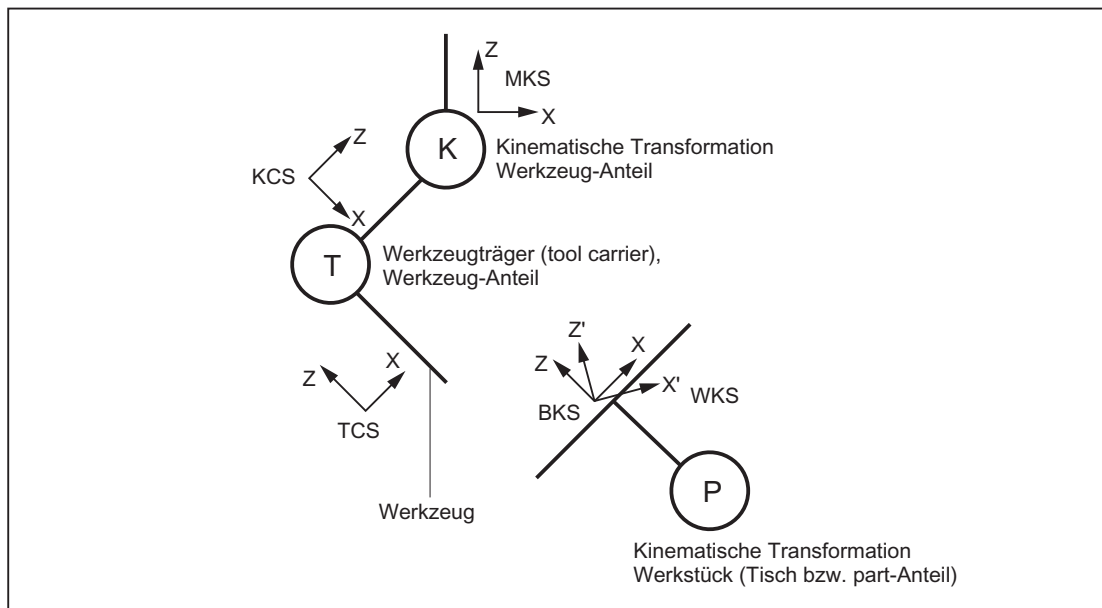


Bild 19-53 Koordinatensystem bei der Bewertung von Werkzeuglängen

19.12.5.2 Funktionalität der einzelnen Verschleißwerte

TOWSTD

Grundstellungswert (Standardbehandlung):

- Die Verschleißwerte werden zu den übrigen Werkzeuglängenkomponenten addiert. Die daraus resultierende Gesamtwerkzeuglänge geht dann gegebenenfalls in weitere Berechnungen ein.

Für den Fall eines aktiven orientierbaren Werkzeugträgers:

- Die Verschleißwerte werden der entsprechenden Drehung unterworfen.

TOWMCS

Verschleißwerte im Maschinenkoordinatensystem (MKS):

Für den Fall einer aktiven Drehung durch einen orientierbaren Werkzeugträger:

- Der Werkzeugträger dreht nur den Vektor der resultierenden Werkzeuglänge ohne Berücksichtigung des Verschleißes.
Anschließend werden der so gedrehte Werkzeuglängenvektor und der Verschleiß addiert.
Der Verschleiß wird der Drehung nicht unterworfen.

Wenn **kein** orientierbarer Werkzeugträger aktiv ist oder dieser keine Drehung bewirkt, sind TOWMCS und TOWSTD identisch.

Lineare Transformation

Die Werkzeuglänge im MKS ist nur dann eindeutig definiert, wenn das MKS aus dem BKS durch eine lineare Transformation hervorgeht.

Dies wäre ebenso der Fall:

- wenn keine kinematische Transformation aktiv ist
- oder eine Orientierungstransformationen (3-, 4- und 5-Achs-Transformationen) wirksam ist.

TOWWCS

Verschleißwerte im Werkstückkoordinatensystem (WKS):

- Bei einem aktiven orientierbaren Werkzeugträger wird der Werkzeugvektor ohne Berücksichtigung des Verschleißes wie bei TOWMCS berechnet.
- Die Angaben des Verschleißes werden im Werkstückkoordinatensystem interpretiert.

Der Verschleißvektor im Werkstückkoordinatensystem wird auf das Maschinenkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

TOWBCS

Verschleißwerte im Basiskoordinatensystem (BKS):

- Bei einem aktiven orientierbaren Werkzeugträger wird der Werkzeugvektor ohne Berücksichtigung des Verschleißes wie bei TOWMCS berechnet.
- Die Angaben des Verschleißes werden im Werkstückkoordinatensystem interpretiert.

Der Verschleißvektor im Basiskoordinatensystem wird auf das Werkstückkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

Nichtlineare Transformation

Ist eine nichtlineare Transformation beispielsweise mit TRANSMIT aktiv und wird das MKS als gewünschtem Koordinatensystem angegeben, dann wird an Stelle des MKS automatisch das BKS verwendet.

Orientierbarer Werkzeugträger

Ein eventuell vorhandener Tischanteil des orientierbaren Werkzeugträgers geht im Gegensatz zu einem Tisch- (bzw. part-) Anteil der kinematischen Transformation nicht unmittelbar in die Koordinatensysteme ein. Eine durch solch einen Anteil beschriebene Drehung wird in einem

Basis- oder Systemframe abgebildet und geht damit in den Übergang vom WKS in das BKS mit ein.

Kinematische Transformation

Der Tisch- (bzw. part-) der kinematischen Transformation wird durch den Übergang vom BKS in das MKS beschrieben.

TOWTCS

Verschleißwerte im Werkzeugkoordinatensystem (TCS):

- Bei einem aktiven orientierbaren Werkzeugträger wird der Werkzeugvektor ohne Berücksichtigung des Verschleißes wie bei TOWMCS berechnet.
- Die Angaben des Verschleißes werden im Werkzeugkoordinatensystem TCS interpretiert.

Der Verschleißvektor im Werkzeugkoordinatensystem (TCS) wird über das Werkzeugkoordinatensystem der kinematischen Transformation (KCS) auf das Maschinenkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

TOWKCS

Die Angaben der Verschleißwerte der kinematischen Transformation werden im dazugehörigen Werkzeugkoordinatensystem (KCS) interpretiert.

Der Verschleißvektor wird über das Werkzeugkoordinatensystem der kinematischen Transformation auf das Maschinenkoordinatensystem umgerechnet und zum Werkzeugvektor addiert.

G-Befehl Wechsel bei aktivem Werkzeug

Ein Wechsel des G-Befehls der Gruppe TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS und TOWKCS beeinflusst ein bereits aktives Werkzeug nicht und wird erst bei der nächsten Anwahl eines Werkzeuges wirksam.

Ein neuer G-Befehl dieser Gruppe wird auch dann wirksam, wenn er im gleichen Satz programmiert wird, in dem auch ein Werkzeug angewählt wird.

Bewertung einzelner Verschleißkomponenten

Die Bewertung der einzelnen Verschleißkomponenten (Zuordnung zu den Geometrieachsen, Vorzeichenbewertung) wird beeinflusst durch:

- Die aktive Ebene
- Die Adaptertransformation
- Die in der folgenden Tabelle dargestellten fünf Settingdaten

Settingdatum	Verschleißkomponenten		
SD42910 \$SC_MIRROW_TOOL_WEAR	TOWSTD	TOWMCS	TOWWCS
SD42920 \$SC_WEAR_SIGN_CUTPOS	X	X	—
SD42930 \$SC_WEAR_SIGN	X	—	—

Settingdatum	Verschleißkomponenten		
SD42940 \$SC_TOOL_LENGTH_CONST	X	X	X
SD42950 \$SC_TOOL_LENGTH_TYPE	X	X	X

Hinweis

Verschleißkomponenten, welche einer aktiven Drehung durch eine Adaptertransformation oder einen orientierbaren Werkzeugträger unterworfen werden, werden als nicht-transformierte Verschleißkomponenten bezeichnet.

Besonderheiten

Ist **TOWMCS** oder **TOWWCS** aktiv, wirkt folgendes Settingdatum nicht auf die nicht-transformierten Verschleißkomponenten:

SD42920 \$SC_WEAR_SIGN_CUTPOS (Vorzeichen des Verschleißes bei Werkzeugen mit Schneideanlagen)

Nicht auf die nicht-transformierten Verschleißkomponenten wirkt bei **TOWWCS** zusätzlich auch das Settingdatum:

SD42910 \$SC_MIRROR_TOOL_WEAR (Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln)

In diesem Fall ist eine eventuell aktive Spiegelung in dem Frame, der zur Bewertung der Verschleißkomponenten herangezogen wird, bereits enthalten.

Bei einem Ebenenwechsel bleibt die Zuordnung der nicht-transformierten Verschleißkomponenten zu den Geometrieachsen erhalten, d.h. diese werden nicht wie bei übrigen Längskomponenten getauscht. Die Zuordnung der Komponenten hängt von der aktiven Ebene bei der Werkzeuganwahl ab.

Beispiel

Es sei ein Fräswerkzeug gegeben, bei dem nur der der Länge L1 zugeordnete Verschleißwert \$TC_DP12 ungleich Null ist.

Ist **G17** aktiv, wirkt diese Länge in Richtung der Z-Achse.

Dieses Maß wirkt auch bei einem Ebenenwechsel nach der Werkzeuganwahl stets in Z-Richtung, wenn **TOWMCS** oder **TOWWCS** aktiv und das Bit 1 gesetzt ist im Settingdatum:

SD42935 \$SC_WEAR_TRANSFORM (Transformationen für Werkzeugkomponenten)

Ist bei der Werkzeuganwahl z.B. **G18** wirksam, wirkt die Komponente stattdessen unter sonst gleichen Bedingungen stets in Y-Richtung.

19.13 Mit Werkzeugumgebungen arbeiten

Übersicht Funktionen

- Werkzeugumgebung speichern (TOOLENV) (Seite 1642)
- Werkzeugumgebung löschen (DELTOOLENV) (Seite 1645)
- T-, D- und DL-Nummer lesen (GETTENV) (Seite 1646)
- Werkzeuglängen bzw. -längenkomponenten lesen (GETTCOR) (Seite 1647)
- Werkzeugkomponenten ändern (SETTCOR) (Seite 1654)

Übersicht Systemvariablen

- Informationen zu gespeicherten Werkzeugumgebungen lesen (\$P_TOOLENVN, \$P_TOOLENV) (Seite 1647)

19.13.1 Werkzeugumgebung speichern (TOOLENV)

Die Funktion TOOLENV dient dazu, alle aktuellen Zustände zu speichern, die für die Bewertung der im Speicher abgelegten Werkzeugdaten von Bedeutung sind.

Das sind im Einzelnen die folgenden Daten:

- Der aktive G-Befehl der Gruppe:
 - 6 (G17, G18, G19)
 - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- Die aktive Planachse
- Die Maschinendaten:
 - MD18112 \$MN_MM_KIND_OF_SUMCORR (Eigenschaften der Summenkorrekturen im TO-Bereich)
 - MD20360 \$MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

- Die Settingdaten:
 - SD42900 \$SC_MIRROR_TOOL_LENGTH (Vorzeichenwechsel Werkzeuglänge beim Spiegeln)
 - SD42910 \$SC_MIRROR_TOOL_WEAR (Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln)
 - SD42920 \$SC_WEAR_SIGN_CUTPOS (Vorzeichen des Verschleißes bei Werkzeugen mit Schneidanlagen)
 - SD42930 \$SC_WEAR_SIGN (Vorzeichen des Verschleißes)
 - SD42935 \$SC_WEAR_TRANSFORM (Transformationen für Werkzeugkomponenten)
 - SD42940 \$SC_LENGTH_CONST (Wechsel der Werkzeuglängenkomponten bei Ebenenwechsel)
 - SD42942 \$SC_TOOL_LENGTH_CONST_T (Wechsel der Werkzeuglängenkomponten für Drehwerkzeuge bei Ebenenwechsel)
 - SD42950 \$SC_TOOL_LENGTH_TYPE (Zuordnung der Werkzeuglängenkomponten unabhängig vom Werkzeugtyp)
 - SD42954 \$SC_TOOL_ORI_CONST_M (Wechsel der Werkzeugorientierungskomponten für Fräswerkzeuge bei Ebenenwechsel)
 - SD42956 \$SC_TOOL_ORI_CONST_T (Wechsel der Werkzeugorientierungskomponten für Drehwerkzeuge bei Ebenenwechsel)
- Der Orientierungsanteil des aktuellen Gesamtframes (Drehungen und Spiegelungen, keine Nullpunktverschiebungen oder Skalierungen)
- Der Orientierungsanteil und die resultierende Länge des aktiven orientierbaren Werkzeugträgers
- Der Orientierungsanteil und die resultierende Länge einer aktiven Transformation

Außer den genannten Daten, die die Umgebung des Werkzeugs beschreiben, werden auch T-Nummer, D-Nummer und DL-Nummer des aktiven Werkzeugs mit abgelegt, damit später auf dieses Werkzeug in der gleichen Umgebung wie beim Aufruf von TOOLENV zugegriffen werden kann, ohne das Werkzeug erneut bezeichnen zu müssen.

Syntax

<Status> = TOOLENV(<Name>)

Bedeutung

TOOLENV (. . .) :	Vordefinierte Funktion zum Speichern einer Werkzeugumgebung	
	Alleine im Satz:	ja

<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Keine freien Speicherplätze für Werkzeugumgebungen mehr vorhanden.
		-3	Nullstring als Name einer Werkzeugumgebung ist nicht zulässig.
-4		Kein Parameter (<Name>) angegeben.	
Parameter			
1	<Name>:	Name, unter dem der aktuelle Datensatz abgelegt werden soll Ist ein Datensatz mit dem gleichen Namen bereits vorhanden, wird dieser überschrieben. Der Status ist in diesem Fall "0".	
	Datentyp:	STRING	

Weitere Informationen

Basismaß/Adaptermaß - Werkzeuglängenkorrektur

Ob die Adapterlänge oder das Basismaß (schneidenspezifische Parameter \$TC_DP21, \$TC_DP22 und \$TC_DP23) in die Berechnung der Werkzeuglänge eingeht, entscheidet bei aktiver Werkzeug-Magazinverwaltung (nur verfügbar mit der Option "Werkzeugverwaltung") der Wert des Maschinendatums:

MD18104 \$MN_MM_NUM_TOOL_ADAPTER (WZ-Adapter im TO-Bereich)

Da eine Veränderung dieses Maschinendatums nur beim Hochlauf der Steuerung wirksam werden kann, wird es nicht in der Werkzeugumgebung gespeichert.

Resultierende Länge von orientierbaren Werkzeugträgern und Transformationen

Hinweis

Sowohl bei orientierbaren Werkzeugträgern als auch bei Transformationen gibt es Systemvariablen bzw. Maschinendaten, die wie zusätzliche Werkzeuglängenkomponenten wirken, und die ganz oder teilweise den von diesen ausführbaren Drehungen unterworfen werden. Die zusätzlichen Werkzeuglängenanteile, die sich dadurch ergeben, müssen beim Aufruf von TOOLENV mit abgelegt werden, da sie einen Teil der Umgebung bilden, in der das Werkzeug eingesetzt wird.

Adaptertransformation

Die Adaptertransformation ist eine Eigenschaft des Werkzeugadapters und damit des Gesamtwerkzeugs. Sie ist deshalb nicht Bestandteil einer Werkzeugumgebung, die auf ein anderes Werkzeug angewendet werden kann.

Durch das Abspeichern der gesamten für die Bestimmung der Werkzeuggesamtlänge erforderlichen Daten ist es möglich, zu einem späteren Zeitpunkt die effektive Länge eines

Werkzeugs zu berechnen, auch wenn es zu diesem Zeitpunkt nicht mehr aktiv ist bzw. wenn sich die Umgebungsbedingungen (z. B. G-Befehle oder Settingdaten) geändert haben. Ebenso kann die effektive Länge eines anderen Werkzeugs berechnet werden mit der Annahme, es würde unter den gleichen Bedingungen eingesetzt wie das Werkzeug, für das der Status abgespeichert wurde.

Maximale Anzahl Datensätze für Werkzeugumgebungen

Mit dem Maschinendatum MD18116 \$MN_MM_NUM_TOOL_ENV wird festgelegt, wie viele Datensätze zur Beschreibung von Werkzeugumgebungen maximal abgelegt werden können. Die Daten liegen im TOA-Bereich. Sie bleiben auch beim Ausschalten der Steuerung erhalten.

Eine Datensicherung ist nicht möglich. Das bedeutet, dass diese Daten nicht zwischen verschiedenen Steuerungen übertragen werden können.

19.13.2 Werkzeugumgebung löschen (DELTOOLENV)

Mit der Funktion DELTOOLENV können Datensätze zur Beschreibung von Werkzeugumgebungen gelöscht werden. Löschen bedeutet, dass auf den unter einem bestimmten Namen abgelegten Datensatz nicht mehr zugegriffen werden kann (ein Zugriffsversuch führt zu einem Alarm).

Hinweis

Datensätze können nur mit der Funktion DELTOOLENV, durch einen INITIAL.INI-Download oder durch einen Kaltstart (NC-Hochlauf mit Standard-Maschinendaten) gelöscht werden. Es gibt keine weiteren automatischen Löschvorgänge.

Syntax

```
<Status> = DELTOOLENV (<Name>)
<Status> = DELTOOLENV ()
```

Bedeutung

DELTOOLENV(...):	Vordefinierte Funktion zum Löschen einer Werkzeugumgebung		
	Alleine im Satz:		ja
<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Ein Werkzeugumgebung mit dem angegebenen Namen existiert nicht.
Parameter			

1	<Name>:	Name des zu löschenden Datensatzes	
		Datentyp:	STRING
DELTOOLENV () :		DELTOOLENV () ohne Angabe eines Namens löscht alle Datensätze zur Beschreibung von Werkzeugumgebungen	

19.13.3 T-, D- und DL-Nummer lesen (GETTENV)

Die Funktion GETTENV dient dazu, die in einer Werkzeugumgebung abgelegte T-, D- und DL-Nummer zu lesen.

Syntax

<Status> = GETTENV (<Name>, <TDDL>)

Bedeutung

GETTENV (. . .) :		Vordefinierte Funktion zum Lesen von T-, D- und DL-Nummern in einem Datensatz zur Beschreibung einer Werkzeugumgebung		
		Alleine im Satz:	ja	
<Status>:		Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
		Datentyp:	INT	
		Wert:	0	Funktion OK
			-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
-2	Ein Werkzeugumgebung mit dem angegebenen Namen existiert nicht.			
Parameter				
1	<Name>:	Name der Datensatzes, aus dem die T-, D- und DL-Nummer gelesen werden sollen		
		Datentyp:	STRING	
2	<TDDL>:	Das Feld dieses Ergebnisparameters enthält die T-, D- und DL-Nummer des Werkzeugs, dessen Werkzeugumgebung im angegebenen Datensatz abgelegt ist: <ul style="list-style-type: none">• <TDDL> [0]: T-Nummer• <TDDL> [1]: D-Nummer• <TDDL> [2]: DL-Nummer		
		Datentyp:	INT[3]	
GETTENV (, <TDDL>) , GETTENV (" " , <TDDL>) :		Es ist zulässig, beim Aufruf der Funktion GETTENV den ersten Parameter wegzulassen oder als ersten Parameter den Nullstring zu übergeben. In diesen beiden Sonderfällen werden in <TDDL> die T-, D- und DL-Nummern des aktiven Werkzeugs zurückgeliefert.		

19.13.4 Informationen zu gespeicherten Werkzeugumgebungen lesen (\$P_TOOLENVN, \$P_TOOLENV)

Informationen zu gespeicherten Werkzeugumgebungen sind über folgende Systemvariablen lesbar:

\$P_TOOLENVN:	Liefert die Anzahl der mittels TOOLENV definierten (und noch nicht gelöschten) Datensätze zur Beschreibung von Werkzeugumgebungen			
	Syntax:	<n> = \$P_TOOLENVN		
	Bedeutung:	<n>:	Anzahl der definierten Datensätze	
			Datentyp:	INT
			Wertebereich:	0 ... MD18116 \$MN_MM_NUM_TOOL_ENV
Der Zugriff auf diese Systemvariable ist auch dann zulässig, wenn keine Werkzeugumgebungen möglich sind (MD18116 = 0). In diesem Fall ist der Rückgabewert "0".				

\$P_TOOLENV:	Liefert den Namen des <i>-ten Datensatzes zur Beschreibung einer Werkzeugumgebung			
	Syntax:	<Name> = \$P_TOOLENV[<i>]		
	Bedeutung:	<Name>:	Name des Datensatzes mit der Nummer <i>	
			Datentyp:	STRING
		<i>:	Nummer des Datensatzes	
			Datentyp:	INT
			Wertebereich:	1 ... \$P_TOOLENVN
			Die Zuordnung der Nummern zu den Datensätzen ist nicht fix, sondern kann sich durch das Löschen oder Neuanlegen von Datensätzen verändern. Die Nummerierung der Datensätze wird intern vorgenommen. Verweist <i> auf einen nicht definierten Datensatz, wird der Nullstring zurückgeliefert. Ist der Index <i> ungültig, d. h. <i> ist kleiner 1 oder größer als die maximale Anzahl Datensätze für Werkzeugumgebungen (MD18116 \$MN_MM_NUM_TOOLENV), wird folgender Alarm ausgegeben: Alarm 17020 "unerlaubter Array-Index1"	

19.13.5 Werkzeuglängen bzw. -längenkompontenten lesen (GETTCOR)

Die Funktion GETTCOR dient dazu, Werkzeuglängen bzw. Werkzeuglängenkompontenten auszulesen.

Dabei kann durch Parametrierung angegeben werden, welche Komponenten berücksichtigt werden sollen, und unter welchen Einsatzbedingungen das Werkzeug betrachtet werden soll.

Syntax

```
<Status> = GETTCOR(<Len>[, <Comp>, <Stat>, <T>, <D>, <DL>])
```

Bedeutung

GETTCOR(...):		Vordefinierte Funktion zum Lesen von Werkzeuglängen bzw. Werkzeuglängenkomponenten	
		Alleine im Satz:	ja
<Status>:		Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.	
		Datentyp:	INT
		Wert:	0 Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Eine Werkzeugumgebung mit dem unter <Stat> angegebenen Namen existiert nicht.
		-3	Ungültiger String im Parameter <Comp>. Ursache dieses Fehlers können ungültige oder doppelt programmierte Zeichen sein.
		-4	Ungültige T-Nummer
		-5	Ungültige D-Nummer
		-6	Ungültige DL-Nummer
		-7	Zugriffsversuch auf nicht vorhandenes Speichermodul.
		-8	Zugriffsversuch auf eine nicht vorhandene Option (z. B. programmierbare Werkzeugorientierung, Werkzeugverwaltung).
		-9	Der String <Comp> enthält einen Doppelpunkt (Kennung für die Spezifikationen eines Koordinatensystems), es folgt jedoch kein gültiges Zeichen zur Bezeichnung des verlangten Koordinatensystems.
Parameter			

1	<Len>:	Ergebnisvektor	
		Datentyp:	REAL[11]
<p>Die Vektorkomponenten sind in folgender Reihenfolge angeordnet:</p> <ul style="list-style-type: none">• <Len> [0]: Werkzeugtyp• <Len> [1]: Schneidenlage• <Len> [2]: Abszisse• <Len> [3]: Ordinate• <Len> [4]: Applikate• <Len> [5]: Werkzeugradius <p>Als Bezugskoordinatensystem für die Längskomponenten gilt das in <Comp> und <Stat> definierte Koordinatensystem. Ist in <Comp> kein Koordinatensystem definiert, werden die Werkzeuglängen im Maschinenkoordinatensystem dargestellt.</p> <p>Die Zuordnung von Abszisse, Ordinate und Applikate zu den Geometrieachsen ist abhängig von der in der verwendeten Werkzeugumgebung aktiven Ebene. D. h., bei G17 ist die Abszisse parallel zu X, bei G18 zu Z usw.</p> <p>Die Komponenten <Len>[6] bis <Len>[10] enthalten die zusätzlichen Parameter, die für die Geometriebeschreibung eines Werkzeugs angegeben werden können (z. B. \$TC_DP7 bis \$TC_DP11 für die Geometrie bzw. die entsprechenden Komponenten für Verschleiß bzw. Summen- und Einrichtekorrekturen).</p> <p>Diese 5 zusätzlichen Elemente und der Werkzeugradius sind nur für die Komponenten E, G, S und W definiert. Deren Bewertung hängt nicht von <Stat> ab. Die entsprechenden Werte in <Len>[6] bis <Len>[10] können deshalb nur dann ungleich null sein, wenn an der Werkzeuglängenberechnung mindestens eine der vier genannten Komponenten beteiligt ist. Die übrigen Komponenten haben keinen Einfluss auf das Ergebnis. Die Maßangaben beziehen sich auf das Grundsystem der Steuerung (inch bzw. metrisch).</p>			

2	<Comp>:	Werkzeuglängenkomponenten (optional)																		
		Datentyp:	STRING																	
		Die Zeichenkette besteht aus zwei Teilstrings, die durch einen Doppelpunkt voneinander getrennt sind.																		
		Allg. Form: "<SubStr_1> [: <SubStr_2>]"																		
		<SubStr_1>:	Der erste Teilstring bezeichnet die Werkzeuglängenkomponenten, die bei der Werkzeuglängenberechnung berücksichtigt werden sollen.																	
		Die Reihenfolge der Zeichen im Teilstring sowie deren Schreibweise (groß oder klein) ist beliebig. Zwischen den Zeichen können beliebig viele Leerzeichen oder Tabulatorzeichen (white spaces) eingefügt werden.																		
		Hinweis: Die Zeichen im Teilstring dürfen nicht doppelt programmiert werden!																		
		Zeichen:	<table><tr><td>-</td><td>Minuszeichen (nur als erstes Zeichen zulässig!) Es wird die gesamte Werkzeuglänge abzüglich der Komponenten berechnet, die im nachfolgenden String spezifiziert sind.</td></tr><tr><td>C</td><td>Adaptermaß oder Basismaß (diejenige der beiden alternativ existierenden Komponenten, die für das verwendete Werkzeug aktiv ist)</td></tr><tr><td>E</td><td>Einrichtkorrekturen</td></tr><tr><td>G</td><td>Geometrie</td></tr><tr><td>K</td><td>Kinematische Transformation (wird nur bei generischer 3-, 4- und 5-Achs-Transformation ausgewertet!)</td></tr><tr><td>S</td><td>Summenkorrekturen</td></tr><tr><td>T</td><td>Orientierbarer Werkzeugträger</td></tr><tr><td>W</td><td>Verschleiß</td></tr></table>		-	Minuszeichen (nur als erstes Zeichen zulässig!) Es wird die gesamte Werkzeuglänge abzüglich der Komponenten berechnet, die im nachfolgenden String spezifiziert sind.	C	Adaptermaß oder Basismaß (diejenige der beiden alternativ existierenden Komponenten, die für das verwendete Werkzeug aktiv ist)	E	Einrichtkorrekturen	G	Geometrie	K	Kinematische Transformation (wird nur bei generischer 3-, 4- und 5-Achs-Transformation ausgewertet!)	S	Summenkorrekturen	T	Orientierbarer Werkzeugträger	W	Verschleiß
		-	Minuszeichen (nur als erstes Zeichen zulässig!) Es wird die gesamte Werkzeuglänge abzüglich der Komponenten berechnet, die im nachfolgenden String spezifiziert sind.																	
		C	Adaptermaß oder Basismaß (diejenige der beiden alternativ existierenden Komponenten, die für das verwendete Werkzeug aktiv ist)																	
E	Einrichtkorrekturen																			
G	Geometrie																			
K	Kinematische Transformation (wird nur bei generischer 3-, 4- und 5-Achs-Transformation ausgewertet!)																			
S	Summenkorrekturen																			
T	Orientierbarer Werkzeugträger																			
W	Verschleiß																			
Ist der erste Teilstring (mit Ausnahme von white spaces) leer, so bedeutet das, dass die gesamte Werkzeuglänge unter Berücksichtigung aller Komponenten berechnet werden soll. Gleiches gilt auch dann, wenn der Parameter <Comp> nicht angegeben wird.																				
<Substr_2>:	Der optionale zweite Teilstring bezeichnet das Koordinatensystem, in dem die Werkzeuglänge ausgegeben werden soll.																			
Der zweite Teilstring besteht nur aus einem einzigen relevanten Zeichen.																				
Zeichen:	<table><tr><td>A</td><td>Einstellbares Koordinatensystem (ACS)</td></tr><tr><td>B</td><td>Basiskoordinatensystem (BKS)</td></tr><tr><td>K</td><td>Werkzeugkoordinatensystem der kinematischen Transformation (KCS)</td></tr><tr><td>M</td><td>Maschinenkoordinatensystem (MKS)</td></tr><tr><td>T</td><td>Werkzeugkoordinatensystem (TCS)</td></tr><tr><td>W</td><td>Werkstückkoordinatensystem (WKS)</td></tr></table>		A	Einstellbares Koordinatensystem (ACS)	B	Basiskoordinatensystem (BKS)	K	Werkzeugkoordinatensystem der kinematischen Transformation (KCS)	M	Maschinenkoordinatensystem (MKS)	T	Werkzeugkoordinatensystem (TCS)	W	Werkstückkoordinatensystem (WKS)						
A	Einstellbares Koordinatensystem (ACS)																			
B	Basiskoordinatensystem (BKS)																			
K	Werkzeugkoordinatensystem der kinematischen Transformation (KCS)																			
M	Maschinenkoordinatensystem (MKS)																			
T	Werkzeugkoordinatensystem (TCS)																			
W	Werkstückkoordinatensystem (WKS)																			
Wird kein Koordinatensystem angegeben, erfolgt die Bewertung im Maschinenkoordinatensystem (MKS). Die gegebenen-																				

			falls zu berücksichtigenden Drehungen werden durch die in <Stat> definierte Werkzeugumgebung festgelegt.
3	<_Stat>:	Name des Datensatzes zur Beschreibung einer Werkzeugumgebung (optional)	
		Datentyp:	STRING
		Ist der Wert dieses Parameters der Nullstring ("") oder wird er nicht angegeben, wird der aktuelle Zustand verwendet. Ist kein Werkzeug spezifiziert, wird das aktuelle Werkzeug verwendet.	
4	<T>:	Interne T-Nummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", wird das in <Stat> abgelegte Werkzeug verwendet.	
		Ist der Wert dieses Parameters "-1", wird die T-Nummer des aktiven Werkzeugs verwendet. Es ist auch zulässig, die Nummer des aktiven Werkzeugs explizit anzugeben.	
		Hinweis: Ist <Stat> nicht angegeben, wird als Werkzeugumgebung der aktuelle Zustand verwendet. Da mit <T> = 0 auf die in der Werkzeugumgebung gespeicherte T-Nummer verwiesen wird, wird in dieser das aktive Werkzeug verwendet, d. h. die Angaben <T> = 0 und <T> = -1 sind in diesem Sonderfall gleichbedeutend.	
5	<D>:	Schneidenummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", richtet sich die verwendete D-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs.	
6	<DL>:	Nummer der ortsabhängigen Korrektur (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben, richtet sich die verwendete DL-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs.	

Beispiele

GETTCOR (_LEN)	Berechnet die Werkzeuglänge des aktuellen aktiven Werkzeugs im Maschinenkoordinatensystem unter Berücksichtigung aller Komponenten.
GETTCOR (_LEN, "CGW:W")	Es wird die Werkzeuglänge bestehend aus Adapter- bzw. Basismaß, Geometrie und Verschleiß für das aktive Werkzeug berechnet. Weitere Anteile wie z. B. orientierbarer Werkzeugträger oder kinematische Transformationen werden nicht berücksichtigt. Die Ausgabe erfolgt im Werkstückkoordinatensystem.

GETTCOR (_LEN, "-K:B")	Es wird die gesamte Werkzeuglänge des aktiven Werkzeugs ohne Berücksichtigung der Längskomponenten einer eventuell aktiven kinematischen Transformation berechnet. Ausgabe im Basiskoordinatensystem.
GETTCOR (_LEN, ":M", "Testenv1", , 3)	Es wird die gesamte Werkzeuglänge für das in der Werkzeugumgebung mit dem Namen "Testenv1" abgelegte Werkzeug im Maschinenkoordinatensystem berechnet. Die Berechnung erfolgt jedoch unabhängig von der abgelegten Schneidenummer für die Schneidenummer D3.

Weitere Informationen

Adaptertransformation / Orientierbare Werkzeugträger / Kinematische Transformation

Die Drehungen und Komponentenvertauschungen, die gegebenenfalls von der Adaptertransformation, dem orientierbaren Werkzeugträger und einer kinematischen Transformation ausgeführt werden, sind Bestandteil der Werkzeugumgebung. Sie werden deshalb immer ausgeführt, auch dann, wenn die entsprechende Längskomponente nicht berücksichtigt werden soll. Ist das nicht gewünscht, müssen Werkzeugumgebungen definiert werden, bei denen die entsprechenden Transformationen nicht aktiv sind. In vielen Fällen (nämlich immer dann, wenn an einer Maschine keine Transformationen bzw. orientierbare Werkzeugträger eingesetzt werden), erfüllen die abgespeicherten Datensätze für Werkzeugumgebungen diese Bedingungen automatisch, sodass sie vom Anwender nicht speziell beachtet werden müssen.

Dreh- und Schleifwerkzeuge: Berechnung der Werkzeuglänge in Abhängigkeit von MD20360 \$MC_TOOL_PARAMETER_DEF_MASK

Wie bei Dreh- und Schleifwerkzeugen der Verschleiß bzw. die Werkzeuglänge in einer eventuell vorhandenen Durchmesserachse zu bewerten ist, wird festgelegt im Maschinendatum:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

Bit	Wert
0	Bei Dreh- und Schleifwerkzeugen wird der Verschleißparameter der Planachse als Durchmesserwert eingerechnet:
	= 0 (Default) nein
	= 1 ja
1	Bei Dreh- und Schleifwerkzeugen wird die Werkzeuglängskomponente der Planachse als Durchmesserwert eingerechnet:
	= 0 (Default) nein
	= 1 ja

Sind die betreffenden Bits gesetzt, wird der zugehörige Eintrag mit dem Faktor 0,5 bewertet. Diese Bewertung schlägt sich auch in der von GETTCOR gelieferten Werkzeuglänge nieder.

Beispiel:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK = 3

MD20100 \$MC_DIAMETER_AX_DEF (Geometrieachse mit Planachsfunktion) = "X"

X ist Durchmesserachse (Standard-Drehmaschinenkonfiguration)

Programmcode	Kommentar
N30 \$TC_DP1[1,1]=500	
N40 \$TC_DP2[1,1]=2	
N50 \$TC_DP3[1,1]=3.0	; Geometrie L1
N60 \$TC_DP4[1,1]=4.0	
N70 \$TC_DP5[1,1]=5.0	
N80 \$TC_DP12[1,1]=12.0	; Verschleiß L1
N90 \$TC_DP13[1,1]=13.0	
N100 \$TC_DP14[1,1]=14.0	
N110 T1 D1 G18	
N120 R1=GETTCOR(_LEN, "GW")	
N130 R3=_LEN[2]	; 17.0 (= 4.0 + 13.0)
N140 R4=_LEN[3]	; 7.5 (= 0.5 * 3.0 + 0.5 * 12.0)
N150 R5=_LEN[4]	; 19.0 (= 5.0 + 14.0)
N160 M30	

Längskomponenten von kinematischer Transformation und orientierbarem Werkzeugträger

Wird bei der Werkzeuglängenberechnung ein **orientierbarer Werkzeugträger** berücksichtigt, gehen folgende Vektoren in die Werkzeuglängenberechnung ein:

Typ	Vektoren
M	I1 und I2
T	I1, I2 und I3
P	Werkzeuglänge wird durch den orientierbaren Werkzeugträger nicht beeinflusst.

Bei der generischen **5-Achstransformation** gehen bei den Trafotypen 24 und 56 folgende Maschinendaten in die Werkzeuglängenberechnung ein:

Trafotyp	Maschinendaten
24	MD24550/24650 \$MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 \$MC_TRAFO5_JOINT_OFFSET_1/2 MD24558/24658 \$MC_TRAFO5_PART_OFFSET_1/2
56	MD24550/24650 \$MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 \$MC_TRAFO5_JOINT_OFFSET_1/2

Der Trafotyp 56 (bewegliches Werkzeug und bewegliches Werkstück) entspricht dem Typ M beim orientierbaren Werkzeugträger.

Bei dieser 5-Achs-Transformation entspricht in den bisherigen Softwareständen der Vektor MD24560/24660 \$MC_TRAFO5_JOINT_OFFSET_1/2 (Vektor des kinematischen Versatzes der 1./2. 5-Achstransformation im Kanal) der Summe der beiden Vektoren I₁ und I₃ beim orientierbaren Werkzeugträger vom Typ M.

Für die Transformation ist in beiden Fällen nur die Summe relevant. Die Art der Zusammensetzung aus den beiden Einzelkomponenten ist bedeutungslos. Bei der Berechnung der Werkzeuglänge ist es jedoch von Bedeutung, welcher Anteil dem Werkzeug und welcher dem Werkzeugschulter zuzuordnen ist. Deshalb wurde das Maschinendatum MD24558/24658 \$MC_TRAFO5_JOINT_OFFSET_PART_1/2 (Vektor kinematischer Versatz

im Tisch) eingeführt. Es entspricht dem Vektor I3. Das Maschinendatum MD24560/24660 \$MC_TRAFO5_JOINT_OFFSET_1/2 entspricht jetzt nicht mehr der Summe aus I1 und I3, sondern nur noch dem Vektor I1. Ist das Maschinendatum MD24558/24658 \$MC_TRAFO5_JOINT_OFFSET_PART_1/2 gleich null, so ist das Verhalten identisch zum bisherigen Verhalten.

Kompatibilität

Die Funktion GETTCOR dient zusammen mit den Funktionen TOOLENV und SETTCOR unter anderem dazu, Teile der Funktionalität zu ersetzen, die bisher extern in den Messzyklen realisiert war.

In den Messzyklen wurde nur ein Teil der Parameter ausgewertet, die letztlich die effektive Werkzeuglänge bestimmen. Die genannten Funktionen können so parametrisiert werden, dass das Verhalten der Messzyklen bezüglich der Werkzeuglängenberechnung reproduziert werden kann.

19.13.6 Werkzeugkomponenten ändern (SETTCOR)

Die Funktion SETTCOR dient dazu, Werkzeugkomponenten unter Berücksichtigung aller Randbedingungen, die in die Bewertung der einzelnen Komponenten eingehen können, zu verändern.

Hinweis

Zur Terminologie: Wenn im Folgenden im Zusammenhang mit der Werkzeuglänge von Werkzeugkomponenten die Rede ist, sind die vektoriell betrachteten Komponenten gemeint, aus denen sich die gesamte Werkzeuglänge zusammensetzt (z. B. Geometrie oder Verschleiß). Eine solche Komponente setzt sich deshalb aus drei Einzelwerten (L1, L2, L3) zusammen, die im Folgenden als Koordinatenwerte bezeichnet werden.

Die Werkzeugkomponente "Geometrie" besteht somit beispielsweise aus den drei Koordinatenwerten \$TC_DP3 bis \$TC_DP5.

Syntax

```
<Status> = SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorMode>, <GeoAx>, <Stat>, <T>, <D>, <DL>])
```

Bedeutung

SETTCOR(...):	Vordefinierte Funktion zum Ändern von Werkzeugkomponenten	
	Alleine im Satz:	ja

<Status>:		Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.	
		Datentyp:	INT
		Wert:	0 Funktion OK
		-1	Kein Speicherplatz für Werkzeugumgebungen reserviert: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 D. h., die Funktionalität "Werkzeugumgebungen" ist nicht vorhanden.
		-2	Eine Werkzeugumgebung mit dem unter <Stat> angegebenen Namen existiert nicht.
		-3	Ungültiger String im Parameter <Comp>. Ursache dieses Fehlers können ungültige oder doppelt programmierte Zeichen sein.
		-4	Ungültige T-Nummer.
		-5	Ungültige D-Nummer.
		-6	Ungültige DL-Nummer.
		-7	Zugriffsversuch auf nicht vorhandenes Speichermodul.
		-8	Zugriffsversuch auf eine nicht vorhandene Option (z. B. programmierbare Werkzeugorientierung, Werkzeugverwaltung).
		-9	Unzulässiger Zahlenwert für den Parameter <CorComp>.
		-10	Unzulässiger Zahlenwert für den Parameter <CorMode>.
		-11	Die Inhalte der Parameter <Comp> und <CorComp> sind widersprüchlich.
		-12	Die Inhalte der Parameter <Comp> und <CorMode> sind widersprüchlich.
		-13	Der Inhalt des Parameters <GeoAx> bezeichnet keine Geometrieachse.
		-14	Schreibversuch auf nicht vorhandene Einrichtekorrekturen.
Parameter			
1	<CorVal>:	<p>Korrekturvektor</p> <p>In dem durch <Stat> definierten Werkstückkoordinatensystem (WKS) gilt folgende Zuordnung:</p> <ul style="list-style-type: none"> • <CorVal> [0]: Abszisse • <CorVal> [1]: Ordinate • <CorVal> [2]: Applikate <p>Soll nur eine Werkzeugkomponente korrigiert werden (d. h. keine vektorielle Korrektur, siehe Parameter <CorMode>), steht der Korrekturwert immer in <CorVal>[0], gleichgültig auf welche Achse er wirkt. Der Inhalt der beiden übrigen Komponenten wird dann nicht ausgewertet.</p> <p>Bezieht sich <CorVal> bzw. eine Komponente von <CorVal> auf die Planachse, so wird die Angabe als Radiusmaß bewertet. Das bedeutet, dass ein Werkzeug z. B. um das angegebene Maß "länger" wird, was entsprechend zu einer doppelt so großen Änderung des Werkstückdurchmessers führt.</p> <p>Die Maßangaben beziehen sich auf das Grundsystem (inch bzw. metrisch) der Steuerung.</p>	
		Datentyp:	REAL[3]

2	<Comp>:	Werkzeugkomponente(n)			
		Datentyp:		STRING	
		Die Zeichenkette besteht aus zwei Teilstrings, die durch einen Doppelpunkt voneinander getrennt sind.			
		Allg. Form: "<SubStr_1> [: <SubStr_2>]"			
		<SubStr_1>:	Der erste Teilstring muss immer vorhanden sein und kann entweder aus ein oder zwei Zeichen bestehen. Dabei steht das erste oder einzige Zeichen für die 1. Komponente (Val ₁) und das zweite Zeichen für die 2. Komponente (Val ₂), die entsprechend den nachfolgenden Parametern <CorComp> und <CorMode> bearbeitet werden.		
			Zeichen:	C	Adaptermaß oder Basismaß (diejenige der beiden alternativ existierenden Komponenten, die für das verwendete Werkzeug aktiv ist)
				E	Einrichtkorrekturen
				G	Geometrie
				S	Summenkorrekturen
				W	Verschleiß
		<SubStr_2>:	Der zweite Teilstring ist optional. Er kann alternativ aus dem (einzigen) Buchstaben "W" oder "T" bestehen.		
			Zeichen:	W	Ist der zweite Teilstring leer oder enthält er den Buchstaben "W", werden die Korrekturwerte so verrechnet, als seien sie im Werkstückkoordinatensystem (WKS) vermessen worden.
				T	Enthält der zweite Teilstring den Buchstaben "T", werden Korrekturwerte so verrechnet, als seien sie im Werkzeugkoordinatensystem (Tool Coordinate System, TCS) vermessen worden.
		Die Schreibweise der Zeichen im String (groß oder klein) ist beliebig. Es können beliebig viele Leerzeichen oder Tabulatorzeichen (white spaces) eingefügt werden.			

3	<CorComp>:	Spezifiziert die Komponente(n) des Werkzeugdatensatzes, die beschrieben werden sollen (optional)		
		Datentyp:	INT	
		Wert:	0	Der Korrekturwert <CorVal>[0] bezieht sich auf die im Parameter <GeoAx> übergebene Geometrieachse im Werkstückkoordinatensystem oder im Werkzeugkoordinatensystem (siehe dazu die Beschreibung des Parameters <Comp>). D. h., der Korrekturwert muss in die bezeichneten Werkzeugkomponenten so eingerechnet werden, dass sich unter Berücksichtigung aller Parameter, die die Werkzeuglängenberechnung beeinflussen können, als Ergebnis eine Änderung der gesamten Werkzeuglänge in der gegebenen Achsrichtung um den vorgegebenen Wert ergibt. Diese Änderung soll erreicht werden durch die Korrektur der unter <Comp> angegebenen Komponente und der unter <CorMode> (siehe nachfolgenden Parameter) angegebenen symbolischen Rechenvorschrift. Die resultierende Korrektur kann sich deshalb in allen drei Achskomponenten auswirken.
		1	Wie "0", jedoch vektoriell. Der Inhalt des Vektors <CorVal> bezieht sich auf Abszisse, Ordinate und Applikate im Werkstückkoordinatensystem oder Werkzeugkoordinatensystem (siehe dazu die Beschreibung des Parameters <Comp>). Der nachfolgende Parameter <GeoAx> wird nicht ausgewertet.	
		2	Vektorielle Korrektur, d. h. L1, L2 und L3 können simultan verändert werden. Im Gegensatz zu den Varianten "0" und "1" beziehen sich die in <CorVal> enthaltenen Korrekturwerte jedoch auf die Koordinaten der Val ₁ -Komponente (siehe nachfolgenden Parameter <CorMode>) des Werkzeugs. Eine eventuell vorhandene Schrägstellung des Werkzeugs gegenüber dem Werkstückkoordinatensystem hat keinen Einfluss auf die Korrektur.	
		3 - 5	Korrektur der Werkzeuglänge L1 bis L3 (\$TC_DP3 bis \$TC_DP5) bzw. der entsprechenden Werte bei Verschleiß, Einrichte- oder Summenkorrekturen. Der Korrekturwert ist in <CorVal>[0] enthalten. Er ist in den Koordinaten der Val ₁ -Komponente (siehe nachfolgenden Parameter <CorMode>) des Werkzeugs gemessen. Eine eventuell vorhandene Schrägstellung des Werkzeugs gegenüber dem Werkstückkoordinatensystem hat keinen Einfluss auf die Korrektur.	
		6	Korrektur des Werkzeugradius (\$TC_DP6) bzw. der entsprechenden Werte bei Verschleiß, Einrichte- oder Summenkorrekturen. Die Bits 10 und 11 (Bewertung der Durchmesser- bzw. Durchmesserverschleißangabe wahlweise als Radius- oder Durchmesserangabe) im Maschinendatum MD20360 \$MC_TOOL_PARAMETER_DEF_MASK werden berücksichtigt.	
		7 – 11	Korrektur von \$TC_DP7 bis \$TC_DP11 bzw. der entsprechenden Werte bei Verschleiß, Einrichte- oder Summenkorrekturen. Diese Parameter werden wie der Werkzeugradius behandelt.	
		Ist dieser Parameter nicht angegeben, ist sein Wert "0".		

4

<CorMode>:

Spezifiziert die Art der auszuführenden Schreiboperation (optional)		
Datentyp:		INT
Wert:	0	$Val_{1neu} = <CorVal>$
	1	$Val_{1neu} = Val_{1alt} + <CorVal>$
	2	$Val_{1neu} = <CorVal>$ $Val_{2neu} = 0$
	3	$Val_{1neu} = Val_{1alt} + Val_{2alt} + <CorVal>$ $Val_{2neu} = 0$

Die Schreibweise $Val_{1alt} + Val_{2alt}$ ist symbolisch zu verstehen. Werden die beiden Komponenten (aufgrund des Zustandes von $<_Stat>$) unterschiedlich bewertet, d. h. ist zwischen beiden Komponenten eine Drehung wirksam, so wird Val_{2alt} vor der Addition so transformiert, dass die resultierende Werkzeuglänge nach dem Löschen von Val_{2neu} und vor der Addition von $<CorVal>$ unverändert bleibt.

$<CorVal>$ bezieht sich immer auf Val_1 . $<CorVal>$ ist ein Wert, der abhängig vom zweiten Teils des Parameters $<Comp>$ im Werkstückkoordinatensystem (WKS) oder im Werkzeugkoordinatensystem (TCS) gemessen wird. Er ist deshalb gegenüber der Werkzeugkomponente, in die er eingerechnet werden soll, gegebenenfalls bereits transformiert. Er kann somit nicht unmittelbar mit dem abgespeicherten Wert verrechnet werden, sondern muss vor der Addition zu Val_1 bzw. Val_2 zurücktransformiert werden. Das kann zur Folge haben, dass sich die Korrektur in einer anderen als in der durch $<CorComp>$ definierten Achse auswirkt, bzw. dass diese auf mehrere Achsen wirkt.

Für den Fall $<CorComp> = 0$, d. h., wenn $<CorVal>$ keinen Vektor, sondern nur einen Einzelwert enthält, werden die beschriebene Operationen in den Koordinaten ausgeführt, in denen $<CorVal>$ gemessen wurde (WKS / TCS). Insbesondere gilt das auch für das Nullsetzen von Val_{2neu} in den Varianten 2 und 3. Dieses Ergebnis wird dann in die Koordinaten des Werkzeugs zurücktransformiert. Das kann zur Folge haben, dass keine der nullzusetzenden Koordinatenwerte (L_1 , L_2 , L_3) Null wird bzw. dass Koordinatenwerte, die vorher Null waren, nun ungleich Null sind. Werden die entsprechenden Operationen jedoch sukzessive für alle drei Geometrieachsen ausgeführt, ist gewährleistet, dass auch alle drei Koordinatenwerte der zu löschenden Komponente Null werden. Ist das Werkzeug gegenüber dem Werkstückkoordinatensystem nicht gedreht, oder aber so gedreht, dass alle Werkzeugkomponenten parallel zu den Koordinatenachsen bleiben (Achsvertauschungen), ist gewährleistet, dass sich jeweils auch nur eine Werkzeugkoordinate ändert.

Die sukzessive Ausführung der gleichen Operation ($<CorMode>$) mit $<CorComp> = 0$ für alle drei Koordinatenachsen in beliebiger Reihenfolge ist identisch zu der einmaligen Ausführung der gleichen Operation mit $<CorComp>=1$.

Für die Parameterwerte "0" und "1" muss der Parameter $<Comp>$ einen und für die Parameterwerte "2" und "3" zwei Zeichen enthalten.

Beispiel:

$<Comp>$ enthält den String "ES", $<CorMode>$ den Wert "2"

⇒ $Einrichtkorrektur_{neu} = <CorVal>$, $Summenkorrektur_{neu} = 0$

Ist der Parameter $<CorMode>$ nicht angegeben, ist sein Wert "0".

5	<GeoAx>:	Gibt den Index der Geometrieachse an, in der der Korrekturwert <CorVal>[0] gemessen wurde (optional)	
		Datentyp:	INT
		Wertebereich:	0 ... 2
		Die Indizes 0 bis 2 beziehen sich auf Abszisse, Ordinate und Applikate in der wirksamen Ebene (G17/G18/G19) der aktuellen Werkzeugumgebung. Der Inhalt dieses Parameters wird nur dann ausgewertet, wenn der Parameter <CorComp> den Wert "0" hat.	
6	<Stat>:	Name des Datensatzes zur Beschreibung einer Werkzeugumgebung (optional)	
		Datentyp:	STRING
		Ist der Wert dieses Parameters der Nullstring (""), oder wird er nicht angegeben, wird der aktuelle Zustand verwendet. Ist kein Werkzeug spezifiziert, wird das aktuelle Werkzeug verwendet.	
7	<T>:	Interne T-Nummer des Werkzeugs (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben oder ist sein Wert "0", wird das in <Stat> abgelegte Werkzeug verwendet.	
		Ist der Wert dieses Parameters "-1", wird die T-Nummer des aktiven Werkzeugs verwendet. Es ist auch zulässig, die Nummer des aktiven Werkzeugs explizit anzugeben.	
8	<D>:	Hinweis:	
		Ist <Stat> nicht angegeben, wird als Werkzeugumgebung der aktuelle Zustand verwendet. Da mit <T> = 0 auf die in der Werkzeugumgebung gespeicherte T-Nummer verwiesen wird, wird in dieser das aktive Werkzeug verwendet, d. h. die Angaben <T> = 0 und <T> = -1 sind in diesem Sonderfall gleichbedeutend.	
		Schneidenummer des Werkzeugs (optional)	
		Datentyp:	INT
9	<TL>:	Ist dieser Parameter nicht angegeben oder ist sein Wert "0", richtet sich die verwendete D-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs.	
		Nummer der ortsabhängigen Korrektur (optional)	
		Datentyp:	INT
		Ist dieser Parameter nicht angegeben, richtet sich die verwendete DL-Nummer nach der Quelle der T-Nummer. Wird die T-Nummer aus der Werkzeugumgebung verwendet, wird auch die D-Nummer der Werkzeugumgebung gelesen, andernfalls die des aktuell aktiven Werkzeugs. Spezifizieren T, D und DL ein Werkzeug ohne ortsabhängige Korrekturen, dürfen im Parameter <Comp> keine Summen- oder Einrichtkorrekturen spezifiziert werden (Fehlercode in <Status>).	

Hinweis

Nicht alle möglichen Kombinationen der drei Parameter <Comp>, <CorComp> und <CorMode> sind sinnvoll. Beispielsweise erfordert die Rechenvorschrift 3 in <CorComp> die Angabe von zwei Zeichen in <Comp>. Wird eine unzulässige Parameterkombination angegeben, wird im <Status> ein entsprechender Fehlercode zurückgegeben.

Beispiele

Beispiel 1

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräswerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=SETTCOR(_CORVAL,"G",0,0,2)	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z1.333
N90 M30	

<CorComp> ist "0", deshalb wird der in Z-Richtung wirkende Koordinatenwert der Geometrie-Komponente durch den Korrekturwert 0,333 ersetzt.

Die resultierende Werkzeuggesamtlänge ist somit: $L1 = 0,333 + 1,000 = 1,333$

Beispiel 2

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräswerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=SETTCOR(_CORVAL,"W",0,1,2)	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> ist "1", deshalb wird der in Z-Richtung wirkende Korrekturwert von 0,333 zum Verschleißwert von 1,0 addiert.

Die resultierende Werkzeuggesamtlänge ist somit: $L1 = 10,0 + 1,333 = 11,333$

Beispiel 3

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräswerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=SETTCOR(_CORVAL,"GW",0,2,2)	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z0.333
N90 M30	

<CorComp> ist "2", deshalb wird die in Z-Richtung wirkende Korrektur in die Geometrie-Komponente eingetragen (der alte Wert wird überschrieben), und der Verschleißwert wird gelöscht.

Die resultierende Werkzeuggesamtlänge ist somit: $L1 = 0,333 + 0,0 = 0,333$

Beispiel 4

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräswerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=SETTCOR(_CORVAL,"GW",0,3,2)	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> ist "3", deshalb werden Verschleißwert und Korrekturwert zur Geometrie-Komponente addiert und die Verschleißkomponente gelöscht.

Die resultierende Werkzeuggesamtlänge ist somit: $L1 = 11,333 + 0,0 = 11,333$.

Beispiel 5

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräswerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP12[1,1]=1.0	; Verschleiß L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=SETTCOR(_CORVAL,"GW",0,3,0)	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-Position X0.333 Y0.000 Z11.000
N90 M30	

<CorComp> ist "3" wie im vorigen Beispiel, die Korrektur wirkt jetzt jedoch auf die Geometrieachse mit dem Index "0" (X-Achse), der wegen G17 bei einem Fräswerkzeug die Werkzeugkomponente L3 zugeordnet ist. Der Aufruf von SETTCOR beeinflusst deshalb die Werkzeugparameter \$TC_DP3 und \$TC_DP12 nicht. Der Korrekturwert wird stattdessen in \$TC_DP5 eingetragen.

Beispiel 6

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Drehwerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP4[1,1]=15.0	; Geometrie L2
N50 \$TC_DP12[1,1]=10.0	; Verschleiß L1

Programmcode	Kommentar
N60 \$TC_DP13[1,1]=0.0	; Verschleiß L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=SETTCOR(_CORVAL,"GW",0,3,1)	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-Position X24.330 Y0.000 Z17.500
N120 M30	

Das Werkzeug ist ein Drehwerkzeug. In Satz N80 wird eine Framedrehung aktiviert, sodass das Basiskoordinatensystem (BKS) gegenüber dem Werkstückkoordinatensystem (WKS) gedreht ist. Der Korrekturwert (N70) wirkt im WKS auf die Geometrieachse mit dem Index "1", d. h., da G18 aktiv ist, auf die X-Achse. Da $\langle \text{CorMode} \rangle = 3$ gilt, muss der Werkzeugverschleiß in Richtung der X-Achse des WKS nach dem Ausführen von N100 null werden.

Der Inhalt der relevanten Werkzeugparameter ist deshalb am Programmende:

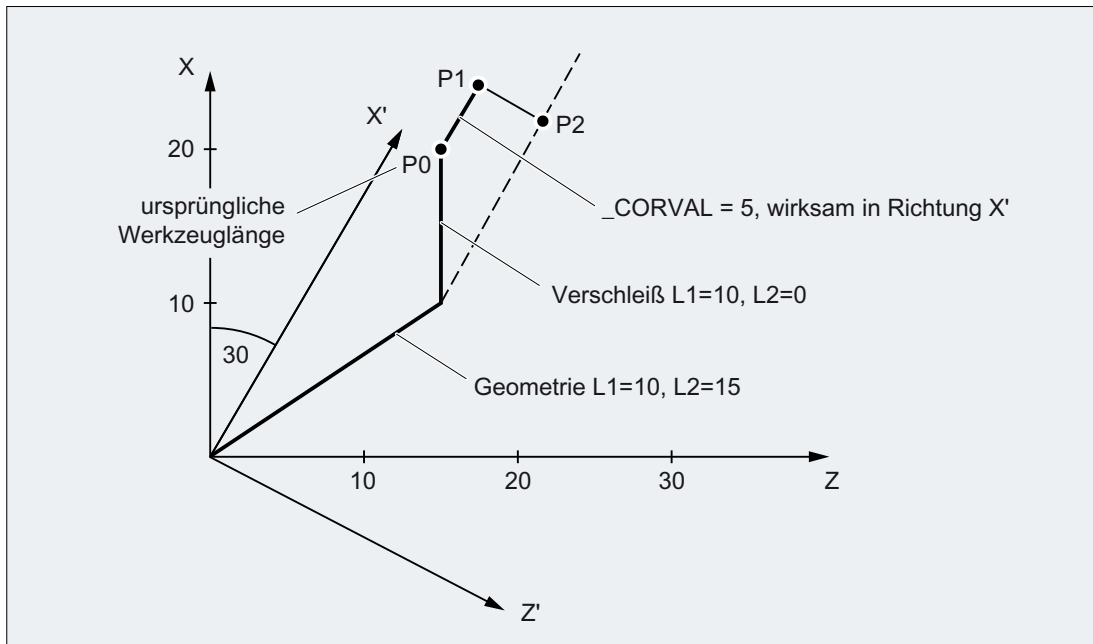
\$TC_DP3[1,1] : 21.830 ; Geometrie L1

\$TC_DP4[1,1] : 21.830 ; Geometrie L2

\$TC_DP12[1,1] : 2.500 ; Verschleiß L1

\$TC_DP13[1,1] : -4.330 ; Verschleiß L2

Die geometrischen Verhältnisse sind im folgenden Bild dargestellt. Der gesamte Verschleiß einschließlich _CORVAL wird auf die X'-Richtung im WKS projiziert. Das ergibt den Punkt P2. Die Koordinaten dieses Punkts (gemessen in X-Y-Koordinaten) werden in die Geometriekomponente des Werkzeugs eingetragen. Im Verschleiß bleibt der Differenzvektor $P_2 - P_1$ übrig. Damit hat der Verschleiß keine Komponente mehr in Richtung von _CORVAL.



Setzt man das Beispielprogramm nach N110 mit den folgenden Anweisungen fort, wird der restliche Verschleiß vollständig in die Geometrie übernommen, da die Korrektur jetzt in der Z'-Achse wirkt (Parameter <GeoAx> = 0):

```
N120 _CORVAL[0]=0.0
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N140 T1 D1 X0 Y0 Z0 ; ==> MKS-Position X24.330 Y0.000 Z17.500
```

Da der neue Korrekturwert "0" ist, darf sich die Werkzeuggesamtlänge und damit auch die in N140 angefahrne Position nicht verändern. Wäre _CORVAL in N120 ungleich "0", würde sich eine neue Werkzeuggesamtlänge und damit auch eine veränderte Position in N140 ergeben, der Verschleißanteil der Werkzeuglänge wäre jedoch in jedem Fall null, d. h., die gesamte Werkzeuglänge ist anschließend in jedem Fall im Geometrieanteil des Werkzeugs enthalten.

Das gleiche Ergebnis wie mit dem zweimaligen Aufruf der Funktion SETTCOR mit dem Parameter <CorComp> = 0 erreicht man auch durch einmaligen Aufruf mit <CorComp> = 1 (vektorielle Korrektur):

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Drehwerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP4[1,1]=15.0	; Geometrie L2
N50 \$TC_DP12[1,1]=10.0	; Verschleiß L1
N60 \$TC_DP13[1,1]=0.0	; Verschleiß L2
N70 _CORVAL[0]=0.0	
N71 _CORVAL[1]=5.0	
N72 _CORVAL[2]=0.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=SETTCOR(_CORVAL,"GW",1,3,1)	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-Position X24.330 Y0.000 Z17.500
N120 M30	

In diesem Fall sind alle Verschleißkomponenten des Werkzeugs sofort nach dem ersten Aufruf von SETTCOR in N100 gleich null.

Beispiel 7

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Drehwerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP4[1,1]=15.0	; Geometrie L2
N50 \$TC_DP12[1,1]=10.0	; Verschleiß L1
N60 \$TC_DP13[1,1]=0.0	; Verschleiß L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=SETTCOR(_CORVAL,"GW",3,3)	

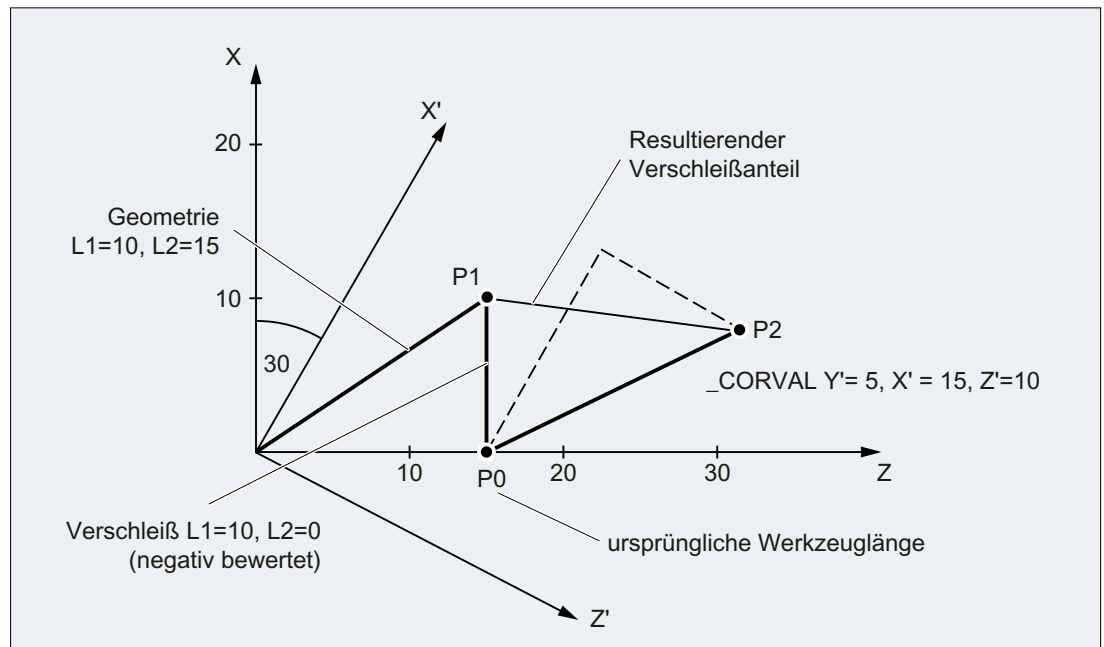
Programmcode	Kommentar
N110 T1 D1 X0 Y0 Z0	; ==> MKS-Position X25.000 Y0.000 Z15.000
N120 M30	

Gegenüber Beispiel 6 ist hier der Parameter <CorComp> = 3, deshalb kann die Angabe des Parameters <GeoAx> entfallen. Der in _CORVAL[0] enthaltene Wert wirkt jetzt unmittelbar auf die Werkzeugkomponente L1, die Drehung in N80 hat auf das Ergebnis keinen Einfluss, die Verschleißkomponenten in \$TC_DP12 wird zusammen mit _CORVAL[0] in den Geometrieanteil übernommen, sodass wegen \$TC_DP13 die gesamte Werkzeuglänge bereits nach dem ersten Aufruf von SETTCOR in N100 im Geometrieteil des Werkzeugs steht.

Beispiel 8

Programmcode	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Drehwerkzeug
N30 \$TC_DP3[1,1]=10.0	; Geometrie L1
N40 \$TC_DP4[1,1]=15.0	; Geometrie L2
N50 \$TC_DP5[1,1]=20.0	; Geometrie L3
N60 \$TC_DP12[1,1]=10.0	; Verschleiß L1
N70 \$TC_DP13[1,1]=0.0	; Verschleiß L2
N80 \$TC_DP14[1,1]=0.0	; Verschleiß L3
N90 \$SC_WEAR_SIGN=TRUE	
N100 _CORVAL[0]=10.0	
N110 _CORVAL[1]=15.0	
N120 _CORVAL[2]=5.0	
N130 ROT Y-30	
N140 T1 D1 G18 G0	
N150 R1=SETTCOR(_CORVAL,"W",1,1)	
N160 T1 D1 X0 Y0 Z0	; ==> MKS-Position X7.990 Y25.000 Z31.160
N170 M30	

In N90 ist das Settingdatum SD42930 \$SC_WEAR_SIGN gesetzt, d. h., der Verschleiß muss mit negativem Vorzeichen bewertet werden. Die Korrektur ist vektoriell (<CorComp> = 1), und der Korrekturvektor muss zum Verschleiß addiert werden (<CorMode> = 1). Die geometrischen Verhältnisse in der Z-X-Ebene sind im folgenden Bild dargestellt:



Wegen $\langle \text{CorMode} \rangle = 1$ bleibt der Geometrieanteil des Werkzeugs unverändert. Der im WKS (Drehung um y-Achse) definierte Korrekturvektor muss so in den Verschleißanteil übernommen werden, dass die gesamte Werkzeuglänge in Bild 3 auf den Punkt P_2 verweist. Der resultierende Verschleißanteil des Werkzeugs ist deshalb durch die Distanz der beiden Punkte P_1 und P_2 gegeben.

Da der Verschleiß wegen des Settingdatums SD42930 $\$SC_WEAR_SIGN$ aber negativ bewertet wird, muss die so ermittelte Korrektur mit negativem Vorzeichen in den Korrekturspeicher eingetragen werden. Der Inhalt der relevanten Werkzeugparameter ist am Programmende deshalb:

```
$TC_DP3[1,1] : 10.000 ; Geometrie L1 (unverändert)
$TC_DP4[1,1] : 15.000 ; Geometrie L2 (unverändert)
$TC_DP5[1,1] : 10.000 ; Geometrie L3 (unverändert)
$TC_DP12[1,1] : 2.010 ; Verschleiß L1 (= 10 - 15 * cos(30) + 10 * sin(30))
$TC_DP13[1,1] : -16.160 ; Verschleiß L2 (= -15 * sin(30) - 10 * cos(30))
$TC_DP14[1,1] : -5.000 ; Verschleiß L3
```

An der L3-Komponente in Y-Richtung lässt sich die Wirkung des Settingdatums SD42930 $\$SC_WEAR_SIGN$ ohne die zusätzliche Komplizierung durch die Framedrehung erkennen.

Weitere Informationen

Dreh- und Schleifwerkzeuge: Berechnung der Werkzeuglänge in Abhängigkeit von MD20360 $\$MC_TOOL_PARAMETER_DEF_MASK$

Wie bei Dreh- und Schleifwerkzeugen der Verschleiß bzw. die Werkzeuglänge in einer eventuell vorhandenen Durchmesserachse zu bewerten ist, wird festgelegt im Maschinendatum:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK (Definition der WZ-Parameter)

Bit	Wert
0	Bei Dreh- und Schleifwerkzeugen wird der Verschleißparameter der Planachse als Durchmesserwert eingerechnet:
	= 0 (Default) nein
	= 1 ja
1	Bei Dreh- und Schleifwerkzeugen wird die Werkzeuglängenkomponente der Planachse als Durchmesserwert eingerechnet:
	= 0 (Default) nein
	= 1 ja

Sind die betreffenden Bits gesetzt, wird der zugehörige Eintrag mit dem Faktor 0,5 bewertet. Die Korrektur mittels SETTCOR wird so ausgeführt, dass die gesamte effektive Werkzeuglängenänderung gleich dem in <CorVal> übergebenen Wert ist. Wird eine Länge auf Grund des Maschinendatums MD20360 \$MC_TOOL_PARAMETER_DEF_MASK bei der Längenberechnung mit dem Faktor 0,5 bewertet, muss die Korrektur dieser Komponente deshalb mit dem doppelten übergebenen Wert erfolgen.

Beispiel:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK = 2 (Werkzeuglänge muss in der Durchmesserachse mit dem Faktor 0,5 bewertet werden)

X ist Durchmesserachse

Programmcode	Kommentar
N10 DEF REAL _LEN[11]	
N20 DEF REAL _CORVAL[3]	
N30 \$TC_DP1[1,1]=500	
N40 \$TC_DP2[1,1]=2	
N50 \$TC_DP3[1,1]=3.	
N60 \$TC_DP4[1,1]=4.	
N70 \$TC_DP5[1,1]=5.	
N80 _CORVAL[0]=1.	
N90 _CORVAL[1]=1.	
N100 _CORVAL[2]=1.	
N110 T1 D1 G18 G0 X0 Y0 Z0	; ==> MKS-Position X1.5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)	
N130 T1 D1 X0 Y0 Z0	; ==> MKS-Position X2.5 Y6 Z5
N140 R3=\$TC_DP3[1,1]	; = 5. = (3.000 + 2.*1.000)
N150 R4=\$TC_DP4[1,1]	; = 5. = (4.000 + 1.000)
N160 R5=\$TC_DP5[1,1]	; = 6. = (5.000 + 1.000)
N170 M30	

Die Korrektur der Werkzeuglänge soll in jeder Achse 1 mm sein (N80 bis N100). In den Längen L2 und L3 wird deshalb zur Originallänge jeweils 1 mm addiert. Zur Originalwerkzeuglänge in L1 wird dagegen der doppelte Korrekturwert (2 mm) addiert, damit sich die Gesamtwerkzeuglänge wie verlangt um 1 mm ändert. Im Vergleich der angefahrenen Positionen in den Sätzen N110 und N130 erkennt man, dass sich jede Achsposition um 1 mm verändert hat.

19.14 Zuordnung der Werkzeuglängen L1, L2, L3 zu den Koordinatenachsen lesen (LENTOAX)

Die Funktion LENTOAX liefert Informationen über die Zuordnung der Werkzeuglängen L1, L2 und L3 des **aktiven** Werkzeugs zu Abszisse, Ordinate und Applikate. Die Zuordnung von Abszisse, Ordinate und Applikate zu den Geometrieachsen wird durch Frames und die aktive Ebene (G17 - G19) beeinflusst.

Betrachtet wird dabei nur der Geometrieanteil eines Werkzeugs (\$TC_DP3[<t>,<d>] bis \$TC_DP5[<t>,<d>]), d. h., eine gegebenenfalls davon abweichende Achszuordnung weiterer Komponenten (z. B. Verschleiß) hat auf das Ergebnis keinen Einfluss.

Syntax

<Status> = LENTOAX(<AxInd>, <Matrix>[, <Coord>])

Prinzip

LENTOAX (. . .):		Vordefinierte Funktion zum Lesen der Zuordnung der Werkzeuglängen L1, L2 und L3 des aktiven Werkzeugs zu den Koordinatenachsen	
Alleine im Satz:		ja	
<Status>:	Rückgabewert der Funktion. Negative Werte zeigen Fehlerzustände an.		
	Datentyp:		INT
	Wert:	0	Funktion OK Information in <AxInd> reicht zur Beschreibung aus (alle Werkzeuglängenkomponenten sind parallel zu Geometrieachsen).
		1	Funktion ist OK, zu einer korrekten Beschreibung muss aber der Inhalt von <Matrix> ausgewertet werden (die Werkzeuglängenkomponenten sind nicht parallel zu den Geometrieachsen).
		-1	Ungültiger String im Parameter <Coord>.
-2		Kein Werkzeug aktiv.	
Parameter			

19.14 Zuordnung der Werkzeuglängen L1, L2, L3 zu den Koordinatenachsen lesen (LENTOAX)

1	<AxInd>:	<p>Sind die Werkzeuglängenkomponenten parallel zu den Geometrieachsen, werden die Achsindizes, die den Längenkomponenten L1 bis L3 zugeordnet sind, im Feld <AxInd> zurückgeliefert:</p> <ul style="list-style-type: none">• <AxInd> [0]: Abszisse• <AxInd> [1]: Ordinate• <AxInd> [2]: Applikate	
	Datentyp:	INT[3]	
	Wert:	0	Keine Zuordnung vorhanden (Achse existiert nicht)
		1 ... 3 bzw. -1 ... -3	Nummer der Länge, die in der entsprechende Koordinatenachse wirkt. Das Vorzeichen ist negativ, wenn die Werkzeuglängenkomponente in negative Koordinatenrichtung zeigt.
	<p>Sind nicht alle Längenkomponenten parallel bzw. antiparallel zu den Geometrieachsen, wird in <AxInd> jeweils der Index der Achse zurückgegeben, die den größten Anteil einer Werkzeuglängenkomponente enthält. In diesem Fall (falls die Funktion nicht aus einem anderen Grund einen Fehler liefert) ist der Rückgabewert <Status> = 1. Die Abbildung der Werkzeuglängenkomponenten L1 bis L3 auf die Geometrieachsen 1 bis 3 wird dann durch den Inhalt des 2. Parameters <Matrix> vollständig beschrieben.</p>		
2	<Matrix>:	<p>Matrix, die den Vektor der Werkzeuglängen (L1=1, L2=1, L3=1) in den Vektor der Koordinatenachsen (Abszisse, Ordinate, Applikate) abbildet, d. h., den Spalten sind die Werkzeuglängenkomponenten in der Reihenfolge L1, L2, L3 zugeordnet, den Zeilen die Achsen in der Reihenfolge Abszisse, Ordinate, Applikate.</p>	
	Datentyp:	REAL	
	<p>In der Matrix sind immer alle Elemente gültig, auch dann, wenn die zu einer Koordinatenachse gehörende Geometrieachse nicht vorhanden ist, d. h. wenn der entsprechende Eintrag in <AxInd> null ist.</p>		
3	<Coord>:	<p>Koordinatensystem, für das die Zuordnung gilt (optional)</p>	
	Datentyp:	STRING	
	Zeichen:	MCS M	Abbildung der Werkzeuglänge in das Maschinenkoordinatensystem
		BCS B	Abbildung der Werkzeuglänge in das Basiskoordinatensystem
		WCS W	Abbildung der Werkzeuglänge in das Werkstückkoordinatensystem (Default)
		KCS K	Abbildung der Werkzeuglänge in das Werkzeugkoordinatensystem der kinematischen Transformation
		TCS T	Abbildung der Werkzeuglänge in das Werkzeugkoordinatensystem
	<p>Die Schreibweise der Zeichen im String (groß oder klein) ist beliebig. Wird der Parameter <Coord> nicht angegeben, wird das WKS verwendet (Default).</p>		

19.14 Zuordnung der Werkzeuglängen L1, L2, L3 zu den Koordinatenachsen lesen (LENTOAX)

Hinweis

Im TCS sind immer alle Werkzeuglängenkomponenten parallel oder antiparallel zu den Achsen.

Antiparallel können die Komponenten nur dann sein, wenn Spiegeln aktiv ist und das folgende Settingdatum gesetzt ist:

SD42900 \$SC_MIRROR_TOOL_LENGTH (Vorzeichenwechsel Werkzeuglänge beim Spiegeln)

Beispiel

Standardfall Fräswerkzeug bei G17.

L1 wirkt in Z (Applikate), L2 wirkt in Y (Ordinate), L3 wirkt in X (Abszisse).

Aufruf der Funktion in der Form:

```
<Status>=LENTOAX(<AxInd>,<Matrix>,"WCS")
```

Der Ergebnisparameter <AxInd> enthält dann die Werte:

```
<AxInd>[0] = 3
```

```
<AxInd>[1] = 2
```

```
<AxInd>[2] = 1
```

oder kurz: (3, 2, 1)

Die zugehörige Matrix (<Matrix>) ist in diesem Fall:

$$\text{<Matrix>} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Ein Wechsel von G17 nach G18 oder G19 ändert am Ergebnis nichts, da die Zuordnung der Längenkomponenten zu den Geometrieachsen sich in gleicher Weise ändert wie die Zuordnung von Abszisse, Ordinate und Applikate.

Es wird nun bei aktivem G17 eine Framedrehung um Z von 60 Grad programmiert, z. B.:

```
ROT Z60
```

Die Richtung der Applikate (Z-Richtung) bleibt unverändert, der Hauptanteil von L2 liegt nun in Richtung der neuen X-Achse, der Hauptanteil von L1 in Richtung der negativen Y-Achse. Der Rückgabewert (<Status>) ist deshalb "1", <AxInd> enthält die Werte (2, -3, 1).

Die zugehörige Matrix (<Matrix>) ist in diesem Fall:

$$\text{<Matrix>} = \begin{pmatrix} 0 & \sin 60^\circ & \cos 60^\circ \\ 0 & \cos 60^\circ & -\sin 60^\circ \\ 1 & 0 & 0 \end{pmatrix}$$

19.15 Randbedingungen

19.15.1 Flache D-Nummern-Struktur

Satzsuchlauf

Wann bei Satzsuchlauf die Hilfsfunktionen an die PLC ausgegeben werden, kann parametriert werden mit:

- MD22080 \$MC_AUXFU_PREDEF_SPEC
- MD22035 \$MC_AUXFU_ASSIGN_SPEC
- MD11110 \$MN_AUXFU_GROUP_SPEC

19.15.2 SD42935 Erweiterungen

Transformation der Verschleißwerte

Welche der Verschleißkomponenten im Zusammenhang mit den Anweisungen `TOWMCS` (Verschleißwerte im Maschinen-Koordinatensystem) und `TOWWCS` (Verschleißwerte im Werkstück-Koordinatensystem) transformiert bzw. nicht transformiert werden sollen, kann festgelegt werden durch:

SD42935 \$SC_WEAR_TRANSFORM (Transformation der Verschleißwerte)

19.15.3 Ankratzen

Ankratzen

Wird die Funktion "Ankratzen" im Reset-Zustand verwendet, so ist bezüglich des Grundstellungswerts Folgendes zu beachten:

- Für die Bewertung der Verschleißkomponenten ist der Grundstellungswert der G-Gruppe TOWSTD, TOWMCS und TOWWCS maßgebend.
- Wird zur korrekten Berechnung ein anderer als der Grundstellungswert benötigt, darf das Ankratzen nur im Stopp-Zustand erfolgen.

19.16 Datenlisten

19.16.1 Maschinendaten

19.16.1.1 NC-spezifische Maschinendaten

Nummer	Bezeichner: \$MN_	Beschreibung
18082	MM_NUM_TOOL	Anzahl der Werkzeuge, die NC verwalten kann (SRAM)
18088	MM_NUM_TOOL_CARRIER	Maximale Anzahl der definierter Werkzeugträger
18094	MM_NUM_CC_TDA_PARAM	Anzahl der Werkzeugdaten (SRAM)
18096	MM_NUM_CC_TOA_PARAM	Anzahl der Daten, pro Werkzeugschneide für Compileryklen (SRAM)
18100	MM_NUM_CUTTING_EDGES_IN_TOA	Werkzeugkorrekturen im TOA-Bereich (SRAM)
18102	MM_TYPE_OF_CUTTING_EDGE	Art der D-Nummer Programmierung (SRAM)
18105	MM_MAX_CUTTING_EDGE_NO	Maximaler Wert der D-Nummer
18106	MM_MAX_CUTTING_EDGE_PERTOOL	Maximale Anzahl der D-Nummern pro Werkzeug
18108	MM_NUM_SUMCORR	Anzahl aller Summenkorrekturen im NC
18110	MM_MAX_SUMCORR_PER_CUTTEDGE	Anzahl von Summenkorrekturen pro Schneide
18112	MM_KIND_OF_SUMCORR	Eigenschaften der Summenkorrekturen im TO-Bereich (SRAM)
18114	MM_ENABLE_TOOL_ORIENT	Werkzeugschneiden Orientierung zuordnen
18116	MM_NUM_TOOL_ENV	Werkzeugumgebungen im TO-Bereich (SRAM)

19.16.1.2 Kanal-spezifische Maschinendaten

Nummer	Bezeichner: \$MC_	Beschreibung
20096	T_M_ADDRESS_EXT_IS_SPINO	Bedeutung der Adresserweiterung bei T, M Werkzeugwechsel
20110	RESET_MODE_MASK	Festlegung der Steuerungs-Grundstellung nach RESET/Teileprogrammende
20120	TOOL_RESET_VALUE	Werkzeug Längenkorrektur im Hochlauf (Reset/TP-Ende)
20121	TOOL_PRESEL_RESET_VALUE	Vorgewähltes Werkzeug bei RESET
20125	CUTMOD_ERR	Fehlerbehandlung für die Funktion CUTMOD
20126	TOOL_CARRIER_RESET_VALUE	Wirksamer Werkzeugträger bei RESET
20127	CUTMOD_INIT	CUTMOD bei POWER ON initialisieren
20130	CUTTING_EDGE_RESET_VALUE	Werkzeugschneide Längenkorrektur im Hochlauf (Reset/TP-Ende)
20132	SUMCORR_RESET_VALUE	Wirksame Summenkorrektur bei RESET
20140	TRAFO_RESET_VALUE	Transformationsdatensatz im Hochlauf (Reset/TP-Ende)

Nummer	Bezeichner: \$MC_	Beschreibung
20180	TOCARR_ROT_ANGLE_INCR[i]	Rundachseninkrement des orientierbaren Werkzeugträgers
20182	TOCARR_ROT_ANGLE_OFFSET[i]	Rundachsoffset des orientierbaren Werkzeugträgers
20184	TOCARR_BASE_FRAME_NUMBER	Nummer des Basisframes für Aufnahme des Tischoffsets
20188	TOCARR_FINE_LIM_LIN	Limit lineare Feinverschiebung TCARR
20190	TOCARR_FINE_LIM_ROT	Limit der rotatorische Feinverschiebung TCARR
20202	WAB_MAXNUM_DUMMY_BLOCKS	Max. Satzanzahl ohne Verfahrbewegung bei WAB
20204	WAB_CLEARANCE_TOLERANCE	Richtungsumkehr bei WAB
20210	CUTCOM_CORNER_LIMIT	Maximalwinkel für Ausgleichssätze bei WRK
20220	CUTCOM_MAX_DISC	Maximaler Wert für DISC
20230	CUTCOM_CURVE_INSERT_LIMIT	Maximalwert für Schnittpunktberechnung bei WRK
20240	CUTCOM_MAXNUM_CHECK_BLOCKS	Sätze für vorausschauende Konturberechnung bei WRK
20250	CUTCOM_MAXNUM_DUMMY_BLOCKS	Maximale Satzanzahl ohne Verfahrbewegung bei WRK
20252	CUTCOM_MAXNUM_SUPPR_BLOCKS	Maximale Satzanzahl mit Korrekturunterdrückung
20256	CUTCOM_INTERPOL_POLY_ENABLE	Schnittpunktverfahren für Polynome möglich
20270	CUTTING_EDGE_DEFAULT	Grundstellung der Werkzeugschneide ohne Programmierung
20272	SUMCORR_DEFAULT	Grundstellung Summenkorrektur ohne Programm
20360	TOOL_PARAMETER_DEF_MASK	Definition der Werkzeug-Parameter
20390	TOOL_TEMP_COMP_ON	Aktivierung der Temperaturkompensation für Werkzeuglängen
20392	TOOL_TEMP_COMP_LIMIT	Maximale Temperaturkompensation für Werkzeuglängen
20610	ADD_MOVE_ACCEL_RESERVE	Beschleunigungsreserve für überlagerte Bewegungen
21080	CUTCOM_PARALLEL_ORI_LIMIT	Minimaler Winkel (Bahntangente und Werkzeugorientierung) bei 3D-Werkzeugradiuskorrektur
22530	TOCARR_CHANGE_M_CODE	M-Code bei Werkzeugträgerwechsel
22550	TOOL_CHANGE_MODE	Neue WZK bei M-Funktion
22560	TOOL_CHANGE_M_CODE	M-Funktion für Werkzeugwechsel
22562	TOOL_CHANGE_ERROR_MODE	Verhalten bei Fehlern im Werkzeugwechsel
24558	TRAFO5_JOINT_OFFSET_PART_1	Vektor kinematischer Versatz im Tisch, Trafo 1
24658	TRAFO5_JOINT_OFFSET_PART_2	Vektor kinematischer Versatz im Tisch, Trafo 2
28085	MM_LINK_TOA_UNIT	Zuordnung der TO-Einheit zu einem Kanal (SRAM)

19.16.1.3 Achs-/Spindel-spezifische Maschinendaten

Nummer	Bezeichner: \$MA_	Beschreibung
32750	TEMP_COMP_TYPE	Temperaturkompensationstyp

19.16.2 Settingdaten

19.16.2.1 Kanal-spezifische Settingdaten

Nummer	Bezeichner: \$SC_	Beschreibung
42442	TOOL_OFFSET_INCR_PROG	Werkzeuglängenkorrekturen
42470	CRIT_SPLINE_ANGLE	Ecken-Grenzwinkel für Kompressor
42480	STOP_CUTCOM_STOPRE	Alarmreaktion bei Werkzeugradiuskorrektur und Vorlaufstopp
42494	CUTCOM_ACT_DEACT_CTRL	An-/Abfahrverhalten bei Wz.-Radiuskorrektur
42496	CUTCOM_CLSDT_CONT	Verhalten der Werkzeugradiuskorrektur bei geschlossener Kontur
42900	MIRROR_TOOL_LENGTH	Vorzeichenwechsel Werkzeuglängen beim Spiegeln
42910	MIRROR_TOOL_WEAR	Vorzeichenwechsel Werkzeugverschleiß beim Spiegeln
42920	WEAR_SIGN_CUTPOS	Vorzeichen des Verschleißes bei Werkzeugen mit Schneidenlage
42930	WEAR_SIGN	Vorzeichen des Verschleißes
42935	WEAR_TRANSFORM	Transformationen für Werkzeugkomponenten
42940	TOOL_LENGTH_CONST	Wechsel der Werkzeuglängenkomponenten bei Ebenenwechsel
42950	TOOL_LENGTH_TYPE	Zuordnung der Werkzeuglängenkompensation unabhängig vom Werkzeugtyp
42960	TOOL_TEMP_COMP	Temperaturkompensationswert bezogen auf das Werkzeug
42974	TCARR_FINE_CORRECTION	Feinverschiebung TCARR ein/aus
42984	CUTDIRMOD	Modifikation von \$P_AD[2] bzw. \$P_AD[11]

19.16.3 Signale

19.16.3.1 Signale von Kanal

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
T-Funktion 1 Änderung	DB21,DBX61.0	-
D-Funktion 1 Änderung	DB21,DBX62.0	-
T-Funktion 1	DB21,DBB116-119	DB250x.DBD2000
D-Funktion 1	DB21,DBB128-129	DB250x.DBD5000
Aktiver G-Befehl der Gruppe 7	DB21,DBB214	DB350x.DBB6
Aktiver G-Befehl der Gruppe 16	DB21,DBB223	DB350x.DBB15
Aktiver G-Befehl der Gruppe 17	DB21,DBB224	DB350x.DBB16

Signalname	SINUMERIK 840D sl	SINUMERIK 828D
Aktiver G-Befehl der Gruppe 18	DB21,DBB225	DB350x.DBB17
Aktiver G-Befehl der Gruppe 23	DB21,DBB230	DB350x.DBB22

Z1: NC/PLC-Nahtstellensignale

Ab Ausgabestand 05/2017 befindet sich ist die ausführliche Beschreibung der NC/PLC-Nahtstellensignale im Listenhandbuch NC-Variablen und Nahtstellensignale.

A.1 Liste der Abkürzungen

A	
A	Ausgang
ADI4	Analog Drive Interface for 4 Axes
AC	Adaptive Control
ALM	Active Line Module
ARM	Asynchroner rotatorischer Motor
AS	Automatisierungssystem
ASCII	American Standard Code for Information Interchange: Amerikanische Code-Norm für den Informationsaustausch
ASIC	Application Specific Integrated Circuit: Anwender-Schaltkreis
ASUP	Asynchrones Unterprogramm
AUXFU	Auxiliary Function: Hilfsfunktion
AWL	Anweisungsliste
AWP	Anwenderprogramm

B	
BA	Betriebsart
BAG	Betriebsartengruppe
BCD	Binary Coded Decimals: Im Binärcode verschlüsselte Dezimalzahlen
BERO	Berührungsloser Näherungsschalter
BI	Binector Input
BICO	Binector Connector
BIN	Binary Files: Binärdateien
BIOS	Basic Input Output System
BKS	Basiskoordinatensystem
BO	Binector Output
BTSS	Bedientafelschnittstelle

C	
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CC	Compile Cycle: Compile-Zyklen
CEC	Cross Error Compensation
CI	Connector Input
CF-Card	Compact Flash-Card

C	
CNC	Computerized Numerical Control: Computerunterstützte numerische Steuerung
CO	Connector Output
CoL	Certificate of License
COM	Communication
CPA	Compiler Projecting Data: Projektierdaten des Compilers
CRT	Cathode Ray Tube: Bildröhre
CSB	Central Service Board: PLC-Baugruppe
CU	Control Unit
CP	Communication Processor
CPU	Central Processing Unit: Zentrale Rechneinheit
CR	Carriage Return
CTS	Clear To Send: Meldung der Sendebereitschaft bei seriellen Daten-Schnittstellen
CUTCOM	Cutter Radius Compensation: Werkzeugradiuskorrektur

D	
DAU	Digital-Analog-Umwandler
DB	Datenbaustein (PLC)
DBB	Datenbaustein-Byte (PLC)
DBD	Datenbaustein-Doppelwort (PLC)
DBW	Datenbaustein-Wort (PLC)
DBX	Datenbaustein-Bit (PLC)
DDE	Dynamic Data Exchange
DDS	Drive Data Set: Antriebsdatensatz
DIN	Deutsche Industrie Norm
DIO	Data Input/Output: Datenübertragungs-Anzeige
DIR	Directory: Verzeichnis
DLL	Dynamic Link Library
DO	Drive Object
DPM	Dual Port Memory
DPR	Dual Port RAM
DRAM	Dynamischer Speicher (ungepuffert)
DRF	Differential Resolver Function: Differential-Drehmelder-Funktion (Handrad)
DRIVE-CLiQ	Drive Component Link with IQ
DRY	Dry Run: Probelaufvorschub
DSB	Decoding Single Block: Dekodierungseinzelsatz
DSC	Dynamic Servo Control / Dynamic Stiffness Control
DW	Datenwort
DWORD	Doppelwort (aktuell 32 Bit)

E	
E	Eingang
EES	Execution from External Storage
E/A	Ein-/Ausgabe
ENC	Encoder: Istwertgeber
EPF	Einfach Peripheriemodul (PLC–E/A–Baugruppe)
EGB	Elektronisch gefährdete Baugruppen/Bauelemente
EMV	Elektromagnetische Verträglichkeit
EN	Europäische Norm
ENC	Encoder: Istwertgeber
EnDat	Geberschnittstelle
EPROM	Erasable Programmable Read Only Memory: Löschbarer, elektrisch programmierbarer nur Lesespeicher
ePS Network Services	Dienste zur internetgestützten Maschinen-Fernwartung
EQN	Typbezeichnung eines Absolutwertgebers mit 2048 Sinussignalen/Umdrehung
ES	Engineering System
ESR	Erweitertes Stillsetzen und Rückziehen
ETC	ETC–Taste ">"; Erweiterung der Softkeyleiste im gleichen Menü

F	
FB	Funktionsbaustein (PLC)
FC	Function Call: Funktionsbaustein (PLC)
FEPROM	Flash–EPROM: Les- und schreibbarer Speicher
FIFO	First In First Out: Speicher, der ohne Adressangabe arbeitet und dessen Daten in derselben Reihenfolge gelesen werden, in der sie gespeichert wurden
FIPO	Feininterpolator
FPU	Floating Point Unit: Gleitpunkteinheit
FRK	Fräsradiuskorrektur
FST	Feed Stop: Vorschub Halt
FUP	Funktionsplan (Programmierungsmethode für PLC)
FW	Firmware

G	
GC	Global Control (PROFIBUS: Broadcast-Telegramm)
GDIR	Globaler Teileprogrammspeicher
GEO	Geometrie, z.B. Geometrieachse
GIA	Gear Interpolation Data: Getriebeinterpolationsdaten
GND	Signal Ground
GP	Grundprogramm (PLC)
GS	Getriebestufe
GSD	Gerätestammdatei zur Beschreibung eines PROFIBUS Slaves

G	
GSDML	Generic Station Description Markup Language: XML-basierte Beschreibungssprache zur Erstellung einer GSD-Datei
GUD	Global User Data: Globale Anwenderdaten

H	
HEX	Kurzbezeichnung für hexadezimale Zahl
HiFu	Hilfsfunktion
HLA	Hydraulischer Linearantrieb
HMI	Human Machine Interface: SINUMERIK-Bedienoberfläche
HSA	Hauptspindelantrieb
HW	Hardware

I	
IBN	Inbetriebnahme
IKA	Interpolatorische Kompensation
IM	Interface-Modul: Anschaltungsbaugruppe
IMR	Interface-Modul Receive: Anschaltungsbaugruppe für Empfangsbetrieb
IMS	Interface-Modul Send: Anschaltungsbaugruppe für Sendebetrieb
INC	Increment: Schrittmaß
INI	Initializing Data: Initialisierungsdaten
IPO	Interpolator
ISA	International Standard Architecture
ISO	International Standard Organization

J	
JOG	Jogging: Einrichtbetrieb

K	
K_v	Verstärkungsfaktor des Regelkreises
K_p	Proportionalverstärkung
K_U	Übersetzungsverhältnis
KOP	Kontaktplan (Programmiermethode für PLC)

L	
LAI	Logic Machine Axis Image: Logisches Maschinenachsen-Abbild
LAN	Local Area Network
LCD	Liquid-Crystal Display: Flüssigkristallanzeige
LED	Light Emitting Diode: Leuchtdiode
LF	Line Feed

L	
LMS	Lagemesssystem
LR	Lageregler
LSB	Least Significant Bit: Niederwertigstes Bit
LUD	Local User Data: Anwenderdaten (lokal)

M	
MAC	Media Access Control
MAIN	Main program: Hauptprogramm (OB1, PLC)
MB	Megabyte
MCI	Motion Control Interface
MCIS	Motion–Control–Information–System
MCP	Machine Control Panel: Maschinensteuertafel
MD	Maschinendatum bzw. Maschinendaten
MDA	Manual Data Automatic: Handeingabe
MDS	Motor Data Set: Motordatensatz
MELDW	Meldungswort
MKS	Maschinenkoordinatensystem
MM	Motor Module
MPF	Main Program File: Hauptprogramm (NC)
MSTT	Maschinensteuertafel

N	
NC	Numerical Control: Numerische Steuerung mit Satzaufbereitung, Verfahrbereich usw.
NCU	Numerical Control Unit: Hardware-Einheit des NC
NRK	Bezeichnung des Betriebssystems des NC
NST	Nahtstellensignal
NURBS	Non-Uniform Rational B-Spline
NV	Nullpunktverschiebung
NX	Numerical Extension: Achserweiterungsbaugruppe

O	
OB	Organisationsbaustein in der PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: Bedieneinrichtung
OPI	Operation Panel Interface: Bedientafel-Anschaltung
OPT	Options: Optionen
OLP	Optical Link Plug: Busstecker für Lichtleiter
OSI	Open Systems Interconnection: Normung für Rechnerkommunikation

P	
PAA	Prozessabbild der Ausgänge
PAE	Prozessabbild der Eingänge
PC	Personal Computer
PCIN	Name der SW für den Datenaustausch mit der Steuerung
PCMCIA	Personal Computer Memory Card International Association: Speichersteckkarten-Normierung
PCU	PC Unit: PC-Box (Rechneinheit)
PG	Programmiergerät
PKE	Parameterkennung: Teil eines PKW
PKW	Parameterkennung: Wert (Parametrierteil eines PPO)
PLC	Programmable Logic Control: Anpass-Steuerung
PN	PROFINET
PNO	PROFIBUS-Nutzerorganisation
PO	POWER ON
POE	Programmorganisationseinheit
POS	Position/Positionieren
POSMO A	Positioning Motor Actuator: Positioniermotor
POSMO CA	Positioning Motor Compact AC: Komplette Antriebseinheit mit integrierter Leistungs- und Reglungsbaugruppe sowie Positioniereinheit und Programmspeicher; Wechselstrom-Einspeisung
POSMO CD	Positioning Motor Compact DC: wie CA, jedoch Gleichstromeinspeisung
POSMO SI	Positioning Motor Servo Integrated: Positioniermotor; Gleichstromeinspeisung
PPO	Parameter Prozessdaten Objekt ; Zyklisches Datentelegramm bei der Übertragung mit PROFIBUS-DP und Profil "Drehzahlveränderbare Antriebe"
PPU	Panel Processing Unit (zentrale Hardware einer Panel-basierten CNC-Steuerung z.B. SINUMERIK 828D)
PROFIBUS	Process Field Bus: Serieller Datenbus
PRT	Programmtest
PSW	Programmsteuerwort
PTP	Point to Point: Punkt zu Punkt
PUD	Program Global User Data: Programmglobale Anwendervariable
PZD	Prozessdaten: Prozessdatenteil eines PPO

Q	
QFK	Quadrantenfehler Kompensation

R	
RAM	Random Access Memory: Schreib-/Lese-Speicher
REF	Funktion Referenzpunkt anfahren
REPOS	Funktion Repositionieren
RISC	Reduced Instruction Set Computer: Prozessortyp mit kleinem Befehlssatz und schnellem Befehlsdurchsatz

R	
ROV	Rapid Override: Eingangskorrektur
RP	R-Parameter, Rechenparameter, vordefinierte Anwendervariable
RPA	R-Parameter Active: Speicherbereich in NC für R-Parameternummern
RPY	Roll Pitch Yaw: Drehungsart eines Koordinatensystems
RTL	Rapid Traverse Linear Interpolation: Lineare Interpolation bei Eilgangbewegung
RTS	Request To Send: Sendeteil einschalten, Steuersignal von seriellen Daten-Schnittstellen
RTCP	Real Time Control Protocol

S	
SA	Synchronaktion
SBC	Safe Break Control: Sichere Bremsenansteuerung
SBL	Single Block: Einzelsatz
SBR	Subroutine: Unterprogramm (PLC)
SD	Settingdatum bzw. Settingdaten
SDB	System Datenbaustein
SEA	Setting Data Active: Kennzeichnung (Dateityp) für Settingdaten
SERUPRO	Search-Run by Program Test: Satzsuchlauf via Programmtest
SFB	System Funktionsbaustein
SFC	System Function Call
SGE	Sicherheitsgerichteter Eingang
SGA	Sicherheitsgerichteter Ausgang
SH	Sicherer Halt
SIM	Single in Line Module
SK	Softkey
SKP	Skip: Funktion zum Ausblenden eines Teileprogrammsatzes
SLM	Synchroner Linearmotor
SM	Schrittmotor
SMC	Sensor Module Cabinet Mounted
SME	Sensor Module Externally Mounted
SMI	Sensor Module Integrated
SPF	Sub Program File: Unterprogramm (NC)
SPS	Speicherprogrammierbare Steuerung = PLC
SRAM	Statischer Speicher (gepuffert)
SRK	Schneidenradiuskorrektur
SRM	Synchron rotatorischer Motor
SSFK	Spindelsteigungsfehlerkompensation
SSI	Serial Synchron Interface: Serielle synchrone Schnittstelle
SSL	Satzsuchlauf
STW	Steuerwort
SUG	Scheibenumfangsgeschwindigkeit
SW	Software

S	
SYF	System Files: Systemdateien
SYNACT	Synchronized Action: Synchronaktion

T	
TB	Terminal Board (SINAMICS)
TCP	Tool Center Point: Werkzeugspitze
TCP/IP	Transport Control Protocol / Internet Protocol
TCU	Thin Client Unit
TEA	Testing Data Active: Kennung für Maschinendaten
TIA	Totally Integrated Automation
TM	Terminal Module (SINAMICS)
TO	Tool Offset: Werkzeugkorrektur
TOA	Tool Offset Active: Kennzeichnung (Dateityp) für Werkzeugkorrekturen
TRANSMIT	Transform Milling Into Turning: Koordinatentransformation für Fräsbearbeitungen an einer Drehmaschine
TTL	Transistor–Transistor–Logik (Schnittstellen–Typ)
TZ	Technologiezyklus

U	
UFR	User Frame: Nullpunktverschiebung
UP	Unterprogramm
USB	Universal Serial Bus
USV	Unterbrechungsfreie Stromversorgung

V	
VDI	Interne Kommunikationsschnittstelle zwischen NC und PLC
VDI	Verein Deutscher Ingenieure
VDE	Verband Deutscher Elektrotechniker
VI	Voltage Input
VO	Voltage Output
VSA	Vorschubantrieb

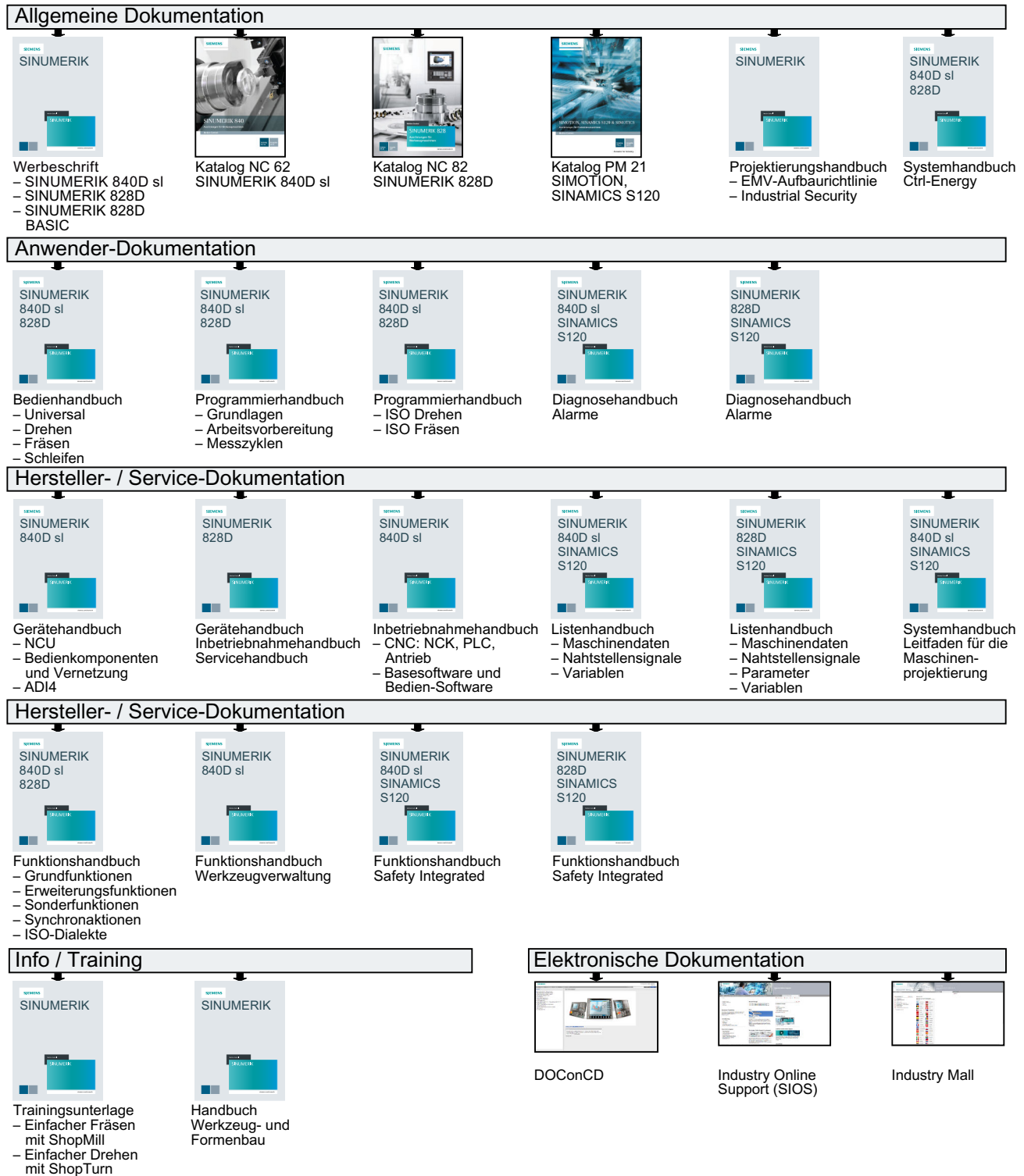
W	
WAB	Funktion Weiches An- und Abfahren
WKS	Werkstückkoordinatensystem
WKZ	Werkzeug
WLK	Werkzeuglängenkorrektur
WOP	Werkstatt-orientierte Programmierung
WPD	Work Piece Directory: Werkstückverzeichnis
WRK	Werkzeug-Radius-Korrektur

W	
WZ	Werkzeug
WZK	Werkzeugkorrektur
WZV	Werkzeugverwaltung
WZW	Werkzeugwechsel

X	
XML	Extensible Markup Language

Z	
ZOA	Zero Offset Active: Kennung für Nullpunktverschiebungen
ZSW	Zustandswort (des Antriebs)

A.2 Dokumentationsübersicht



Glossar

Absolutmaß

Angabe des Bewegungsziels einer Achsbewegung durch ein Maß, das sich auf den Nullpunkt des momentan gültigen Koordinatensystems bezieht. Siehe → Kettenmaß.

Achsadresse

Siehe → Achsname

Achsen

Die CNC-Achsen werden entsprechend ihres Funktionsumfangs abgestuft in:

- Achsen: interpolierende Bahnachsen
- Hilfsachsen: nicht interpolierende Zustell- und Positionierachsen mit achsspezifischem Vorschub. Hilfsachsen sind an der eigentlichen Bearbeitung nicht beteiligt, z. B. Werkzeugzubringer, Werkzeugmagazin.

Achsname

Zur eindeutigen Identifikation müssen alle Kanal- und → Maschinenachsen der Steuerung mit kanal- bzw. steuerungsweit eindeutigen Namen bezeichnet werden. Die → Geometrieachsen werden mit X, Y, Z benannt. Die um die Geometrieachsen drehenden → Rundachsen werden mit A, B, C benannt.

Adresse

Eine Adresse ist die Kennzeichnung für einen bestimmten Operanden oder Operandenbereich, z. B. Eingang, Ausgang usw.

Alarme

Alle → Meldungen und Alarme werden auf der Bedientafel im Klartext mit Datum und Uhrzeit und dem entsprechenden Symbol für das Löschkriterium angezeigt. Die Anzeige erfolgt getrennt nach Alarmen und Meldungen.

1. Alarme und Meldungen im Teileprogramm
Alarme und Meldungen können direkt aus dem Teileprogramm im Klartext zur Anzeige gebracht werden.
2. Alarme und Meldungen von PLC
Alarme- und Meldungen der Maschine können aus dem PLC-Programm im Klartext zur Anzeige gebracht werden. Dazu sind keine zusätzlichen Funktionsbaustein-Pakete notwendig.

Antrieb

Der Antrieb ist diejenige Einheit der CNC, welche die Drehzahl- und Momentenregelung aufgrund der Vorgaben der NC ausführt.

Anwenderdefinierte Variable

Anwender können für beliebige Nutzung im → Teileprogramm oder Datenbaustein (globale Anwenderdaten) anwenderdefinierte Variablen vereinbaren. Eine Definition enthält eine Datentypangabe und den Variablennamen. Siehe → Systemvariable.

Anwenderprogramm

Anwenderprogramme für Automatisierungssysteme S7-300 werden mit der Programmiersprache STEP 7 erstellt. Das Anwenderprogramm ist modular aufgebaut und besteht aus einzelnen Bausteinen.

Die grundlegenden Bausteintypen sind:

- Code-Bausteine
Diese Bausteine enthalten die STEP 7-Befehle.
- Datenbausteine
Diese Bausteine enthalten Konstanten und Variablen für das STEP 7-Programm.

Anwenderspeicher

Alle Programme und Daten wie Teileprogramme, Unterprogramme, Kommentare, Werkzeugkorrekturen, Nullpunktverschiebungen/Frames sowie Kanal- und Programmanwenderdaten können in den gemeinsamen CNC-Anwenderspeicher abgelegt werden.

Arbeitsfeldbegrenzung

Mit der Arbeitsfeldbegrenzung kann der Verfahrbereich der Achsen zusätzlich zu den Endschaltern eingeschränkt werden. Je Achse ist ein Wertepaar zur Beschreibung des geschützten Arbeitsraumes möglich.

Arbeitsraum

Dreidimensionaler Raum, in den die Werkzeugspitze aufgrund der Konstruktion der Werkzeugmaschine hineinfahren kann. Siehe → Schutzraum.

Arbeitsspeicher

Der Arbeitsspeicher ist ein RAM-Speicher in der → CPU, auf den der Prozessor während der Programmbearbeitung auf das Anwenderprogramm zugreift.

Archivieren

Auslesen von Dateien und/oder Verzeichnissen auf ein **externes** Speichergerät.

Asynchrones Unterprogramm

Teileprogramm, das asynchron (unabhängig) zum aktuellen Programmzustand durch ein Interruptsignal (z. B. Signal "schneller NC-Eingang") gestartet werden kann.

Automatik

Betriebsart der Steuerung (Satzfolgebetrieb nach DIN): Betriebsart bei NC-Systemen, in der ein → Teileprogramm angewählt und kontinuierlich abgearbeitet wird.

Bahnachse

Bahnachsen sind alle Bearbeitungsachsen des → Kanals, die vom → Interpolator so geführt werden, dass sie gleichzeitig starten, beschleunigen, stoppen und den Endpunkt erreichen.

Bahngeschwindigkeit

Die maximal programmierbare Bahngeschwindigkeit ist abhängig von der Eingabefeinheit. Bei einer Auflösung von beispielsweise 0,1 mm beträgt die maximal programmierbare Bahngeschwindigkeit 1000 m/min.

Bahnsteuerbetrieb

Ziel des Bahnsteuerbetriebes ist es, ein größeres Abbremsen der → Bahnachsen an den Teileprogramm-Satzgrenzen zu vermeiden und mit möglichst gleicher Bahngeschwindigkeit in den nächsten Satz zu wechseln.

Bahnvorschub

Bahnvorschub wirkt auf → Bahnachsen. Er stellt die geometrische Summe der Vorschübe der beteiligten → Geometrieachsen dar.

Basisachse

Achse, deren Soll- oder Istwert für die Berechnung eines Kompensationswertes herangezogen wird.

Basiskoordinatensystem

Kartesisches Koordinatensystem, wird durch Transformation auf das Maschinenkoordinatensystem abgebildet.

Im → Teileprogramm verwendet der Programmierer Achsnamen des Basiskoordinatensystems. Es besteht, wenn keine → Transformation aktiv ist, parallel zum → Maschinenkoordinatensystem. Der Unterschied zu diesem liegt in den → Achsnamen.

Baudrate

Geschwindigkeit bei der Datenübertragung (Bit/s).

Baustein

Als Bausteine werden alle Dateien bezeichnet, die für die Programmerstellung und Programmverarbeitung benötigt werden.

Bearbeitungskanal

Über eine Kanalstruktur können durch parallele Bewegungsabläufe Nebenzeiten verkürzt werden, z. B. Verfahren eines Ladeportals simultan zur Bearbeitung. Ein CNC-Kanal ist dabei als eigene CNC-Steuerung mit Dekodierung, Satzaufbereitung und Interpolation anzusehen.

Bedienoberfläche

Die Bedienoberfläche (BOF) ist das Anzeigemedium einer CNC-Steuerung in Gestalt eines Bildschirms. Sie ist mit horizontalen und vertikalen Softkeys gestaltet.

Beschleunigung mit Ruckbegrenzung

Zur Erzielung eines optimalen Beschleunigungsverhaltens an der Maschine bei gleichzeitiger Schonung der Mechanik kann im Bearbeitungsprogramm zwischen sprunghafter Beschleunigung und stetiger (ruckfreier) Beschleunigung umgeschaltet werden.

Betriebsart

Ablaufkonzept für den Betrieb einer SINUMERIK-Steuerung. Es sind die Betriebsarten → Jog, → MDA, → Automatik definiert.

Betriebsartengruppe

Technologisch zusammengehörige Achsen und Spindeln können zu einer Betriebsartengruppe (BAG) zusammengefasst werden. Achsen/Spindeln einer BAG können von einem oder mehreren → Kanälen gesteuert werden. Den Kanälen der BAG ist immer die gleiche → Betriebsart zugeordnet.

Bezeichner

Die Wörter nach DIN 66025 werden durch Bezeichner (Namen) für Variable (Rechenvariable, Systemvariable, Anwendervariable), für Unterprogramme, für Schlüsselwörter und Wörter mit mehreren Adressbuchstaben ergänzt. Diese Ergänzungen kommen in der Bedeutung den Wörtern beim Satzaufbau gleich. Bezeichner müssen eindeutig sein. Derselbe Bezeichner darf nicht für verschiedene Objekte verwendet werden.

Booten

Laden des Systemprogramms nach Power On.

C-Achse

Achse, um die eine gesteuerte Drehbewegung und Positionierung mit der Werkstückspindel erfolgt.

CNC

Siehe → NC

Computerized Numerical Control: umfasst die Komponenten → NC, → PLC, HMI, → COM.

CNC

Siehe → NC

Computerized Numerical Control: umfasst die Komponenten → NC, → PLC, HMI, → COM.

COM

Komponente der NC-Steuerung zur Durchführung und Koordination von Kommunikation.

CPU

Central Processing Unit, siehe → Speicherprogrammierbare Steuerung

C-Spline

Der C-Spline ist der bekannteste und am meisten verwendete Spline. Die Übergänge an den Stützpunkten sind tangential- und krümmungstetig. Es werden Polynome 3. Grades verwendet.

Datenbaustein

1. Dateneinheit der → PLC, auf die → HIGHSTEP-Programme zugreifen können.
2. Dateneinheit der → NC: Datenbausteine enthalten Datendefinitionen für globale Anwenderdaten. Die Daten können bei der Definition direkt initialisiert werden.

Datenübertragungsprogramm PCIN

PCIN ist ein Hilfsprogramm zum Senden und Empfangen von CNC-Anwenderdaten über die serielle Schnittstelle, wie z. B. Teileprogramme, Werkzeugkorrekturen etc. Das PCIN-Programm ist unter MS-DOS auf Standard-Industrie-PCs lauffähig.

Datenwort

Zwei Byte große Dateneinheit innerhalb eines → Datenbausteins.

Diagnose

1. Bedienbereich der Steuerung
2. Die Steuerung besitzt sowohl ein Selbstdiagnose-Programm als auch Testhilfen für den Service: Status-, Alarm- und Serviceanzeigen

DRF

Differential Resolver Function: NC-Funktion, die in Verbindung mit einem elektronischen Handrad eine inkrementale Nullpunktverschiebung im Automatik-Betrieb erzeugt.

Editor

Der Editor ermöglicht das Erstellen, Ändern, Ergänzen, Zusammenschieben und Einfügen von Programmen/Texten/Programmsätzen.

Eilgang

Schnellste Verfahrgeschwindigkeit einer Achse. Sie wird z. B. verwendet, wenn das Werkzeug aus einer Ruhestellung an die → Werkstückkontur herangefahren oder von der Werkstückkontur zurückgezogen wird. Die Eilganggeschwindigkeit wird maschinenspezifisch über Maschinendatum eingestellt.

Externe Nullpunktverschiebung

Von der → PLC vorgegebene Nullpunktverschiebung.

Fertigteilkontur

Kontur des fertig bearbeiteten Werkstücks. Siehe → Rohteil.

Festpunkt-Anfahren

Werkzeugmaschinen können feste Punkte wie Werkzeugwechsellpunkt, Beladepunkt, Palettenwechsellpunkt etc. definiert anfahren. Die Koordinaten dieser Punkte sind in der Steuerung hinterlegt. Die Steuerung verfährt die betroffenen Achsen, wenn möglich, im → Eilgang.

Frame

Ein Frame stellt eine Rechenvorschrift dar, die ein kartesisches Koordinatensystem in ein anderes kartesisches Koordinatensystem überführt. Ein Frame enthält die Komponenten → Nullpunktverschiebung, → Rotation, → Skalierung, → Spiegelung.

Führungssachse

Die Führungssachse ist die → Gantry-Achse, die aus Sicht des Bedieners und des Programmiers vorhanden und damit entsprechend wie eine normale NC-Achse beeinflussbar ist.

Funktionalität

Die Begrenzung des Bahnruks kann durch Programmierung des Settingdatums ein/ ausgeschaltet werden.

Parameter: *Wert*

- Wertebereich: TRUE, FALSE

Anwendbarkeit:

- Teileprogramm
- Statische Synchronaktion

Genauhalt

Bei programmierter Genauhalt-Anweisung wird die in einem Satz angegebene Position genau und ggf. sehr langsam angefahren. Zur Reduktion der Annäherungszeit werden für Eilgang und Vorschub → Genauhaltsgrenzen definiert.

Genauhaltgrenze

Erreichen alle Bahnachsen ihre Genauhaltgrenze, so verhält sich die Steuerung als habe sie einen Zielpunkt exakt erreicht. Es erfolgt Satzweilerschaltung des → Teileprogramms.

Geometrie

Beschreibung eines → Werkstücks im → Werkstückkoordinatensystem.

Geometrieachse

Die Geometrieachsen bilden das 2- bzw. 3-dimensionale → Werkstückkoordinatensystem in dem in → Teileprogrammen die Geometrie des Werkstücks programmiert wird.

Geradeninterpolation

Das Werkzeug wird auf einer Geraden zum Zielpunkt verfahren und dabei das Werkstück bearbeitet.

Geschwindigkeitsführung

Um bei Verfahrbewegungen um sehr kleine Beträge je Satz eine akzeptable Verfahrgewindigkeit erreichen zu können, kann vorausschauende Auswertung über mehrere Sätze (→ Look Ahead) eingestellt werden.

Gewindebohren ohne Ausgleichsfutter

Mit dieser Funktion können Gewinde ohne Ausgleichsfutter gebohrt werden. Durch das interpolierende Verfahren der Spindel als Rundachse und der Bohrachse werden Gewinde exakt auf Endbohrtiefe geschnitten, z. B. Sacklochgewinde (Voraussetzung: Achsbetrieb der Spindel).

Gleichlaufachse

Die Gleichlaufachse ist die → Gantry-Achse, deren Sollposition stets von der Verfahrbewegung der → Führungsachse abgeleitet und damit synchron verfahren wird. Aus Sicht des Bedieners und des Programmierers ist die Gleichlaufachse "nicht vorhanden".

Grenzdrehzahl

Maximale/minimale (Spindel-)Drehzahl: Durch Vorgaben von Maschinendaten, der → PLC oder → Settingdaten kann die maximale Drehzahl einer Spindel begrenzt sein.

Hauptprogramm

Die Bezeichnung Hauptprogramm stammt noch aus der Zeit, als Teileprogramm fest in Haupt- und → Unterprogramme unterteilt waren. Diese feste Einteilung besteht mit der heutigen SINUMERIK NC-Sprache nicht mehr. Prinzipiell kann jedes Teileprogramm im Kanal angewählt und gestartet werden. Es läuft dann in der → Programmebene 0 (Hauptprogramm-Ebene) ab. Im Hauptprogramm können weitere Teileprogramme oder → Zyklen als Unterprogramme aufgerufen werden

Hauptsatz

Durch ":" eingeleiteter Satz, der alle Angaben enthält, um den Arbeitsablauf in einem → Teileprogramm starten zu können.

HIGHSTEP

Zusammenfassung der Programmiermöglichkeiten für die → PLC des Systems AS300/AS400.

Hilfsfunktionen

Mit Hilfsfunktionen können in → Teileprogrammen → Parameter an die → PLC übergeben werden, die dort vom Maschinenhersteller definierte Reaktionen auslösen.

Hochsprache CNC

Die Hochsprache dient zum Schreiben von NC-Programmen, → Synchronaktionen und → Zyklen. Sie bietet: Kontrollstrukturen, → Anwenderdefinierte Variable, → Systemvariable, → Makrotechnik.

HW-Konfig

SIMATIC S7-Tool zum Konfigurieren und Parametrieren von Hardware-Komponenten innerhalb eines S7-Projekts.

Interpolator

Logische Einheit des → NC, die nach Angaben von Zielpositionen im Teileprogramm Zwischenwerte für die in den einzelnen Achsen zu fahrenden Bewegungen bestimmt.

Interpolatorische Kompensation

Über interpolatorische Kompensationen wie → Spindelsteigungsfehler-, Durchhang-, Winkligkeits- und Temperaturkompensation werden mechanische Fehler der Maschine kompensiert.

Interruptroutine

Interruptroutinen sind spezielle → Unterprogramme, die durch Ereignisse (externe Signale) vom Bearbeitungsprozess gestartet werden können. Ein in Abarbeitung befindlicher Teileprogrammsatz wird abgebrochen, die Unterbrechungsposition der Achsen wird automatisch gespeichert.

JOG

Betriebsart der Steuerung (Einrichtebetrieb): In der Betriebsart JOG kann die Maschine eingerichtet werden. Einzelne Achsen und Spindeln können über die Richtungstasten im Tippbetrieb verfahren werden. Weitere Funktionen in der Betriebsart JOG sind das → Referenzpunktfahren, → Repos sowie → Preset (Istwert setzen).

Kanal

Ein Kanal ist dadurch gekennzeichnet, dass er unabhängig von anderen Kanälen ein → Teileprogramm abarbeiten kann. Ein Kanal steuert exklusiv die ihm zugeordneten Achsen und Spindeln. Teileprogrammabläufe verschiedener Kanäle können durch → Synchronisation koordiniert werden.

Kettenmaß

Auch Inkrementmaß: Angabe eines Bewegungsziels einer Achse durch eine zu verfahrenende Wegstrecke und Richtung bezogen auf einen bereits erreichten Punkt. Siehe → Absolutmaß.

Kompensationsachse

Achse, deren Soll- oder Istwert durch den Kompensationswert modifiziert wird.

Kompensationstabelle

Tabelle von Stützpunkten. Sie liefert für ausgewählte Positionen der Basisachse die Kompensationswerte der Kompensationsachse.

Kompensationswert

Differenz zwischen der durch den Messgeber gemessenen Achsposition und der gewünschten, programmierten Achsposition.

Kontur

Umriss des → Werkstücks

Konturüberwachung

Als Maß für die Konturtreue wird der Schleppfehler innerhalb eines definierbaren Toleranzbandes überwacht. Ein unzulässig hoher Schleppfehler kann sich z. B. durch Überlastung des Antriebs ergeben. In diesem Fall kommt es zu einem Alarm und die Achsen werden stillgesetzt.

Koordinatensystem

Siehe → Maschinenkoordinatensystem, → Werkstückkoordinatensystem

Korrekturspeicher

Datenbereich in der Steuerung, in dem Werkzeugkorrekturdaten hinterlegt sind.

Kreisinterpolation

Das → Werkzeug soll zwischen festgelegten Punkten der Kontur mit einem gegebenen Vorschub auf einem Kreis fahren und dabei das Werkstück bearbeiten.

Krümmung

Die Krümmung k einer Kontur ist das Inverse des Radius r des anschmiegenden Kreises in einem Konturpunkt ($k = 1/r$).

KÜ

Übersetzungsverhältnis

KV

Kreisverstärkungsfaktor, regelungstechnische Größe eines Regelkreises

Ladespeicher

Der Ladespeicher ist bei der CPU 314 der → SPS gleich dem → Arbeitsspeicher.

Linearachse

Die Linearachse ist eine Achse, welche im Gegensatz zur Rundachse eine Gerade beschreibt.

Look Ahead

Mit der Funktion **Look Ahead** wird durch das "Vorausschauen" über eine parametrierbare Anzahl von Verfahrssätzen ein Optimum an Bearbeitungsgeschwindigkeit erzielt.

Losekompensation

Ausgleich einer mechanischen Maschinenlose, z. B. Umkehrlose bei Kugelrollspindeln. Für jede Achse kann die Losekompensation getrennt eingegeben werden.

Makrotechnik

Zusammenfassung einer Menge von Anweisungen unter einem Bezeichner. Der Bezeichner repräsentiert im Programm die Menge der zusammengefassten Anweisungen.

Maschinenachsen

In der Werkzeugmaschine physikalisch existierende Achsen.

Maschinenfestpunkt

Durch die Werkzeugmaschine eindeutig definierter Punkt, z. B. Maschinen-Referenzpunkt.

Maschinenkoordinatensystem

Koordinatensystem, das auf die Achsen der Werkzeugmaschine bezogen ist.

Maschinennullpunkt

Fester Punkt der Werkzeugmaschine, auf den sich alle (abgeleiteten) Messsysteme zurückführen lassen.

Maschinensteuertafel

Bedientafel der Werkzeugmaschine mit den Bedienelementen Tasten, Drehschalter usw. und einfachen Anzeigeelementen wie LEDs. Sie dient der unmittelbaren Beeinflussung der Werkzeugmaschine über die PLC.

Maßangabe metrisch und inch

Im Bearbeitungsprogramm können Positions- und Steigungswerte in inch programmiert werden. Unabhängig von der programmierbaren Maßangabe (G70/G71) wird die Steuerung auf ein Grundsystm eingestellt.

Masse

Als Masse gilt die Gesamtheit aller untereinander verbundenen inaktiven Teile eines Betriebsmittels, die auch im Fehlerfall keine gefährliche Berührungsspannung annehmen können.

MDA

Betriebsart der Steuerung: Manual Data Automatic. In der Betriebsart MDA können einzelne Programmsätze oder Satzfolgen ohne Bezug auf ein Haupt- oder Unterprogramm eingegeben und anschließend über die Taste NC-Start sofort ausgeführt werden.

Meldungen

Alle im Teileprogramm programmierten Meldungen und vom System erkannte → Alarmer werden auf der Bedientafel im Klartext mit Datum und Uhrzeit und dem entsprechenden Symbol für das Löschkriterium angezeigt. Die Anzeige erfolgt getrennt nach Alarmen und Meldungen.

Metrisches Messsystem

Genormtes System von Einheiten: für Längen z. B. mm (Millimeter), m (Meter).

NC

Numerical Control Komponente der → CNC, die → Teileprogramme abarbeitet und die Bewegungsvorgänge der Werkzeugmaschine koordiniert.

Nebensatz

Durch "N" eingeleiteter Satz mit Informationen für einen Arbeitsschritt, z. B. eine Positionsangabe.

Netz

Ein Netz ist die Verbindung von mehreren S7-300 und weiteren Endgeräten, z. B. einem PG, über → Verbindungskabel. Über das Netz erfolgt ein Datenaustausch zwischen den angeschlossenen Geräten.

NRK

Numeric Robotic Kernel (Betriebssystem des → NC)

Nullpunktverschiebung

Vorgabe eines neuen Bezugspunkts für ein Koordinatensystem durch Bezug auf einen bestehenden Nullpunkt und ein → Frame.

1. Einstellbar
Es steht eine projektierbare Anzahl von einstellbaren Nullpunktverschiebungen für jede CNC-Achse zur Verfügung. Die über G-Befehle anwählbaren Verschiebungen sind alternativ wirksam.
2. Extern
Zusätzlich zu allen Verschiebungen, die die Lage des Werkstücknullpunkts festlegen, kann eine externe Nullpunktverschiebung durch Handrad (DRF-Verschiebung) oder von der PLC überlagert werden.
3. Programmierbar
Mit der Anweisung `TRANS` sind für alle Bahn- und Positionierachsen Nullpunktverschiebungen programmierbar.

NURBS

Die steuerungsinterne Bewegungsführung und Bahninterpolation wird auf Basis von NURBS (**N**on **U**niform **R**ational **B**-**S**plines) durchgeführt. Damit steht steuerungsintern für alle Interpolationen ein einheitliches Verfahren zur Verfügung.

OEM

Für Maschinenhersteller, die ihre eigene Bedienoberfläche erstellen oder technologiespezifische Funktionen in die Steuerung einbringen wollen, sind Freiräume für individuelle Lösungen (OEM-Applikationen) vorgesehen.

Orientierter Spindelhalt

Halt der Werkstückspindel in vorgegebener Winkellage, z. B. um an bestimmter Stelle eine Zusatzbearbeitung vorzunehmen.

Override

Manuelle bzw. programmierbare Eingriffsmöglichkeit, die es dem Bediener gestattet, programmierte Vorschübe oder Drehzahlen zu überlagern, um sie einem bestimmten Werkstück oder Werkstoff anzupassen.

Peripheriebaugruppe

Peripheriebaugruppen stellen die Verbindung zwischen CPU und Prozess her.

Peripheriebaugruppen sind:

- → Digital-Ein-/Ausgabebaugruppen
- → Analog-Ein-/Ausgabebaugruppen
- → Simulatorbaugruppen

PLC

Programmable Logic Control: → Speicherprogrammierbare Steuerung. Komponente der → NC: Anpass-Steuerung zur Bearbeitung der Kontroll-Logik der Werkzeugmaschine.

PLC-Programmierung

Die PLC wird mit der Software **STEP 7** programmiert. Die Programmiersoftware STEP 7 basiert auf dem Standardbetriebssystem **WINDOWS** und enthält die Funktionen der STEP 5 - Programmierung mit innovativen Weiterentwicklungen.

PLC-Programmspeicher

SINUMERIK 840D sl: Im PLC-Anwenderspeicher werden das PLC-Anwenderprogramm und die Anwenderdaten gemeinsam mit dem PLC-Grundprogramm abgelegt.

Polarkoordinaten

Koordinatensystem, das die Lage eines Punktes in einer Ebene durch seinen Abstand vom Nullpunkt und den Winkel festlegt, den der Radiusvektor mit einer festgelegten Achse bildet.

Polynom-Interpolation

Mit der Polynom-Interpolation können die unterschiedlichsten Kurvenverläufe erzeugt werden, wie **Gerade-, Parabel-, Potenzfunktionen** (SINUMERIK 840D sl).

Positionierachse

Achse, die eine Hilfsbewegung an einer Werkzeugmaschine ausführt. (z. B. Werkzeugmagazin, Palettentransport). Positionierachsen sind Achsen, die nicht mit den → Bahnachsen interpolieren.

Programmbaustein

Programmbausteine enthalten die Haupt- und Unterprogramme der → Teileprogramme.

Programmebene

Ein im Kanal gestartetes Teileprogramm läuft als → Hauptprogramm auf Programmebene 0 (Hauptprogramm-Ebene). Jedes im Hauptprogramm aufgerufene Teileprogramm läuft als → Unterprogramm auf einer eigenen Programmebene 1 ... n.

Programmierbare Arbeitsfeldbegrenzung

Begrenzung des Bewegungsraumes des Werkzeugs auf einen durch programmierte Begrenzungen definierten Raum.

Programmierbare Frames

Mit programmierbaren → Frames können dynamisch im Zuge der Teileprogramm-Abarbeitung neue Koordinatensystem-Ausgangspunkte definiert werden. Es wird unterschieden nach absoluter Festlegung anhand eines neuen Frames und additiver Festlegung unter Bezug auf einen bestehenden Ausgangspunkt.

Programmierschlüssel

Zeichen und Zeichenfolgen, die in der Programmiersprache für → Teileprogramme eine festgelegte Bedeutung haben.

Pufferbatterie

Die Pufferbatterie gewährleistet, dass das → Anwenderprogramm in der → CPU netzausfallsicher hinterlegt ist und festgelegte Datenbereiche und Merker, Zeiten und Zähler remanent gehalten werden.

Quadrantenfehlerkompensation

Konturfehler an Quadrantenübergängen, die durch wechselnde Reibverhältnisse an Führungsbahnen entstehen, sind mit der Quadrantenfehlerkompensation weitgehend eliminierbar. Die Parametrierung der Quadrantenfehlerkompensation erfolgt durch einen Kreisformtest.

Referenzpunkt

Punkt der Werkzeugmaschine, auf den sich das Messsystem der → Maschinenachsen bezieht.

Rohteil

Teil, mit dem die Bearbeitung eines Werkstücks begonnen wird.

Rotation

Komponente eines → Frames, die eine Drehung des Koordinatensystems um einen bestimmten Winkel definiert.

R-Parameter

Rechenparameter, kann vom Programmierer des → Teileprogramms für beliebige Zwecke im Programm gesetzt oder abgefragt werden.

Rundachse

Rundachsen bewirken eine Werkstück- oder Werkzeugdrehung in eine vorgegebene Winkellage.

Rundungsachse

Rundungsachsen bewirken eine Werkstück- oder Werkzeugdrehung in eine einem Teilungsraster entsprechende Winkellage. Beim Erreichen eines Rasters ist die Rundungsachse "in Position".

Satzsuchlauf

Zum Austesten von Teileprogrammen oder nach einem Abbruch der Bearbeitung kann über die Funktion "Satzsuchlauf" eine beliebige Stelle im Teileprogramm angewählt werden, an der die Bearbeitung gestartet oder fortgesetzt werden soll.

Schlüsselschalter

Der Schlüsselschalter auf der → Maschinensteuertafel besitzt 4 Stellungen, die vom Betriebssystem der Steuerung mit Funktionen belegt sind. Zum Schlüsselschalter gehören drei verschiedenfarbige Schlüssel, die in den angegebenen Stellungen abgezogen werden können.

Schlüsselwörter

Wörter mit festgelegter Schreibweise, die in der Programmiersprache für → Teileprogramme eine definierte Bedeutung haben.

Schneidenradiuskorrektur

Bei der Programmierung einer Kontur wird von einem spitzen Werkzeug ausgegangen. Da dies in der Praxis nicht realisierbar ist, wird der Krümmungsradius des eingesetzten Werkzeugs der Steuerung angegeben und von dieser berücksichtigt. Dabei wird der Krümmungsmittelpunkt um den Krümmungsradius verschoben äquidistant um die Kontur geführt.

Schnellabheben von der Kontur

Beim Eintreffen eines Interrupts kann über das CNC-Bearbeitungsprogramm eine Bewegung eingeleitet werden, die ein schnelles Abheben des Werkzeugs von der gerade bearbeiteten Werkstückkontur ermöglicht. Zusätzlich kann der Rückzugwinkel und der Betrag des Weges parametrisiert werden. Nach dem Schnellabheben kann zusätzlich eine Interruptroutine ausgeführt werden.

Schnelle digitale Ein-/Ausgänge

Über die digitalen Eingänge können z. B. schnelle CNC-Programmroutinen (Interruptroutinen) gestartet werden. Über die digitalen CNC-Ausgänge können schnelle, programmgesteuerte Schalfunktionen ausgelöst werden.

Schrägenbearbeitung

Bohr- und Fräsbearbeitungen an Werkstückflächen, die nicht in den Koordinatenebenen der Maschine liegen, können mit Unterstützung der Funktion "Schrägenbearbeitung" komfortabel ausgeführt werden.

Schraubenlinien-Interpolation

Die Schraubenlinien-Interpolation eignet sich besonders zum einfachen Herstellen von Innen- oder Außengewinden mit Formfräsern und zum Fräsen von Schmiernuten.

Dabei setzt sich die Schraubenlinie aus zwei Bewegungen zusammen:

- Kreisbewegung in einer Ebene
- Linearbewegung senkrecht zu dieser Ebene

Schrittmaß

Verfahrweglängenangabe über Inkrementanzahl (Schrittmaß). Inkrementanzahl kann als → Settingdatum hinterlegt sein bzw. durch entsprechend beschriftete Tasten 10, 100, 1000, 10000 gewählt werden.

Schutzraum

Dreidimensionaler Raum innerhalb des → Arbeitsraumes, in den die Werkzeugspitze nicht hineinreichen darf.

Settingdaten

Daten, die Eigenschaften der Werkzeugmaschine auf durch die Systemsoftware definierte Weise der NC-Steuerung mitteilen.

Sicherheitsfunktionen

Die Steuerung enthält ständig aktive Überwachungen, die Störungen in der → CNC, der Anpass-Steuerung (→ PLC) und der Maschine so frühzeitig erkennen, dass Schäden an Werkstück, Werkzeug oder Maschine weitgehend ausgeschlossen werden. Im Störfall wird der Bearbeitungsablauf unterbrochen und die Antriebe werden stillgesetzt, die Störungsursache gespeichert und als Alarm angezeigt. Gleichzeitig wird der PLC mitgeteilt, dass ein CNC-Alarm ansteht.

Skalierung

Komponente eines → Frames, die achsspezifische Maßstabsveränderungen bewirkt.

Softkey

Taste, deren Beschriftung durch ein Feld im Bildschirm repräsentiert wird, das sich dynamisch der aktuellen Bediensituation anpasst. Die frei belegbaren Funktionstasten (Softkeys) werden softwaremäßig definierten Funktionen zugeordnet.

Software-Endschalter

Software-Endschalter begrenzen den Verfahrbereich einer Achse und verhindern ein Auffahren des Schlittens auf die Hardware-Endschalter. Je Achse sind 2 Wertepaare vorgebar, die getrennt über die → PLC aktiviert werden können.

Speicherprogrammierbare Steuerung

Speicherprogrammierbare Steuerungen (SPS) sind elektronische Steuerungen, deren Funktion als Programm im Steuerungsgerät gespeichert ist. Aufbau und Verdrahtung des Gerätes hängen also nicht von der Funktion der Steuerung ab. Die speicherprogrammierbare Steuerung hat die Struktur eines Rechners; sie besteht aus CPU (Zentralbaugruppe) mit Speicher, Ein-/Ausgabebaugruppen und internem Bus-System. Die Peripherie und die Programmiersprache sind auf die Belange der Steuerungstechnik ausgerichtet.

Spiegelung

Bei Spiegelung werden die Vorzeichen der Koordinatenwerte einer Kontur bezüglich einer Achse vertauscht. Es kann bezüglich mehrerer Achsen zugleich gespiegelt werden.

Spindelsteigungsfehler-Kompensation

Ausgleich mechanischer Ungenauigkeiten einer am Vorschub beteiligten Kugelrollspindel durch die Steuerung anhand von hinterlegten Messwerten der Abweichungen.

Spline-Interpolation

Mit der Spline-Interpolation kann die Steuerung aus nur wenigen vorgegebenen Stützpunkten einer Sollkontur einen glatten Kurvenverlauf erzeugen.

Standardzyklen

Für häufig wiederkehrende Bearbeitungsaufgaben stehen Standardzyklen zur Verfügung:

- Für die Technologie Bohren/Fräsen
- Für die Technologie Drehen

Im Bedienbereich "Programm" werden unter dem Menü "Zyklenunterstützung" die zur Verfügung stehenden Zyklen aufgelistet. Nach Anwahl des gewünschten Bearbeitungszyklus werden die notwendigen Parameter für die Wertzuweisung im Klartext angezeigt.

Synchronachsen

Synchronachsen benötigen für ihren Weg die gleiche Zeit wie die Geometrieachsen für ihren Bahnweg.

Synchronaktionen

1. Hilfsfunktionsausgabe
Während der Werkstückbearbeitung können aus dem CNC-Programm heraus technologische Funktionen (→ Hilfsfunktionen) an die PLC ausgegeben werden. Über diese Hilfsfunktionen werden beispielsweise Zusatzeinrichtungen der Werkzeugmaschine gesteuert, wie Pinole, Greifer, Spannfutter etc.
2. Schnelle Hilfsfunktionsausgabe
Für zeitkritische Schaltfunktionen können die Quittierungszeiten für die → Hilfsfunktionen minimiert und unnötige Haltepunkte im Bearbeitungsprozess vermieden werden.

Synchronisation

Anweisungen in → Teileprogrammen zur Koordination der Abläufe in verschiedenen → Kanälen an bestimmten Bearbeitungsstellen.

Syntax

`$SC_IS_SD_MAX_PATH_JERK = Wert`

Systemspeicher

Der Systemspeicher ist ein Speicher in der CPU, in der folgende Daten abgelegt werden:

- Daten, die das Betriebssystem benötigt
- Die Operanden Zeiten, Zähler, Merker

Systemvariable

Ohne Zutun des Programmierers eines → Teileprogramms existierende Variable. Sie ist definiert durch einen Datentyp und dem Variablennamen, der durch das Zeichen **\$** eingeleitet wird. Siehe → Anwenderdefinierte Variable.

Teileprogramm

Folge von Anweisungen an die NC-Steuerung, die insgesamt die Erzeugung eines bestimmten → Werkstücks bewirken. Ebenso Vornahme einer bestimmten Bearbeitung an einem gegebenen → Rohteil.

Teileprogrammsatz

Teil eines → Teileprogramms, durch Line Feed abgegrenzt. Es werden → Hauptsätze und → Nebensätze unterschieden.

Teileprogrammverwaltung

Die Teileprogrammverwaltung kann nach → Werkstücken organisiert werden. Die Größe des Anwenderspeichers bestimmt die Anzahl der zu verwaltenden Programme und Daten. Jede

Datei (Programme und Daten) kann mit einem Namen von maximal 24 alphanumerischen Zeichen versehen werden.

Text-Editor

Siehe → Editor

TOA-Bereich

Der TOA-Bereich umfasst alle Werkzeug- und Magazindaten. Standardmäßig fällt der Bereich bzgl. der Reichweite der Daten mit dem Bereich → Kanal zusammen. Über Maschinendaten kann jedoch festgelegt werden, dass sich mehrere Kanäle eine → TOA-Einheit teilen, so dass diesen Kanälen dann gemeinsame WZV-Daten zur Verfügung stehen.

TOA-Einheit

Jeder → TOA-Bereich kann mehrere TOA-Einheiten enthalten. Die Anzahl der möglichen TOA-Einheiten wird über die maximale Anzahl aktiver → Kanäle begrenzt. Eine TOA-Einheit umfasst genau einen WZ-Daten-Baustein und einen Magazindaten-Baustein. Zusätzlich kann noch ein WZ-Trägerdaten-Baustein enthalten sein (optional).

Transformation

Additive oder absolute Nullpunktsverschiebung einer Achse.

Unterprogramm

Die Bezeichnung Unterprogramm stammt noch aus der Zeit, als Teileprogramm fest in → Haupt- und Unterprogramme unterteilt waren. Diese feste Einteilung besteht mit der heutigen SINUMERIK NC-Sprache nicht mehr. Prinzipiell kann jedes Teileprogramm oder jeder → Zyklus innerhalb eines anderen Teileprogramms als Unterprogramm aufgerufen werden. Es läuft dann in der nächsten → Programmebene (x+1) (Unterprogrammebene (x+1)) ab.

Urlöschen

Beim Urlöschen werden folgende Speicher der → CPU gelöscht:

- → Arbeitsspeicher
- Schreib-/Lesebereich des → Ladespeichers
- → Systemspeicher
- → Backup-Speicher

V.24

Serielle Schnittstelle für die Dateneingabe/-ausgabe. Über diese Schnittstelle können Bearbeitungsprogramme sowie Hersteller- und Anwenderdaten geladen und gesichert werden.

Variablendefinition

Eine Variablendefinition umfasst die Festlegung eines Datentyps und eines Variablennamens. Mit dem Variablennamen kann der Wert der Variablen angesprochen werden.

Verfahrbereich

Der maximal zulässige Verfahrbereich bei Linearachsen beträgt ± 9 Dekaden. Der absolute Wert ist abhängig von der gewählten Eingabe- und Lageregelfeinheit und dem Einheitensystem (inch oder metrisch).

Vorkoinzidenz

Satzwechsel bereits, wenn Bahnweg um ein vorgegebenes Delta der Endposition nahe gekommen ist.

Vorschub-Override

Der programmierten Geschwindigkeit wird die aktuelle Geschwindigkeitseinstellung über → Maschinensteuertafel oder von der → PLC überlagert (0-200%). Die Vorschubgeschwindigkeit kann zusätzlich im Bearbeitungsprogramm durch einen programmierbaren Prozentfaktor (1-200%) korrigiert werden.

Vorsteuerung, dynamisch

Ungenauigkeiten der → Kontur, bedingt durch Schleppfehler, lassen sich durch die dynamische, beschleunigungsabhängige Vorsteuerung nahezu eliminieren. Dadurch ergibt sich auch bei hohen → Bahngeschwindigkeiten eine hervorragende Bearbeitungsgenauigkeit. Die Vorsteuerung kann achsspezifisch über das → Teileprogramm an- und abgewählt werden.

Werkstück

Von der Werkzeugmaschine zu erstellendes/zum bearbeitendes Teil.

Werkstückkontur

Sollkontur des zu erstellenden/bearbeitenden → Werkstücks.

Werkstückkoordinatensystem

Das Werkstückkoordinatensystem hat seinen Ausgangspunkt im → Werkstücknullpunkt. Bei Programmierung im Werkstückkoordinatensystem beziehen sich Maße und Richtungen auf dieses System.

Werkstücknullpunkt

Der Werkstücknullpunkt bildet den Ausgangspunkt für das → Werkstückkoordinatensystem. Er ist durch Abstände zum → Maschinennullpunkt definiert.

Werkzeug

An der Werkzeugmaschine wirksames Teil, das die Bearbeitung bewirkt (z. B. Drehmeißel, Fräser, Bohrer, LASER-Strahl ...).

Werkzeugkorrektur

Berücksichtigung der Werkzeug-Abmessungen bei der Berechnung der Bahn.

Werkzeugradiuskorrektur

Um eine gewünschte → Werkstückkontur direkt programmieren zu können, muss die Steuerung unter Berücksichtigung des Radius des eingesetzten Werkzeugs eine äquidistante Bahn zur programmierten Kontur verfahren (G41/G42).

WinSCP

WinSCP ist ein frei verfügbares Open Source-Programm für Windows zum Transferieren von Dateien.

Zeitreziproker Vorschub

Anstelle der Vorschubgeschwindigkeit kann für die Achsbewegung auch die Zeit programmiert werden, die der Bahnweg eines Satzes benötigen soll (G93).

Zoll-Maßsystem

Maßsystem, das Entfernungen in "inch" und Bruchteilen davon definiert.

Zwischensätze

Verfahrbewegungen mit angewählter → Werkzeugkorrektur (G41/G42) dürfen durch eine begrenzte Anzahl Zwischensätze (Sätze ohne Achsbewegungen in der Korrektorebene) unterbrochen werden, wobei die Werkzeugkorrektur noch korrekt verrechnet werden kann. Die zulässige Anzahl Zwischensätze, die die Steuerung vorausliest, ist über Systemparameter einstellbar.

Zyklen

Geschützte Unterprogramme zur Ausführung von wiederholt auftretenden Bearbeitungsvorgängen am → Werkstück.

Index

\$

\$AA_ACC, 1467
\$AA_ATOL, 254
\$AA_FGREF, 1443
\$AA_FGROUP, 1443
\$AA_MOTEND, 1468
\$AA_S, 1300
\$AA_SCPAR, 1469
\$AA_VLFCT, 1393
\$AA_VMAXB, 1393
\$AA_VMAXM, 1393
\$AC_ACT_PROG_NET_TIME, 692
\$AC_ACTUAL_PARTS, 698, 702
\$AC_ASUP, 607
\$AC_AUXFU_EXT, 463
\$AC_AUXFU_M_EXT, 463
\$AC_AUXFU_M_STATE, 463
\$AC_AUXFU_M_TICK, 444
\$AC_AUXFU_M_VALUE, 463
\$AC_AUXFU_PREDEF_INDEX, 449, 462
\$AC_AUXFU_SPEC, 460, 463
\$AC_AUXFU_STATE, 463
\$AC_AUXFU_TYPE, 462
\$AC_AUXFU_VALUE, 463
\$AC_CTOL, 254
\$AC_CUT_INV, 1602
\$AC_CUTMOD, 1602
\$AC_CUTMOD_ANG, 1595, 1602
\$AC_CUTTING_TIME, 694
\$AC_CYCLE_TIME, 694
\$AC_DELAYFST, 590
\$AC_FGROUP_MASK, 1443
\$AC_OLD_PROG_NET_TIME, 692
\$AC_OLD_PROG_NET_TIME_COUNT, 693
\$AC_OPERATING_TIME, 694
\$AC_OTOL, 254
\$AC_PATHACC, 276, 288
\$AC_PATHJERK, 287, 288, 289
\$AC_PROG_NET_TIME_TRIGGER, 693
\$AC_REQUIRED_PARTS, 698, 702
\$AC_SGEAR, 1351, 1372
\$AC_SMAXACC, 1393
\$AC_SMAXACC_INFO, 1393
\$AC_SMAXVELO, 1393
\$AC_SMAXVELO_INFO, 1393
\$AC_SMINVELO, 1393
\$AC_SMINVELO_INFO, 1393
\$AC_SPECIAL_PARTS, 698, 702
\$AC_SPIND_STATE, 1393
\$AC_STOLF, 258
\$AC_TOTAL_PARTS, 698, 702
\$AN_AUXFU_LIST_ENDINDEX, 453
\$AN_POWERON_TIME, 691
\$AN_SETUP_TIME, 691
\$C_AUX_EXT, 671
\$C_AUX_IS_QUICK, 671
\$C_AUX_VALUE, 671
\$C_D, 672
\$C_D_PROG, 672
\$C_DL, 672
\$C_DL_PROG, 672
\$C_DUPLO, 672
\$C_DUPLO_PROG, 672
\$C_M, 671
\$C_M_PROG, 671
\$C_ME, 671
\$C_MTL, 668, 672
\$C_MTL_PROG, 668, 672
\$C_T, 671
\$C_T_PROG, 671
\$C_TCA, 672
\$C_TE, 672
\$C_THNO, 672
\$C_THNO_PROG, 672
\$C_TS, 672
\$C_TS_PROG, 672
\$P_AD, 1603
\$P_CHANNO, 582
\$P_CTOL, 255
\$P_CUT_INV, 1602
\$P_CUTMOD, 1602
\$P_CUTMOD_ANG, 1595, 1602
\$P_DELAYFST, 590
\$P_FGROUP_MASK, 1444
\$P_GEAR, 1372
\$P_GFRNUM, 788
\$P_IFRAME, 784
\$P_ISTEST, 496
\$P_OTOL, 255
\$P_PROG_EVENT, 582
\$P_REPINF, 601
\$P_SEARCH_S, 446, 509
\$P_SEARCH_SDIR, 446, 509
\$P_SEARCH_SGEAR, 446, 509, 1351
\$P_SEARCH_SMODE, 509
\$P_SEARCH_SPOS, 446, 509

\$P_SEARCH_SPOSMODE, 446, 509
\$P_SGEAR, 1351
\$P_SIM, 502
\$P_STOLF, 258
\$P_SUB_AUTOGEAR, 681
\$P_SUB_AXFCT, 680, 681
\$P_SUB_CA, 681
\$P_SUB_GEAR, 680
\$P_SUB_LA, 681
\$P_SUB_M19, 681
\$P_SUB_SPOS, 681
\$P_SUB_SPOSA, 681
\$P_SUB_SPOSIT, 681
\$P_SUB_SPOSMODE, 681
\$P_SUB_STAT, 670, 672
\$P_TOOLENV, 1647
\$P_TOOLENVN, 1647
\$P_UIFR, 784
\$P_UIFRNUM, 784
\$P_WORKAREA_CS_COORD_SYSTEM, 118
\$PA_ATOL, 255
\$PA_FGREF, 1443
\$PA_FGROUP, 1444
\$SC_IS_SD_MAX_PATH_ACCEL, 275
\$SC_IS_SD_MAX_PATH_JERK, 286
\$SC_PA_ACTIV_IMMED, 145, 157
\$SC_PA_CENT_ABS, 146
\$SC_PA_CENT_ORD, 146
\$SC_PA_CONT_ABS, 146
\$SC_PA_CONT_NUM, 145
\$SC_PA_CONT_ORD, 146
\$SC_PA_CONT_TYP, 146
\$SC_PA_LIM_3DIM, 145
\$SC_PA_MINUS_LIM, 145
\$SC_PA_ORI, 145
\$SC_PA_PLUS_LIM, 145
\$SC_PA_T_W, 145
\$SC_SD_MAX_PATH_ACCEL, 274
\$SC_SD_MAX_PATH_JERK, 286
\$SN_PA_ACTIV_IMMED, 145, 157
\$SN_PA_CENT_ABS, 146
\$SN_PA_CENT_ORD, 146
\$SN_PA_CONT_ABS, 146
\$SN_PA_CONT_NUM, 145
\$SN_PA_CONT_ORD, 146
\$SN_PA_CONT_TYP, 146
\$SN_PA_LIM_3DIM, 145
\$SN_PA_MINUS_LIM, 145
\$SN_PA_ORI, 145
\$SN_PA_PLUS_LIM, 145
\$SN_PA_T_W, 145
\$TC_DP1, 1592

\$TC_DP10, 1593
\$TC_DP11, 1592
\$TC_DP2, 1592, 1596
\$TC_DP24, 1593
\$TC_DPCE[t,d], 1485
\$TC_DPNT, 1417
\$TC_DPV, 1608
\$TC_DPV3, 1608
\$TC_DPV4, 1608
\$TC_DPV5, 1608
\$VA_ABSOLUTE_ENC_DELTA_INIT, 108
\$VA_ENC_ZERO_MON_ERR_CNT, 105, 108
\$VA_TORQUE_AT_LIMIT, 335
\$VC_SGEAR, 1351

—

_N_STRTLK, 494
_N_STRTUL, 494

A

Ablage von Winkeln in den
Werkzeugträgerdaten, 1560
ACC, 1380, 1467
Access MyMachine (AMM), 1226
Achse/Spindel steht, 53
Achsensperre, 45, 1446
Achskonfiguration, 728
Achsüberwachungen
 Drehzahlsollwert, 99
 Istgeschwindigkeit, 100
 Randbedingungen, 139
 Schleppabstand, 84
 Stillstand, 88
ACN, 1379
ACP, 1379
Adaptermaß, 1511
ADDFRAME, 826
AG_SEND, AG_RECV, 922
Aktions-Einzelsatz, 497
Aktionssätze, 506
Aktivierung
 von Maschinensteuertafel, Bedienhandgerät, 932
Alarm bei Vorlaufstopp, 1549
ALF, 603, 1430
Anlauf und Synchronisation NCK-PLC, 914
Anpassfaktor
 der Bahndynamik, 223
Anpassungen der Motor/Last-Verhältnisse, 366
Antrieb ready, 41

Antriebe im zyklischenBetrieb, 41
 Antriebtest Fahr Anforderung, 53
 Antriebtest Fahr freigabe, 44
 Anwahl der Schneide bei WZ-Wechsel, 1479
 ANY, 1139
 ANY im FB, 1141
 ANY im FC, 1139
 Anzeigefinheit, 344
 Anzeigesatz, Aufbau (DIN), 631
 Arbeitsfeldbegrenzung, 112
 im BKS, 114
 im WKS/ENS, 116
 Arbeitsfeldbegrenzungsgruppe, 117
 ASUP
 Aktivierung, 593
 bei Anwenderalarmen, 608
 interne, 606
 Priorität, 603
 Reorganisation, 594
 SERUPRO-Ende, 454
 ATOL, 251
 ATRANS, 717
 Ausblendebeinen, 625
 Ausgabe
 auf externes Gerät/Datei, 653
 -sequenz, 450
 -verhalten einer Hilfsfunktion, 422
 -zähler, 450
 Autarke Einzelachsvorgänge, 548
 AUXFUDEL, 455
 AUXFUDELG, 455
 AUXFUMSEQ, 444
 AUXFUSYNC, 454

B

BAG, 915
 Bahn
 -geschwindigkeit, Maximale, 1415
 -vorschub F, 1414
 Bahnachsen, 723
 Bahnsteuerbetrieb, 192
 Impliziter, 195
 Bahnvorschub, 342
 Ballige Gewinde, 1434
 Basis-Anzeige
 Größe des Anzeigebuffers, 629
 Basiskoordinatensystem (BKS), 715, 748
 Basismaß, 1511
 Basis-Satzanzeige
 aktivieren, 629
 konfigurieren, 629

Bearbeitung in Richtung der
 Werkzeugorientierung, 1606
 Bearbeitungszeit, 694
 Benutzerdefinierten ASUP
 nach SERUPRO-Vorgang, 519
 Beschreibung einer Drehung, 1568
 Besonderheiten im Zielsatz
 STOPPRE-Satz, 541
 Betriebsart
 AUTOMATIC, 483
 JOG, 484
 JOG in AUTOMATIC, 483
 MDA, 484
 Betriebsarten
 der Betriebsartengruppe, 483
 Prioritäten, 484
 -übergreifende Synchronaktionen, 484
 -Überwachungen, 488
 -Verriegelungen, 488
 -wechsel, 483, 489
 Betriebsartengruppe (BAG), 480
 Betriebsartenwechsel
 von/nach Betriebsarten AUTOMATIK, JOG,
 MDA, 489
 Betriebszustände, 485
 Bewegungsendekriterium
 bei Einzelachsen, 1468
 Bewertung einzelner Verschleißkomponenten, 1640
 Bezugsachse
 für G96 / G961 / G962, 871
 Bildschirm dunkel steuern, 42
 Bildschirm hell, 42
 BLSYNC, 603
 BRISK, 269
 BRISKA, 270

C

Cancelalarme, 43
 CFC, 1415
 CFCIN, 1415
 CFINE, 717
 CFTCP, 1415
 CLRINT, 604
 CNC-Sperrfunktion, 1226
 COMPCAD, 237, 247
 COMPCURV, 236, 247
 COMPOF, 247
 COMPON, 236, 247
 COMPSURF, 237, 247
 CPROT, 154
 CPROTDEF, 150

CTOL, 251
CTRANS, 717

D

D/DL-Funktionsersetzung, 668
Darstellungsweise der Programmanzeige, 628
Datenaustausch
 mit Bedientafel, 896
Datenkanal, schneller, 61
Datenschnittstelle, 899

DB10

DBB4-7, 866
DBB71, 352
DBX103.0, 41
DBX104.3, 934
DBX104.4, 934
DBX104.7, 41, 918
DBX106.1, 865, 867
DBX107.7, 352
DBX108.3, 41
DBX108.5, 41
DBX108.6, 41
DBX108.7, 41
DBX109.0, 42
DBX109.6, 42
DBX109.7, 42
DBX56.1, 50, 865, 866, 868
DBX56.2, 865, 866
DBX56.4, 68
DBX56.5, 68
DBX56.6, 68
DBX56.7, 68
DBX92.0, 917
DBX92.1, 917
DBX92.2, 917
DBX92.4, 918
DBX92.5, 918
DBX92.6, 918

DB11

DBX0.0, 484, 486
DBX0.1, 484, 1244, 1246
DBX0.2, 484, 1244, 1246
DBX0.4, 490
DBX0.5, 482
DBX0.6, 482
DBX0.7, 482, 561, 866, 1246, 1247, 1248
DBX1.0 - DBX1.2, 485
DBX1.2, 1244, 1246
DBX26.4, 487
DBX26.5, 487
DBX4.0 - DBX4.2, 484

DBX46.4, 487
DBX46.5, 487
DBX5.0 - DBX5.2, 485
DBX6.0, 485, 487
DBX6.1, 485
DBX6.2, 485
DBX6.3, 483, 865, 867
DBX6.4, 487
DBX6.5, 487
DBX6.7, 483
DBX7.0, 487
DBX7.0 - DBX7.2, 485

DB1600, 1153

DB19

DBB13, 43
DBB16, 43
DBB17, 43, 44
DBB26, 44
DBB27, 44
DBB33, 945
DBB36, 945
DBX 0.3, 43
DBX 0.4, 43
DBX0.0, 42
DBX0.1, 42, 43
DBX0.2, 42, 43
DBX0.7, 43
DBX20.3, 43
DBX20.4, 43
DBX32.0 - .5, 945
DBX32.6, 945
DBX32.7, 945

DB1900

DBB5011, 1224
DBB5021, 1224
DBB5022, 1224
DBX5021.0-.5, 1224
DBX5021.6, 1224
DBX5021.7, 1224

DB21

DBX24.4, 431
DBX378.1, 601

DB21, ...

D35.0, 559
D35.5, 559
DBB116 - DBB136, 438
DBB140 - DBB190, 438
DBB194, 1381
DBB194 - DBB206, 438
DBB202, 1381
DBB376, 577
DBB4, 1447

DBB5, 1447
 DBB58 - DBB67, 438
 DBB68 - DBB112, 438
 DBB68ff., 1381
 DBX0.4, 497
 DBX0.6, 500, 1452
 DBX1.0, 1247
 DBX1.1, 171
 DBX1.6, 505, 508
 DBX1.7, 495
 DBX10.0 - DBX11.1, 147
 DBX12.3, 1446
 DBX16.3, 1446
 DBX2.0, 501, 559, 586, 587
 DBX2.0 - 7, 626
 DBX20.3, 1446
 DBX24.6, 500, 1452
 DBX25.7, 495
 DBX26.0, 501
 DBX272.0 - DBX273.1, 147
 DBX274.0 - DBX275.1, 147
 DBX276.0 - DBX277.1, 149, 169
 DBX278.0 - DBX279.1, 149, 169
 DBX29.0, 1460
 DBX29.1, 1460
 DBX29.2, 1460
 DBX29.3, 1460
 DBX30.5, 431
 DBX31.0 - 2, 525
 DBX31.0 - DBX31.2, 526
 DBX31.0-31.2, 531
 DBX31.4, 524, 526, 527
 DBX31.6 - 7, 626
 DBX318.0, 595
 DBX318.5, 431
 DBX318.6, 1453
 DBX319.0, 527
 DBX319.1 - DBX319.3, 526, 527, 528
 DBX319.5, 527, 529
 DBX32.3, 505
 DBX32.4, 505, 506
 DBX32.6, 447, 505
 DBX33.0, 1247
 DBX33.4, 505
 DBX33.7, 495
 DBX35.0, 558, 562
 DBX35.0 - DBX35.4, 601
 DBX35.1, 562
 DBX35.2, 562
 DBX35.3, 498, 562
 DBX35.4, 562
 DBX35.5, 558, 563
 DBX35.5 - DBX35.7, 601
 DBX35.6, 563
 DBX35.7, 561, 563
 DBX36.2, 1245, 1247
 DBX36.6, 42
 DBX36.7, 42, 508
 DBX384.0, 566
 DBX39.0, 148
 DBX4.3, 1446
 DBX6.0, 1445
 DBX6.1, 1305, 1345
 DBX6.2, 44
 DBX6.6, 1447
 DBX6.7, 1447
 DBX7.1, 495, 560
 DBX7.2, 559, 586
 DBX7.3, 559, 586
 DBX7.4, 559, 586, 1347, 1392
 DBX7.5, 493
 DBX7.7, 561, 867, 1246, 1247, 1248
 DBX8.0 - DBX9.1, 147
 DB21, ... DB32.6, 508
 DB21, ... DBB4, 94, 95
 DB21, ... DBX0.4, 506, 600
 DB21, ... DBX36.6, 508
 DB21, ... DBX6.1, 599
 DB21, ...
 DBX378.0, 594
 DB21, ...
 DBX317.1, 698, 702
 DB31
 DBX84.0, 1386
 DB31, ...
 DBB0, 1449
 DBB1.5, 1285
 DBB1.6, 1285
 DBB19, 1449
 DBB2.1, 1285
 DBB60.4, 1269, 1279, 1282
 DBB60.5, 1269, 1279, 1282
 DBB68ff., 1381
 DBX1.0, 44, 53
 DBX1.1, 323
 DBX1.2, 319
 DBX1.3, 45, 319, 1392
 DBX1.4, 45, 48, 51, 98, 1273
 DBX1.5, 49, 103, 122, 361, 1336
 DBX1.6, 49, 103, 122, 361, 1336
 DBX1.7, 1448, 1449
 DBX10.0, 522, 525, 526, 527, 531
 DBX102.5, 122, 125
 DBX102.6, 122, 125

DBX12.0, 109
DBX12.1, 109
DBX12.2, 110
DBX12.3, 110
DBX12.7, 1254, 1256
DBX16.0 - 2, 1358
DBX16.0 - DBX16.2, 1301, 1341, 1346, 1347, 1348, 1352, 1364
DBX16.3, 1297, 1301, 1341, 1346, 1347, 1348, 1357, 1359, 1361, 1364, 1368
DBX16.4, 1336
DBX16.5, 1336
DBX16.7, 1392
DBX17.0, 1367
DBX17.4 - DBX17.5, 1363
DBX18.4, 1363
DBX18.5, 1301, 1341, 1363, 1392
DBX2.1, 45, 48, 49, 50, 122, 124, 319, 1273, 1363
DBX2.2, 52, 523, 1300, 1308, 1347, 1392
DBX2.3, 98
DBX2.4 - DBX2.7, 1261
DBX20.1, 1348
DBX21.0-4, 54, 55
DBX21.5, 54
DBX21.6, 54
DBX21.7, 50, 55, 323
DBX28.7, 492
DBX3.1, 318, 322, 323
DBX3.2, 1460
DBX3.3, 1460
DBX3.4, 1460
DBX3.5, 1460
DBX3.7, 548
DBX30.0, 1385
DBX30.1, 1385
DBX30.2, 1385
DBX30.3, 1385
DBX30.4, 1385
DBX39.0, 169, 173
DBX4.3, 1361, 1392
DBX4.6, 1244
DBX4.7, 1244
DBX60.2, 104, 1397
DBX60.3, 104, 1397
DBX60.4, 48, 122, 1245, 1246, 1247, 1248, 1261, 1268, 1271, 1273, 1274, 1397
DBX60.4 - DBX60.5, 1353, 1358, 1362, 1363, 1364
DBX60.5, 48, 122, 867, 1245, 1246, 1247, 1248, 1261, 1268, 1271, 1273, 1274, 1397
DBX60.6, 48, 88, 98, 1304, 1311, 1315, 1316, 1398, 1399
DBX60.7, 48, 88, 98, 1311, 1315, 1316, 1368, 1398
DBX61.0, 44, 53
DBX61.3, 45, 46, 53, 1272, 1273
DBX61.4, 52, 53, 1273, 1300, 1301, 1306, 1341, 1390
DBX61.5, 50, 53, 1299, 1357, 1358, 1359, 1363
DBX61.6, 50, 53
DBX61.7, 54
DBX62.4, 318, 322, 323
DBX62.5, 323, 326
DBX64.6, 88, 442
DBX64.7, 88, 442
DBX69.0, 54, 76
DBX69.0 - DBX69.2, 1352
DBX69.1, 54, 76
DBX69.2, 54, 76
DBX7.7, 1392
DBX70.0, 527, 528, 529
DBX70.1, 527, 528, 529
DBX70.2, 527, 528
DBX71.4, 1336
DBX71.5, 1337
DBX72.0, 526
DBX76.0, 54
DBX76.4, 527, 530
DBX82.0 - DBX82.2, 1341, 1343, 1344, 1346, 1356, 1357, 1358, 1359, 1363, 1366
DBX82.3, 1341, 1343, 1344, 1357, 1358, 1359, 1363, 1366
DBX83.1, 1300, 1311, 1343, 1345, 1391, 1392
DBX83.2, 1343, 1345, 1391, 1392
DBX83.5, 1300, 1301, 1305, 1306, 1307, 1391
DBX84.5, 1363, 1368
DBX84.6, 1297, 1357, 1358, 1359, 1363, 1366, 1368
DBX84.7, 1299, 1341, 1364
DBX85.5, 1316
DBX9.0, 52, 54, 75
DBX9.1, 52, 54, 75
DBX9.2, 52, 54, 75
DBX9.3, 53
DBX92.1, 55
DBX93.5, 41, 50, 55
DBX93.6, 55, 56
DBX93.7, 55, 56
DBX94.0, 56
DBX94.1, 56
DBX94.2, 57
DBX94.3, 57

DBX94.4, 57
 DBX94.5, 57
 DBX94.6, 57
 DB31, ... DBX1.1, 319
 DB31, ... DBX1.4, 98
 DB31, ... DBX102.3, 89, 91
 DB31, ... DBX2.3, 89, 90
 DB31, ... DBX61.3, 91
 DB31, ... DBX62.5, 319
 DB31, ... DBX64.6, 91, 93, 95
 DB31, ...
 DBX9.3, 1469
 DB4500, 1154
 DB9000 - DB9063, 1154
 DC, 1379
 Debugmode für SBL2, 617
 Default-Kennwort, 67
 Dekodier-Einzelsatz, 497
 DELAYFSTOF, 588
 DELAYFSTON, 588
 DELTOOLENV, 1645
 D-Funktionen, 404, 1479
 DIACYCOFA, 878
 DIAM90, 878
 DIAM90/DIAM90A[AX], 873
 DIAM90A, 878
 DIAMCHAN, 878
 DIAMCHANA, 878
 DIAMCYCOF, 878
 DIAMCYCOF/DIACYCOFA[AX], 873
 DIAMOF, 877
 DIAMOFA, 878
 DIAMON, 877
 DIAMON/DIAMONA[AX], 872
 DIAMONA, 878
 DILF, 1430
 Direkttasten
 Adresse, 934
 Alarmer, 935
 OPs am Ethernet-Bus, 934
 DISABLE, 604
 DISC, 1537
 DITE, 1428
 DITS, 1428
 DL-Funktionen, 404
 D-Nummern
 Vergabe von freien ..., 1484
 D-Nummern-Struktur
 - flache (ohne WZV), 1494
 Drehzahl
 -vorsteuerung, 385
 Drehzahlbegrenzung bei G96, G961, 1421

Drehzahlregelkreis, 380
 Drehzahlregler aktiv, 53
 Drehzahlsollwertanpassung, 369
 Drehzahlsollwertausgabe, 369
 DRIVE, 302
 DRIVEA, 302
 Durchmesserprogrammierung
 achs-spezifisch, 878
 kanalspezifisch, 877
 Dynamik
 -anpassung, 222
 Dynamikanpassung, 381
 DYNFINISH, 230
 DYNNORM, 230
 DYNPOS, 230
 DYNROUGH, 230
 DYNSEMIFIN, 230

E

EES, 640
 Eilgang
 -korrektur, 1447
 Eingabefeinheit, 344
 Eingeschränkte Orientierung des
 Werkzeuträgers, 1560
 Einzelachsdynamik, 1465
 Einzelsatz
 -betrieb, 496
 SBL1, 616
 SBL2, 616
 SBL3, 616
 -unterdrückung, 618
 ENABLE, 604
 ENDLABEL, 568
 Endschalte-Überwachung, 109
 Ereignisgesteuerte Programmabläufe, 573
 Ersatznullmarke, 1283
 Ersetzungsunterprogramm, 665
 Ethernet
 -Ankopplung, 933
 EXTCLOSE, 653
 Externe Nullpunktverschiebung, 750
 Externe Programmspeicher, 635
 Externes Unterprogramm Abarbeiten, 635
 EXTOPEN, 653

F

FA, 1380, 1444
 FA-Funktionen, 405

Fahren auf Festanschlag, 315
 Abwahl, 323
 Anwahl, 318
 Festanschlag wird erreicht, 319
 Funktionsabbruch, 328
 Konturüberwachung, 328
 Positionierachsen, 328
 Satzsuchlauf, 324
 Überwachungsfenster, 320
 FB, 1464
 FB1 RUN_UP (Grundprogramm, Anlaufteil), 980
 FB10 Sicherheits-Relais, 1048
 FB11 Bremsentest, 1051
 FB2 GET (NC-Variable lesen), 989
 FB29 Diagnose (Signalrekorder und Datentrigger), 1057
 FB3 PUT (NC-Variable schreiben, 997
 FB4 PI_SERV (PI-Dienste), 1005
 Verfügbare PI-Dienste, 1009, 1010
 FB5 GETGUD (GUD-Variable lesen), 1034
 FB7 PI_SERV2 (PI Dienste), 1043
 FB9 MzuN (Bedieneinheitenumschaltung), 1044
 FC10 AL_MSG, 1080
 FC1005 AG_SEND, 1133
 FC1006 AG_RECV, 1134
 FC12 AUXFU, 1083
 FC13 BHGDisp, 1084
 FC17 YDelta, 1088
 FC18 SpinCtrl, 1092
 FC19 MCP_IFM, 1102
 FC2 GP_HP (Grundprogramm, zyklischer Teil), 1060
 FC21 Transfer, 1108
 FC22 TM_DIR, 1116
 FC24 MCP_IFM2, 1118
 FC25 MCP_IFT, 1122
 FC26 HPU_MCP, 1125
 FC3 GP_PRAL (Grundprogramm, alarmgesteuerter Teil), 1062
 FC5 GP_DIAG (Grundprogramm, Diagnose), 1064
 FC6 TM_TRANS2, 1066
 FC7 TM_REV, 1067
 FC8 TM_TRANS, 1071
 FC9 ASUP, 1078
 FCUB, 1461
 Feininterpolation, 380
 Feinverschiebung, 717
 Ferndiagnose, 41
 Festvorschübe, 1459
 F-Funktionen, 405
 FGROUP, 723, 726
 FIFO-Puffer, 635

Firmware, 1151
 FL, 1415
 Flache D-Nummern-Struktur, 1494
 FLIN, 1461
 FNORM, 1461
 FOC, 316
 FOCOF, 316
 FOCON, 316
 FPO, 1461
 FPRAOF, 1379
 FPRAON, 1379
 FRAME, 722
 Framedrehungen, 718
 in Werkzeugrichtung, 835, 836
 mit Raumwinkeln, 832
 FRC, 1462
 FRCM, 1462
 Freiformflächen, 233
 -modus, 186, 233
 Freischneidwinkel, 1513
 Freiwinkel, 1593
 Funktionsschnittstelle, 900
 FXS, 316
 FXS-REPOS, 327
 FXST, 316
 FXSW, 316
 FZ, 1417

G

G0-Toleranzfaktor, 256, 257
 G25, 114, 1380
 G26, 114, 1380
 G33, 1423
 G331, 1439
 G332, 1439
 G335, 1434
 G336, 1434
 G40, 1516
 G41, 1516
 G42, 1516
 G450/G451, 1536
 G451, 1538
 G460, 1551
 G461, 1477, 1551
 G462, 1477, 1552
 G58, 717
 G59, 717
 G60, 188
 G601, 189
 G602, 189
 G603, 189

G63, 1330, 1441
 G64, 195
 G642, 200
 G643, 200
 G644, 203
 G645, 207
 G70, 357
 G700, 357
 G71, 357
 G710, 357
 G74, 1248
 G9, 188
 G91 Erweiterung, 1605
 Nullpunktverschiebung, 1606
 G93, 1416
 G94, 1417
 G95, 1417
 G96, 1419
 G961, 1419
 G97, 1421
 G971, 1421
 Geber direkt am Werkzeug, 367
 Geberüberwachungen, 103
 Geberfrequenz, 103
 Genauhalt, 188
 Impliziter, 192
 Genauhaltbedingungen, 188, 189, 190
 Geometrieachsen, 722, 748
 Geschwindigkeiten, 341
 Getriebestufe
 bei M70, 1369
 Manuelle Vorgabe, 1346
 Vorgabe durch PLC, 1345
 Getriebestufe vorgeben, 1339
 Getriebestufen, 1337
 Getriebestufenauswahl
 Automatische, 1399
 GETTCOR, 1647
 GETTENV, 1646
 Gewinde
 -schneiden, 1430
 Gewindebohren
 mit Ausgleichsfutter, 1330
 Gewindeschneiden G33, 1423
 GFRAME0 ... GFRAME100, 788
 G-Gruppen, 552
 Glättung
 der Bahngeschwindigkeit, 218
 Grenz
 -geschwindigkeit, für Bahnachsen, 1415
 Grobverschiebung, 717
 Grundsystem, 349

H

Halterwinkel, 1593
 Hängende Achsen, 328
 Hardware-Endschalter, 109
 Hauptlauf, 490
 -Achsen, 1467
 Hauptlaufachsen, 724
 Helixinterpolation, 727
 H-Funktionen, 403
 HHH"; "UUU, 91
 Hilfsfunktion
 Adresserweiterung, 422
 anwenderdefiniert, 399
 Assoziierte, 430
 Ausgabeverhalten, 422
 Definition, 400
 Typ, 421
 vordefiniert, 399
 Wert, 422
 Hilfsfunktionen
 Anwenderspezifische, 428
 Vordefinierte, 406
 Hilfsfunktions
 -zähler, 450
 Hilfsfunktionsausgabe, 552
 HMI - CPU ab BTSS eady, 41
 HMI-Monitor, 959
 Hochlaufvorgang beendet, 57

I

IC, 1379
 Identitätsvergleich, 700
 Impliziter Bahnsteuerbetrieb, 195
 Impliziter Genauhalt, 192
 impliziter Vorlaufstopp, 542
 Inkrementell programmierte Korrekturwerte, 1605
 interpolarischer Achsverbund, 50
 Interpolator-Ende, 189
 Interrupt
 -routine, 592
 -signal, 593
 -sperre, 604
 Interruptroutine
 Ende, 594
 IR, 1434
 Istwert synchronisieren, 51
 Istwerte in Werkstückkoordinatensystem, 43
 Istwerterfassung, 361

Istwertkorrektur, 362
Istwertsystem
 werkstücknahes, 846
Istwertverarbeitung, 369

J

Joggen
 in Betriebsart AUTOMATIK, 486
JR, 1434

K

Kanal
 Aktueller, 582
 Eigenschaften, 490
 Grundeinstellungen, 552
 Konfiguration, 491
 -Zustand anzeigen, 563
 -Zustände, 563
Kanalachsen, 722
Kanalspezifischer NCK-Alarm steht an, 42
Kanalzustand
 Kanal aktiv, 485
 Kanal unterbrochen, 485
 Kanal-Reset, 485
Kaskadierter Satzsuchlauf, 512
Kennwort, 66
Kinematik der Maschine, 1560
Kinematiktyp, 1560
kinematische Transformation, 748
Kollisionsüberwachung, 1542
Konkurrierende Achsen, 920
Konstante Drehzahl, 1421
Konstante Schnittgeschwindigkeit, 1419
KONT, 1517
KONTC, 1517
KONTT, 1517
Kontur
 -abtastfaktor, 234
 -abtastzeit, 234
 -toleranz, 250
Konturfehler, 83
Korrekturnummer, 1485
KR, 1434
Krümmung, 232
krümmungsstetig, 1518
Kv-Faktor, 381

L

Label, 568
Laderachsen, 723
Lage der Koordinatensysteme und
Referenzpunkte, 733
Lagedifferenz-Aufschaltung, 392
Lagemesssystem, 49
Lageregelkreis, 380
Lageregler aktiv, 53
Laufzeiten
 Programm-, 691
LENTOAX, 1667
LFOF, 1430
LFON, 1430
LFPOS, 1430
LFTXT, 1430
LFWP, 1430
LIFTFAST, 603
LIMS, 1380
Lineare Signalverzerrungen, 83
Lineurvorschub (G94), 1417
Lockset-Datei
 zum Aktivieren der CNC-Sperrfunktion, 1228
 zum Deaktivieren der CNC-Sperrfunktion, 1233
 zum Entsperren der Steuerung, 1235, 1237
 zum Verlängern der CNC-Sperrfunktion, 1231
LookAhead, 208
 An- und Abwahl, 210
Luft-Temperatur-Alarm, 42

M

M1, 465
M17, 465
M19, 1303, 1379
M2, 465
M3, 1378
M30, 465
M4, 1378
M40, 1381, 1399
M41, 1381
M42, 1381
M43, 1381
M44, 1381
M45, 1381
M5, 1378
M70, 1379
Maschinen mit aufgelöster Kinematik, 1567
Maschinen mit drehbarem Werkstück, 1566

Maschinen mit drehbarem Werkzeug, 1565
Maschinenachsen, 721
Maschinenkoordinatensystem (MKS), 43, 715, 737
Maschinennullpunkt, 1243
Maschinennullpunkt M, 731
Maßsystem, 349, 357
Master-Slave Umschaltung bei G96, G961, 1423
MCP-Identifizierung, 935
Md < Mdx, 57
MD10000, 714
MD10010, 480
MD10050, 1286
MD10070, 1286
MD10125, 646
MD10131, 58
MD10192, 1355, 1356
MD10200, 342, 343, 345, 355
MD10210, 342, 343, 345, 376, 378
MD10220, 347
MD10230, 347
MD10240, 349, 353, 1470
MD10250, 353
MD10260, 354
MD10270, 355
MD10290, 356
MD10292, 356
MD10366, 1156
MD10368, 1156
MD10600, 762, 834
MD10602, 798, 800, 808, 812
MD10610, 718, 793
MD10612, 793, 794
MD10615, 841
MD10618, 150
MD10651, 1435
MD10680, 235
MD10682, 235
MD10700, 630
MD10702, 498, 506, 600, 617
MD10704, 1452
MD10707, 519
MD10708, 518
MD10710, 116, 1422, 1429
MD10712, 232, 689
MD10713, 439
MD10714, 417, 667
MD10715, 666
MD10716, 666
MD10719, 669
MD10735, 486
MD10804, 667
MD10806, 668
MD10814, 668
MD1103, 329
MD1104, 329
MD1105, 329
MD11100, 428
MD11110, 434, 1670
MD11220, 355
MD11300, 1245
MD11346, 1494
MD11410, 1341, 1424, 1543
MD11411, 493
MD11450, 446, 508, 509, 510, 512, 520, 546
MD11470, 521, 523, 548
MD11550, 588, 1440
MD11600, 597
MD11602, 596, 598, 599, 600
MD11604, 546, 596, 599
MD11610, 607
MD11620, 578
MD11625, 647
MD11626, 647
MD12000, 1449
MD12010, 1449
MD12020, 1447
MD12030, 211, 212, 1448
MD12040, 1447
MD12050, 1448
MD12060, 1449
MD12070, 1450
MD12080, 1450
MD12082, 1448
MD12090, 1440
MD12100, 211, 212, 1450
MD12200, 1460
MD12202, 1460
MD12204, 1460
MD1230/1231, 329
MD14504, 928
MD14506, 928
MD14508, 928
MD14516, 1195
MD14518, 1196
MD15700, 678
MD15702, 678
MD17200, 629
MD17300, 1230
MD18080, 1624
MD18088, 1561, 1576, 1581
MD18100, 1480, 1495
MD18102, 1487, 1495, 1496
MD18104, 1644
MD18105, 1485, 1486, 1487, 1489

MD18106, 1485, 1486	MD20184, 836, 1575
MD18108, 1627	MD20188, 1568
MD18112, 1628, 1642	MD20190, 1568
MD18114, 1610	MD20191, 601
MD18116, 1644, 1645, 1646, 1648, 1655	MD20192, 580
MD18150, 61	MD20193, 580
MD18190, 149	MD20194, 596, 609
MD18360, 636	MD20200, 1463
MD18362, 636	MD20201, 1462
MD18600, 760	MD20202, 1533
MD18602, 783, 786, 789, 790	MD20204, 1525
MD18960, 269, 284	MD20210, 1539
MD20000, 491	MD20220, 1537
MD20050, 661, 714, 818	MD20230, 1539
MD20060, 714	MD20240, 1541, 1542
MD20070, 714	MD20250, 1517
MD20080, 714, 726	MD20252, 1556
MD20090, 464, 1378	MD20256, 1549
MD20092, 1375	MD20270, 404, 1479, 1496, 1497, 1633, 1634
MD20094, 417, 668	MD20272, 404, 1628, 1633, 1634
MD20095, 417, 668	MD20310, 534
MD20096, 1482, 1483, 1484	MD20360, 875, 1493, 1494, 1642, 1652, 1666
MD20100, 874, 1420, 1423	MD20390, 1619, 1621, 1623
MD20105, 546, 605	MD20392, 1620
MD20106, 510, 579, 617	MD20400, 211
MD20107, 510, 579	MD20430, 212
MD20108, 577	MD20440, 212
MD20109, 578	MD20443, 216
MD20110, 351, 596, 840, 841, 843, 850, 876, 877, 1478, 1498, 1499, 1583, 1606, 1633	MD20450, 213
MD20112, 519, 596, 661, 845, 850, 877, 1498, 1583	MD20460, 219, 220
MD20115, 546, 599, 605	MD20462, 220
MD20116, 600	MD20465, 223, 224
MD20117, 600, 617	MD20480, 201, 204, 250
MD20118, 661	MD20482, 238, 250
MD20120, 661, 1497, 1583	MD20484, 238
MD20121, 661, 1478, 1497	MD20485, 238
MD20124, 464	MD20486, 238
MD20125, 1598	MD20487, 238
MD20126, 1582, 1599	MD20488, 249
MD20127, 1598	MD20490, 195
MD20130, 661, 1497	MD20500, 271, 272
MD20140, 661	MD20550, 191
MD20144, 663	MD20552, 191
MD20150, 116, 258, 269, 284, 302, 303, 356, 491, 552, 661, 775, 843, 844, 876, 1470	MD20560, 256
MD20152, 661, 843, 876	MD20600, 283
MD20170, 238	MD20602, 280, 281
MD20171, 238	MD20606, 234
MD20172, 238, 1461	MD20610, 273
MD20173, 238	MD20624, 876
MD20180, 1573	MD20700, 599, 1280, 1281
	MD20750, 1421
	MD20800, 401, 465, 466

MD20850, 1303
MD21015, 59
MD21016, 60
MD21020, 113
MD21158, 305
MD21159, 305
MD21166, 304
MD21168, 305
MD21220, 1455
MD21230, 1454
MD21330, 661
MD22000, 428, 1303
MD22010, 429, 1303
MD22020, 429, 1303
MD22030, 429, 1303
MD22035, 430, 1670
MD22040, 421, 428
MD22050, 421
MD22060, 422
MD22070, 422
MD22080, 422, 679, 1670
MD22100, 439
MD22110, 403, 405, 406
MD22200, 432
MD22210, 402, 432
MD22220, 403, 432
MD22230, 403, 432
MD22240, 405, 432, 1445
MD22250, 404, 432
MD22252, 405, 432
MD22254, 417, 431, 668
MD22256, 417, 431, 668
MD22410, 1445, 1451, 1471
MD22510, 552, 1137
MD22530, 460, 1583
MD22532, 460
MD22534, 460
MD22550, 669, 1478, 1491, 1492, 1496, 1497, 1498
MD22560, 417, 464, 668, 1478, 1497, 1498
MD22562, 1492, 1493
MD22600, 532
MD22620, 550
MD24004, 841
MD24006, 753, 775, 842
MD24007, 844
MD24008, 753, 841
MD24010, 793
MD24020, 781
MD24040, 818
MD24100, 1622
MD24110, 1622
MD24120, 1622
MD24558, 1654
MD24570, 1622
MD24572, 1622
MD24658, 1654
MD24805, 807, 808
MD24855, 807, 808
MD24905, 801
MD24955, 801
MD25574, 1622
MD26008, 417, 668
MD26012, 668
MD27100, 630, 876
MD27800, 492
MD27850, 695
MD27860, 566, 695
MD27880, 567, 698
MD27882, 699
MD28060, 626, 629
MD28070, 249
MD28071, 239
MD28072, 239
MD28082, 775, 795, 827, 828, 836, 1574
MD28085, 1480, 1576, 1581
MD28150, 63
MD28200, 149
MD28210, 149
MD28212, 149
MD28400, 629
MD28402, 629
MD28530, 61, 203
MD28533, 215
MD28560, 845
MD28600, 117
MD28610, 233
MD28620, 1546
MD30100, 363
MD30110, 363
MD30120, 363
MD30130, 362, 363
MD30200, 361, 370
MD30210, 363, 371
MD30220, 364, 371
MD30230, 364, 371
MD30240, 362, 364, 371
MD30242, 362, 365, 371, 1271
MD30244, 371
MD30250, 371, 1276
MD30260, 371
MD30270, 371
MD30300, 370, 372, 376, 378, 1470
MD30310, 111, 370
MD30320, 370

MD30330, 370, 1269
 MD30340, 371, 1269
 MD30350, 362, 1200
 MD30455, 1269, 1380
 MD30460, 874, 878
 MD30550, 1246
 MD31000, 372, 376, 378
 MD31010, 372
 MD31020, 372, 376, 378, 1286
 MD31025, 376
 MD31030, 139, 371, 372, 376
 MD31040, 372, 376, 378, 392, 1313, 1363
 MD31044, 367, 372, 1354
 MD31046, 125
 MD31050, 76, 139, 367, 372, 373, 376, 378, 1353, 1362
 MD31060, 76, 139, 367, 372, 373, 376, 378, 1353
 MD31064, 367, 371, 1354
 MD31066, 367, 371, 1354
 MD31070, 139, 372, 376, 378
 MD31080, 139, 372, 376, 378
 MD31090, 356
 MD31122, 1335
 MD31123, 1335
 MD31200, 356
 MD32000, 84, 301, 341, 370, 371, 1397, 1415
 MD32060, 548, 1466
 MD32074, 841
 MD32100, 369
 MD32200, 52, 76, 84, 87, 381
 MD32210, 394
 MD32220, 394
 MD32250, 139, 369
 MD32260, 139
 MD32300, 84, 91, 268, 278, 279, 280, 301, 1466, 1467
 MD32301, 304
 MD32310, 194, 279
 MD32320, 1467
 MD32400, 295, 388
 MD32402, 295, 389
 MD32410, 295, 389
 MD32420, 270, 284, 285
 MD32430, 284
 MD32431, 283, 289
 MD32432, 290
 MD32433, 278
 MD32434, 277, 278, 289
 MD32435, 289
 MD32436, 304
 MD32439, 291
 MD32440, 220, 223
 MD32610, 84
 MD32620, 385
 MD32630, 385
 MD32640, 387
 MD32711, 357
 MD32750, 1619, 1621, 1623
 MD32800, 76, 84
 MD32810, 76, 84, 139, 385
 MD32890, 390, 391
 MD32900, 381
 MD32910, 76, 381, 387
 MD32950, 392
 MD33050, 54
 MD33100, 201, 234, 238, 250
 MD33120, 207, 250
 MD34000, 1258, 1268
 MD34010, 1244, 1254, 1256, 1257, 1266, 1278
 MD34020, 1254, 1257
 MD34040, 1256, 1257, 1258, 1266, 1267, 1335
 MD34050, 1256
 MD34060, 1259, 1267, 1269, 1314
 MD34070, 1260, 1268
 MD34080, 1260
 MD34090, 371, 1260, 1263, 1264, 1277, 1279, 1280, 1281
 MD34092, 1258
 MD34093, 1259
 MD34100, 1261, 1268, 1278, 1280
 MD34102, 1270, 1271
 MD34104, 1272
 MD34110, 1246
 MD34200, 1264, 1271, 1272, 1278, 1282, 1335
 MD34210, 1274, 1276, 1277, 1278, 1279, 1280
 MD34300, 1267
 MD34320, 371, 372, 1263
 MD34330, 1282
 MD34990, 1415
 MD35000, 714, 770, 1375, 1381
 MD35010, 1338, 1342, 1357, 1363, 1368
 MD35012, 1337, 1363, 1366
 MD35014, 1369
 MD35020, 1323
 MD35030, 1324
 MD35035, 1304, 1371, 1373
 MD35040, 1300
 MD35090, 1337
 MD35092, 1342
 MD3510, 1368
 MD35100, 341, 1394
 MD35110, 1337, 1399
 MD35112, 1342
 MD35120, 1337, 1399

MD35122, 1342
 MD35130, 76, 1337, 1343, 1345, 1392
 MD35135, 1337
 MD35140, 1337, 1343, 1345, 1392, 1422, 1424
 MD35150, 1256, 1305, 1307, 1391, 1395
 MD35200, 1299, 1307, 1310, 1314, 1338, 1367
 MD35210, 1299, 1307, 1310, 1311, 1314, 1315, 1338, 1367
 MD35212, 1342
 MD35220, 298, 301
 MD35230, 298, 301
 MD35240, 203, 301
 MD35242, 296
 MD35300, 1286, 1307, 1309, 1310, 1314, 1315, 1338, 1367, 1380
 MD35310, 1338, 1363, 1366
 MD35350, 1314, 1366
 MD35400, 1094, 1357, 1358, 1359
 MD35410, 1357, 1358, 1359
 MD35430, 1357, 1358
 MD35500, 1301, 1305
 MD35510, 1305, 1390
 MD35550, 1338, 1343
 MD35590, 52, 75, 1338, 1352
 MD36000, 188, 1312, 1315, 1316, 1398
 MD36010, 87, 188, 1311, 1315, 1316, 1398
 MD36012, 89, 190
 MD36020, 87
 MD36030, 88, 89
 MD36040, 88, 335
 MD36042, 335
 MD36050, 89, 90, 98
 MD36051, 89, 90
 MD36052, 90, 91, 93, 94, 97
 MD36060, 52, 53, 1300, 1306, 1390
 MD36100, 110
 MD36110, 110
 MD36120, 110
 MD36130, 110
 MD36200, 100
 MD36210, 99, 370
 MD36220, 100
 MD36300, 103, 1285, 1286
 MD36302, 1309
 MD36310, 102, 104
 MD36312, 105
 MD36400, 84
 MD36500, 361
 MD36510, 361
 MD36600, 109
 MD36610, 50, 85, 87, 88, 91, 100, 101, 104, 106, 865

MD36620, 50, 865
 MD37002, 323
 MD37010, 318
 MD37012, 323
 MD37020, 320, 330
 MD37030, 319
 MD37040, 319
 MD37060, 318, 322, 323
 MD37080, 334
 MD51029, 625
 MD51074, 699
 MD51230, 1157
 MD51231, 1157
 MD9004, 345, 631
 MD9006, 43
 MD9010, 631
 MD9011, 345, 631
 MD9424, 631, 756
 MD9440, 849
 M-Dekodierung nach Liste, 924
 Mehrere Planachsen
 Achstausch, 878
 Meldesignale in DB2, 1138
 Messsysteme, 361, 1243
 M-Funktionersetzung, 666
 MKS, 43
 Modale Aktivierung (FOCON/FOCOF), 334
 Motor-/Last-Getriebe, 367
 MTL, 668
 Multi-Instanz-DB, 1144
 Multitool, 1031

N

Nachführen aktiv, 53
 Nahtstelle
 MCP/PLC und BHG/PLC, 898
 PLC/HMI, 898, 905
 PLC/HMI-Meldungen, 906
 PLC/MCP, 910
 PLC/NCK, 899
 NC
 -Ausfall, 918
 -Sprachumfang, 552
 -Start, 557
 -Variable lesen/schreiben, 921
 NC/PLC-Nahtstelle, 490
 NCK-Alarm mit Bearbeitungsstillstand, 42
 NCK-Alarm steht an, 42
 NCK-Batterie-Alarm, 42
 NCK-CPU ready, 41

NC-VAR-Selector, 968
 Bedienoptionen, 972
 Inbetriebnahme, Installation, 980
negative Adresserweiterung, 466
Nicht quittierter Getriebestufenwechsel, 1355
Nichtlineare Signalverzerrungen, 83
nist, 57
NORM, 1517
Normen, 863
Not-Halt
 Ablauf, 865
 Nahtstelle, 864
 Quittierung, 866
Not-Halt-Stellteile, 864
NPROT, 154
NPROTDEF, 150
Nullmarken-Diagnose, 105
Nullpunkte, 731
Nullpunktverschiebung
 Externe Nullpunktverschiebung, 750
Nullvektoren, 1564
Nutsäge, 1504

O

Orientierbare Werkzeugträger, 1476, 1557
 Beispiele, 1585
 Berechnung der aktiven Werkzeuglänge, 1572
 Kinematische Kette, 1564
 Neuanlegen, 1576
Orientierungs
 -toleranz, 250
OTOL, 251
OVR, 1451
OVRA, 1380, 1451

P

Paket
 -zähler, 450
Parametersatz
 bei Achsen, 384
 orientierbare Werkzeugträger, 1560
Parametrierung der Drehachsen, 1561
Parken, 122, 1248
Passwörter, 67
PCOF, 1379
Phasenfilter, 390
Physikalische Größen, 346
Planachse, 871
 Maßangaben, 877

PLC

-Achsen, 724
Grundprogramm-Funktionen, 885
HMI-Monitor, 959
Programmliste, 945, 946, 1223
Tastensperre, 949
-Variable lesen und schreiben, 61
-Versionen, 885
POINTER, 1139
POINTER im FB, 1141
POINTER im FC, 1139
POLF, 1430
POLFMASK, 1430
POLFMLIN, 1430
Polynome, Schnittpunktverfahren, 1549
POS, 723, 724
POSA, 723, 724
Positionierachsen, 343, 723
Positioniergenauigkeit, 344
PRESETON, 738, 1248
PRESETONS, 743
Probelaufvorschub, 499, 1452
Process DataShare, 648, 653
PROFIBUS
 -Ankopplung, 936
PROFIBUS-Ankopplung, 937
PROFIBUS-Diagnose, 917
PROFINET
 -Ankopplung, 939
Programm
 -aktion, 564
 -laufzeiten, 691
 -test, 494
 -zustand anzeigen, 562
 -zustände, 562
Programmbearbeitung ohne Sollwertausgaben, 494
Programmbeeinflussung, Nahtstellen-Signale, 624
Programmbetrieb, 551
Programmiergerät
 Hardware-Voraussetzungen, 966
Programmteil
 -wiederholung, 568
Prozessalarm, 917

R

Radius bezogene Daten, 873
Recallalarme, 43
Rechenfeinheit, 344
Referenzierarten, 1243

- Referenzieren
 - bei inkrementellen Messsystem, 1252
 - bei rotatorischen Absolutwertgebern, 1283
- Referenzpunkt R, 731
- Referenzpunkte, 731
- Regelsinn, 369
- Regelung, 380
- Reglerfreigabe, 49
- Regler-Parametersatz-Umschaltung, 75
- REPEAT, 568
- REPEATB, 568
- REPOS-Verschiebung, 326
- RESET
 - Kommando, 561
 - verhalten,
- Restweg löschen, 44
- Restzeit
 - für ein Werkstück, 693
- Rotationsanteil, 829
- Ruck
 - überhöhung, geschwindigkeitsabhängige, 290
- Ruckbegrenzung, 205
- Rückzug
 - richtung beim Gewindeschneiden, 1431
- Rundachsen, 723

- S**
- S..., 1378
- Satz
 - ausblenden, 625
- Satzbezogene Begrenzung (FOC), 334
- Satzsuchlauf
 - Kaskadierter, 504
 - mit Berechnung an der Kontur (Typ 2), 503
 - mit Berechnung an Satzendpunkt (Typ 4), 504
 - mit Berechnung im Modus Programmtest SERUPRO (Typ 5), 504
 - ohne Berechnung (Typ 1), 503
- Satzsuchlauf mit Berechnung
 - aufgesammlte Spindelfunktionen, 509
- Satzsuchlauf SERUPRO
 - Bahnachsen, 525
 - Bedingungen für Achsfunktionen, 547
 - Getriebestufenwechsel, 549
 - Grundeinstellung, 550
 - Positionierachsen Wiederanfahren, 524
 - REPOS mit NC/PLC-Nahtstellensignale steuern, 526
 - REPOS-Quittungen, 527
 - REPOS-Verhalten einstellen, 521
 - REPOS-Verschiebung bei Synchronspindelkopplung, 530
 - REPOS-Verschiebung im Gültigkeitsbereich, 529
 - REPOS-Verschiebung nach einem Achstausch, 529
 - Soll- und Istwertkopplungen, 544
 - Überlagerte Bewegungen, 549
 - Zeitlicher Ablauf, 518
- SAVE, 604
- SBLOF, 618
- SBLON, 618
- SCC, 877
- Schlüsselschalter, 68
- Schmierimpuls, 54
- Schneiden
 - bezugspunkt, 1595
 - form, 1594
 - lage, 1592, 1596
 - mittelpunkt, 1595
- Schneidenlage, 1505
 - relevante, 1515
- Schneidennummer, 1485
- Schneller Datenkanal, 61
- Schnitttrichtung, 1592
- Schraubenlinieninterpolation, 727
- Schutzbereiche, 143, 150
 - Einschränkungen, 172
- Schutzstufe
 - bei Anwender-ASUP, 607
- Schutzstufen, 65
 - parametrierbare, 69
- SCPARA, 1469
- SD41100, 1418
- SD42010, 1427, 1429
- SD42100, 499, 1453
- SD42101, 1453
- SD42200, 618
- SD42440, 719, 1605
- SD42442, 1605
- SD42444, 506
- SD42465, 201, 251
- SD42466, 201, 251
- SD42470, 239
- SD42471, 239
- SD42472, 239
- SD42473, 239
- SD42475, 239
- SD42476, 239
- SD42477, 239
- SD42480, 1549
- SD42496, 1548
- SD42500, 274

- SD42502, 274
- SD42510, 285, 286
- SD42512, 286
- SD42600, 1388, 1418
- SD42676, 251
- SD42678, 251
- SD42700, 637
- SD42750, 628
- SD42800, 1375
- SD42900, 1476, 1613, 1635, 1643, 1669
- SD42910, 1613, 1635, 1640, 1641, 1643
- SD42920, 1613, 1614, 1635, 1640, 1641, 1643
- SD42930, 1613, 1615, 1635, 1640, 1643
- SD42935, 1590, 1613, 1636, 1641, 1643, 1670
- SD42940, 1601, 1611, 1613, 1616, 1635, 1641, 1643
- SD42942, 1611, 1643
- SD42950, 1592, 1610, 1613, 1635, 1641, 1643
- SD42954, 1643
- SD42956, 1643
- SD42960, 1476, 1613, 1620, 1621
- SD42974, 1568
- SD42977, 1545
- SD42984, 1600
- SD42990, 626
- SD43200, 1384, 1386
- SD43202, 1385, 1386
- SD43206, 1385, 1387
- SD43210, 1422, 1424
- SD43220, 1422
- SD43230, 1421
- SD43235, 1396
- SD43240, 1303, 1379
- SD43250, 1303
- SD43300, 1388, 1418
- SD43400, 115
- SD43410, 115
- SD43420, 114
- SD43430, 114
- SD43500, 329
- SD43510, 330
- SD43520, 330
- Sekundenfehler, 234
- Self-Acting SERUPRO, 536
- SERUPRO
 - automatischer Unterbrechungszeiger, 540
 - Ende-ASUP, 454
 - programmierbarer Unterbrechungszeiger, 537
- SERUPRO-Anfahren
 - vom PLC beeinflussen, 526
- SERUPRO-ASUP, 326
 - Besonderheiten, 533
- seruproMasterChan, 536
- SETINT, 592, 602
- SETMS, 1378
- SETTCOR, 1654
- S-Funktionen, 402
- Signalaustausch
 - Ereignisgesteuerter, 883
 - Zyklischer, 883
- Signale
 - Achs-/Spindelspezifische (DB31, ...), 40
 - Alarm-Signale, 41
 - Compile-Zyklen, 901
 - Kanalspezifische (DB21, ...), 40
 - NCK / PLC, 902
 - PLC / Achsen, Spindeln, 904
 - PLC / BAG, 902
 - PLC / NCK-Kanäle, 903
 - PLC/NCK, 901
 - Ready-Signale, 41
- Signalverzerrungen, 83
- Simulation, 502
- Simulationsachsen, 362
- Simulierten Zielpunkt für LEAD mit JOG erreichen, 545
- SOFT, 283
- SOFTA, 285
- Software-Endschalter, 110
- Sollwertausgabe, 360
- Sollwertsystem, 360
- SPCOF, 1379
- SPCON, 1379
- Speicherbedarf
 - des PLC-Grundprogramms, 963
- Sperrdatum, 1226
- SPI, 1381
- Spiegelung
 - Frames, 770
- Spindel
 - drehzahl, Maximale, 1392
 - drehzahl, Minimale, 1392
 - drehzahlbegrenzungen, 1393
 - getriebestufe 0, 1348
 - korrektur, 1447, 1449
 - solldrehzahl, 1391
- Spindeldrehzahl, 343
- Spindelfunktionen über PLC, 492
- Spindelsperre, 45, 1446
- Spline, 187
- SPOS, 1301, 1378
- SPOSA, 1379
- Sprungmarke
 - bei Programmteiwiederholungen, 568

Stern-/Dreieck Umschaltung mit FC17, 1348

Stern-Dreieck-Umschaltung, 1088

Steuerungsverhalten

bei Hochlauf, 658

bei Reset, 658

bei Teileprogrammende, 658

bei Teileprogrammstart, 658

Stirnfräsen 3D, 1510

STOLF, 257

Stop-Delay-Bereich, 584

Stopp-Ereignisse, 585

STRINGIS, 553

Strings, 1145

Stromregler aktiv, 54

SVC, 1378

Symbolische Programmierung, 922

Synchronachsen, 726

T

tangentenstetig, 1518

Tastensperre, 43, 949

TCARR, 1573

TCOABS, 1573

TCOFR, 1573

TCP-Tool Center Position, 731

TEACH IN, 484

Teileprogramm

Anwahl, 557

-sätze ausblenden, 501

T-Funktion, 403, 1478

T-Funktionsersetzung, 668

Tick, 450

TOA

-Daten, 1504

-Einheit, 1480

Toleranz

-faktor für Spindeldrehzahl, 1391

Toleranz";"bei G0, 256

TOOLENV, 1642

Torsion, 232

TOWBCS, 1639

TOWKCS, 1640

TOWMCS, 1639

TOWSTD, 1638

TOWTCS, 1640

TOWWCS, 1639

TRANS, 717

TRANSMIT, 748

U

Überlastfaktor, 194

Überschleifen, 195

UDT-Zuordnungen, 923

Umdrehungsvorschub (G95), 1417

UTD-Bausteine, 922

V

VELOLIM, 1380

Verfahrbereiche, 343

Verfahrbewegungsfreigabe, 1282

Verriegelbare Datenbereiche, 69

Vorlauf, 490

Vorsatz-Getriebe, 367, 1354

Vorschub

-arten, 1413

-arten (G93, G94, G95), 1416

Bahnvorschub F, 1414

für Fase/Rundung, 1462

-korrektur, 1447

Linear- (G94), 1417

Umdrehungs- (G95), 1417

Zahn-, 1417

Zeitreziproker (G93), 1416

Vorschübe

Probelaufvorschub, 1452

Vorschub/Spindel Halt, 1445

Vorschubsperr, 1445

Vorschubkorrektur

Programmierbare, 1451

Vorschubsperr

kanalspezifisch, 1445

W

WAITENC, 663

WAITS, 1380

WALCS0, 118

WALIMOF, 116

WALIMON, 116

Wechsel des Frames, 1559

Wegkriterium, 198

Weiches An- und Abfahren

Bedeutung, 1522

Teilbewegungen, 1522

Werkstück

-simulation, 502

-zähler, 697, 702

Werkstückkoordinatensystem (WKS), 43, 715, 716, 757
Werkstücknullpunkt W, 731
Werkzeug, 1477
 anwählen, 1477
 Basismaß/Adaptermaß, 1511
 DISC, 1537
 -form, 1508
 -form, aktive, 1511
 -größe, aktive, 1511
 -korrekturdaten, 1504
 -länge, 1507
 -parameter, 1500
 -rückzug, 662
 Schneide, 1500
 -schneide, 1500
 T-Funktion, 1478
 -typ, 1502
 -verschleiß, 1510, 1511
 -verwaltung, 479
 Wechsel, 1478
 Wechsel mit M06, 1478
 Werkzeugradiuskorrektur-2D (WRK), 1516
 Werkzeugschneide, 1479
Werkzeuggrundorientierung, 1608
Werkzeugkorrektur
 Typen, 1559
 Verrechnung in der NC, 1482
Werkzeuglängenkorrektur
 Geometrie-, 1507
 Verschleiß-, 1510
 werkstückspezifisch einrechnen, 1634
 werkstückspezifisch einrechnen, Beispiel, 1588
Werkzeugmagazinachsen, 723
Werkzeugradiuskorrektur
 konstant halten, 1553
 Verschleiß-, 1511
Werkzeugradiuskorrektur-2D, 1515
 Abwahl, 1536
 An - und Abfahrverhalten, 1517
 Anwahl, 1516
 geändertes Alarmverhalten, 1549
 Geometrie-, 1508
 Kollisionsüberwachung, 1542
 Korrektur an Außenecken, 1536
 Korrektur an Innenecken, 1540
 Schnittpunkt G451, 1538
 Übergangskreis, 1537
 veränderlicher Korrekturwert, 1547
 Weiches An- und Abfahren, 1522
Werkzeugrevolverachsen, 723

Werkzeugträger, orientierbare, 1557
 drehbarer Werk Tisch, 1572
 Neuanlegen, 1576
 Programmierung, 1580
 Randbedingungen, 1582
 Schrägbearbeitung, 1571
 Steuerungsverhalten bei Reset, Programmstart, Repos, 1582
Werkzeugträgeranwahl, 1557
Werkzeugträgerbezugspunkt T, 731
Werkzeugwechsel
 D-Funktion, 1479
 Korrekturspeicher, 1480
Winkel
 Frei-, 1593
 Halter-, 1593
WKS, 43

X

XE * MERGEFORMAT,

Z

Zählimpuls, 700
Zahnvorschub, 1417
Zeitreziproker Vorschub (G93), 1416
Zugriffsmerkmale, 66
Zugriffsrechte, 65
Zugriffsschutz, 65
Zuordnung Werkzeug / Werkzeugträger, 1557
Zusammenfassen von Hilfsfunktionen, 422
Zusatzachsen, 722
Zyklischer Betrieb, 914
Zyklischer Signalaustausch, 40