SIPLUS RIC IEConS7

Functional description
Version V1.6 Upd4

Safety instructions

Warning notices

These Operating Instructions contain information that you should observe in order to ensure your own personal safety, as well to avoid material damage. The notices referring to your personal safety are highlighted in the manual by a warning triangle. Notices referring only to equipment damage have no safety alert symbol. Warnings are shown in descending order according to the degree of danger as follows.

DANGER
indicates that death or serious injury **will** result if proper precautions are not taken.

WARNING
indicates that death or serious injury **may** result if proper precautions are not taken.

CAUTION
indicates that minor personal injury **may** result if proper precautions are not taken.

NOTICE
means that material damage **can** occur if the appropriate precautions are not taken.

Note:
highlights important information about the product, handling the product, or part of the documentation that is of particular importance.

Qualified personnel

Commissioning and operation of equipment described in this manual (module, device) may only be carried out by qualified personnel. Qualified personnel in the meaning of the technical safety instructions in this manual are persons authorized to commission, isolate, ground and tag devices, systems and power circuits in accordance with the standards of security technology.

Please observe also the required basic knowledge mentioned in the preamble.

Trademarks

SIMATIC®, SIMATIC HMI®, SIMATIC NET®, SIROTEC®, SINUMERIK® and USS® are registered trademarks of Siemens AG. Any other names used in this document may be brand names, the use of which by third parties for their own purposes may infringe the rights of the legal owners.

Correct usage of Siemens products

Please observe the following:

WARNING

Siemens products may only be used for the applications indicated in the catalog and in the relevant technical description. If third-party products and components are used, these must be recommended or approved by Siemens. To ensure trouble-free and safe operation of the products, they must be appropriately transported, stored, assembled, installed, commissioned, operated and maintained. The permitted environmental and ambient conditions must be adhered to. Notices in the relevant documentation must be observed.

CAUTION

Changes to cabinet wiring!

Changes to cabinet wiring may only be performed in **zero-voltage** state! An additional or modified wiring realized outside of the manufacturing enterprise requires an **additional** functional and insulation test.

Electrostatic Sensitive Devices ESD

Almost all SIMATIC modules are equipped with highly integrated components or elements in MOS technology. For technological reasons, these electronic components are very sensitive to overvoltage and, consequently, to electrostatic discharge:

The short designation for such electrostatic sensitive components/modules is: "ESD", which is the commonly used international abbreviation of ”ESD” (Electrostatic Sensitive Device).

NOTICE

Electrostatic Sensitive Devices ESD

The presence of this symbol on cabinet, rack or packaging labels indicates the use of electrostatically sensitive devices and thus the touch sensitivity of these components.

These modules can be destroyed by voltage and energy far below the limits of human perception. Voltages of this kind occur as soon as a device or an assembly is touched by a person who is not electrostatically discharged.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.
In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions only form one element of such a concept.

Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.

Additionally, Siemens' guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under http://www.siemens.com/industrialsecurity.

Copyright © Siemens AG 2017 All rights reserved

The forwarding and reproduction of this document, and the reuse and distribution of its content is not permitted, unless explicit permission has been granted. Damages will be sought in all cases of infringement. All rights reserved, in particular in the event of patents being granted or a utility model being registered.

Siemens AG
DF FA SE
P.O. Box 23 55
90713 Fürth

Germany

Disclaimer:

We have checked the content of this printed document in accordance with the hardware and software described. Nevertheless, the risk of deviations cannot be excluded completely, which is why we do not accept liability for complete conformity. The details provided in this printed document are checked on a regular basis, however, and any corrections necessary are included in subsequent editions. We would be happy to receive your suggestions for improvement.

Technical data are subject to change.

Preamble

Purpose of the functional description

This functional description describes all the steps required to use the software application *SIPLUS RIC IEConS7*. It assists in the rapid and effective familiarization of personnel in the functionality of the application.

Contents of the functional description

This functional description contains the following topics

- Principles of communication with the telecontrol protocols conforming to IEC 60870-5-101 and IEC 60870-5-104
- Configuration of communication
- Description of the communication blocks and application blocks
- Parameterization of the blocks
- Appendices

Basic knowledge required

Comprehension of this manual requires basic knowledge in the field of telecontrol and the IEC 60870-5 protocols as well as general knowledge of automation technology with SIMATIC S7. Users should also have sufficient knowledge of computers, or of tools similar to PCs (e.g. programming devices), and of the Windows operating system.

As the S7 is configured using the STEP 7 basic software, you should also have sufficient experience in handling the basic software in SIMATIC Manager or TIA-Portal.

Target group

This manual is aimed at people with the required qualifications to commission, operate and maintain the products described:

- Installation engineers
- Programmers
- Commissioning engineers
- Servicing and maintenance personnel

Validity of the functional description

This functional description contains the description of the software application valid at the time of publication of the manual. We reserve the right to describe changes in the functionality of the software application in a special product information document.

 ©SIEMENS AG 2017

Table of Contents

1. General 11
1.1. IEC 60870-5-101 and IEC 60870-5-104 11
1.2. IEC 60870-5-103 Master 13
1.3. IEC on WinAC 13
1.3.1. Transmission of the Telecontrol protocol IEC 60870-5-101 13
1.3.2. Transmission of the Telecontrol protocol IEC 60870-5-104 13
1.4. IEC on Software Controller 14
1.4.1. Transmission of the telecontrol protocol IEC 60870-5-101 14
1.4.2. Transmission of the telecontrol protocol IEC 60870-5-104 14
2. Ordering information SIPLUS RIC IEConS7 15
2.1. SIPLUS RIC Software library 15
2.2. Add On for SIMATIC PCS 7 AS 15
3. Communication – IEC 60870-5-101 16
3.1. Overview 16
3.2. Settings in the SIMATIC Manager and TIA Portal 19
3.2.1. Hardware configurations for CPx41 and 1SI 19
3.2.2. Project planning CM PtP 20
3.2.3. Hardware configuration for WinAC in the SIMATIC Manager 21
3.2.4. Connection configuration Communication Processors 22
3.2.5. Project planning CP441 connection 22
3.3. Configuration of Communication-FB 'S7_IEC_Config' 23
3.3.1. IEC 60870-5-101 Parameters 25
3.3.2. Parameter description 27
3.3.3. Use of the Windows COM-Interface with WinAC 33
Use of several Interfaces 35
3.3.4. Link-Address parameter-DB for line operation 36
4. Communication with – IEC 60870-5-104 39
4.1. Overview 39
4.2. Settings in the SIMATIC Manager and TIA-Portal 43
4.2.1. Hardware configuration CP343 and CP443 43
4.2.2. Hardware configuration CPU31x PN, CPU41x PN and ET 200S IM151-8 PN 43
4.2.3. Hardware configuration for WinAC in the SIMATIC Manager 44
4.2.6. Connection configuration CP343-1 and CP443-1 45
4.2.7. Project planning Speed-Communication CP443 46
4.3. Configuration of Communication-FB 'S7_IEC_Config' 47
4.3.1. IEC 60870-5-104 parameter 50
4.3.2. Parameter description 53
4.3.3. Use of the Windows Ethernet-Interface under WinAC 59
4.4. Message buffering 61
4.4.1. Use of the Message Memory 61
4.4.2. T104-Confirmation buffer 63
4.4.3. Projection 63
4.5. Improvement of the IEC-Master performance 65
5. Multiple channel/Redundancy 67
5.1. Implementation 67
5.1.1. Cascade_Mode_0: Multiple Channel on basis of Layer 2 69
5.1.2. Cascade_Mode_1: Multiple Channel on basis of Layer 7 70
5.1.3. Cascade_Mode_2: Multiple Channel on basis of Layer 7 70
5.1.4. Cascade_Mode_3: Extension of the T104-Redundancy group and T101-Redundancy according to NUC101 71
5.1.5. Combination of Cascadings 72

5.2. Connection of several Control Centers ... 72
6. Application blocks ... 73
6.1. Slave blocks SL ... 74
6.1.1. Organization block - SL_Org_ASDU_1 (FB121) ... 75
6.1.2. Blocks for process data capture SLi ... 78
6.1.2.1. Single point and double point information - SLi_SP_DP_s128 (FB130) ... 79
6.1.2.2. Step position information - SLi_ST_s8 (FB131) ... 85
6.1.2.3. Bit pattern messages - SLi_BO_s8 (FB132) ... 91
6.1.2.4. Measured values - SLi_ME_ABC_s32 (FB133) ... 98
6.1.2.5. Integrated totals - SLi_IT_s8 (FB134) ... 106
6.1.3. Blocks for the process output SLo ... 113
6.1.3.1. Single, double and regulating step commands - SLo_SC_DC_RC_sx (FB135) ... 114
6.1.3.2. Setpoint commands - SLo_SE_ABC_sx (FB136) ... 122
6.1.3.3. Bit pattern commands - SLo_BO_sx (FB137) ... 129
6.1.3.4. Return_Values of the Slo-Blocks ... 135
6.1.4. Individual IEC-Addressing ... 136
6.2. Calling SLi-Blocks from time interrupts (Cyclic Interrupts) ... 138
6.3. Symbol access HMI ... 138
6.4. Master Blocks MA ... 139
6.4.1. Organization block - MA_Org_ASDU_n (FB122) ... 140
6.4.2. ASDU-Address Parameter-DB ... 144
6.4.3. Output blocks - MAo_xyz_pDB ... 147
6.4.3.1. Single point information - MAo_SP_IM_pDB (FC140) ... 153
6.4.3.2. Double point information - MAo_DP_IM_pDB (FC141) ... 158
6.4.3.3. Step position information - MAo_ST_IM_pDB (FC142) ... 162
6.4.3.4. Bit pattern messages - MAo_BO_IM_pDB (FC143) ... 166
6.4.3.5. Measured values - MAo_ME_NA_IM_pDB (FC144) ... 171
6.4.3.6. Measured values - MAo_ME_NB_IM_pDB (FC145) ... 175
6.4.3.7. Measured values - MAo_ME_NC_IM_pDB (FC146) ... 179
6.4.3.8. Integrated totals - MAo_IT_IM_pDB (FC147) ... 184
6.4.4. Input blocks - MAi_xyz_pDB ... 189
6.4.4.1. Single, double and regulating step commands - MAi_SC_DC_RC_pDB (FB148) ... 191
6.4.4.2. Setpoint commands - MAi_SE_ABC_1 (FB149) ... 196
6.4.4.3. Bit pattern commands - MAi_BO (FB150) ... 199
7. AddOn blocks ‚misc' ... 202
7.1. AddOn-block S7_IEC_Set_Threshold_1 (FC125) ... 203
7.2. Clock synchronization ... 204
7.2.1. Clock synchronization via the IEC-Protocol ... 204
7.2.2. Clock synchronization via the NTP-Protocol and SIMATIC Mode ... 204
7.2.3. Time in the S7-1500 ... 205
7.3. Auxiliary block S7_IEC_REM_MEM (FB126) for S7-15xx ... 206
8. Using the NON_Retain Memory only S7-300 ... 207
8.1.1. AddOn-block S7_IEC_Buf_Manager (FC100) ... 208
9. SIPLUS RIC IEConS7 in SIMATIC H-Systems ... 210
9.1. Configurations ... 210
9.1.1. IEC 60870-5-101 Master ... 211
9.1.2. IEC 60870-5-101 Slave ... 211
9.1.3. IEC 60870-5-104 Master ... 212
9.1.4. IEC 60870-5-104 Slave ... 212
9.1.5. H-CPUs V6 – Use of the integrated PN-Interface ... 213
9.2. Licensing in the H-System ... 214
10. SIPLUS RIC under TIA Step7 ... 215
10.1. Library concept ... 215
10.1.1. Procedure for using the libraries in TIA ... 215
10.1.2. Memory improvement ... 218

10.1.3. First Steps ... 218
11. SIPLUS RIC IEConS7 in S7-1500- and ET 200SP-systems ... 220
11.1. Available Libraries ... 220
11.2. Details and notes ... 220
11.2.1. Development and project planning environment ... 220
11.2.2. Manual ... 220
11.2.3. Use of several protocols at the same time ... 220
11.2.4. Activation ... 220
11.2.5. PN-Interfaces ... 220
11.2.6. Project planning CM PtP ... 221
11.2.7. PUT/GET – Access protection ... 221
11.2.8. Notice for loading ... 221
11.2.9. Data blocks created during run time ... 221
12. Software protection and demo mode ... 223
12.1. General activation of SIPLUS RIC IEConS7 ... 223
12.2. Activation of SIPLUS RIC IEConS7 on H-systems and other systems ... 224
12.3. Activation of SIPLUS RIC IEConWinAC ... 225
12.4. Activation of SIPLUS RIC IEConS7-1500 ... 226
13. Contacts/Addresses ... 227
14. Appendix ... 228
14.1. Resource assignment ... 228
14.2. Re-wiring blocks (only SIMATIC Manager) ... 235
14.2.1. Data blocks: ... 235
14.2.2. Independent FBs/FCs: ... 235
14.2.3. FBs/FCs with dependency ... 235
14.2.3.1. Possible problems at re-wiring ... 236
14.2.3.2. Solution/Workaround: ... 236
14.3. Interoperability list for T101/T104 ... 236
14.4. New Features and Functions ... 237
14.4.1. Delivery Version V1.1 ... 237
14.4.2. Delivery Version V1.2 ... 240
14.4.3. Delivery Version V1.3 ... 241
14.4.4. Delivery Version V1.4 ... 243
14.4.5. Delivery Version V1.5 ... 244
14.4.6. Delivery Version V1.6 ... 247
14.4.7. Delivery Version V1.6 Upd4 ... 250

 ©SIEMENS AG 2017

List of abbreviations used

Abbreviation	Meaning
ACD	access demand
ACK	positive acknowledge
APCI	application protocol control information
APDU	application protocol data unit
ASDU	application service data unit
BCR	binary counter reading
BL-Bit	blocked bit: 1 = blocked - 0 = not blocked
BO	bit pattern
CA	counter was adjusted
CI	counter interrogation
CON	control direction
COT	cause of transmission
CP	communication processor
CPU	central processing unit
CTS	clear to send
CY	carry
DB	data block
DCD	data carrier detect
DCE	data circuit terminating equipment
DCO	double command
DCS	double command state
DIR-Bit	direction bit: 1 = control direction - 0 = monitoring direction
DP/DPI	double point information
DSR	data set ready
DST	daylight saving time
DTE	data terminal equipment
DTR	data terminal ready
FB	functional block
FCB	frame count bit
FCV	frame count bit valid
FRZ	freeze
GI	general interrogation
IC	interrogation command
IEC	International Electrotechnical Commission
IM	image
IOA	information object address
IP	inter networking protocol
ISO	International Organization for Standardization
IT	integrated totals
IV-Bit	invalid bit: 1 = invalid - 0 = valid
MAi	master input block
MAo	master output block
ME	measured value
NACK	negative acknowledge
NIM	network image
NT-Bit	topicality bit: 1 = not topical - 0 = topical
NVA	normalized value
OB	organization block
OC	open controller
OSI	open systems interconnection
OV-Bit	overflow bit: 1 = overflow - 0 = no overflow

 ©SIEMENS AG 2017

Abbreviation	Meaning
PAA	Process image of the outputs
PAE	Process image of the inputs
PRM	primary message
PtP	point-to-point
QCC	qualifier field counter interrogation command
QDS	quality descriptor
QOI	qualifier of interrogation
QOS	qualifier of set-point command
RCO	regulating step command
RCS	regulating step command state
RIC	remote interface control
RQT	request
RTS	request to send
RTU	remote telecontrol unit
S/E	select/execute
SB-Bit	substitution bit: 1 = substituted - 0 = not substituted
SC	software controller
SCO	single command
SCS	single command state
SLi	slave input block
SLo	slave output block
SP	single point information
STARTDT	start data transmission
STOPDT	stop data transmission
SVA	scaled value
TCP	transmission control protocol
TI	type identification
TI-Bit	time bit: 1 = taking over of time status byte - 0 = none taking over of time status byte
TIA	Totally Intergrated Automation
TIA-Portal	Totally Intergrated Automation Portal
UTC	universal time coordinated
VTI-Bit	transient state bit: 1 = value with transient state indication - 0 = none transient state indication

 ©SIEMENS AG 2017

1. General

SIPLUS RIC is a range of hardware and software products facilitating the transmission of information in conformity with the IEC 60870-5 standard.

The standardized IEC 60870-5 protocols are sub-divided into the following versions:

- serial protocol IEC 60870-5-101 for telecontrol
- serial protocol IEC 60870-5-102 for remote meter reading
- serial protocol IEC 60870-5-103 for protection data acquisition
- network protocol IEC 60870-5-104 for telecontrol

Notice:

For ‘first steps’ with SIPLUS RIC see Chapter 10.

Notice:

In TIA-Portal (concerning S7-1500/ET 200SP/Software Controller) it’s committed to deactivate the attribute 'optimized block access' in user-generated data blocks!

- Data block for registration codes

- Offline user-generated buffer-DBs

- Image- and parameter-data blocks

- Source- and Destination-data blocks – to be read / written

Notice:

In TIA-Portal the comments and hints are only displayed for selected project languages.

1.1. IEC 60870-5-101 and IEC 60870-5-104

This manual describes the functionality of the SIPLUS RIC IEConS7 plus SIPLUS RIC IEConSC (Software Controller) and includes the functions of the protocols IEC 60870-5-101 and IEC 60870-5-104.

SIPLUS RIC IEConS7 uses only standard components of SIMATIC S7-1500, ET 200SP, Software Controller, ET 200S, S7-300, S7-400 and S7-400H. SIPLUS RIC IEConS7 comprises a S7 code library (S7 program).

The IEC protocols use 3 layers of the ISO/OSI layer model:

- Layer 1: Physical layer
- Layer 2: Link layer
- Layer 7: User layer

Layer 1 is the hardware channel of the SIMATIC S7 and is therefore dependent on the interface used. This produces different functional blocks for the various configurations.

Layer 2 is the link layer which creates monitors and secures the link to the communication partner regardless of the hardware being used. With the network-based T104 protocol the real layer 2 forms part of the network layer. When T104 layer 2 is referred to below, the 'APCI' transport interface defined in the standard is meant.

Layer 7 is the user layer which controls the defined user data and represents the link element between the protocol and the user program/process.

 ©SIEMENS AG 2017

SIPLUS RIC IEConS7 software packages each includes a Communication-FB (S7_IEC_Config) in which layers 1 and 2 required for the hardware and protocol type, as well as a base layer 7, are summarized and the necessary parameters are adjustable.

The 'Communication-FB' implements a 'telecontrol channel' on which the application blocks included in delivery as well can be coupled for the process monitoring and process control.

If multiple telecontrol channels are required, the Communication-FB must be called up multiple times (another instance) and the relevant parameters entered. The channels then function completely independently of one another.

The link redundancy with the IEC 60870-5-104 protocol is already integrated into the '-104 Slave' and '-104 Master' product versions. Redundancy groups each with 2 TCP links are supported which have to be different in the connection parameters:
IP-Address (module), Port number or active link connection on the part of SIPLUS RIC IEConS7.

More link redundancies can be projected (see chapter 5)

Each software package needs to be activated. This is achieved by entering a **registration code** at the input of a Communication-FB input. The registration code is based on the serial number of the SMC (S7-1500 and ET 200SP), the MMC (ET 200S and S7-300) resp. MC (S7-400) or serial number of the CPUs (S7-400H) and can be supplied with the order, in case a SMC/MMC is included in the order, or the serial number has been specified. Alternatively, you can request a registration code later on via email.

The delivered CD contains runnable example projects for many configurations, which are pre-allocated with basic parameters and running unrestricted for 15 minutes in demo mode. For permanent operation the registration code is required. The example projects can be adapted with low effort.

 ©SIEMENS AG 2017

1.2. IEC 60870-5-103 Master

From V1.2 also the protection devices protocol IEC 60870-5-103 Master with several devices on one polling line is supported.

The functionality of the SIPLUS RIC IEConS7 application for the protocol IEC 60870-5-103 Master is described in the SIPLUS RIC IEC 103 on S7 manual.

1.3. IEC on WinAC

From V1.3 WinAC based systems are supported from SIPLUS RIC IEConS7. On these computers the software package WinAC from RTX 2010 must be installed. WinAC contains a Windows logic controller (WinLC) RTX from V4.6. With it the IEConS7 application can be operated like on a hardware CPU (S7-300/400).

Up to now the mEC31-, Nanobox PC- and Microbox PC- hardware is tested and released.

The registration code is based on the installed license-key of the RTX. For further information we refer you to chapter 12.3.

1.3.1. Transmission of the Telecontrol protocol IEC 60870-5-101

The transmission of the telecontrol protocol IEC 60870-5-101 is carried out via the COM interface of the PCs. The interface is administrated from Windows. For using it by WinAC first of all the installation of a corresponding driver is required (see chapter 3.2.3).

When using mEC31 the communication can be carried out also via a CP340.

Note:

Though a transmission of the telecontrol protocol IEC 60870-5-101 via further interfaces (e.g. USB) is possible, but it cannot be ensured for all applications.

1.3.2. Transmission of the Telecontrol protocol IEC 60870-5-104

WinLC supports a LAN resp. PN-Interface, which can be used also for the transmission of the telecontrol protocol IEC 60870.5.104. Only one interface however is assignable.

Optionally further network adapters administrated from Windows can be used for the transmission. In this case first of all the installation of a corresponding driver on the Nanobox PC, Microbox PC resp. mEC31 is required (see chapter 4.2.3).

Note:

Though a transmission of the telecontrol protocol IEC 60870-5-104 via further network adapters (e.g. USB adapter) is possible, but it cannot be ensured for all applications.

Note:

The use of WinAC in the TIA-Portal is not recommended.

1.4. IEC on Software Controller

From V1.6 Upd4 computer based systems are now supported by SIPLUS RIC IEConS7, on which the software package SIMATIC S7-1500 Software Controller CPU 1507S is installed, as well as on the ET 200SP PC with the related Software Controller CPU 1505SP.

On the Software Controller the application IEConS7 is operated like on a hardware-CPU (S7-1500).

Until now the platforms ET 200SP OC (Open Controller) and IPC427D SC are tested and released.

The registration code is based on a serial number, which is provided by the Software Controller. For further information we refer you to chapter 12.4

1.4.1. Transmission of the telecontrol protocol IEC 60870-5-101

The transmission of the telecontrol protocol IEC 60870-5-101 is done by the CM PTP communication module, which is centrally or decentrally projected with ET 200SP in IO-systems (always required with IPC).

1.4.2. Transmission of the telecontrol protocol IEC 60870-5-104

The transmission of the telecontrol protocol IEC 60870-5-104 can be used over a LAN or rather PN interface. The application of the CP1543SP with the ET 200SP PC is, determined to the system, not yet possible.

 ©SIEMENS AG 2017

2. Ordering information SIPLUS RIC IEConS7

2.1. SIPLUS RIC Software library

SIPLUS RIC are software libraries which are provided together with a MicroMemory-Card MMC in case of S7-300 and ET 200S or with a SIMATIC Memory-Card SMC in case of S7-1500 and ET 200SP including the corresponding activation codes, for the IEC 60870-5-101/103/104 protocols.

For components of the S7-400 and S7-400H as well as Software Controller, SIPLUS RIC is provided without the corresponding activation codes for the IEC 60870-5-101/103/104 protocols.

The activation is carried out by means of Email to “siplus-ric.automation@siemens.com”.

You’ll find the ordering numbers on the SIPLUS web pages.
Website: http://www.automation.siemens.com/siplus/index_00.htm

2.2. Add On for SIMATIC PCS 7 AS

As Add On for the assignment in SIMATIC PCS 7 equipment products can be ordered direct via MLFB

For components of the S7-400 and S7-400H as well as Software Controller, SIPLUS RIC is delivered without the corresponding activation codes for the IEC 60870-5-101/103/104 protocols.

The activation is carried out by means of Email to: “siplus-ric.automation@siemens.com”

You’ll find the ordering numbers on the SIPLUS web pages.
Website: http://www.automation.siemens.com/siplus/index_00.htm

Note:

The blocks of the SIPLUS RIC IEConS7 library can be used in combination with automation systems of the process control system SIMATIC PCS 7 V7. The utilization of the library SIPLUS RIC IEConS7 for the telecontrol configurations shown above is independent from the SIMATIC PCS 7 telecontrol.

For further information please refer to **Catalog ST PCS 7.1, Add Ons for the Process control system SIMATIC PCS 7**

3. Communication – IEC 60870-5-101

3.1. Overview

IEC 60870-5-101 is a telecontrol protocol for bit-serial transmission.

SIPLUS RIC IEConS7 supports communication via the communication modules CM PTP, CP1540, CP1541 with Freeport-protocol and CP 1SI, CP340, CP341 resp. CP441 in ASCII- mode and the properties of the interfaces (RS232, RS422/485), baud rates, etc.

This means the CPs of the ET 200S- and S7-300 range can be used 'directly'. They are allocated to the respective IEC Communication-FB module via the hardware address.

In the S7-400 system 'Connection Configuration' using NETPRO is required in addition to the hardware configuration. The link ID (local ID) of the point-to-point connection (PtP) created must be assigned to the respective IEC communication-FB module.

For details of the hardware configuration and connection parameterization, refer to the chapter 3.2

In the S7-1500-System (S7-1500 and ET 200SP) for the bit-serial transmission the CP PTP, CP1540 and CP1541 are used in the Freeport modus.

SIPLUS RIC IEC on WinAC supports the communication via mEC31-RTX (WinAC 2010) resp. Nanobox-PC/Microbox-PC with RTX.

In the ET 200SP PC with the Software Controller the communication modules can be used for the bit-serial transmission CM PTP in the Freeport modus in central or distributed systems.

For the IPC427D the bit-serial transmission the CM PTP in an IO-device is possible with the Software controller.

SIPLUS RIC IEConS7 resp. SIPLUS RIC IEC on WinAC Software Controller is available for IEC 101 communication with the following properties:

- as **Master** or **Slave**
- in both traffic types balanced or unbalanced mode
- in leased line operation – VFT control is a CP property
- to one or several connection partner each
  (new in V1.2: polling control of multiple stations on one line is available)

 ©SIEMENS AG 2017

The following interfaces are available:

Software-Variant	Released Hardware	
S7LIB: S7IEC_M101_CP340/341 SIPLUS RIC IEConS7 IEC 60870-5-101 Master via CP340 or CP341 for standard CPUs S7-300 and S7-400H **S7LIB: S7IEC_S101_CP340/341** SIPLUS RIC IEConS7 IEC 60870-5-101 Slave via CP340 or CP341 for standard CPUs S7-300 and S7-400H	CP340-1 RS232 CP340-1 RS485/422 CP341-1 RS232 CP341-1 RS485/422	6ES7340-1AH02-0AE0 6ES7340-1CH02-0AE0 6ES7341-1AH02-0AE0 6ES7341-1CH02-0AE0
S7LIB: S7IEC_M101_CP441 SIPLUS RIC IEConS7 IEC 60870-5-101 Master via CP441 for standard CPUs S7-400 **S7LIB: S7IEC_S101_CP441** SIPLUS RIC IEConS7 IEC 60870-5-101 Slave via CP441 for standard CPUs S7-400	CP441-1 CP441-2 IF963-RS232 IF963-RS485/422	6ES7441-1AA04-0AE0 6ES7441-2AA04-0AE0 6ES7963-1AA00-0AA0 6ES7963-3AA00-0AA0
S7LIB: S7IEC_M101_CP1SI SIPLUS RIC IEConS7 IEC 60870-5-101 Master via CP1SI for ET 200S CPUs **S7LIB: S7IEC_S101_CP1SI** SIPLUS RIC IEConS7 IEC 60870-5-101 Slave via CP1SI for ET 200S CPUs	Interface modules 1SI ASCII (8Byte)	6ES7138-4DF01-0AB0
S7LIB:S7IEC_M101_WinAC SIPLUS RIC WinAC IEC 60870-5-101 Master for mEC31, nanobox-PC and microbox-PC IEC-Comm. via CP340, resp. COM-Interfaces **S7LIB:S7IEC_S101_WinAC** SIPLUS RIC WinAC IEC 60870-5-101 Slave for mEC31, nanobox-PC and microbox-PC IEC-Comm. via CP340, resp. COM-Interfaces	EC31-RTX (WinAC 2010) Microbox-PC with RTX Nanobox-PC with RTX	6ES7677-1DD10-0BB0 6ES7675-1DF30-0DB0

Software-Variant	Released Hardware	
S7LIB: **IEConS7 15xxCPU_Vx.y / S7IEC_101_MS_CM_PtP** SIPLUS RIC IEConS7 IEC 60870-5-101 Master/Slave via CM1540 or CM1541 for standard CPUs S7-1500	CM1540 RS232 CM1540 RS485/422 CM1541 RS232 CM1541 RS485/422	6ES7540-1AD00-0AA0 6ES7540-1AB00-0AA0 6ES7541-1AD02-0AB0 6ES7541-1AB00-0AB0
S7LIB: **IEConS7 15xxCPU_Vx.y / S7IEC_101_MS_CM_PtP** SIPLUS RIC IEConS7 IEC 60870-5-101 Master/Slave via CM PTP for standard CPUs ET 200SP	CM PTP	6ES7137-6AA00-0BA0
S7LIB: **IEConS7 15xxSC_Vx.y / S7IEC_101_MS_CM_PtP** SIPLUS RIC IEConS7 IEC 60870-5-101 Master/Slave via CM PTP for Standard CPUs ET 200SP PC	CM PTP	6ES7137-6AA00-0BA0
S7LIB: **IEConS7 15xxSC_Vx.y / S7IEC_101_MS_CM_PtP** SIPLUS RIC IEConS7 IEC 60870-5-101 Master/Slave via IO-Device mit CM PTP for sStandard IPC427_SC	IO-Device with CM PTP	6ES7137-6AA00-0BA0

In each case the library contains a central Communication-FB 'S7_IEC_Config' resp. S7_IEC_101_MS_CM_PtP (FB101) at S7-15xx, corresponding subordinated auxiliary blocks, as well as the associated 'application blocks' for the process connection.

The Communication-FB has to be programmed with your parameters (see chapter 3.3) and takes care of the entire IEC communication with the connection partner.

A pointer in DWord format establishes the connection/interface to the 'application blocks'. You use only the blocks with the required functionality which can be used on the other hand multiple, too.

Limits are only set here by the memory and resource consumption (cycle time). The application blocks (see chapter 6) are separated by master- (e.g. sending commands, output of indication and measurement values) and slave-functionality (e.g. record of indication and measurement values, output of commands).

The blocks for CP340 and CP341 are usable in standard S7-300 systems as well as in S7-400H systems. For more details using the blocks in an H-system, refer to chapter 9.

3.2. Settings in the SIMATIC Manager and TIA Portal

The necessary settings in the hardware configuration and in the connection configuration are described here.

3.2.1. Hardware configurations for CPx41 and 1SI

Note: For the CP441-2 modules the settings have to be done for each interface used.

The following settings must be carried out with serial communication for CP341, CP441 and 1SI:

Protocol: ASCII

- End ID of a Receive Message Frame:
  → On Expiry of Character Delay Time (the Character Delay Time depends on the baud rate, the default setting can be used)
- Transmission speed
  → Baud rate 300-115200 bit/s (depending on the used CP)
- Character frame (specified in standard):
  → 8 data bits
  → 1 stop bit
  → Parity even

Transmission parameters:

- Data Flow Control:
  → XON/XOFF is not permissible!
  → 'No data flow control' makes sense in the default
  → Alternatively RTS/CTS data flow control or operation of the V24 signals in accordance with the 'Online help' permitted
  These settings are only available in combination with the RS232 interface (module with CP441, module version with CP341)

Receiving data (at CPx41):

- Receive buffer on CP
  → Prevent overwriting
  → Buffered Receive Message Frames: 2

The RS422/485 interface version has other setting options in a separate card 'interface'.

Receiving data (at 1SI):

- Receive buffer on CP
  → Dynamic Message frame buffer: yes
  → Prevent message frame buffer overwriting: yes
  → Delete receive buffer during CP Start-up: yes

 ©SIEMENS AG 2017

3.2.2. Project planning CM PtP

The following settings must be carried out with serial communication (protocol T101/T103):

Protocol: Freeport

- Interface parameter
  → Parity: even
  → data bits: 8
  → stop bits: 1
  → Data flow control: none, or with RTS-handling if required

- Configuration of message transfer:
  → send idle line: active
  → Duration of the line: at least 33 Bit times
- Configuration of the message receipt
  → Frame start detection: Start on any character
  → End of frame detection:
    - After character delay time elapses: at least 33 Bit times
    - After receipt of a maximum number of characters: 300 bytes
- Receive buffer
  → Buffered received frames: 2
  → Prevent overwriting: no
  → Clear receive buffer on startup: yes

 ©SIEMENS AG 2017

3.2.3. Hardware configuration for WinAC in the SIMATIC Manager

The SIMATIC Manager Step 7 V5.5 is mandatory required.

The SIMATIC Microbox PC (Nanobox PC as well) and the embedded controller have to be inserted as PC-Station.

WinLC RTX on SIMATIC Mircobox PC (Nanobox PC)

The V4.6 of WinAC RTX2010, must be selected in the hardware catalog via the path SIMATIC PC station / controller / WinLC RTX / 6ES7611-4SB00-0YB7 with the corresponding CP.

WinLC RTX on SIMATIC S7 modular Embedded Controller EC31

The V1.1 of WinAC RTX2010 can be selected in the hardware catalog via the path SIMATIC PC Station / Controller / S7-modular Embedded Controller / EC31 / 6ES7677-1DD00-0Bx0.

Setup of Windows COM-Interface

The transmission of the telecontrol protocol IEC 60870-5-101 is carried out via the COM interface on the Nanobox PC, Microbox PC and mEC31 (in this case an additional EM PC is required). The interface is administrated from Windows. For using it by WinAC first of all the installation of a corresponding driver is required.

You receive the driver for the COM interface on a CD. It is in the ‚WinAC_COM_Driver' directory together with an ‚Install.bat' file. The parameterizing of the communication characteristics is carried out in a parameter block (default DB90) see chapter 3.3.3

When executing the **Install.bat** file the driver-dll (WinLC_SER_COM.dll) will be installed in the Windows Sytsem32 directory of the Microbox PC (Nanobox PC) resp. mEC31 and is now ready for use.

The use of WinAC in the TIA-Portal is not recommended.

 ©SIEMENS AG 2017

3.2.4. Connection configuration Communication Processors

The communication processors (CP1540, CP1541, CM PTP, CP340, CP341) and ET 200S CP 1SI ASCII are operated exclusively via the hardware address and therefore do not require any connection configuration.

3.2.5. Project planning CP441 connection

Only the CP441 modules require a connection projection

- SIMATIC Manager
  - Use the Simatic Manager/NETPRO
  - to generate a point-to-point connection
  (PtP = Point to Point)
  - to an **unspecified** partner.

If multiple 'connection paths' (CPs, or interfaces) are available, select the required path.

Note down the 'Local ID'.
You will need this later on for programming the parameters.

- TIA-Portal
  - in the TIA-Portal the PTP- connection projection of the CP441 with unknown partner is to be made automatically with inserting the interface modules.
  - The local ID is displayed under device configuration / connection.

Note down the 'Local ID'.
You will need this later on for programming the parameters.

 ©SIEMENS AG 2017

3.3. Configuration of Communication-FB ‘S7_IEC_Config’

(FB100 / FB101 at CPU15xx)

The Communication-FB can be called up directly from the OB1 as well as included in CFC.

Note:

The **Communication-FB block** ('S7_IEC_Config') is designed exclusively for the **OB1 priority level**. The application blocks also need to be operated in priority level OB1.

The block library contains in each case an instance DB of the Communication-FB as pre-defined instance DB for the Communication-FB with symbolic name ‚S7_IEC…’. When calling up the Communication-FB manually you can use this DB directly (e.g. Call FB100, DB100). If you prefer or require other DB numbers, or if you are working with CFC (automatic DB assignment), you can delete the instance- DB100.

In addition, you need to refer to the enclosed variable table 'VAT_S7_IEC'. If you want to use this, by assigning the 'new instance DB a symbolic name and change the DB numbers in the variable table (e. g.DB100.x -> DBnew.x).

In the figure below the exemplary illustration of some S7_IEC-Config-blocks in the CFC plan and in the TIA-Portal are shown, and that followed a detailed list of the input and output variables with their brief comments, and a detailed description.

Attention when using in the S7-1500:

The instance DB must be stored in the remanent memory, so that the data blocks are not generated several times after stop start or voltage reversal

Input: Open the data block and activate the Remanence column.

Alternatively, the necessary areas of the DBs can be placed into the remanent memory by means of S7_IEC_Rem_Mem (FB126).

IEC60870-5-101Master and Slave	IEC60870-5-101 Master	IEC60870-5-101 Slave

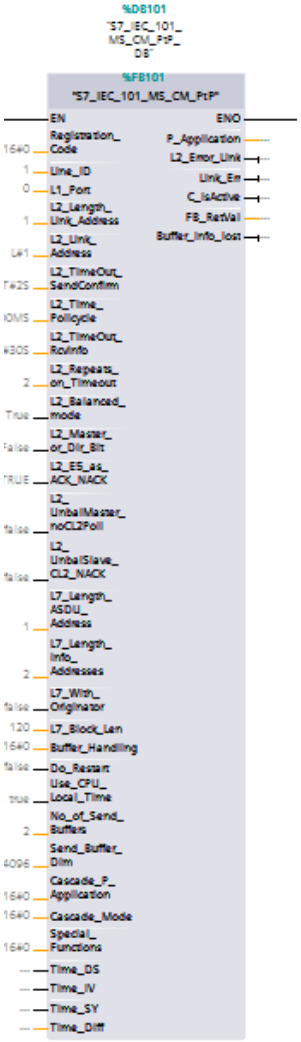
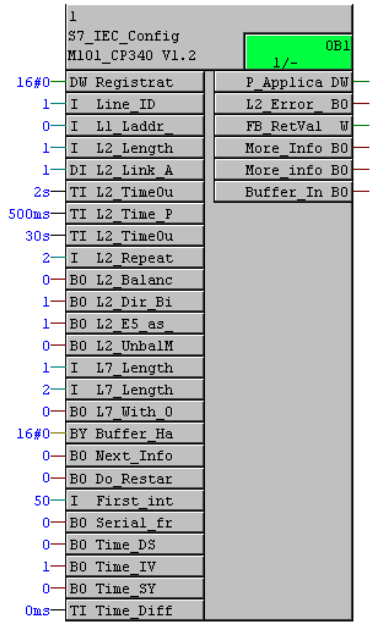
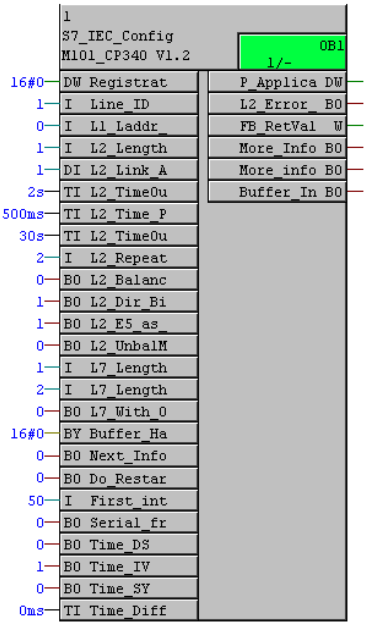

		

Illustration as FUP-block with default values	Illustration as CFC-block with default values	Illustration as CFC-block with default values

Note to the CFC plan view

The CFC views included in the manual are only used for illustration purposes. CFC is **not** required for using the blocks. The standard programming options in SIMATIC are sufficient.

NOTICE:

In case you project with CFC (e.g. within PCS7) please observe that data blocks created from SIPLUS RIC during runtime are not overwritten from CFC compiler. See also parameter ‘First_internal_DB_No’.

3.3.1. IEC 60870-5-101 Parameters

Communication-FB VAR_INPUT		Type	Default	Block-Comment/brief description	
Registration_Code	IN	DWORD	DW#16#0	!!! IMPORTANT !!! See product information	
Line_ID	IN	INT	1	For differentiation of more than one line;1-x (see manual); Default: 1;	
L1_Local_ID	IN	WORD	W#16#1000	see NetPro Connection Property 'ID' of your PtP-Connection	*1)
L1_Laddr_HW	IN	INT	0	see Address of CP-Module in your HW-Configuration	*2)
L1_Data_Wait_Time	IN	TIME	T#10ms	Time to wait for partner to set CTS on after setting RTS (Def: T#10ms)	*3)
L1_Time_RTS_OFF	IN	TIME	T#10ms	Time to elapse after transmission before RTS turns off (Def: T#10ms)	*3)
L1_COM_or_Laddr	IN	INT	1	COM-Port (ext. ser. drv) or address of CP-Module in your HW-Configuration	*4)
L1_Port	IN	UINT	0	see Address of CP-Module in your HW-Configuration	*6)
L2_Length_Link_Address	IN	INT	1	Length of Link-Address (0,1,2 Octets); Default: 1	
L2_Link_Address	IN	DINT	L#1	Link-Address: 1-254, 1-65534; Default: 1	
L2_TimeOut_SendConfirm	IN	TIME	T#2S	Timeout for answers; Default: 2 sec (T#2s)	
L2_Time_Pollcycle	IN	TIME	T#500ms	only unbal.mode master: time between two polls, if no ACD is in answer(T#500ms)	
L2_TimeOut_RcvInfo	IN	TIME	T#30S	unbal.Slave -> Cycle Error, bal mode -> Linktest, 0 -> not active, Default 30s	
L2_Repeats_on_Timeout	IN	INT	2	No. of Repeats on Timeout; Default: 2	
L2_Balanced_mode	IN	BOOL	FALSE	0 -> unbalanced mode (slave) 1-> balanced mode; Default: 0	
L2_Master_or_Dir_Bit	IN	BOOL	TRUE FALSE	Def: 1/0; 0/1 -> unbal. Mode: Slave/Master; bal.mode: DIR-Bit=0/1 in Send Tel	M/S
L2_E5_as_ACK_NACK	IN	BOOL	TRUE	1 -> E5 will be accepted as ACK (FC0) and NACK (FC9); Default: 1	
L2_UnbalMaster_noCL2Poll	IN	BOOL	FALSE	only unbalanced master: Polling only with Class1-Request, no Class2-Requests	
L2_UnbalSlave_CL2_NACK	IN	BOOL	FALSE	only unbal. slave: 1-> Answer to Class2-Request always with NACK not with Data	
L7_Length_ASDU_Address	IN	INT	1	Length of ASDU-Address (1 or 2 Octets); Default: 1	
L7_Length_Info_Addresses	IN	INT	2	Length of Infoobject-Addresses (1, 2 or 3 Octets); Default: 2	
L7_With_Originator	IN	BOOL	FALSE	0 -> without Originator; 1 -> with Originator; Default: 0	
L7_Block_Len	IN	INT	120	max. length of T101/104-ASDU (1..255, see manual); Default 120	
Buffer_Handling	IN	BYTE	B#16#0	0(def): compl, 1: compl. if ov 2: oldest Info will deleted; 3: no act.	
Next_Info	IN	BOOL	FALSE	Only allowed for additional calls in a Cycle ->reduced operation -> next info	*5)
Do_Restart	IN	BOOL	FALSE	a rising edge restarts the IEC-Application	
Use CPU Local Time	IN	BOOL	TRUE	Use Local Time of CPU incl. DST-Bit New V1.6 Upd4	*6)
Serial_from_CPU	IN	BOOL	FALSE	For Registration use serial number from CPU instead of memory card	*1)

 ©SIEMENS AG 2017

First_internal_DB_No	IN	INT	50	First DB-Number for DBs created by the Runtime Software; Default: 50;	
No_of_Send_Buffers	IN	INT	1 2	Default 1/2;Number of Send buffers (with send priority 1 to this para)	M/S
Send_Buffer_Dim	IN	DINT	L#4096	Dimension [size in byte] of the send buffers; Default 4096	
Cascade_P_Application	IN	DWORD	DW#16#0	Connect to P_Application from an other S7_IEC_Config for realizing mult. channel	
Cascade_Mode	IN	BYTE	B#16#0	0 (Def): L2 copy, 1: L7 copy, 2: L7 copy, 3: red channel	
Special_Functions	IN	DWORD	DW#16#0	Default 16#0; Only for special cases; Please refer to manual	
Para_DB_No	IN	INT	90	DB-Number which contains the settings for ext. serial driver (Default: 90)	*4)

Communication-FB VAR_OUTPUT		**Type**	**Default**	**Block-Comment/brief description**	
P_Application	OUT	DWORD	DW#16#0	Pointer to Application Functions (SL_x, MA_x)	
L2_Error_Link	OUT	BOOL	FALSE	Set, if no Connection (incl. Casc) is active longer than L2_Error_DelayTime	
Link_Err	OUT	BOOL	FALSE	Set, if this Connection is not established	
C_isActive	OUT	BOOL	FALSE	Set, if Channel is active (normally invers to Err_Link - if not Casc_Mode_3)	
FB_RetVal	OUT	INT	0	positive Values -> ok; negative Values (16#8xxx) -> Error	
Z_Info	OUT	WORD	W#16#0	additional information in case of errors	*4)
More_Info_available *4)	OUT	BOOL	FALSE	Set, if more infos are available for processing in the actual OB1-cycle.	*5)
More_info_is_spo *4)	OUT	BOOL	FALSE	More info is spontaneous	*5)
Buffer_Info_lost	OUT	BOOL	FALSE	in case of Link_Err one ore more Info were deleted from the buffers	

Communication-FB VAR_IN_OUT		**Type**	**Default**	**Block-Comment/brief description**	
Time_DS	IN/OUT	BOOL	FALSE	Time-Qualifier DaylightSaving (Summertime)	
Time_IV	IN/OUT	BOOL	TRUE	Time-Qualifier Time is InValid	
Time_SY	IN/OUT	BOOL	FALSE	Time-Qualifier Time is synchronized (actually only internal used)	
Time_Diff	IN/OUT	Time	T#0MS	Time difference between CPU-Time and desired IEC-Time-Stamps	

*1) Parameter only present in product version for S7-400
*2) Parameter only present in product version for S7-300 and ET 200S
*3) Parameter only present in product version for ET 200S
*4) Parameter only present in product version for WinAC
*5) Parameter only present up to variant V1.4
*6) Parameter only present in product version for S7-15xx
M/S: From V1.6 the Master and Slave blocks are standardized and can be switched over in its function. They include always all parameter and differ only in the pre-allocation of individual values.

3.3.2. Parameter description

Registration_Code

The MMC (S7-300, ET 200S) or MC (S7-400), SMC (CPU15xx) is used as a dongle. Depending on its serial number, you will receive from Siemens a registration/activation code which you need to specify here.
No entries or the incorrect entries will permit the unrestricted operation in demo mode for 15 min. After this period, the telecontrol communication will be stopped.
For more details, refer to '**FB_RetVal'** and the software protection chapter

Alternatively also the number of the data block containing a list of registration codes can be entered here.
The permissible range for entering a DB
1 up to 2000 (decimal); e.g. DB1000 -> DW#16#3E8

For further details please refer to ‚**FB_RetVal**' and chapter 1110.

Line_ID

Is only relevant if you call up the Communication-FB multiple times (as another instance with a separate DB) to realize an additional IEC link.
The calls then need to be identified using different Line_IDs.

The Line_ID is used for the derivation of the internal Connection_ID (Con_ID) which has to be unique for each TCP_Connection of the CPU when using PN-Interfaces. Therefore Line-IDs from Communication-FB, using the same interface have to be selected that way that from this unique Con_IDs can be generated.
The values are generated as follows:

CON_ID = LINE_ID for C1 (if only one channel exists)
CON_ID = LINE_ID + x for C2
CON_ID = LINE_ID + 2x for C3
CON_ID = LINE_ID + 3x for C3

x = 16 for WinAC-Variants
x = 64 for S7-Variants with 4 channels
x = 128 for S7 Variants with 2 channels

For diagnostics purposes additional a Comp_ID is used. It has the same value like the Con_ID and concerns the Layer 1, 2 and 7 which are (internal) called in the Communication-FB

L1_COM_or_Laddr

Number of the COM Interface (in the hardware configuration) which is used for the transmission of the IEC 60870-5-101 protocol. In case of using a CP340 for transmission the corresponding address has to be entered (still not released)
<=255 Interpretation as Windows COM port. The external interface driver is used. The parameterizing of the communication characteristics is carried out in a parameter data block (default DB90) see chapter 3.3.3

L1_Local_ID

Is the ID from the connection configuration (NETPRO) e.g. w#16#1000.
The value can be taken from NETPRO/properties TCP connection/module parameters.

L1_Laddr_HW

Is the address of the CP allocated/shown in the hardware configuration.

L1_Port

This is the hardware identifier of the CP resp. its interface at S7-15xx systems. It is available in the device configuration.
It can be found as well at the system constants and taken over from there.

L1_Data_Wait_Time

RTS-lead time.
Waiting time until data are sent after RTS has been set if the partner has not set CTS = ON before.

Only for ET 200S (1SI) with adjusted data flow control „Automatically use of the V.24-signal“. At other CPs the parameter is located in the hardware configuration.

L1_Time_RTS_OFF

RTS-Follow-up time
Time which has to expire after transmission before RTS is switched OFF.

Only for ET 200S (1SI) with adjusted data flow control „Automatically use of the V.24-signal“. At other CPs the parameter is located in the hardware configuration.

L2_Length_Link_Address	Is the number of the address octet in the link layer Permissible range in balanced mode 0, 1 and 2 Permissible range in unbalanced mode 1 and 2
L2_Link_Address	Link_address is the address value of the link layer Permissible range with length of link address 0 is fixed at 0 Permissible range with length of link address 1 is 1 to 254 Permissible range with length of link address 2 is 1 to 65534 If there is more than one station connected to one channel (line operation) you have to use a link address parameter data block. Detailed description see chapter 3.3.4
L2_TimeOut_SendConfirm	Maximum time for responses in balanced mode and unbalanced mode master, incl. the message transfer time itself, in other words, depending on the baud rate. After the time elapses, the last message is repeated. If the repetitions are still unsuccessful (para **L2_Repeats_on_Timeout**), the link is marked as faulted (**L2_Error_Link)** and re-established
L2_Time_Pollcycle	Only with unbalanced mode master: Time [ms] between 2 call messages if there is no high priority data.
L2_TimeOut_RcvInfo	Every correctly received message re-triggers the monitoring timer. When the time set here elapses, the response is as follows, depending on the traffic type: Unbalanced mode slave: -> link error-> (**L2_Error_Link)** balanced mode -> send link test FC2 (whose response must be received, which is monitored)
L2_Repeats_on_Timeout	Refer to **L2_TimeOut_SendConfirm**
L2_Balanced_mode	Setting of the required traffic type: FALSE: -> unbalanced mode TRUE: -> balanced mode
L2_Master_or_Dir_Bit (from V1.6) **L2_Dir_Bit (up to V1.6)**	In balanced mode: DIR-Bit The DIR-Bit is used to differentiate between the two partners. Normally the DIR-Bit is set for masters (main station), and not set for slaves (sub-station). In unbalanced mode: Switch over Master/Slave Determines whether the communication is working as Master (calling station/control center) or as Slave (answering station/substation).
L2_E5_as_ACK_NACK	If activated, individual characters E5 are accepted depending on the situation as the follows: - positive acknowledgement (ACK FC0) - no user data available (NACK FC9) Short messages are always used for sending, regardless of this setting, which means that no individual characters are used.
L2_UnbalSlave_CL2_NACK	Only relevant to unbalanced mode slave: If TRUE, all data being set is handled as class 1 data, and class 2 interrogations are rejected with NACK 'No user data available' (but with the ACD bit set). With FALSE, data to be sent (in addition to class 1) are also transferred to class 2 queries.
L2_UnbalMaster_noCL2Poll	Only relevant with unbalanced mode master: If TRUE, instead of the normal cyclical class 2 interrogations, only class 1 interrogations are implemented. This increases the data through-put with relevant partners.
L7_Length_ASDU_Address	is the number of the octet of the application service data unit Station address Permissible range is 1 and 2
L7_Length_Info_Addresses	is the number of the octet of the information object address Permissible range in 1, 2 and 3
L7_With_Originator	FALSE means transmission cause without originator TRUE means transmission cause with originator
L7_Block_Len	The max. length of the T101/104-ASDU will be adjusted. The possible ASDU-length is depending on the protocol type, at approx. 253 signs. However an adjusted value which is too high will be limited automatically. Default value = 120 Setting options 1…255

Buffer_Handling

Defines the modality of buffering the messages (information). Generally the process described below is carried out only in case of link error

B#16#00 Clear all buffer and keep them empty
B#16#01 buffer first,
in case a buffer is filled up
-> clear all buffer and keep them empty
B#16#02 buffer and in case of overflow clear the oldest information.
B#16#03 buffer and in case of overflow stuffing backward
(meets the behavior in V1.0)

In normal operation (no ‚Link_Error') the information are buffered. In case of impending buffer overflow ‚backlog' occurs in the collection blocks (excepting organizational messages and command acknowledgements)
For the Master blocks we recommend the adjustment B#16#00 with it commands and setpoint values are not stored.)

For further details about message buffering please refer to chapter 4.4.

Next_Info
(only up to V1.4)

Only relevant at Master-Applications.

The parameter **Next_Info** informs the config. block that a repeated request occurs within one (OB) cycle. This serves only for handing over the next blocked information. In this case a reduced cycle through the ‚Master chain' occurs.

For further details of increasing the flow rate at Master-Applications please refer to chapter 4.5.

Do_Restart

Not all parameter are effective in case of online changes (without CPU Stop/Start).

The control input enables the save transfer from online changed parameters without CPU Stop/Start.

With rising edge the IEC-application is new initialized with the values programmed at the block inputs and started.

Use_CPU_Local_Time

True: use of the local time of the CPU inclusive daylight saving time-Bit
as of SIPLUS RIC V1.6 Upd4 at S7-1500
False: use of the system time
(see also chapter 7.2.3)

First_internal_DB_No

The IEC application generates during start-up different data blocks (diag DB, message memory. etc.) The parameter defines the first DB number for creating this (default: 50).

NOTICE:
If you project with CFC (e.g. within PCS7) please observe that data blocks created from SIPLUS RIC during runtime are not overwritten from CFC compiler. Use this parameter alternatively to put internal data blocks into a secure area.

No_of_Send_Buffers

Number of send buffers to be created (1…16).
From V1.6 the parameter is available in all libraries. In the block variant Slave the default adjustment is 2, in Master variants the default adjustment is 1.
See also chapter 8

Send_Buffers_Dim

Size of the send buffers to be created. The default value is 4096.
From V1.6 the parameter is available in all libraries. Individual DB length can be adjusted using an AddOn block (see chapter 8.1.1)

Cascade_P_Application

Possibility of coupling another (or more) Communication-FB to an existing Communication-FB (cascading/casc.). Main_FB is the first Communication-FB which is used for the connection of further Communication-FB (Casc_FB)

Cascade_Mode

Selection of the desired redundancy mode: (see chapter 5)
Mode 0: Multiple channel on the basis of layer 2
Mode 1: Multiple channel on layer 7 with partner fault per channel
Mode 2: Multiple channel on layer 7 with partner fault at Main_FB
Mode 3: Extension of the redundancy group (T104) resp. switch over to/from T101-channels

 ©SIEMENS AG 2017

Para_DB_No

When using the COM interface on the mEC31 or microbox PC (nanobox PC) for WinAC an external interface driver is required. The communication performance of this external driver is provided in a parameter data block. It's DB No. (default DB90) has to be assigned to the Communication-FB input ‚Para-DB_No'.

Serial_from_CPU

Selection whether the serial number of the Memory card or the serial number of the CPU is used for the licensing.
FALSE: Use serial number of the Memory card
TRUE: Use serial number of the CPU

Special_Functions (from V1.6)

Allows the activation of diverse special functions
Available functions are:

Bit 0	Switching off the filtering on cause of transmission in the Master blocks. This enables e.g. the non standard conformance receive and evaluation of indications with cause of transmission cyclic.
Bit 4	Deactivation of the reaction if receiving an unknown message type identification in control direction (reflection with COT 44)
Bit 5	Deactivation of the reaction if receiving an unknown common ASDU address in control direction (reflection with COT 46)
Bit 6	Deactivation of the reaction if receiving an unknown cause of transmission in control direction (reflection with COT 45)
Bit 7	Deactivation of the reaction if receiving an unknown information object address in control direction (reflection with COT 47)
Bit 16	Project specific special function at T104 protocol: Repetition of k messages instead of number of unreceipt messages

P_Application

Pointer (DWORD) for the interconnection of the application blocks.
The interconnection can be carried out directly in the CFC plan.
With manual programming you assign the output, e.g. to a free memory (MD) or to a temporary variable which you then specify on the relevant input variables of the application blocks.

L2_Error_Link
Link_Err (from V1.6)

Signals the communication status to the connection partner, resp. error, as follows:
FALSE: communication established
TRUE: communication error
Errors are formed as follows depending on the traffic type:

With Point to point connections:
True: if the connection partner can't be contacted anymore.

As unbalanced mode slave:
TRUE: No message reception within the programmable time period **L2_TimeOut_RcvInfo**

As unbalanced mode master or in balanced mode:
Please refer to the parameter description **L2_TimeOut_SendConfirm**

With line operation (unbalanced mode Master):
SET if no station does answer
RESET if at least one station does answer.
Station specific Link-Error will be written to the link address parameter data block.

Link_Err signals the communication status of the own channel.

L2_Error_Link (name unchanged for compatibility reason) signals from version V1.6 the communication status via all channels (in case of redundancy by means of cascading). In case of simple applications without redundancy L2_Error_Link follows the output Link_Err time delayed by the (internal) parameter ‚L2_Error_DelayTime', fixed on T#1s

C_isActive

Shows whether the data transmission on the connection is active.
This is always the case at a non redundant connection as soon as Link_Err is going.
If redundancy is projected (cascading in mode 3) it depends on the real activation.
See also chapter 5.

FB_RetVal

The signaling of the functional block is as follows:

W#16#0000	No error
W#16#0001 – 0900	block in demo mode. Remaining runtime in seconds in BCD
-	> can be read directly in hex illustration
W#16#7FFF	Initialization cycle

Error:

W#16#8101	Error when generating data blocks. Cause could be working memory too less
W#16#8201	Problem with registration code and demo time expired -> Enter correct registration code on the parameter input **Registration_Code**.

Further error codes at WinAC applications:

W#16#8301	Error from P_RCV_WinAC
W#16#8302	Error from P_SEND_WinAC
W#16#8305	Error from Com_INIT_WinAC
W#16#84xA	General initialization fault L1
W#16#84xB	Error Instance-DB allocation (not existing or 0)

Z_Info

Additional information for error code W#16#84xx from FB_RetVal

FB_RetVal error code	Additional information (Z_Info)
W#16#8301	Status of the FB P_RCV_WinAC or P_RCV *)
W#16#8302	Status of the FB P_SEND_WINAC or P_SEND *)
W#16#8305	Status of the FB Com_INIT_WINAC *)
W#16#830A	1 = Para DB = 0 or not existing
	2 = Para DB wrong version
	3 = Para DB Para for Com not found
	4 = Checksum fault Instance DBs
W#16#830B	1 = Instance DB No_P_RCV
	2 = Instance DB No_P_SEND
	3 = Instance DB No_V24_STAT
	4 = Instance DB No_V24_SET
	5 = Instance DB COM_INIT
	6 = Instance DB P_RESET

*) See also WinAC IP driver documentation (WinAcIpDoku) or S7 manual

More_Info_available (up to variant V1.4)

Only relevant at Master-Applications

The parameter **More_Info_available** signalizes that further information (from a blocked message) are directly available which could be handed over in one of the following cycle (refer also to the parameter **Next_Info)**.

For further details of increasing the flow rate at Master-Applications please refer to chapter 4.5.

.

More_info_is_spo (up to variant V1.4)

Only relevant at Master-Applications

More_Info_is_spo is an additional information permitting repetition cycles in dependency on the cause of transmission (refer also to the parameter **Next_Info)**

For further details of increasing the flow rate at Master-Applications please refer to chapter 4.5.

Buffer_Info_lost

Loss of information:
Indicates that minimum one message from the message memories has been lost (deleted).

With detected Link_Error applies:
The output is set as soon as information has been cleared.
The output is reset with disappearing Link_Error.

Note:
If the parameter **Buffer_Handling** = B#16#03: the **Buffer_Info_lost** is never set because in this case no information are cleared from the message buffers.

Time_DS	IN/OUT variable: Summer time bit of the current time. Displays the current time status, if the variable is being read exclusively. It can be affected when the assigned variable is forced. The bit is transferred in IEC messages with a time stamp direct into SU
Time_IV	IN/OUT variable: Invalidity bit of the current time. Displays the current time status, if the variable is being read exclusively. It can be affected when the assigned variable is forced. The bit is transferred in IEC messages with a time stamp direct into IV
Time_SY	IN/OUT variable: Synchronization status of the current time. Displays the current time status, if the variable is being read exclusively. It can be affected when the assigned variable is forced. The bit is not used in IEC messages
Time_Diff	Time difference in comparison to CPU base time. The value specified here [ms] is added to the current time in the CPU and the result is used as a time stamp for IEC messages. This means that time corrections such as local time calculations with CPU time on the basis of GMT can be implemented.

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

 ©SIEMENS AG 2017

3.3.3. Use of the Windows COM-Interface with WinAC

The selection of the Windows administrated interfaces for the use by WinAC occurs with the Communication-FB parameter ‚L1_Com_or_Laddr'. It is interpreted as follows: <=255: Interpretation as (Windows-) COM-Port. In this case the external interface driver is used. The parameterizing of the communication characteristics is carried out in a parameter data block (default DB90). It's DB-No. has to be assigned to the Communication-FB input ‚Para-DB_No'.

Additionally the external interface driver needs a number of instance DBs for its functional blocks. The allocation of the numbers is carried out in the parameter DB as well. The parameter DB is supplied with pre-settings which allow the direct use in conjunction with the delivered FC-, FB- and DBs without further adjustments.

The header part of the parameter DB serves for the general management (DB_Management, Information management) and global adjustments which are pre-adjusted and may not be changed.

Now one (default) or more parameter blocks are succeeding with the following design/contents:

- **Com-Port**:
  Corresponds with the Communication-FB input ‚L1_Com_or_Laddr' and indicates for which interface the succeeding parameters have to be used. The value 0 (default) means the use for all interfaces.
- **IDB_No.**:
  The numbers listed here refer to the instance data blocks of the CC Cologne driver blocks. They correspond with the DBs provided with the runtime software. Adjustments are only required if the DB numbers are changed.
- **COM_PARAMS**:
  Original parameter format of the provided driver for adjustment of the serial interface aligned to the setting options of a CP340. Pre-settings are: 9600 Baud, data format 8E1 (8 data bits, even parity 1 stop bit) without handshake.
  Note: In contrast to the CP340 the minimum character delay time has to be at least 20 ms.

DB90 Parameter

Addr.	Name	Type	Initial value	Actual value	Comment
0.0	DB_Manag.Pos_of_DB	BYTE	B#16#0	B#16#0	Reserved for future - do not change
1.0	DB_Manag.No_of_all_DBs	BYTE	B#16#0	B#16#0	Reserved for future - do not change
2.0	DB_Manag.DB_No_Act	WORD	W#16#0	W#16#0	Reserved for future - do not change
4.0	DB_Manag.DB_No_Prev	WORD	W#16#0	W#16#0	Reserved for future - do not change
6.0	DB_Manag.DB_No_Next	WORD	W#16#0	W#16#0	Reserved for future - do not change
8.0	DB_Manag.Reserved	WORD	W#16#0	W#16#0	
10.0	Info_Manag.P_Byte_First_Info	INT	26	26	(Byte-) Position of first Para-Block do not change
12.0	Info_Manag.No_of_Infos	INT	0	0	0; unspecified (DB is filled up to the end);otherwise size n of Array 1...n
14.0	Info_Manag.Len_Info	BYTE	B#16#46	B#16#46	Difference in byte between two Para Blocks do not change
15.0	Info_Manag.Len_Sort_Key	BYTE	B#16#0	B#16#0	0: without sorting; >0: Data sets are sorted ascending with x bytes
16.0	Info_Manag.Re_internal_usage	DWORD	DW#16#0	DW#16#0	
20.0	DB_Ident	WORD	W#16#232C	W#16#232C	Do not change
22.0	DB_Ver	WORD	W#16#100	W#16#100	Do not change
24.0	Checksum_IDBs	INT	820	820	Do not change
26.0	Para [1]. Com Port	INT	0	0	0: unspec.(para for any COM), 1-8 (future 1-255) para only for this COM
28.0	Para [1].IDB_No. P_RCV	INT	91	91	Instance DB for FB P_RCV_WinAC
30.0	Para [1].IDB_No. P_SEND	INT	92	92	Instance DB for FB P_SEND_WinAC
32.0	Para [1].IDB_No.V24_STAT	INT	93	93	Instance DB for FB V24_STAT_WinAC
34.0	Para [1].IDB_No.V24_SET	INT	94	94	Instance DB for FB V24_SET_WinAC
36.0	Para [1].IDB_No.COM_INIT	INT	95	95	Instance DB for FB COM_INIT_WinAC
38.0	Para [1].IDB_No.P_RESET	INT	96	96	Instance DB for FB P_RESET_WinAC
40.0	Para [1].IDB_No.res14	INT	0	0	

Addr.	Name	Type	Initial value	Actual value	Comment
42.0	Para [1].IDB_No. res16	INT	0	0	
44.0	Para [1].Res18	INT	0	0	
46.0	Para [1].COM_PARAMS. DB_Identifier	WORD	W#16#232C	W#16#232C	Identifier for this DB - Do not change
48.0	Para [1]. COM_PARAMS.COM_PROTOCOL End_Code_ofRcv_Mes_Frame	BTE	B#16#0	B#16#0	0-2=Expiry of Char Delay Time, Receipt of End Char(s), Receipt of fix.No. chars
50.0	Para [1]. COM_PARAMS.COM_PROTOCOL Character_Delay_Time	DNT	L#20	L#20	20-65530 ms - Caution: Will be used in all cases -
54.0	Para [1]. COM_PARAMS.COM_PROTOCOL End_ID_1	BYTE	B#16#2	B#16#2	7Bit: 00-7F (Hex)/8Bit: 00-FF (Hex) - Relevant, if End_Code_ofRcv_Mes_Fra = 1 -
55.0	Para [1]. COM_PARAMS.COM_PROTOCOL Enable End_ID_2	BYTE	B#16#0	B#16#0	0-1= deactivate, activate - Relevant, if End_Code_ofRcv_Mes_Fra = 1 -
56.0	Para [1]. COM_PARAMS.COM_PROTOCOL End_ID_2	BYTE	B#16#0	B#16#0	7Bit: 00-7F (Hex)/8Bit: 00-FF (Hex) - Relevant, if Enable_End_ID_2 = 1 -
58.0	Para [1]. COM_PARAMS.COM_PROTOCOL Message_Frame_Length	DINT	L#240	L#240	0-1024 Bytes - Relevant, if End_Code_ofRcv_Mes_Fra = 2 -
62.0	Para [1]. COM_PARAMS.COM_BAUDRATE_ CHARFRAME.Baud_Rate	DINT	L#2	L#2	0-6= 2400, 4800, 9600, 19200, 38400, 57600, 115200
66.0	Para [1]. COM_PARAMS.COM_BAUDRATE_ CHARFRAME.CharSize	BYTE	B#16#8	B#16#8	7-8 Bits
67.0	Para [1]. COM_PARAMS.COM_BAUDRATE_ CHARFRAME.StopBits	BYTE	B#16#1	B#16#1	1-2
68.0	Para [1]. COM_PARAMS.COM_BAUDRATE_ CHARFRAME.EnableParity	BYTE	B#16#1	B#16#1	0-1= deactivated, activated
69.0	Para [1]. COM_PARAMS.COM_BAUDRATE_ CHARFRAME.Parity	BYTE	B#16#2	B#16#2	0-3= none, odd, even, any
70.0	Para [1]. COM_PARAMS.COM_BAUDRATE_ CHARFRAME.Break_Monitoring	BYTE	B#16#0	B#16#0	0-1= deactivated, activated
72.0	Para [1]. COM_PARAMS.COM_FLOW CONTROL.Data_Flow_Control	BYTE	B#16#0	B#16#0	0-3= without, XON/XOFF, RTX/CTS, Automat. Use of V24-Signals
73.0	Para [1]. COM_PARAMS.COM_FLOW CONTROL..XonChar	BYTE	B#16#11	B#16#11	7Bit: 00-7F, 8Bit: 00-FF
74.0	Para [1]. COM_PARAMS.COM_FLOW CONTROL..XoffChar	BYTE	B#16#13	B#16#13	7Bit: 00-7F, 8Bit: 00-FF
76.0	Para [1]. COM_PARAMS.COM_FLOW CONTROL.Wait_Time_1	DINT	L#2000	L#2000	Wait for XON after XOFF (Wait Time for CTS=ON) 20-65530 ms (10 Steps
80.0	Para [1]. COM_PARAMS.COM_FLOW CONTROL.Wait_Time_2	DINT	L#70	L#70	Time to RTS OFF (only if Data_Flow_Control = Auto) 0-65530 ms (10 Steps)
84.0	Para [1]. COM_PARAMS.COM_FLOW CONTROL.Wait_Time_3	DINT	L#10	L#10	Data Output Waiting Ti (only if Data_Flow_Control = Auto) 0-65530 ms (10 Steps)
88.0	Para [1]. COM_PARAMS.COM_RECEIVE BUFFER_Delete_Buffer	BYTE	B#16#1	B#16#1	Delete Receive Buffer on Startup 0-1: yes, no
90.0	Para [1]. COM_PARAMS.COM_RECEIVE BUFFER_Buffer_Size	DINT	L#2	L#2	Buffered Receive Message Frames 1-250
94.0	Para [1]. COM_PARAMS.COM_RECEIVE BUFFER_Prevent_Overwriting	BYTE	B#16#1	B#16#1	0-1= no, yes 0, only if Buffer_Size = 1

Use of several Interfaces

If several interfaces are used, by calling multiple instances of the (Communication-FB) software the following information has to be observed:

If no other interface characteristics are required the existing parameter block can be used 1 to 1

If other communication adjustments are required, either an additional parameter block has to be established in the existing parameter DB (extension of the array ‚Para') or the complete DB has to be copied and its new number has to be handed over to the Communication-FB.

In both cases the number of the allocated instance DB may remain unchanged, because the interface drivers can operate with one and the same instance.

If using only one DB with then several parameter blocks these have to be arranged in a way that the parameter block can certainly be found. If any parameter block with unspecified Com-Port (COM=0) is registered it should be at the end.

Note:

Changes in the parameter DB values always have to be carried out in the ***actual values*** in the ***data view***.

3.3.4. Link-Address parameter-DB for line operation

From version V1.2 also line operation with several stations at a polling line is supported.

The link addresses of the stations at the line have to be different.

If there is more than one station connected to one channel (line operation) the parameterizing of the link addresses is carried out via a link address parameter data block. The number of this data block has to be stated at the input parameter of the ‚L2_Link_Address' from the S7_IEC_Config-Block.

If the address setting has to be carried out via a link address parameter DB, an offset of 100000000 (eight zeros) has to be entered there.

If the addresses are taken from the DB111 for example, in the ‚L2_Link_Address' the value L#100000111 has to be entered.

In the parameter DB the link addresses have to be entered in the sequence of the stations to be polled. Single stations can be tagged as reserve by setting the entry ‚Link_Reserved' in the parameter DB to ‚TRUE'.

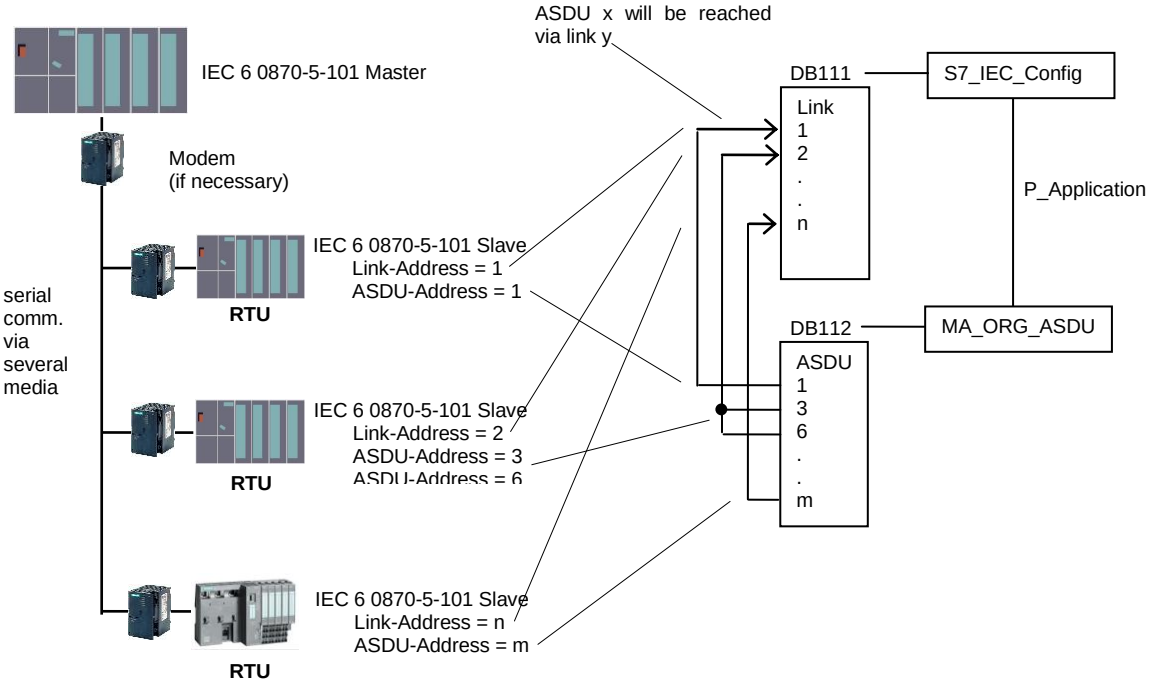
In case of processing information with different ASDU addresses from one device (defined via the link address) the use of an ASDU address parameter DB is provided. For this you find a detailed description in chapter 6.4.2.

System related the maximum number of the link and ASDU addresses is only limited by the DB length. Up to 8 link and ASDU addresses have been tested.

A broad summary of the physically and program technically coherences is shown in the graphic below.

The assignment from the ASDU addresses to the link addresses is necessary:

- in control direction for searching the path,
- in monitoring direction for trouble shooting of single sub stations (RTU)



Polling structure for several stations (links) and several ASDU-Addresses per station.

Structure and values of the Parameter-Data block:

The Parameter-DB has the following basic structure:

Group/ Parameter	Addr. Rel.	Addr. Abs.	Type	Initial value	Comment
DB_Manag	0	0	STRUCT		
DB_Manag. **Pos_of_DB**	+0.0	0.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **No_of_all_DBs**	+1.0	1.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Act**	+2.0	2.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Prev**	+4.0	4.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Next**	+6.0	6.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **Reserved**	+8.0	8.0	WORD	W#16#0	reserved for future - do not change!
	+10.0	10.0	END_STRUCT		END_STRUCT (10 BYTE)
Info_Manag			STRUCT		
Info_Manag. **P_Byte_First_Info**	+0.0	10.0	INT	34	(Byte-) Position of first Para-Block - do not change!
Info_Manag. **No_of_Infos**	+2.0	12.0	INT	0	0: unspecified (DB is filled up to the end); otherwise size n of ARRAY [1..n]
Info_Manag. **Len_Info**	+4.0	14.0	BYTE	B#16#4	Difference in [byte] between two Para-Blocks - do not change!
Info_Manag. **Len_Sort_Key**	+5.0	15.0	BYTE	B#16#0	0: without sorting; > 0: Data sets are sorted ascending with x Bytes
DB_Manag. **Re_internal_usage**	+6.0	16.0	DWORD	DW#16#0	
	+10.0	20.0	END_STRUCT		END_STRUCT (10 BYTE)
Global			STRUCT		
Global. **Link_Para_Type**	0.0	20.0	BYTE	B#16#0	reserved for future - do not change
Global. **Class_1_Requests**	1.0	21.0	BYTE	B#16#3	Parameter: Max no. of class 1 requests for a Station before going to the next
Global. **Act_Parablock**	2.0	22.0	INT	0	actual processed Para block at runtime -> 'read only'
	4.0	24.0	END_STRUCT		END_STRUCT (4 BYTE)
Link_Para			ARRAY [0..n] STRUCT		n = number -1 of the Parameter entries
Link_Para **Link_Error**	+0.0	24.0	BOOL	TRUE	actual state of this link connection at runtime -> 'read only'
Link_Para **Link_Reserved**	+0.1	24.1	BOOL	FALSE	Parameter: Set if you actually don't want to use this link
Link_Para **Res_1**	+1.0	25.0	BYTE	B#16#0	
Link_Para **Link_Address**	+2.0	26.0	DINT	L#0	Parameter: Link address
Link_Para **Res_2**	+6.0	30.0	INT	0	
Link_Para **PRM_State**	+8.0	32.0	BYTE	B#16#0	Internal
Link_Para **L2_Send_FCB**	+9.0	33.0	BOOL	FALSE	Internal
Link_Para **L2_PRM0_ACD**	+9.1	33.1	BOOL	FALSE	Internal
Link_Para **Res_3**	+10.0	34.0	DINT	L#0	
Link_Para **Res_4**	+14.0	38.0	INT	0	
	16.0	40.0	END_STRUCT		END_STRUCT (16 BYTE)

	These values are pre-adjusted in the sample DB and have not to be changed. A sample DB is enclosed in the particular block libraries. DB111 = ‚P_LinkAdr_n'.
	!!! These parameters have to be adjusted !!!
	Note: Changes have to be performed always in the ‚Data view', and not in the ‚Declaration view' (except dimensioning of the array).

 ©SIEMENS AG 2017

DB_Management	Parameter blocks which include an enumeration are created so, that more than one data block can be used if required. The following information is required for management. Currently the DB management is not used – for this reason all parameters can be left at their default settings (0)
Pos_of_DB	Number (1 – n) of the current data block for the parameter block 0: not used
No_of_all_DBs	Total number (n) of data blocks for the parameter block 0: not used
DB_No_Act	DB number of the current data block 0: not used
DB_No_Prev	DB number of the previous data block, if one exists. Otherwise: 0
DB_No_Next	DB number of the subsequent data block, if there is another one. Otherwise: 0
Info_Management	Parameter blocks containing an enumeration have one information block 'Info management'. It contains following information The values of the information management parameters are already preset and therefore do not need to be modified.
P_Byte_First_Info	Byte position of the first data record in this DB, may not be changed
No_of_Infos	Number of data records contained in this DB 0: unspecified -> the data block end marks the last data record
Len_Info	Length of the data records: In this case 4: may not be changed
Len_Sort_Key	0: no sorting (default setting) Other sort lengths are not supported when using as parameter DB for Individual IEC addressing.
Global Parameter	
Class_1_Requests	Max. number of class 1 requests until switching forward to the next station.
Act_Parablock	This value gives information about the actual processed parameter block. -> access only reading.
Link_Para	The dimensioning of the array happens in the declaration layer of the data block. It has to be carried out according to the number of stations which have to be requested. n+1 stations are requested.
Link_Reserved	Deactivates the request for this station. FALSE: The station will be requested TRUE: The station will be not requested (reserve)
Link_Address	Link_address is the value of the link layer address The permissible range for link address length 0 is 0 The permissible range for link address length 1 is 1 up to 254 The permissible range for link address length 2 is 1 up to 65534

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

4. Communication with – IEC 60870-5-104

4.1. Overview

Main feature of IEC 60870-5-104 is its network capability. SIPLUS RIC IEConS7 supports the network link via:

- CPs (communication processors) CP343 and CP443
- PROFINET®-interfaces of the CPUs 31xPN,41xPN
- PROFINET®-interface of the ET 200S IM151-8PN/DP CPU
- PROFINET®-interfaces of the Software Controllers (ET 200SP-CPU; IPC427D, WinAC) and Ethernet interface(s) of the host computer.
- PROFINET®-interfaces of the CPUs 15xx (S7-1500/ET 200SP), CM1542 and CP1543-1 / CP1543SP-1

The different hardware results in an important difference in the 'connection configuration'. Whereas TCP connections are implemented via the interfaces integrated into the CPU and the CP1543 exclusively via programming (in this case, parameter inputs on Communication-FB, multiple connections are each differentiated by *_C1, _C2*). Configuration in NETPRO is required for connections via CP343 and CP443.

Parameters such as connection partners, etc. are set here. For more information we refer you to the relevant chapter 4.2.1. Separate IP addresses, and where applicable, subnet masks, gateway details, etc. generally need to be carried out in the hardware configuration.

The master/slave distinction with regard to telecontrol connection with the T104 protocol means:

- On the network level:
  whether TCP connections are established actively (master) or are passively maintained on incoming connections (slave).

  Multiple simultaneous connections are permissible. This 'connection redundancy' is designated as redundancy group and supported from V1.1 by the Slave and Master versions.
  2 TCP connections can be held. In case of passive TCP link connection via one interface they must have different port numbers (default 2404 and 2405).

  The Profinet-interfaces of all current CPUs (S7-1500, ET 200SP, S7-300/ET 200S V3.2, S7-400 V6), as well as the CM1542/CP1543 permit the multiple use of Port 2404.

NOTICE

Interfaces of the **Communication-FBs** (‘S7_IEC_Config’) which are not in use shall be locked; otherwise there might be a security issue!

In case of Communication-FBs for S7-1500 / ET 200SP the interfaces will be locked by input ‘0’ at the parameter ‘L_InterfaceID_Cx. The parameter input for all other CPU-types is named ‘L1_TCP_Port_Number_Cx’ (x=1 or 2).

Alternatively there is the option:

- to project active TCP link connection (from V1.1 possible in all variants)
- to communicate via 2 CPs with different IP addresses – the port number in this case may remain the same.

- On the link level, more specifically APCI (transport interface):
  It is used for data security and control of the data flow.

  Masters initiate data traffic using 'START_DT'. They are also called ‘Controlling Station’. Slaves accept user data after receiving 'START_DT' and send their own user data via the connection which received 'START_DT' last. Slaves are ‘Controlled Stations’.

SIPLUS RIC on Software Controller with WinAC supports the network connection via the PN-Interface from

- mEC31-RTX (WinAC 2010)
- nanobox-PC with RTX
- microbox-PC with RTX

further LAN interfaces administrated from Windows via an additional driver WinAC_IP_V217 which has to be installed on mEC31 resp. microbox-PC (nanobox-PC).

SIPLUS RIC on Software Controller supports the network connection over LAN or rather PN interfaces of:

- ET 200SP PC with open controller
- IP427D with Software Controller

The selection is carried out via the Communication-FB parameter ‚L1_IF_Slot_or_Para_ID_Cx' which is available for each connection

 ©SIEMENS AG 2017

The following interfaces are available:

Software-Variant	Released Hardware	
S7LIB: S7IEC_M104_CP343 SIPLUS RIC IEConS7 IEC 60870-5-104 Master via CP343-1 for standard CPUs S7-300 **S7LIB: S7IEC_S104_CP343** SIPLUS RIC IEConS7 IEC 60870-5-104 Slave via CP343-1 for standard CPUs S7-300	CP 343-1 LEAN CP343-1 CP 343-1 ADVANCED	6GK7343-1CX10-0XE0 6GK7343-1EX30-0XE0 6GK7343-1GX31-0XE0
S7LIB: S7IEC_M104_CP443 SIPLUS RIC IEConS7 IEC 60870-5-104 Master via CP443-1 for Standard CPUs S7-400 and S7-400H **S7LIB: S7IEC_S104_CP443** SIPLUS RIC IEConS7 IEC 60870-5-104 Slave via CP443-1 for standard CPUs S7-400 and S7-400H	CP443-1 CP443-1 Advanced	6GK7443-1EX30-0XE0 6GK7443-1GX30-0XE0
S7LIB: S7IEC_M104_31xPN SIPLUS RIC IEConS7 IEC 60870-5-104 Master via CPU PN interface for standard CPUs S7-300 with PN interface **S7LIB: S7IEC_S104_31xPN** SIPLUS RIC IEConS7 IEC 60870-5-104 Slave via CPU PN interface for standard CPUs S7-300 with PN interface	CPU314C-2PN/DP CPU315-2 PN/DP CPU315F-2 PN/DP CPU317-2 PN/DP CPU317F-2 PN/DP CPU319-3 PN/DP CPU319F-3 PN/DP	6ES7314-6EH04-0AB0 6ES7315-2EH14-0AB0 6ES7315-2FJ14-0AB0 6ES7317-2EK14-0AB0 6ES7317-2FK14-0AB0 6ES7318-3EL01-0AB0 6ES7318-3FL01-0AB0
S7LIB: S7IEC_M104_41xPN SIPLUS RIC IEConS7 IEC 60870-5-104 Master via CPU PN interface for standard CPUs S7-400 with PN interface **S7LIB: S7IEC_S104_41xPN** SIPLUS RIC IEConS7 IEC 60870-5-104 Slave via CPU PN interface for standard CPUs S7-400 with PN interface	CPU 412-2 PN CPU 414-3 PN/DP CPU 414F-3PN/DP CPU 416-3 PN/DP CPU 416F-3 PN/DP CPU417-4 PN/DP CPU 412-5H CPU 414-5H CPU 416-5H CPU 417-5H CPU 410-5H	6ES7412-2EK06-0AB0 6ES7414-3EM06-0AB0 6ES7414-3FM06-0AB0 6ES7416-3ES06-0AB0 6ES7416-3FS06-0AB0 6ES7417-4XT05-0AB0 6ES7412-5HK06-0AB0 6ES7414-5HM06-0AB0 6ES7416-5HS06-0AB0 6ES7417-5HT06-0AB0 6ES7410-5HX08-0AB0
S7LIB: S7IEC_M104_151PN SIPLUS RIC IEConS7 IEC 60870-5-104 Master via CPU PN interface for ET 200S IM151-8 PN/DP CPUs **S7LIB: S7IEC_S104_151PN** SIPLUS RIC IEConS7 IEC 60870-5-104 Slave via CPU PN interface for ET 200S IM151-8 PN/DP CPUs	IM151-8 PN/DP CPU IM151-8F PN/DP CPU	6ES7151-8AB01-0AB0 6ES7151-8FB01-0AB0
S7LIB:S7IEC_M104_WinAC SIPLUS RIC WinAC IEC 60870-5-104 Master for mEC31, nanobox-PC and microbox-PC IEC-Communication via PN-Interface **S7LIB:S7IEC_S104_WinAC** SIPLUS RIC WinAC IEC 60870-5-104 Slave for mEC31, nanobox-PC und microbox-PC IEC-Communication via PN-Interface	EC31-RTX (WinAC 2010) Microbox-PC with RTX Nanobox-PC with RTX	6ES7677-1DD10-0BB0 6ES7675-1DF30-0DB0

Software-Variant	Released Hardware	
S7LIB: **S7onIEC 15xxCPU_Vx.y / S7IEC_104_MS_CPU_PN** SIPLUS RIC IEConS7 IEC 60870-5-104 Master/Slave via CPU PN interface for standard CPUs S7-1500 with PN interface	CPU1511-1 PN CPU 1511C-1PN CPU1511F-1 PN CPU 1512C-1PN CPU1513-1 PN CPU1513F-1 PN CPU1515-2 PN CPU1515F-2 PN CPU1516-3 PN/DP CPU1516F-3 PN/DP CPU1517-3 PN/DP CPU1517F-3 PN/DP CPU1518-4 PN/DP CPU1518F-4 PN/DP	6ES7511-1AK00-0AB0 6ES7511-1CK00-0AB0 6ES7511-1FK00-0AB0 6ES7512-1CK00-0AB0 6ES7513-1AL00-0AB0 6ES7513-1FL00-0AB0 6ES7515-2AM00-0AB0 6ES7515-2FM00-0AB0 6ES7516-3AN00-0AB0 6ES7516-3FN00-0AB0 6ES7517-3AP00-0AB0 6ES7517-3FP00-0AB0 6ES7518-4AP00-0AB0 6ES7518-4FP00-0AB0
S7LIB: **S7onIEC 15xxCPU_Vx.y / S7IEC_104_MS_CPU_PN** SIPLUS RIC IEConS7 IEC 60870-5-104 Master/Slave via CPU PN interface for standard CPUs ET 200SP with PN interface	CPU1510SP-1 PN CPU1510SP F-1 PN CPU1512SP-1 PN CPU1512SP F-1 PN	6ES7510-1DJ00-0AB0 6ES7510-1SJ00-0AB0 6ES7512-1DK00-0AB0 6ES7512-1SK00-0AB0
S7LIB: **S7onIEC 15xxSC_Vx.y / S7IEC_104_MS_CPU_PN** SIPLUS RIC IEConS7 IEC 60870-5-104 Master/Slave for Standard CPUs ET 200SP PC	CPU 1515SP PC	6ES7 677-2AAxx-0xB0
S7LIB: **S7onIEC 15xxSC_Vx.y / S7IEC_104_MS_CPU_PN** SIPLUS RIC IEConS7 IEC 60870-5-104 Master/Slave for Standard CPUs IPC427D with Software Controller (SC)	IPC427D PN/IE mit CPU 1507S	6AG4140-xxxxx-xxxx 6ES7 672-7AC00-0YA0

In each case the library contains a central Communication-FB 'S7_IEC_Config' (Communication-FB or rather S7IEC_104_MS_CPU_PN (FB104) with S7-15xx, its corresponding subordinated auxiliary blocks, as well as the associated 'application blocks' for the process connection.

The Communication-FB has to be programmed with your parameters (see chapter 4.2) and takes care of the entire IEC communication with the connection partner.

A pointer in DWord format establishes the connection/interface to the 'application blocks'. You use only the blocks with the required functionality which can be used on the other hand multiple too.

Limits are only set here by the memory and resource consumption (cycle time). The application blocks (see chapter 6) are separated by master (e.g. sending commands, output of indication and measurement values) and slave functionality (e.g. record of indication and measurement values, output of commands).

The blocks for CP443-1 are usable in standard S7-400 systems as well as in S7-400H systems. For more details using the blocks in an H-system, refer to chapter 9.

4.2. Settings in the SIMATIC Manager and TIA-Portal

The necessary settings in the hardware configuration and in the connection configuration are described here.

4.2.1. Hardware configuration CP343 and CP443

For T104 connections it is normally not necessary to carry out any other settings (other than the default settings) with the exception of the following properties dialog. If necessary, optional according to the notes in the online help.

The following settings must be carried out:

According to the allocation by your system administrator:

- Own IP address
- Subnet mask
- Router parameter
- Port number 2404

The port number 2404 can be used only once with passive connection establishment with the CP343-1.

CP443-1 can use the port number 2404 with passive connection establishment since firmware status V3.2.12 several times.

CM1542 and CP1543 can use the port number several times.

4.2.2. Hardware configuration CPU31x PN, CPU41x PN and ET 200S IM151-8 PN

As a minimum, the settings in the shown properties dialog for the integrated PN-IO interface has to be carried out:

According to the allocation by your system administrator:

- Own IP address
- Subnet mask
- Router parameter
- Port number 2404

Other settings can be carried out (only if required) in the 'PN-IO/Options' properties dialog. Refer also to the relevant part of the online help.

 ©SIEMENS AG 2017

4.2.3. Hardware configuration for WinAC in the SIMATIC Manager

The SIMATIC Mircobox PC (Nanobox PC) and the embedded controller must be inserted as PC-station

WinLC RTX on SIMATIC Microbox PC (Nanobox PC)

The SIMATIC Manager Step 7 V5.5 is mandatory required

In the Hardware catalog under SIMATIC PC station/controller/WinLC RTX/6ES7611-4SB00-0YB7/V4.6

you have to insert the corresponding CP. The V4.6 corresponds to WinAC RTX2010.

WinLC RTX on SIMATIC S7 modular Embedded Controller EC31

The SIMATIC Manager Step 7 V5.5 is mandatory required

In the Hardware catalog under SIMATIC PC Station/Controller/S7-modular embedded Controller/EC31/6ES76777-1DD00-0Bx0/V1.1/CP5603. The V41-1 corresponds to WinAC RTX2010.

Setup of Windows LAN-Interface

WinLC supports a LAN resp. PN-Interface on the Microbox PC, Nanobox PC and mEC31, which can be used also for the transmission of the telecontrol protocol IEC 60870.5.104. However, only one interface is assignable.

Optionally further LAN-Interfaces administrated from Windows can be used for the transmission. In this case first of all the installation of a corresponding driver on the Microbox PC (Nanobox PC) resp. mEC31 is required.

You'll receive the driver for the LAN interface on a CD. It is in the ‚WinAC_TCP_Driver' directory together with an ‚Install.bat' file. The corresponding connection parameters are provided by default in the DB80 (see chapter 4.3.3).

When executing the **Install.bat** file the driver-dll (WinLC_IP_TCOM.dll) will be installed in the Windows Sytsem32 directory of the Microbox PC (Nanobox PC) resp. mEC31 and is now ready for use.

4.2.4. Hardware configuration for S7-1500

As a minimum, the settings in the shown properties dialog for the integrated PN-IO interface has to be carried out:

According to the allocation by your system administrator:

- Own IP address
- Subnet mask
- Router parameter
- Port number 2404

Other settings can be carried out (only if required) in the 'PN-IO/Options' properties dialog. Refer also to the relevant part of the online help.

4.2.5. S7-1500 Software Controller

The communication is possible via the PN interface assigned to the software controller and the PC interface, and simultaneously within a redundancy group.

Note

Currently (Stand Software Controller V1.8), no 'multiport usage' is supported within an interface. Several (passive) TCP connections within an interface must therefore differ in the port number.

4.2.6. Connection configuration CP343-1 and CP443-1

SIMATIC Manager

- Start NETPRO from the Simatic Manager (CPU/connections).
- Select the CPU
- Insert a new connection (Menu/Insert/New connection)
- The 'new connection'; is a **TCP connection** to an **unspecified** partner

From the properties dialog that will appear, you can transfer/note the module parameters 'ID' and 'LADDR' required later on.

TIA-Portal

- Go to device configuration and start Network view
- Select the CPU
- Activate connections
- Select the TCP-connection
- add new connection (CPU right mouse button)
- the “new connection” is a TPC-connection to an unspecific partner

From the the properties, which are hidden now, you can assume/note the required block parameters “ID” and LADDR”.

Settings for passive link connection (standard slave configurations):

- No active connection establishment
- Local port: → typical for IEC: 2404
- Partner IP:
  - None → any partner
  - Set → only this partner is permissible
- Partner-Port:
  - None → any port
  - Set → only this port is permissable

The CP443 is able to handle the port number 2404 several times as from firmware V3.2.12.

Settings for active link connection (standard master configuration)

- Active connection establishment
- Partner IP and port number must be entered
- Local port number: irrelevant with an active connection established, but must be explicit assigned → any number.

4.2.7. Project planning Speed-Communication CP443

The CP443 modules support a so called speed-communication for an accelerated data transmission between CPU and CP. The CPU needs minimum FW status V5.1. For further information please refer to the module manuals.

SIPLUS RIC supports the function from version V1.6 by providing of own libraries;

- S7IEC_S104_CP443_Sp
- S7IEC_M104_CP443_Sp

Generally the project planning is carried out like at CP443. Additionally the following adjustments have to be carried out.

SIMATIC Manager

In the hardware configuration CP443:

- activate adress settings for Lock/UNLOCK with FETCH/WRITE, SPEED Send/Receive

In the project planning for the connection in NetPro:

- activate Mode SPEED Send/Recv

TIA-Portal

In the device configuration / device view/E/A-adresses:

- - activate address settings for Lock / UNLOCK with FETCH / WRITE, SPEED Send / Receive

In Network View / Connections / Options:

- Operating mode: activate SPEED Send / Recv

 ©SIEMENS AG 2017

4.3. Configuration of Communication-FB 'S7_IEC_Config'

(FB100/ FB104 with CPU15xx)

The Communication-FB can be called up directly from the OB1 or also integrated in the CFC.

Note:

The **Communication-FB block** ('S7_IEC_Config') is designed exclusively for the **OB1 priority level**. The application blocks also need to be operated in priority level OB1.

The block library contains one DB100 as a pre-defined instance DB for Communication-FB with the symbolic name 'S7_IEC'. When calling up the Communication-FB manually you can use this DB directly (e. g. Call FB100, DB100). If you prefer or require other DB numbers, or you are working with CFC (automatic DB assignment), you can delete DB100.

In addition, you need to update the accompanying variable table 'VAT_S7_IEC' if you want to use this. In this case you have to assign a symbolic name to the 'new instance DB and change the DB numbers in the variable table (DB100.x → Dbnew.x).

In the figure below the exemplarily illustration of some S7_IEC config blocks in the CFC plan are shown, and that followed a detailed list of the input and output variables with their brief comments, and a detailed description.

Attention when using in the S7-1500:

The instance DB must be stored in the remanent memory, so that the data blocks are not generated several times after stop start or voltage reversal

Input: Open the data block and activate the Remanence column.

Alternatively, the necessary areas of the DBs can be placed into the remanent memory by means of S7_IEC_Rem_Mem (FB126).

NOTICE

The CPU410 uses a maximum of 256kB RAM memory for online generated data blocks.

The loading and operating of the CPU410 is only possible in PCS7-system

 ©SIEMENS AG 2017

IEC60870-5-104 Master und Slave

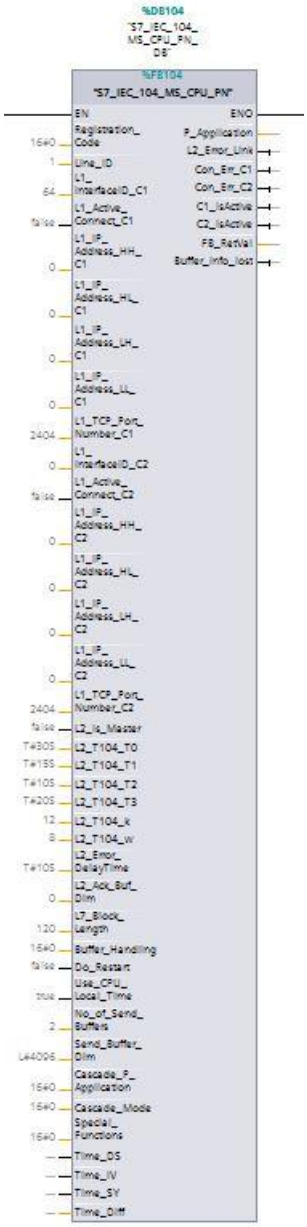


Illustration as FUP-block with default values

IEC60870-5-104 Master

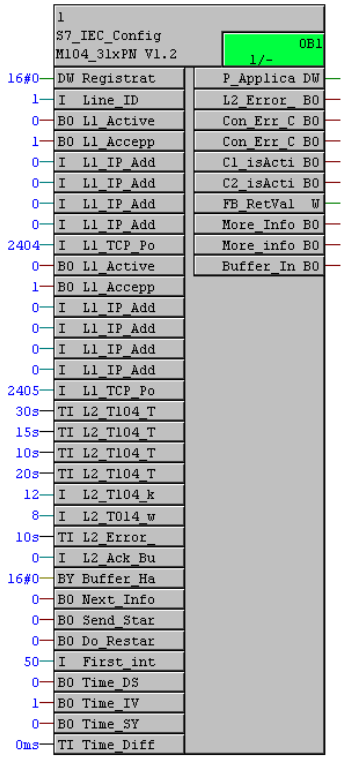


Illustration as CFC-block with default values

IEC60870-5-104 Slave

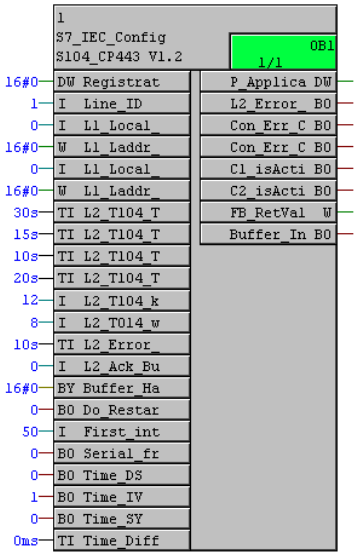


Illustration as CFC-block with default values

Note to the CFC plan view

The CFC views included in the manual are only used for illustration purposes. CFC is **not** required for using the blocks. The standard programming options in SIMATIC are sufficient.

NOTICE:

In case you project with CFC (e.g. within PCS7) please observe that data blocks created from SIPLUS RIC during runtime are not overwritten from CFC compiler. See also parameter ‘First_internal_DB_No’.

For each Communication-FB two (TCP-) connections are supported. The parameters are marked with -C1 resp. –C2.

At the CP variants only the allocation of the connection description created in NetPro and the module address of the CPs is necessary. In all other variants the connection parameter are parameterized at the block. For each connection separately the following adjustments are possible:

- Connection set-up passive (typically Slave) or active (typically Master)
- In case of active connection set-up the target address is the IP-address of the partner. In case of passive connection set-up the IP-address of the partner is the only accepted IP-address.
- Own port number (passive set-up) resp. Port number of the partner (active set-up)

Both connections form a redundancy group.

In case of Master (controlling station) the block transmits a StartDT_Act-message via the first successfully set-up connection. With receiving StartDT_Con the connection is process leading. This status is signaled at the output Cx_isActive. Data are exclusively exchanged on this connection. Further set-up connections are monitored with test frames.

In case of Slave (controlled station) the block waits on for the activation of the data transmission by StartDT_Act. The receiving of StartDT_act is confirmed and signaled at the output Cx_isActive. Data traffic is carried out with the communication partner which has transmitted StartDT at last – this means always via exactly one connection.

From V1.6 Master and Slave blocks are standardized and can be switched over in its function (parameter L2_is_Master)

Note

By cascading of several S7_IEC_Config blocks it is possibe to manage further TCP connections. Either to expand the standard conformable redundancy group or to communicate via several connections in parallel. For further information see chapter 5.

Note

A passive connection set-up of several connections via the same physical channel with the same port number (typically 2404) is only possible via the PN interfaces of the following hardware: S7-1500, ET 200SP, S7-300/ET 200S from V3.2, S7-400 from V6, WinAC, Station Controller from V2.0.
With older versions, CP343 and CP443 and LAN interfaces of WinAC laying under Windows this function is not supported. There the used port numbers have to be different (default 2404 and 2405). If applicable an active connection set-up can be used.

NOTICE

On WinAC-platforms (mEC31, microbox-PC, nanobox-PC) a change of the Windows firewall setup is mandatory!

The standard setup of the windows firewall blocks all incoming TCP connections.

Because of that the L1_TCP_Port_Number_C1 (Default 2404) and L1_TCP_Port_Number_C2 (Default 2405) first of all have to be unlocked.

The unlocking is carried out under: <System control/Windows Security center/Windows Firewall/Exceptions/Add port>

Enter the name here (e.g. IEC) and the port number (2404 resp. 2405) and complete the entry with OK. The port will be recorded in the firewall under “Exceptions“ and can be used now.

4.3.1. IEC 60870-5-104 parameter

Communication-FB VAR_INPUT		Type	Default	Block-comment/Short description	
Registration_Code	IN	DWORD	DW#16#0	!!! IMPORTANT !!! See product information	
Line_ID	IN	INT	1	for differentiation of more than one line;1-x(see manual); Default: 1;	
L1-Parameter at Communication via* CP343 and CP443*:					
L1_Local_ID_C1	IN	INT	0	see NetPro Connection Property 'ID' of your TCP-Connection	
L1_Laddr_C1	IN	WORD	W#16#0	see NetPro Connection Property 'LADDR' of your TCP-Connection	
L1_Local_ID_C2	IN	INT	0	see NetPro Connection Property 'ID' of your TCP-Connection	
L1_Laddr_C2	IN	WORD	W#16#0	see NetPro Connection Property 'LADDR' of your TCP-Connection	
L1-Parameter at Communication via PN-interface of CPU or WinAC:					
L1_Active_Connect_C1	IN	BOOL	FALSE	0 (default): we are passive (listen); 1: we (active) try to establish the con...	
L1_Acceppt_All_IP_Adr_C1	IN	BOOL	TRUE	0:Only one defined IP-Addr. ... 1 (Default): Each IP-Address will be accepted	
L1_IF_Slot_or_Para_ID_C1	IN	BYTE	B#16#1	IF_Slot (1-4) of PLC Ethernet Contr. or Para_ID for ext IP Drv (≥ 80); Def: 1	
L1_IP_Address_HH_C1	IN	INT	0	xxx-Part of IP-Address xxx.0.0.0 ; only needed if 'accept all' = false	
L1_IP_Address_HL_C1	IN	INT	0	xxx-Part of IP-Address 0.xxx.0.0 ; only needed if 'accept all' = false	
L1_IP_Address_LH_C1	IN	INT	0	xxx-Part of IP-Address 0.0.xxx.0 ; only needed if 'accept all' = false	
L1_IP_Address_LL_C1	IN	INT	0	xxx-Part of IP-Address 0.0.0.xxx ; only needed if 'accept all' = false	
L1_TCP_Port_Number_C1	IN	INT	2404	Connections will be accepted or established with this port (Def:2404)	
L1_Active_Connect_C2	IN	BOOL	FALSE	0 (default): we are passive (listen); 1: we (active) try to establish the con...	
… _C2	IN			Parameter für 2te TCP-Verbindung (wie _C1)	
L1_TCP_Port_Number_C2	IN	INT	2405	Connections will be accepted or established with this port (Def:2405)	
L1-Parameter at Communication via PN-Interface(s) of S7-15xx (incl. CM1542 and CP1543):					
L1_InterfaceID_C1	IN	HW_ANY	64	ID of the HW-Interface you would like to use for tcp communication; 0: Channel deactivated; Default: 64=Local_PN_1	
Further L1-Parameter at S7-400 for selection of the PN-interface (CPU410, H-System):					
L1_Local_Device_Id_C1	IN	BYTE	B#16#5	ID for the local PN/IE interface: 1.IF Rack0/1:16#05(def)/15; 2.IF:16#08/18	
L1_Local_Device_Id_C2	IN	BYTE	B#16#5	ID for the local PN/IE interface: 1.IF Rack0/1:16#05(def)/15; 2.IF:16#08/18	
Further L1-Parameter at Communication via WinAC:					
L1_IF_Slot_or_Para_ID_C1	IN	BYTE	B#16#1	IF_Slot (1-4) of PLC Ethernet Contr. or Para_ID for ext IP Drv (≥ 80); Def: 1	
L1_IF_Slot_or_Para_ID_C2	IN	BYTE	B#16#1	IF_Slot (1-4) of PLC Ethernet Contr. or Para_ID for ext IP Drv (≥ 80); Def: 1	

Communication-FB VAR_INPUT		Type	Default	Block-comment/Short description	
L2_is_Master	IN	Bool	True/False	0: Slave; 1: Master; Default: 0	M/S
L2_T104_T0	IN	Time	T#30S	1-255 sec/Def:30/Time-out of connection establishment	
L2_T104_T1	IN	Time	T#15S	1-255 sec/Def:15/Time-out of send or test APDUs	
L2_T104_T2	IN	Time	T#10S	1-255 sec/Def:10/Time-out for acknowledges in case of no data messages t2<t1	
L2_T104_T3	IN	Time	T#20S	1s-48h/Def:20s/Time-out for sending test frames in case of a long idle state	
L2_T104_k	IN	INT	12	>1/Def:12/Maximum difference receive sequence number to send state variable	
L2_T014_w	IN	INT	8	>1/Def:8/Latest acknowledge after receiving w I-format APDUs	
L2_Error_DelayTime	IN	TIME	T#10S	Def: 10 sec: after this time L2_Error_Link will be set, if no con. is active	
L2_Ack_Buf_Dim	IN	INT	0	0(def): without; <>0 (we suggest 1600): dimension [byte] of ack-buffer;	
L7_Block_Len	IN	INT	120	max. length of T101/104-ASDU (1..255, see manual); Default 120	
Buffer_Handling	IN	BYTE	B#16#0	0(def): compl, 1:compl. if ov 2:oldest Info will deleted; 3:no act.	
Next_Info	IN	BOOL	FALSE	Only allowed for additional calls in a Cycle ->reduced operation -> next info	*5)
Send_StartDT	IN	BOOL	FALSE	a rising edge will send a StartDT telegram (only as master and if tcp-conn)	
Use_CPU_Local_Time	IN	BOOL	TRUE	1(def): Local Time of CPU incl. DST-Bit 0: system time New V1.6 Upd4	*6)
Do_Restart	IN	BOOL	FALSE	a rising edge restarts the IEC-Application	
Serial_from_CPU	IN	BOOL	FALSE	for Registration use serial number from CPU instead of memory card	*1)
First_internal_DB_No	IN	INT	50	First DB-Number for DBs created by the Runtime Software; Default: 50;	
No_of_Send_Buffers	IN	INT	2	Default 2;Number of Send buffers (with send priority 1 to this para)	
Send_Buffer_Dim	IN	DINT	L#4096	Dimension [size in byte] of the send buffers; Default 4096	
Cascade_P_Application	IN	DWORD	0	Connect to P_Application from an other S7_IEC_Config for realizing mult. channel	
Cascade_Mode	IN	BYTE	B#16#0	0 (Def): L2 copy, 1: L7 copy, 2: L7 copy, 3: red channel	
Special_Functions	IN	DWORD	DW#16#0	Default 16#0; Only for special cases; Please refer to manual	
Para_DB_No	IN	INT	80	DB-Number which contains the settings for ext. TCP driver (Default: 80)	*4)

Communication-FB VAR OUTPUT		Type	Default	Block-comment/Short description	
P_Application	OUT	DWORD	DW#16#0	Pointer to Application Functions (SL_x, MA_x)	
L2_Error_Link	OUT	BOOL	FALSE	Set, if no Connection is active longer than L2_Error_DelayTime.	
Con_Err_C1	OUT	BOOL	FALSE	Set, if tcp connection 1 is not established	
Con_Err_C2	OUT	BOOL	FALSE	Set, if tcp connection 2 is not established	
C1_isActive	OUT	BOOL	FALSE	Set, if tcp connection 1 is controlling the process data (StartDT)	

 ©SIEMENS AG 2017

C2_isActive	OUT	BOOL	FALSE	Set, if tcp connection 2 is controlling the process data (StartDT)	
FB_RetVal	OUT	WORD	W#16#0	positive Values -> ok; negative Values (16#8xxx) -> Error	
Z_Info	OUT	WORD	W#16#0	additional information in case of error	*4)
More_Info_available	OUT	BOOL	FALSE	Set, if more infos are available for processing in the actual OB1-cycle.	*5)
More_info_is_spo	OUT	BOOL	FALSE	More info is spontaneous	*5)
Buffer_Info_lost	OUT	BOOL	FALSE	in case of Link_Err one ore more Info were deleted from the buffers	

Communication-FB VAR_IN_OUT		**Type**	**Default**	**Block-comment/Short description**	
Time_DS	IN/OUT	BOOL	FALSE	Time-Qualifier DaylightSaving (Summertime)	
Time_IV	IN/OUT	BOOL	TRUE	Time-Qualifier Time is InValid	
Time_SY	IN/OUT	BOOL	FALSE	Time-Qualifier Time is synchronized (actually only internal used)	
Time_Diff	IN/OUT	Time	T#0MS	Time difference between CPU-Time and desired IEC-Time-Stamps	

*1) Parameter only present in product version for S7-400
*2) Parameter only present in product version for S7-300 and ET 200S
*3) Parameter only present in product version for ET 200S
*4) Parameter only present in product version for WinAC
*5) Parameter only present up to variant V1.4
*6) Parameter only present in product version for S7-15xx
M/S: From V1.6 the Master and Slave blocks are standardized and can be switched over in its function. They include always all parameter and differ only in the pre-allocation of individual values.

4.3.2. Parameter description

Registration_Code

The MMC (S7-300) or MC (S7-400), SMC (CPU15xx) is used as a dongle. Depending on its serial number, you will receive from Siemens a registration/activation code which you need to enter here.
No entry or a mismatched entry of the information will permit unrestricted operation in demo mode for 15 min. After this period, the telecontrol communication is stopped.
For more details, we refer you to 'FB_RetVal' and the software protection chapter

Alternatively also the number of the data block containing a list of registration codes can be entered here.
The permissible range for entering a DB
1 up to 2000 (decimal); e.g. DB1000 -> DW#16#3E8

For further details please refer to ‚**FB_RetVal**' chapter 4.3 and chapter 11.

Line_ID

Is only relevant if you call up the Communication-FB multiple times (as another instance with a separate DB) to realize an additional IEC link.
The calls then need to be identified using different Line_IDs.

The Line_ID is used for the derivation of the internal Connection_ID (Con_ID) which has to be unique for each TCP_Connection of the CPU when using PN-Interfaces. Therefore Line-IDs from Communication-FB using the same interface have to be selected that way that from this unique Con_IDs can be generated.
The values are generated as follows:

CON_ID = LINE_ID for C1 (if only one channel exists)
CON_ID = LINE_ID + x for C2
CON_ID = LINE_ID + 2x for C3
CON_ID = LINE_ID + 3x for C3

x = 16 for WinAC-Variants
x = 64 for S7-Variants with 4 channels
x = 128 for S7 Variants with 2 channels

For diagnostics purposes additional a Comp_ID is used. It has the same value like the Con_ID and concerns the Layer 1, 2 and 7 which are (internal) called in the Communication-FB

L1 parameter with communication via CP:

L1_Local_ID (_C1, _C2)

This is the ID from the connection configuration (NETPRO) e.g. 1.
The value can be taken from NETPRO/properties TCP connection/module parameters.
T104 connections which are not used should be specific switched off with the value 0.

L1_Laddr_C1 (_C1, _C2)

This is the hardware address of the CP module used for establishment of the TCP connection. It can be taken from the hardware configuration or from NETPRO/properties TCP connection/module parameters.

L1 parameter with communication via PN interface of the CPU:

L1_Active_Connect (_C1 _C2)

Determines whether the TCP link connection is established active or passive.
In case of active link connection the connection partner has to be specified with the parameters ‚L1_IP_Address_xy (_C1 _C2)' und ‚L1_TCP_Port_Number (_C1 _C2)'

L1_Acceppt_All_IP_Adr (_C1 _C2)

When establishing passive connections (as a slave), this parameter can be used to determine whether connections are permitted by any partner or only by a specified partner (IP address below).
Can be parameterized for each connection.

L1_IP_Address_xy (_C1 _C2)

Passive connection establishment (IEC slave): In the case of selective connection partners (parameter 'L1_Acceppt_All_IP_Adr' = FALSE), this will set its IP address.
Active connection establishment (IEC master): IP address of the partner for TCP connection establishment.

L1_TCP_Port_Number (_C1 _C2)

Passive connection establishment (IEC slave): A connection can only be established using the port no. set here. The port number 2404 intended for IEC-T104 is preset for connection 1 (_C1). Any other connection (_C2) can only be enabled using a different port number (default 2405).
Active connection establishment (IEC master): Number of the port used for the connection establishment to the partner is.
T104 connections which are not used should be specific switched off with the value 0.

L1-Parameter at Communication via PN-Interface(s) of S7-15xx (incl. CP1543):

L1_InterfaceID (_C1,_C2)

This is the hardware identifier of the profinet interface which is used for communication at S7-15xx systems. It is available in the device configuration.
It can be found as well at the system constants and taken over from there - symbolically as well.

Further L1-Parameter at S7-400 for selection of the PN-Interface (CPU410, H-System):

L1_Local_Device_Id (C1..C2)

Communication-FB parameter for the selection of the interface on the (S7-400) CPU used for the transmission, resp. selection of the CPU within an H-system.

The following selection is possible with the subsequent identifiers:

	CPU in Rack 0	CPU in Rack 1
First PN-interface	16#5	16#15
Second PN-interface	16#8	18#18

Further L1 Parameter with communication via WinAC

L1_IF_Slot_or_Para_ID_(C1_C2)

Communication-FB parameter for the selection of the interface on mEC31 resp. microbox PC (nanobox PC) used for transmission.
1-4 interpretation as slot containing the ‚installed' LAN interface. Default =1.
IF1 = B#16#01
IF2 = B#16#02
IF3 = B#16#03
IF4 = B#16#04
>=B#16#80 = Interpretation as LAN-interface/Network adapter supported from Windows logic controller (WinLC). In this case the setup of an external interface driver (provided) is required.
The value is at the same time a parameter ID (>=80)
The parameterizing of the communication characteristics of the interface driver is carried out in a parameter data block (default DB80) see chapter 4.3.3. These DB-No. has to be assigned to a Communication-FB input 'Para-DB_No'.

L2_is_Master (from V1.6)

Determines whether the block is working as Master (Controlling Station) or as Slave (Controlled Station).
0: Slave
1: Master

L2_T104_T0

Standard parameter T0: Time monitoring of the connection establishment
Normally this parameter does not need to be changed

L2_T104_T1

Standard parameter T1: Time monitoring for sent *APDU* or test *APDU*
Normally this parameter does not need to be changed

L2_T104_T2

Standard parameter T2: Time monitoring for acknowledgements, if no data messages have been transferred
Normally this parameter does not need to be changed

L2_T104_T3

Standard parameter T3: Time monitoring for sent test messages in the case of long idle states
Normally this parameter does not need to be changed

L2_T104_k

Standard parameter k: Maximum difference in the number of receive sequences and the number of send sequences
Normally this parameter does not need to be changed

L2_T014_w

Standard parameter w: Latest acknowledgement after receipt of *w* APDU in the I-format
Normally this parameter does not need to be changed

L2_Error_DelayTime

Delay time for L2_Error_Link
After this time **L2_Error_Link** is set if no IEC connection is active (no StartDT).

L2_Ack_Buf_Dim

The parameter defines the size of the T104 confirmation buffer
0 without confirmation buffer (like V1.0)
>0 size of the confirmation buffer in Byte recommended adjustment: 1600

For further details about rating the confirmation buffer please refer to chapter 4.4.2.

L7_Block_Len

The max. length of the T101/104-ASDU will be adjusted.
The possible ASDU-length is depending on the protocol type, at approx. 253 signs. However an adjusted value which is too high will be limited automatically.
Default value = 120
Setting options 1…255

Buffer_Handling

Defines the modality of buffering the messages (information). Generally the processing described below is carried out only in case of link error

B#16#00 Clear all buffer and keep them empty
B#16#01 buffer first,
in case a buffer is filled up
-> clear all buffer and keep them empty
B#16#02 buffer and in case of overflow clear the oldest information.
B#16#03 buffer and in case of overflow stuffing backward
(meets the behavior in V1.0)

In normal operation (no ‚Link_Error') the information are buffered. In case of impending buffer overflow ‚backlog' occurs in the collection blocks (excepting organizational messages and command acknowledgements)
For the Master blocks we recommend the adjustment B#16#00 with it commands and setpoint values are not stored.)
For further details about message buffering please refer to chapter 4.4

Next_Info
(only up to V1.4)

Only relevant at Master-Applications.

The parameter **Next_Info** informs the config. block that a repeated request occurs within one (OB) cycle. This serves only for handing over the next blocked information. In this case a reduced cycle through the ‚Master chain' occurs.

For further details of increasing the flow rate at Master-Applications please refer to chapter 4.5.

Send_StartDT

With rising edge a StartDT-message is send.

Assumption:
The application is working as Master and the TCP-connection is assembled.

The input permits the manual transmission of a StartDT-message. For start of the communication after an L2_Link_Error the wiring is not necessary because the StartDT-message is sent automatically in this case.

Do_Restart

Not all parameter are effective in case of online changes (without CPU Stop/Start). The control input enables the save transfer of online changed parameters without CPU Stop/Start.

With rising edge the IEC-application is new initialized with the values parameterized at the block inputs and started.

Use_CPU_Local_Time

True: use of the local time of the CPU inclusive daylight saving time-Bit
as of SIPLUS RIC V1.6 Upd4 at S7-1500
False: use of the system time
V1.6 Upd4 with S7-1500

First_internal_DB_No

The IEC application generates during start-up different data blocks (diag DB, message memory. etc.) The parameter defines the first DB number for creating this (default: 50).

NOTICE:
If you project with CFC (e.g. within PCS7) please observe that data blocks created from SIPLUS RIC during runtime are not overwritten from CFC compiler. Use this parameter alternatively to put internal data blocks into a secure area.

No_of_Send_Buffers

Number of send buffers to be created (1…16).
From V1.6 the parameter is available in all libraries. In the block variant Slave the default adjustment is 2, in Master variants the default adjustment is 1.
See also chapter 8.

Send_Buffers_Dim
Size of the send buffers to be created. The default value is 4096.
From V1.6 the parameter is available in all libraries. Individual DB length can be adjusted using an AddOn block (see chapter 8.1.1).

Cascade_P_Application
Possibility of coupling another (or more) Communication-FB to an existing FB Communication-FB (cascading/casc.). Main_FB is the first Communication-FB which is used for the connection of further Communication-FB (Casc_FB)

Cascade_Mode
Selection of the desired redundancy mode:
Mode 0: Multiple channel on layer 2 basis
Mode 1: Multiple channel on layer 7 with partner fault per channel
Mode 2: Multiple channel on layer 7 with partner fault at Main_FB
Mode 3: Extension of the redundancy group (T104) resp. switch over to/from T101-channels

Para_DB_No
When using further LAN interfaces on the mEC31 or microbox PC (nanobox PC) for WinAC an external interface driver is required. The communication performance of this external driver is provided in a parameter data block. It's DB No. (default DB80) has to be assigned to the Communication-FB input ‚Para-DB_No'.

Serial_from_CPU
Selection whether the serial number of the Memory card or the serial number of the CPU is used for the licensing.
FALSE: Use serial number of the Memory card
TRUE: Use serial number of the CPU

Special_Functions (from V1.6)
Allows the activation of diverse special functions
Available functions are:

Bit 0 Switching off the filtering on cause of transmission in the Master blocks. This enables e.g. the nonstandard conformance receive and evaluation of indications with cause of transmission cyclic.

Bit 4 Deactivation of the reaction if receiving an unknown message type identification in control direction (reflection with COT 44)

Bit 5 Deactivation of the reaction if receiving an unknown common ASDU address in control direction (reflection with COT 46)

Bit 6 Deactivation of the reaction if receiving an unknown cause of transmission in control direction (reflection with COT 45)

Bit 7 Deactivation of the reaction if receiving an unknown information object address in control direction (reflection with COT 47)

Bit 16 Project specific special function at T104 protocol: Repetition of k messages instead of number of unreceipted messages

P_Application
Pointer (DWORD) for the interconnection of the application blocks.
The interconnection can be carried out directly in the CFC plan.
With manual programming you assign the output, e.g. to a free memory (MD) or to a temporary variable which you then specify on the relevant input variables of the application blocks.

L2_Error_Link
Signals the communication status to the connection partner, resp. error, as follows:
FALSE: communication established
TRUE: communication error
Errors are formed as follows depending on the traffic type:

Set if no active IEC-connection was established to the connection partner within ‚L2_Error_DelayTime' (no StartDT).

Con_Err_C1
Connection fault TCP-connection 1
Set if the TCP connection 1 is not established.

Con_Err_C2
Connection fault TCP-connection 2
Set if the TCP connection 2 is not established.

C1_isActive
TCP-connection 1 is active
Set if the process data are exchanged via TCP-connection 1 (StartDT).

C2_isActive
TCP-connection 2 is active
Set if the process data are exchanged via TCP-connection 2 (StartDT).

FB_RetVal

The signaling of the functional block is as follows:

W#16#0000	No error
W#16#0001 – 0900	block in demo mode. Remaining runtime in seconds in BCD -> can be read directly in hex illustration
W#16#7FFF	Initialization cycle

Error:

W#16#8101	Error when generating data blocks. Cause could be working memory too less
W#16#8201	Problem with registration code and demo time expired -> Enter correct registration code on the parameter input **Registration_Code** (page 55).
W#16#85x1	Error from L1 (x=Channel_No)
W#16#85A0	Fatal Error (L7) -> Restart

Further error codes at WinAC applications

W#16#84x2	Error from TINT_WINAC
W#16#84x3	Error from TSEND_WINAC
W#16#84x4	Error from TRCV_WINAC
W#16#84x5	Error from TCON_WINAC
W#16#84x6	Error from TDISCON_WINAC
W#16#84xA	Gen. initialization fault L1
W#16#84xB	Error instance-DB allocation (not existing or 0)

x = channel number (1,2)

Z_Info

Additional information for error code W#16#84xx from FB_RetVal

FB_RetVal error code	Additional information (Z_Info)
W#16#84x2	Status of the FB TINT_WINAC *)
W#16#84x3	Status of the FB TSEND_WINAC or TSEND *)
W#16#84x4	Status of the FB TRCV_WINAC or TRCV *)
W#16#84x5	Status of the FB TCON_WINAC or TCON *)
W#16#84x6	Status of the FB TDISCON_WINAC or TDISCON *)
W#16#84xA	1 = Para DB = 0 or not existing
	2 = Para DB wrong version
	3 = Para DB ID not found
	4 = Checksum fault instance DBs
W#16#84xB	1 = Instance DB TINIT_WINAC
	2 = Instance DB TSEND_WINAC
	3 = Instance DB TRCV_WINAC
	4 = Instance DB TCON_WINAC
	5 = Instance DB TDISCON_WINAC

*) See also WinAC IP driver documentation (WinAcIpDoku)

More_Info_available (up to variant V1.4)

Only relevant at Master-Applications

The parameter **More_Info_available** signalizes that further information (from a blocked message) are directly available which could be handed over in one of the following cycle (refer also to the parameter **Next_Info)**.

For further details of increasing the flow rate at Master-Applications please refer to chapter 4.5.

.

More_info_is_spo (up to variant V1.4)

Only relevant at Master-Applications

More_Info_is_spo is an additional information permitting repetition cycles in dependency on the cause of transmission (refer also to the parameter **Next_Info)**

For further details of increasing the flow rate at Master-Applications please refer to chapter 4.5.

Buffer_Info_lost

Loss of information:
Indicates that minimum one message from the message memories has been lost (deleted).

With detected Link_Error applies:
The output is set as soon as information has been cleared.
The output is reset with disappearing Link_Error.

Note
If the parameter **Buffer_Handling** = B#16#03,: the **Buffer_Info_lost** is never set because in this case no information are cleared from the message buffers.

 ©SIEMENS AG 2017

Time_DS	IN/OUT variable: Summer time bit of the current time. Displays the current time status, if the variable is being read exclusively. It can be affected when the assigned variable is forced. The bit is transferred in IEC messages with a time stamp direct into SU
Time_IV	IN/OUT variable: Invalidity bit of the current time. Displays the current time status, if the variable is being read exclusively. It can be affected when the assigned variable is forced. The bit is transferred in IEC messages with a time stamp direct into IV
Time_SY	IN/OUT variable: Synchronization status of the current time. Displays the current time status, if the variable is being read exclusively. It can be affected when the assigned variable is forced. The bit is not used in IEC messages
Time_Diff	Time difference in comparison to CPU base time. The value specified here [ms] is added to the current time in the CPU and the result is used as a time stamp for IEC messages. This means that time corrections such as local time calculations with CPU time on the basis of GMT can be implemented.

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

 ©SIEMENS AG 2017

4.3.3. Use of the Windows Ethernet-Interface under WinAC

For using further LAN interfaces for WinAC on mEC31 or Microbox PC (Nanobox PC) an external interface driver is used. The communication characteristics of this driver are provided in a parameter data block (default DB80).

DB80 Parameter

Address	Name	Type	Initial value	Actual value	Comment
0.0	DB_Manag.Pos_of_DB	BYTE	B#16#0	B#16#0	Reserved for future - do not change
1.0	DB_Manag.No_of_all_DBs	BYTE	B#16#0	B#16#0	Reserved for future - do not change
2.0	DB_Manag.DB_No_Act	WORD	W#16#0	W#16#0	Reserved for future - do not change
4.0	DB_Manag.DB_No_Prev	WORD	W#16#0	W#16#0	Reserved for future - do not change
6.0	DB_Manag.DB_No_Next	WORD	W#16#0	W#16#0	Reserved for future - do not change
8.0	DB_Manag.Reserved	WORD	W#16#0	W#16#0	
10.0	Info_Manag.P_Byte_First_Info	INT	26	26	(Byte-) Position of first Para-Block do not change
12.0	Info_Manag.No_of_Infos	INT	0	0	0; unspecified (DB is filled up to the end);otherwise size n of Array 1...n
14.0	Info_Manag.Len_Info	BYTE	B#16#26	B#16#26	Difference in byte between two Para Blocks do not change
15.0	Info_Manag.Len_Sort_Key	BYTE	B#16#0	B#16#0	0: without sorting; >0: Data sets are sorted ascending with x bytes
16.0	Info_Manag.Re_internal_usage	DWORD	DW#16#0	DW#16#0	
20.0	DB_Ident	WORD	W#16#AC01	W#16#AC01	Do not change
22.0	DB_Ver	WORD	W#16#100	W#16#100	Do not change
24.0	Checksum_IDBs	INT	1092	1092	Do not change
26.0	Para [1]. Para_ID	Byte	B#16#0	B#16#80	0: unspec. (para for any connection), 80-FF for spec. connection
28.0	Para [1].DLL_ID	INT	0	0	0: Default-DLL
30.0	Para [1].Res_04	INT	0	0	
32.0	Para [1].IDB_No.TINIT_WINAC	INT	82	82	Instance DB for FB TINIT_WINAC
34.0	Para [1].IDB_No.TSEND_WINAC	INT	83	83	Instance DB for FB TSEND_WINAC
36.0	Para [1].IDB_No.TRCV_WINAC	INT	84	84	Instance DB for FB TRCV_WINAC
38.0	Para [1].IDB_No.TCON_WINAC	INT	85	85	Instance DB for FB TCON_WINAC
40.0	Para [1].IDB_No.TDISCON_WINAC	INT	86	86	Instance DB for FB TDISCON_WINAC
42.0	Para [1].IDB_No.TUSEND_WINAC	INT	87	87	Instance DB for FB TUSEND_WINAC
44.0	Para [1].IDB_No.TURCV_WINAC	INT	88	88	Instance DB for FB TURCV_WINAC
46.0	Para [1].IDB_No.res_14	INT	0	0	
48.0	Para [1].Local_IP_Address.HH	INT	0	192	xxx-Part of local IP-Address xxx.0.0.0 only needed if more netw. Adapt. exist
50.0	Para [1].Local_IP_Address.HL	INT	0	168	xxx-Part of local IP-Address 0.xxx.0.0 only needed if more netw. Adapt. exist
52.0	Para [1].Local_IP_Address.LH	INT	0	1	xxx-Part of local IP-Address 0.0.xxx.0 only needed if more netw. Adapt. exist
54.0	Para [1].Local_IP_Address.LL	INT	0	4	xxx-Part of local IP-Address 0.0.0.xxx. only needed if more netw. Adapt. Exist
56.0	Para [1].Local_IP_Address.Res_08	INT	0	0	
58.0	Para [1].Local_IP_Address.Res_10	INT	0	0	
60.0	Para [1].Res_34	DWORD	DW#16#0	DW#16#0	
64.0	Para [2].Para_ID	Byte	B#16#0	B#16#81	0: unspec. (para for any connection), 80-FF for spec. connection
66.0	Para [2].DLL_ID	INT	0	1	0: Default-DLL
68.0	Para [2].Res_04	INT	0	0	
70.0	Para [2].IDB_No.TINIT_WINAC	INT	82	1082	Instance DB for FB TINIT_WINAC
72.0	Para [2].IDB_No.TSEND_WINAC	INT	83	1083	Instance DB for FB TSEND_WINAC
74.0	Para [2].IDB_No.TRCV_WINAC	INT	84	1084	Instance DB for FB TRCV_WINAC
76.0	Para [2].IDB_No.TCON_WINAC	INT	85	1085	Instance DB for FB TCON_WINAC
78.0	Para [2].IDB_No.TDISCON_WINAC	INT	86	1086	Instance DB for FB TDISCON_WINAC
80.0	Para [2].IDB_No.TUSEND_WINAC	INT	87	1087	Instance DB for FB TUSEND_WINAC
82.0	Para [2].IDB_No.TURCV_WINAC	INT	88	1088	Instance DB for FB TURCV_WINAC
84.0	Para [2].IDB_No.res_14	INT	0	0	
86.0	Para [2].Local_IP_Address.HH	INT	0	192	xxx-Part of local IP-Address xxx.0.0.0 only needed if more netw. Adapt. exist
88.0	Para [2].Local_IP_Address.HL	INT	0	168	xxx-Part of local IP-Address 0.xxx.0.0 only needed if more netw. Adapt. exist
90.0	Para [2].Local_IP_Address.LH	INT	0	1	xxx-Part of local IP-Address 0.0.xxx.0 only needed if more netw. Adapt. exist
92.0	Para [2].Local_IP_Address.LL	INT	0	116	xxx-Part of local IP-Address 0.0.0.xxx.

Ad-dress	Name	Type	Initial value	Actual value	Comment
					only needed if more netw. Adapt. Exist
94.0	Para [2].Local_IP_Address.Res_08	INT	0	0	
96.0	Para [2].Local_IP_Address.Res_10	INT	0	0	
98.0	Para [2].Res_34	DWORD	DW#16#0	DW#16#0	

Additionally the external interface driver needs a number of instance DBs for its functional blocks. The allocation of the numbers is carried out in the parameter DB as well. The parameter DB is supplied with pre-settings which allow the direct use in conjunction with the delivered FC-, FB- and DBs without further adjustments if only **one** network adapter exists.

Communication-FB Parameter L1_IF_Slot_or_Para_ID_Cx = B#16#80

The adjustment of the IP address for the network adapter is carried out under the DB80 address 48 – 54.

If another network adapter has to be used the corresponding address for this adapter has to be adjusted in the DB80:

Communication-FB Parameter L1_IF_Slot_or_Para_ID_Cx = B#16#81

The adjustment of the IP address for this network adapter is carried out under the DB80 address 86 – 94.

 ©SIEMENS AG 2017

4.4. Message buffering

4.4.1. Use of the Message Memory

From V1.0 already 4 data blocks are created as message buffer (more exactly: event buffer between process blocks and telecontrol protocol) and used as follows:

- Diagnosis memory: Default length 1000h = 4096 Byte
  for Diagnosis information
- TX_Buf_1 (message memory 1): Default length 1000h=4096 Byte
  for organizational entries (start-up indication, confirmation messages,…)
- TX_Buf_2 (message memory 2): Default length 1000h=4096 Byte
  Entries from process blocks (Sly,…) with Priority High
- TX_Buf_3 (message memory 3): Default length 1000h=4096 Byte
  Entries from process blocks (Sly,…) with Priority Low
- TX_Buf_4 (message memory 4): Default length 200h=512 Byte
  reserved for low prior applications, currently not used

The Buffer-DBs are created during start-up. The lengths are singular given in the Instance-DB from S7_IEC_Config. With it they are available as a kind of background parameter'.

Basically also in V1.1 the size of a buffer is limited to one DB (no DB comprehensive buffer). In larger systems (S7-400, CP317PN) the maximum DB size is 64 kB, in S7-300 systems normally 16 kB (refer to the S7 manual).

As a guiding value for the memory consumption per information approx. 40 Bytes can be assumed. In this case you can enter per buffer:

- with 4096 Byte: approx. 100 Information
- with 16 Kbyte: approx. 400 Information
- with 64 Kbyte: approx. 1600 Information

An entry in one of the buffers occurs independently from the cause of transmission, also at general interrogations and cyclic transmission procedures. However cyclic entries are only carried out from the SLi-blocks if the buffer is filled less 50 %. Requested entries are carried out only if the buffer is filled less 70 %. The remaining space is used for spontaneous changes (like in V1.0).

In V1.0 the message memories in all Master packages were cleared continuously (to prevent the storage of commands). In the Slave packages the accumulated information during connection disturbances were remaining (oldest information – see above). While in Master Systems preferred should not be buffered, the requirements in the Slave are arranged customer specific as follows:

a) After disappearing connection disturbances no old information is desired
b) No loss of information should occur during connection disturbances

From Version 1.1 the desired behavior regarding the message buffering can be parameterized via ‚Buffer_Handling' (Parameter input from S7_IEC_Config).

For more details refer to chapter 4.4.3.

From Version V1.3, instead of two defined buffers, up to maximum 16 priority buffers can be set, adjustable at Communication-FB with the parameter „No_of_Send_Buffers". With the Parameter "Send_Buffer_Dim" the size of the single buffers can be adjusted. Thereby the limits mentioned above apply referenced to the size of the DB.

- Diagnosis memory: Default length 1000h = 4096 Byte
  for Diagnosis information

  TX_Buf_org: Default length 1000h=4096 Byte
  for organizational entries (start-up indication, confirmation messages,…)
- TX_Buf_0 (message memory 0): Default length 1000h=4096 Byte
  Entries from process blocks (Sly,…) with Priority 0
- TX_Buf_1 (message memory 1): Default length 1000h=4096 Byte
  Entries from process blocks (Sly,…) with Priority 1

Every buffer is treated with a defined priority, that is informations with higher priority will be sended earlier than informations with lower priority. The first buffer (buffer 0) is disposed with highest priority, while the 16th (buffer 15) is treated with lowest priority.

The priority, with which information should be transmitted, is adjusted at the respective collection blocks (SLi_xyz) via the input parameter 'Tx_Prio' range of value 0…15.

If a priority is adjusted at the collection block that is not available – for example 5 priority buffers are created, but the adjusted priority at the collection block is 8 – the next lower order priority is used automatically, in this example 4.

 ©SIEMENS AG 2017

4.4.2. T104-Confirmation buffer

At IEC 60870-5-104 transmission not every single message will get a confirmation. In order to transmit unconfirmed messages again if necessary (connection fault); all messages (I-Formats) have to be temporary buffered. For this purpose a confirmation buffer can be created from V1.1 with the following behavior:

- An existing receive buffer is used from all T104 connections of a redundancy group (several TCP connections, process data exchange only via one connection)
- Sent messages (I-Formats) are generally entered in parallel into the receive buffer. The reading pointer is switched forward according to the received messages.
- In case of a connection switch over (StartDT to another TCP-connection) or interrupt and anew link connection, first non acknowledged information from the receive buffer are sent.

Via a parameter input ‚L2_Ack_Buf_Dim' at S7_IEC_Config block the configuration is carried out whether a corresponding receive buffer should be created and controlled.

With the default value (0) **no** receive buffer is created/controlled.

- Messages once sent cann**ot** be repeated.
- This adjustment should be used for central units (Master functions) so that commands are not stored during a longer period.

Values <>0 define the desired size of the receive buffer:

- The optimum buffer size can be calculated as follows:
- L = 30 + ((22Bytes + 6 + ‚*max_Block_Len*') x parameter ‚*L2_T104_k*')
  30 = administration part in the DB itself
  22 = length of the internal Job-Header
  6 = length of one T104-Message-Header
  In conjunction with the default values for ‚max_Block_Len' = 120 and ‚*L2_T104_k*' = 12 it results in a value of 1806 (Bytes).
- too small values (but not less 30) in the worst case lead to the state that not all unconfirmed information can be entered, however no fault occurs.

4.4.3. Projection

For projection of the desired buffer characteristics the following new Communication-FB -Parameter have been established in Version V1.1:

- Input: ‚L2_Ack_Buf_Dim' (T104-Variants only)
- Input: ‚Buffer_Handling'
- Output: ‚Buffer_Info_lost' (for diagnostics)

L2_Ack_Buf_Dim
(only T104-Variants)

The parameter defines the size of the T104 confirmation buffer
0 without confirmation buffer (like V1.0)
>0 size of the confirmation buffer in Byte recommended adjustment: 1600

Buffer_Handling

Defines the modality of buffering the messages (information). Generally the processing described below is carried out only in case of link error. In normal operation (no ‚Link_Error'), backlog' occurs (excepting ORG- and command blocks).

B#16#00 Clear all buffer and keep them empty
B#16#01 buffer first,
in case a buffer is filled up
-> clear all buffer and keep them empty
B#16#02 buffer and in case of overflow clear the oldest information.
B#16#03 buffer and in case of overflow stuffing backward
(meets the behavior in V1.0)

 ©SIEMENS AG 2017

Buffer_Info_lost	Loss of information: Indicates that minimum one message from the message memories has been lost (deleted). With detected Link_Error applies: The output is set as soon information has been cleared The output is reset with disappearing Link_Error Note If the parameter **Buffer_Handling** = B#16#03: the **Buffer_Info_lost** is never set because in this case no information is cleared from the message buffers.

Generally the processing is carried out only in case of Link_Error. In normal operation ‚backlog' occurs always.

‚Buffer_Info_Lost' appears with the first cleared information and disappears with disappearing Link_Error.

An existing T104 receive buffer is always cleared in Mode 0 (‚Buffer_Handling' = 0). In the other modes it is cleared only with erasing information (‚Buffer_Info_lost').

With ‚Buffer_Handling' = 0 the buffering is deactivated.

Deactivating means continuous clearing of the buffers during a discovered link error. In case of Master all buffers are cleared, in case of Slave all buffers with exception of the organizational buffer. In this case a possible existing TI70 message is kept (further entries aren't to be expected).
Using T104 the receive buffer is controlled analog resp. it is not existing (‚L2_Ack_Buf_Dim').

The following table gives an overview of the effect from the possible buffer variants (‚Buffer_Handling').

Buffer_ Handling	**Function**	**Concerns**	**Buffer_Info_lost appears**
00	Clear all buffer/keep it empty	Message memory 2,3,4 Message memory 1 (ORG) only if Master T104-receive buffer	As soon as information was cleared
01	Buffer first in case a buffer is filled up -> clear all buffer and keep them empty	Message memory 2,3,4 Message memory 1 (ORG) only if Master T104-receive buffer	As soon as information was cleared
02	Buffer and clear the oldest information in case of overflow	Each individual buffer Message memory 1 (ORG) only if Master T104- receive buffer	As soon as information was cleared
03	Buffer and possibly backlog*		Is NOT set here

* **Meaning of backlog in this case:**

Fill the buffer and in case the buffer is filled don't enter new messages.
=> Possible new (actual) information can't be entered into the message buffer and can get lost.

Note:

The message buffering is switched off in case of parameterization with default values. This corresponds with the behavior in V1.0. If necessary the corresponding parameterization has to be consciously activated.

Note:

If necessary the size of the buffer DBs in the Communication-FB's instance DB can be adapted. For this adjust the parameters **TX_Buf_n_Len** according to your requirements.

4.5. Improvement of the IEC-Master performance

In normal case in each OB1 cycle one information is processed. This may lead to problems in case of receiving blocked information and longer OB1 cycle times.

In order to process more than one information per cycle the ‚Master-Chain', this means the S7_IEC_Config and all evaluation blocks, principally could be called several times (but always the complete Master-Chain).

In this case all functions would be passed every time. This is not necessary in order to evaluate only a further information from a blocked message.

Therefore parameter have been established in the S7_IEC_Config (Communication-FB) from the Master applications which allow a reduced cycle of the ‚Master-Chain'.

This can be used as follows:

- **‚More_Info_Available'** signals that directly further information (from a blocked message) are existing which could be handed over in a subsequent cycle.
- **‚More_Info_is_spo'** is an additional information, allowing repeated cycles in dependency of the cause of transmission (= spontaneous, <> spontaneous).
- **‚Next_Info'** informs the Config-block that a repeated call is executed within a (OB-) cycle only for the purpose to hand over the next blocked information.

With programming a suitable loop an improvement of the performance when receiving blocked messages can be achieved.

Example for the programming:

```
// Master
      SET
      R     #Next_Info

      L     10
Loop: T     #L_Count

      CALL  "S7_IEC_Config" , DB100
       Next_Info              :=#Next_Info
       More_Info_available    :=#More_Info_Available
      CALL  "MA_Org_Asdu_1" , DB122

      CALL  "MAo_SP_IM_pDB"
… (further evaluation blocks)

// set Rep after the first cycle
      SET
      U     #More_Info_available
      S     #Next_Info
      SPBN  WWW
      L     #L_Count
      LOOP  Loop
WWW:  NOP   0
```

The loop shown above is causing the processing and output of up to 10 (#L_Count) information from a received blocked message within one OB1-cycle.

The drop is carried out if *#More_Info_available=FALSE* or the loop counter is *#L_Count=0.*

Wiring the Communication-FB input ‘Next_Info' with the variable #Next_Info is causing thereby the reduced cycle of the ‚Master-Chain'. In this case an explicit enhanced information performance has been reached with only slight increased cycle time.

The size of the loop counter should be selected in such a way to reach a possible optimum ratio between performance improvement and max. cycle time extension. Normally 10 should be a practicable value.

Remark for the supervisory time L2_T104_T2 at T104-Communication

In case of Master and an expected high data appearance the time ***L2_T104_T2*** should be possibly small adjusted (1 sec) (T2 ->send receive).

Note:

From version V1.5 up to 10 information objects are internal provided per cycle and edited from the MAo blocks. This is carried out independently of the use of the blocks, in Master or Slave communication in standard or reversed direction.

The possibility to improve the performance described in this chapter for the Master library is no longer necessary. It is inapplicable including the related parameter.

 ©SIEMENS AG 2017

5. Multiple channel/Redundancy

The IEC 60870-5-104 standard specifies for the operation of redundant connections the following rules:

1. Central station and substation must be able to process several logical connections.
2. These logical connections form a redundancy group.
3. Only one logical connection per redundancy group is started and transmits and receives application data.
4. The central station decides which of the existing connections is started.
5. All logical connections of a redundancy group have to be monitored with test messages.
6. A redundancy group has to be configured on one process image only (data inventory/event buffer)
7. If more than one central station simultaneously needs access to the same substation each central station has to be assigned to another redundancy group (process image).

However not all control centers support this redundancy rules. Instead of data transmission on only one connection which was activated with StartDT, they expect when operating with several connections also data from all connections. Up to now this could only be realized with starting several instances of the (Communication-FB) communication and the complete collection (SL-Blocks).

5.1. Implementation

From version V1.4 IEConS7 offers the possibility of cascading Communication-FBs . The basic idea is the possibility of coupling one (or more) Communication-FBs to an existing Communication-FB.

For this the Communication-FB are extended by the inputs ‚Cascade_P_Application' and ‚Cascade_Mode'. The first Communication-FB (Main) is leading like until now. As usual all application blocks (SL-Blocks) are coupled to it via P_Application.

At cascading further Communication-FB (Casc) respectively can be interconnected to P_Application via its new input ‚Cascade_P_Application'.

In monitoring direction the subordinated (Casc) Communication-FB transmit identical messages like the respective superior Communication-FB by writing either directly into the transfer box of layer 7 to layer 2 (Casc_Mode 0) or into a send buffer (Casc_Mode 1, 2).

With Casc_Mode 3 the redundancy group is extended by additional 2 connections. Within the entire redundancy group furthermore only one channel transmits the present send data from the Main- Communication-FB.

In control direction each Communication-FB checks alternating on self received messages or on received messages existing from a subordinated block and transfers them to the superior Communication-FB resp. to P_Application as Main Communication-FB.

It is only buffered once (at the Main Communication-FB) and only if no more connection exists. The subordinated Communication-FB do not need own buffer blocks in Casc_Modes 0 and 3. Here the parameter ‚No_of_Send_Buffers' can be adjusted to 0. In Mode 1 the parameter can be reduced to 1.

Note:

In order to have a consistent time management in the Communication-FB blocks the time bits based on the FC121 (Time_DS, Time_IV, Time_SY) should be interconnected.

Note:

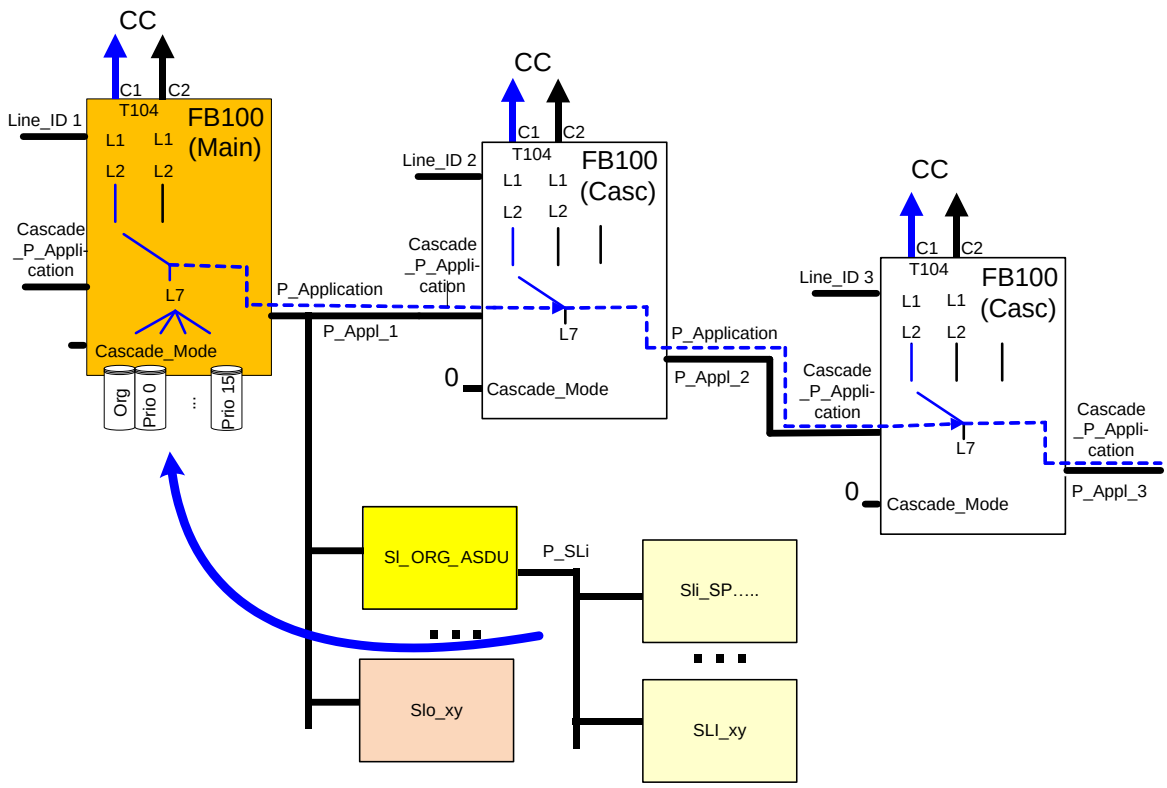
Please don't forget to allocate a definite (ascending) number to each Communication-FB input 'Line_ID'

 ©SIEMENS AG 2017

In conjunction with the already mentioned parameter ‚Cascade_Mode' the following functions can be realized:

Cascade_ Mode	Function	Application
0	Multiple channel on the basis of layer 2 → Transmission of a message copy (application data)	Standard for transmitting parallel In T104 Slave, where applicable also in T101 Slave, for redundant/parallel data supply for control center (control center compound) if identical transmission path (e.g. LAN) are existing. The transmission of a message must be completed on all channels before transmission of the next message.
1	Multiple channel on the basis of layer 7 – with partner fault per channel	Link_Error is generated from each channel separately. In case of fault a specific additional block (per channel) is necessary for the buffering of data. Should be only used project specific resp. after consulting!
2	Multiple channel on the basis of layer 7 – with partner fault at Main FB	In T101 Slave, where applicable also in T104 Slave, for redundant/parallel data supply for control center (control center compound) and different transmission path/data rates are available because of decoupling via a buffer. Theoretical also communication parameter like the length of the IOA may be different in this mode.
3	Extension of the redundancy group (T104) resp. T101 redundancy according to NUC101	Extension of the redundancy group from T104 Master or Slaves and T101 redundancy only if transmitting application data via one channel/connection.

5.1.1. Cascade_Mode_0: Multiple Channel on basis of Layer 2



Data flow in monitoring direction

- Virtual ‚parallel' transmission for the supply of one control center via several active connections.

  Messages created in the Main Communication-FB (only ASDU = application service data unit) are handed over to all connections which are marked as non disturbed, and sent. A short congestion may occur if it was not possible to send on one connection. For example confirmation is not possible because a breakdown of the line arises and the disturbance is not yet recognized.

  With different protocols (e.g. 104, 101) the slowest partner specifies the flow-rate.
- A total disturbance (Link_Error at Main Communication-FB) and an associated congestion occur only if there is no more connection to the control center.
- Cascade shape: Line
- No buffer-DBs and T104-acknowledgement buffer are required in the Communication-FB (Casc).
- In control direction messages from all channels are accepted. In case of high message appearance a message from the own connection is alternating processed with a message from the next subordinated block. The more FBs have been cascaded the longer it takes (OB1 cycles) for the handover of messages from the FB with the lowest hierarchy.
- The Communication-FB (Casc) must have the same L7-Message adjustments like the Communication-FB (main).

The operation mode is designed for minimum one valid and active connection. In case of a total disturbance the buffer handling adjusted with the parameter ‘Buffer_Handling’ is executed.

Example: Buffer_Handling 2 is adjusted and 4 connections are projected. At last only connection 3 was active. In case of breakdown of connection 3 as well, buffering is carried out in the Communication-FB (Main). If now a connection which is different from connection 3 operates again all messages from the Communication-FB (Main) are transmitted via this connection.

It may occur that some messages are obviously missing. These are still in the send buffer of connection 3, because they have not been yet transmitted or acknowledged from the control center. If connection 3 returns, these missing messages are transmitted as well.

5.1.2. Cascade_Mode_1: Multiple Channel on basis of Layer 7

(With partner disturbance per channel)

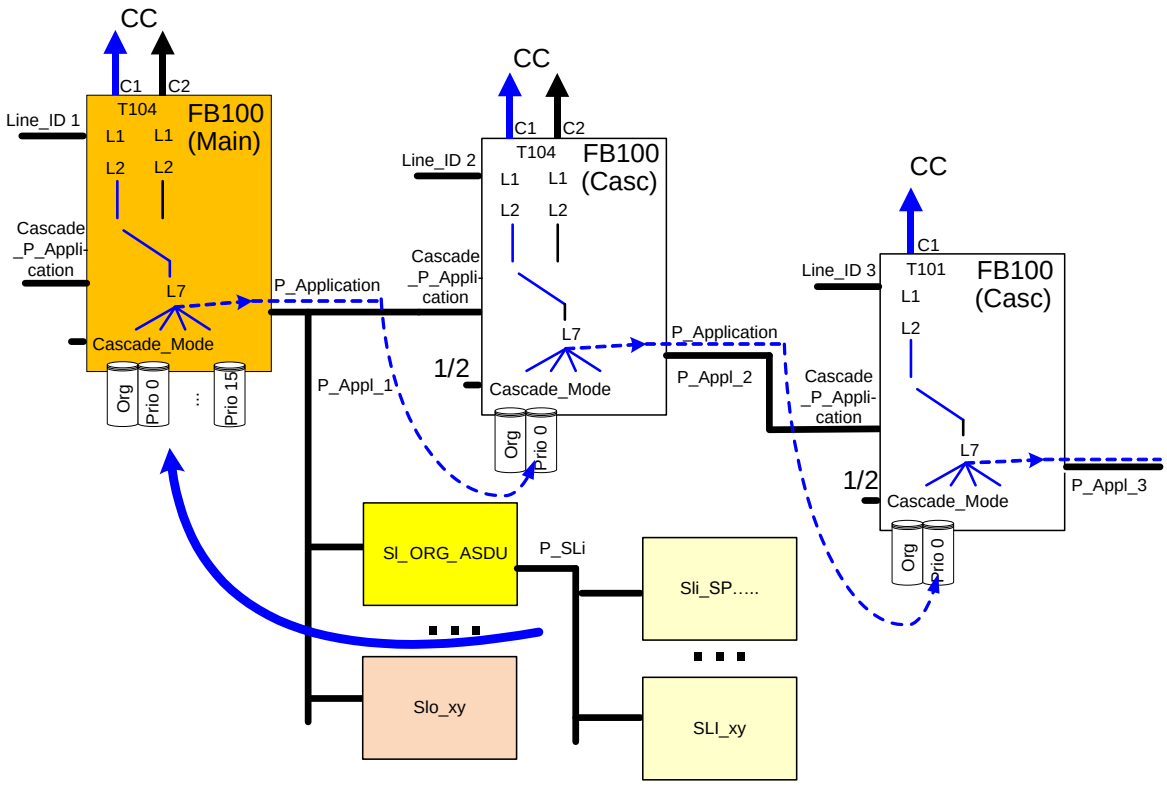
This mode is released from version V1.6 however it should be only used project specific resp. after consulting!

The basic behavior and interconnection is like Mode 2. Deviant is the independent generation of L2_error_Link from each channel. A complete disturbance at the main and back pressure into the send buffers is not derived.

In case of disturbance optionally a specific additional block (per channel) for buffering can be used. For that an AddOn block (S7IEC_AddOn_Ext_Buf_CP4 / Ext_Buffer_CP_4 – only available for SIMATIC manager, not TIA) for S7-400 is currently available. The block saves (also a large amount of data) in the available data memory of the CPU and in connection with CP443 Adv modules in its RAM.

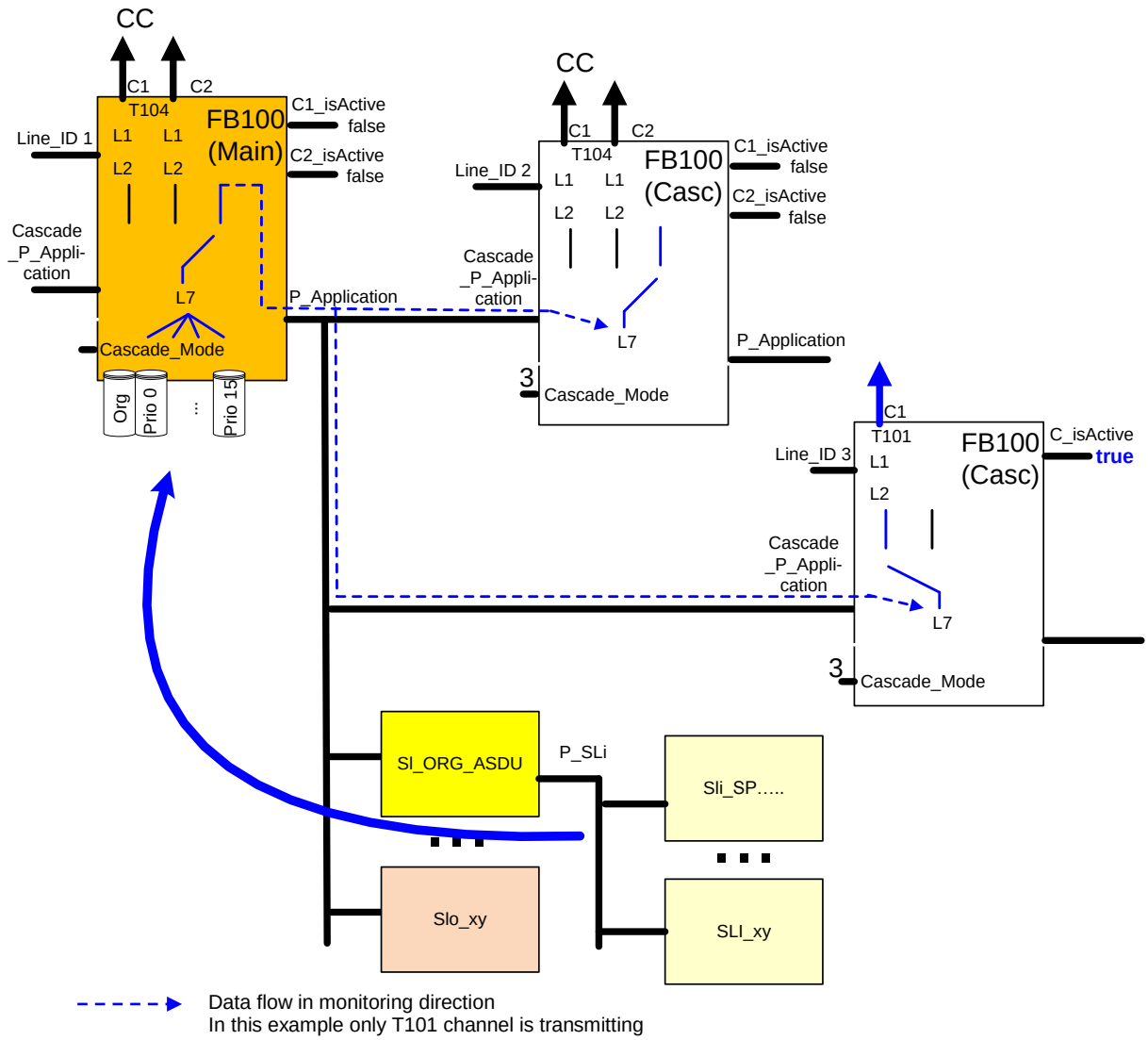
5.1.3. Cascade_Mode_2: Multiple Channel on basis of Layer 7

(With partner disturbance at the main FB)



- Information which has to be transmitted is additional placed to another channel by writing into the send buffer DB with the highest priority (like from a SLi-Block).
- Thus a decoupled processing of single information, including parameterized properties like with/without originator, IOA length (at T101 protocol etc.).
- Cascade shape: Line
- Control direction like Cascade_mode 0
- With larger amount of data the send buffer with the highest priority from the respective Communication-FB (Casc) serves as decoupling buffer. If it is full a congestion into the Communication-FB (Main) occurs in case of slow processing. In case of fault the buffer is kept empty resp. data are transferred to another Communication-FB (Casc) if existing.
- Link_Error at the Communication-FB (main) is the collective message also from subordinated Communication-FB (Casc) Link Errors. In case of fault congestion is possible!
- Sequential extendable …

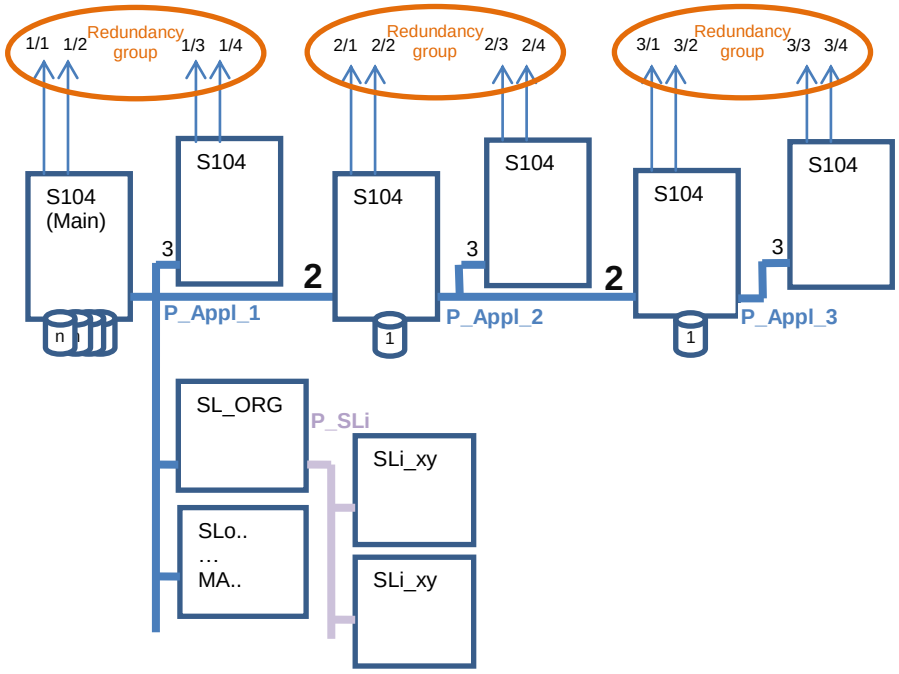
5.1.4. Cascade_Mode_3: Extension of the T104-Redundancy group and T101-Redundancy according to NUC101



- **T104-Protocol:**
  Step-by-step extension of the T104 redundancy group from Communication-FB (Main) by respectively 2 further connections. According to **definition in the standard** application data are only sent and accepted on **one** connection (which was activated with StartDT).
- **T101-Protocol:**
  Step-by-step extension by respectively one further T101 channel. Also at T101 only **one** channel will be process leading. The switchover/channel activation is carried out from V1.6 according to Norwegian User Convention (NUC101) in dependence on the kind of traffic.
  - Unbalanced mode -> activation by ‘Reset of remote Link’
  - Balanced mode -> activation by any ASDU (user message)
- The functionality is supported at both protocols as Slave as well as Master
- Mixed operation of T101 and T104 protocol and with different interfaces is possible as well (as shown in the example above).
- Cascade shape: Several Communication-FB (Casc) at **one** Communication-FB (Main)! This means in parallel resp. star-shaped.
- No buffers are required at the Communication-FB (Casc) (however the internal Org and Diag are always created).

5.1.5. Combination of Cascadings

From V1.6 SIPLUS RIC also supports the combination of cascading in Mode 0, 1 or 2 with Mode 3. Subsequently it is schematically represented for the channel multiplication in Mode 2.



5.2. Connection of several Control Centers

The connection of several (independent) control centers to SIPLUS RIC can be implemented by multiple projecting of all required RIC blocks. The channels can be configured independently from each other and also individually (e.g. concerning protocol or quantity structure). Disadvantage is the higher demand of resources, because the same information has to be recorded multiple. Please plan it for dimensioning of your hardware.

If several control centers have to be connected by a multiple channel mode described in this chapter (sensible are only Mode 0, 1, 2) please check carefully whether the following restrictions/features are sufficient for your application.

- Each message is transmitted to all (non disturbed) participants – also organizational messages like command confirmations and answers on general interrogations etc.
- Received messages of all channels are transferred to the application blocks. Selection is not possible.
- At loss of individual channels no message buffering is carried out (exception in Casc_Mode 1 and usage of an AddOn buffer block). Buffering according to parameter buffer handling starts only if no channel is available for communication (not Casc_Mode 1). The transmission starts again via the first rebuild communication connection. It is not possible to repeat for later established connections.
- In Casc_Mode 1 the AddOn buffer block can be used per channel (Communication-FB). It will buffer in case of connection fault and transmit the information from buffer after return of the communication.
- At disruption of a connection short-time delays may occur on the other connections – until the disruption has been recognized.

6. Application blocks

Basically there are two packets available: Slave and master functions.

Slave (SL) blocks fulfill traditional 'substation functions' such as indication/measured value recording or command output.

Master (MA) blocks carry out 'control center functions, e.g. the transfer of status conditions to images, initiation of commands, etc.

Although each protocol version can work in principle together with both master and slave blocks, the available versions only include the master or slave application packet.

NOTICE

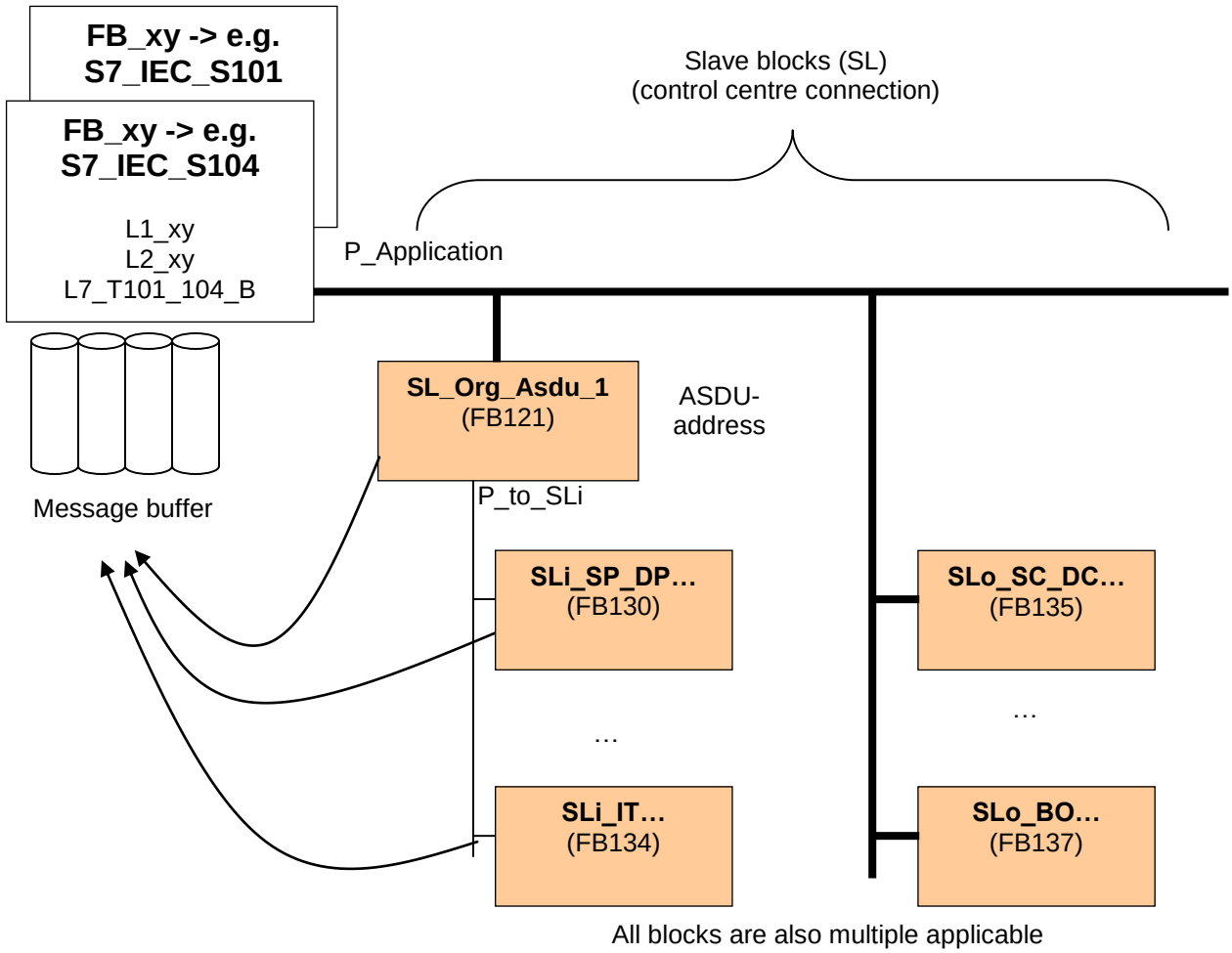
All application blocks are designed exclusively for the **priority class OB1**. The communication blocks Communication-FB ('S7_IEC_Config') also need to be operated in this priority class!

Note to the CFC plan view

The CFC views included in the manual are used exclusively for explanation purposes. CFC is **not** required for using the blocks.

The standard programming options in SIMATIC are sufficient.

6.1. Slave blocks SL



The slave application blocks (SL) are connected via the pointer 'P_Application' to the protocol version supplied, i.e. the function block 'SL_Org_Asdu_1' and the functional blocks for the outputs in the control direction (SLo) are coupled directly to the IEC interface connection. The data capture blocks (SLi) are each 'interconnected' to an organization block.

The organization of an ASDU address in monitoring direction is carried out in the function block 'SL_Org_Asdu_1' which supplies the pointer 'P_to_SLi' for the data capture blocks.

The data capture blocks are connected to the pointer 'P_to_SLi' and write its messages to the relevant send buffer.

Additional ASDU addresses can be managed in the monitoring direction via the multiple call-up of the block SL_Org_Asdu_1 and subordinated SLi blocks.

In the control direction the ASDU address(es) are managed directly in the parameter DB.

6.1.1. Organization block - SL_Org_ASDU_1 (FB121)

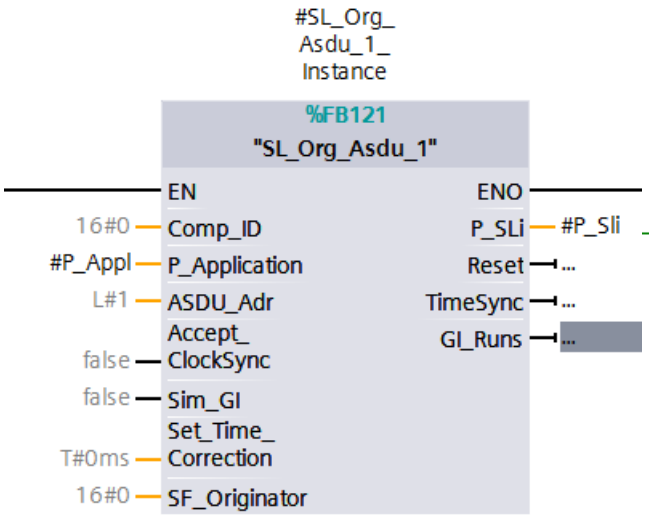


Illustration as FUP block with default values

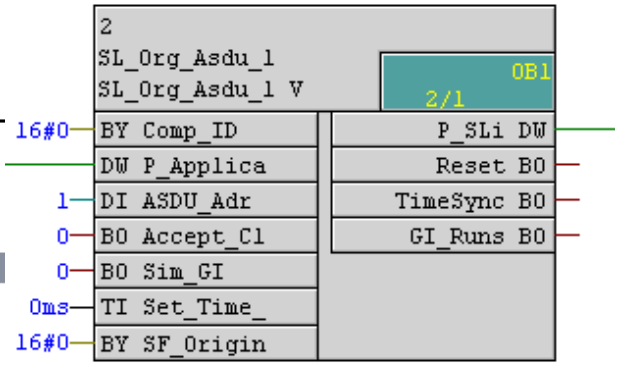


Illustration as CFC block with default values

The block has been created as a functional block (FB), and therefore requires an instance DB (e.g. Call FB121, DB121) or can be used within a higher level FBs as a 'multi-instance'.

The tasks of SL_ORG_ASDU_1 always relate to the parameterized ASDU address and are:

- Provision of an interface to subordinate SLi data capture blocks
- Sending of TI70 (start-up message)
- Management of TI100 interrogation (general interrogation)
- Management of counter interrogation commands TI101
- Support of single interrogation TI102
- Response of test messages TI104 and TI107
- TI105 remote reset
- TI103 time synchronizing message
- Negative command receipt, in case no command block has processed the message

➢ From Version V1.5 applicable also in time interrupts (see chapter 6.2)

Block parameters with their default assignment and brief comment

FB121		Type	Default	Block-Comment/brief description.
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	DW#16#0	Pointer which must be connected to 'S7_IEC_Config' – Module
ASDU_Adr	IN	DINT	L#1	ASDU-Address which will be controlled by this module
Accept_ClockSync	IN	BOOL	FALSE	if true, time of received clock sync. commands will accepted (Default: False)
Sim_GI	IN	BOOL	FALSE	A rising edge starts a station (general) interrogation
Set_Time_Correction	IN	TIME	T#0MS	the set time of clock sync. commands will be corrected with this time (Def. 0ms)
SF_Originator	IN	BYTE	B#16#0	Originator which will be overtaken by all SLi-blocks (Default: b#16#0)
P_SLi	OUT	DWORD	DW#16#0	Pointer, which must be connected to SLi - Application modules (SLi_x)
Reset	OUT	BOOL	FALSE	Signals a received reset command (TI105) for about 5 sec
TimeSync	OUT	BOOL	FALSE	Signals a received (and accepted) TimeSync-Telegr.(TI103) for one cycle
GI_Runs	OUT	BOOL	FALSE	Signals a running general interrogation

Parameter and function details:

Comp_ID
Block identifier for (currently only internal) diagnostic functions

P_Application
Pointer which **has to** be connected to the output 'P_Application' on Communicatons-FB. It refers to all data required internally, such as send and receive buffer, etc.

ASDU_Adr
Value of the address of the application service data unit which is processed by this block.
Permissible range with ASDU address length 1 is 1 to 254
Permissible range with ASDU address length 2 is 1 to 65534.
The broadcast addresses 255 (FFh with length 1) or 65535 (FFFFh with length 2) are always accepted.

Accept_ClockSync
TRUE: Received time setting jobs (TI103) are accepted and confirmed positively. The time transferred is used as the system time.
FALSE: TI103 messages are not accepted and confirmed negatively

Sim_GI
A rising edge at this input simulates the initiation of a general interrogation (with originator 0) (from version V1.3). Like at a real interrogation the GI-Confirmation and the GI-End message will be transmitted as well.

Set_Time_Correction
The time from the clock adjustment task (TI103) will be corrected by this value (runtime correction)

SF_Originator
Special function from V1.4: The value denoted here is taken over as originator in messages of the own block and from the SLi-blocks. This occurs as well at interrogations, if the originator = 0 in the interrogation message or equal the value parameterized here.

P_SLi
Pointer (DWORD) for the interconnection of the SLi blocks.
The interconnection can be carried out directly in the CFC plan.
With manual programming you assign the output to a free marker (MD) or a temporary variable for example. Then this has to be assigned again to the corresponding input variables of the SLi blocks.

Reset
Signals a remote reset job received for approx. 5 seconds (TI105)

TimeSync
Signals a received and accepted clock adjustment task (TI103) for one OB1-cycle.

GI_Runs
These output signals a received GI-Request (processed from SL_Org_ASDU). The output remains set until the general interrogation is terminated (GI-End). (From V1.3).

 ©SIEMENS AG 2017

Additional Notes for Command Processing

In Version V1.0 commands are only confirmed (and terminated) if the command was processed from an output block (SLo) thus the combination ASDU, TI and information object address (IOA) was known as well.

Commands with unknown IOA or unknown command types were rejected.

From V1.1 negative command confirmations are sent ‚substitutional' from SL_ORG_ASDU if no command block has processed the message within the next OB1-cycle after receipt of a command message (with COT=6).

The confirmation is carried out with:

- COT 44: unknown TI, if there is no output block programmed for the received TI.
- COT 47: unknown IOA, if there is no output block programmed for the received IOA.
- SL_ORG_ASDU reacts only at messages with ASDU-address, which are suitable to its parameterization
- if the SL_ORG_ASDU confirms negative, the original message is mirrored 1:1 (including probably existing time stamp) only the Cause Of Transmission (COT) will be changed.

From V1.5 the mirroring for all ASDU marked with ‚CON' in the standard (confirmed application services) is carried out centrally if the message cannot be ‚executed'.

The following causes of transmission are possible and executed in the subsequent order:

<44>:= unknown type identification:
If there is no evaluation block for the message type ‚CON'. This applies generally without interconnection at P_Application.

<46>:= unknown common address of the ASDU:
Broadcast addresses are not mirrored.

<45>:= unknown cause of transmission (COT)
In case of receiving unexpected or not supported cause of transmission. Received messages with one of the cause of transmission mentioned here (44-47) are not mirrored.

<47>:= unknown address of the information object

 ©SIEMENS AG 2017

6.1.2. Blocks for process data capture SLi

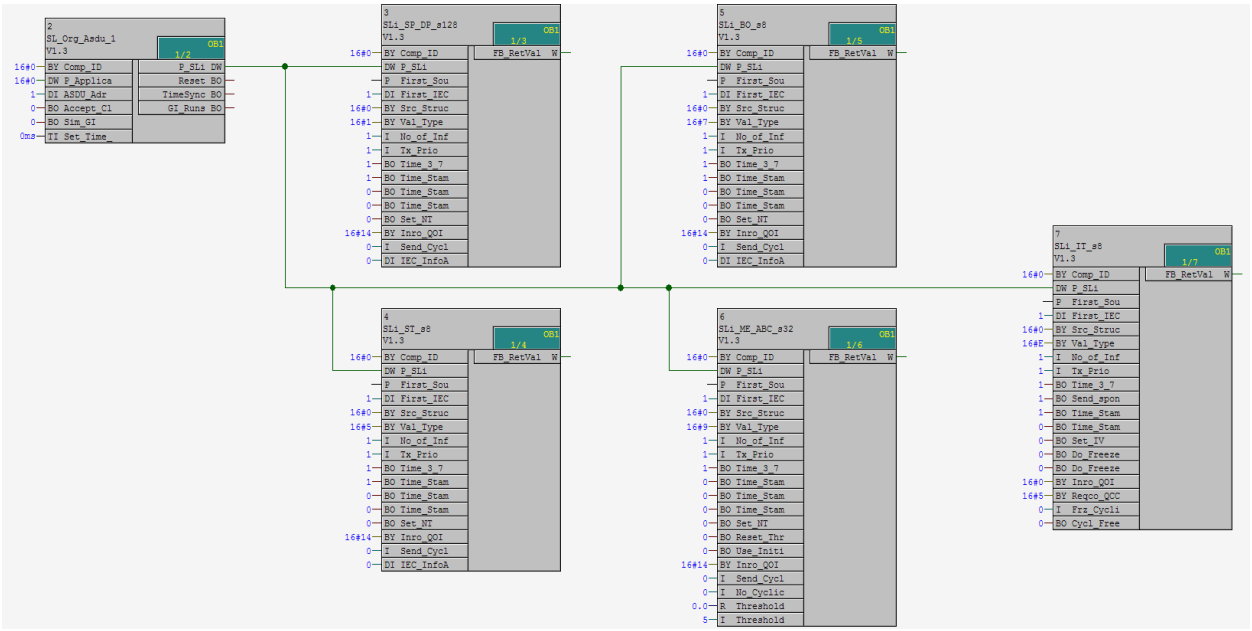
The symbol name encrypts the most important block properties as follows:

SLi_SP_DP_s128	**SLi** is the abbreviation for slave blocks input, in other words process capture blocks
SLi_**SP_DP**_s128	Information types supported by the block with the brief designation: **SP** = Single Point information according to IEC standard
SLi_SP_DP_**s**128	**s** is the abbreviation for sequential processing, in other words consecutive image data capture and IEC addresses each starting with a base position, resp. start address. **p** would be the abbreviation for programmable allocation of data capture point and IEC address.
SLi_SP_DP_s**128**	The (maximum) number of processable information points of the block (here **128**). The number of information points can be increased by multiple call-ups (instances).

The available SLi functional blocks can be found in the following overview.

Block	Block No.	Function
SLi_SP_DP_s128	FB130	For processing max. 128 single (SP) or double points (DP)
SLi_ST_s8	FB131	For processing max. 8 step position (ST)
SLi_BO_s8	FB132	For processing max. 8 bit pattern (BO)
SLi_ME_ABC_s32	FB133	For processing max. 32 measured values, standardized (ME_A), scaled (ME_B) or floating point (ME_C)
SLi_IT_s8	FB134	For processing of max. 8 integrated totals (IT)

Illustration of the necessary interconnection - here in a CFC plan as an example:



6.1.2.1. Single point and double point information - SLi_SP_DP_s128 (FB130)

The block only works in combination with 'SL_ORG_ASDU'.

In the simplest possible case, it monitors a bit field (starting with First_Source_Pos) on changes and tele-controls the contained values as single or double point information.

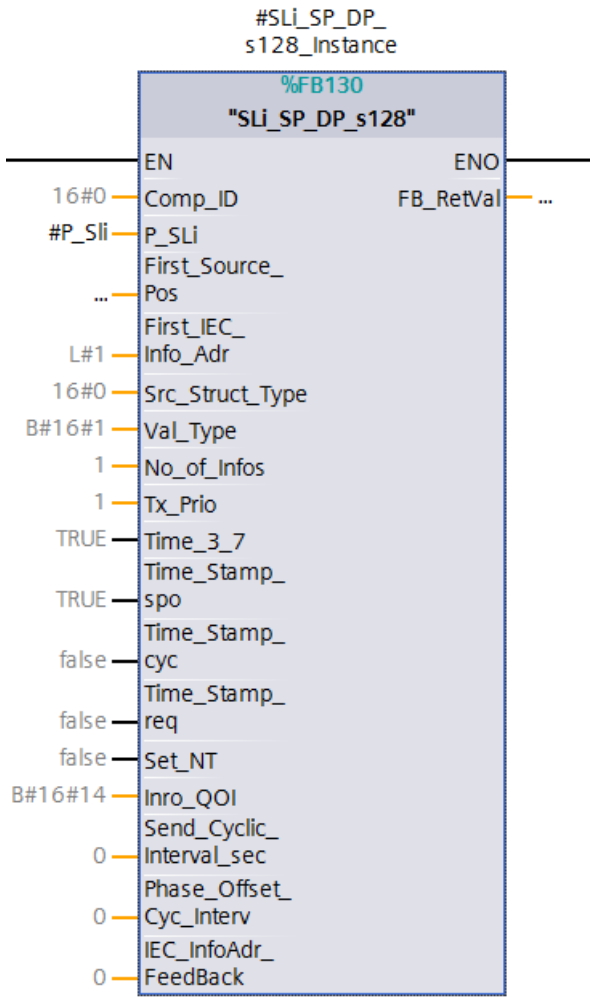


Illustration as FUP block with default values

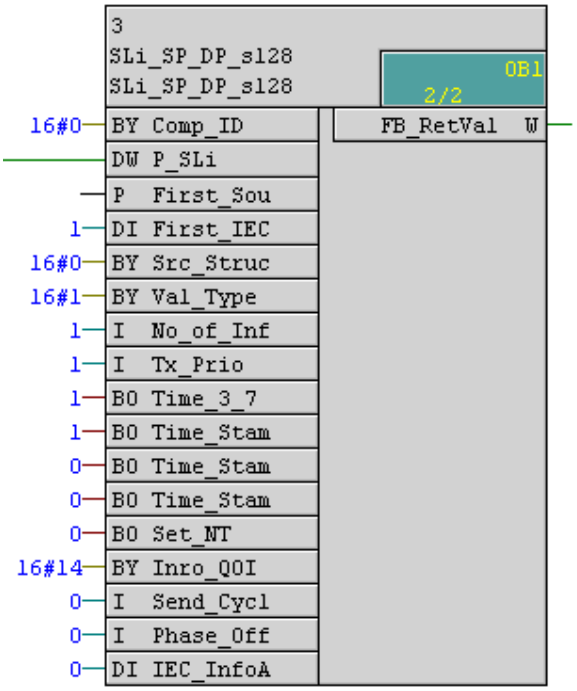


Illustration as CFC block with default values

Important features are:

- With minor effort of parameterization you can telecontrol up to 128 information (No_of_Infos):
  single point information with TI1, 2, 30 (depending on 'Val_Type' + 'Time-Para')
  double point information with TI3, 4, 31 (depending on 'Val_Type' + 'Time-Para')
- Default settings can be read in the respective variable comment; on the other hand they require a minimum of mandatory inputs.
  The minimum requirement is the interconnection of P_SLi and FirstSource_Pos.
- The information object addresses (IOA) are automatically calculated in ascending/consecutive order starting with the programmable basic address ('First_IEC_Info_Adr').
- Any type of image sources starting with 'First_Source_Pos; can be addressed (E, M, DB...)
- Numerous data capture structures can be selected with 'Src_Struct_Type':
  - One-pole or two-pole bit field
  - IEC format (+ status bits)
  - Optional time stamp and/or additional filler bytes
- In addition to spontaneous operation cyclically sending is possible (from V1.6 with cause of transmission 2 = background scan).
  A group interrogation is supported in parallel to the general interrogation (GI).

- Selectable transmission priority (high/low).
- Time stamping can be activated separately for spontaneous, cyclically and interrogated transmission
- A global error input permits the fast identification of information with the status not topical (NT) and simultaneous data capture blocking.
- Input for external realization of feedback processing available (IEC_IOA_Feedback)
- Individual address setting via address parameter-DBs possible.
- From Version V1.5 applicable also in time interrupts (see chapter 6.2)

Block parameters with their default assignment and brief comment

FB130		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_SLi	IN	DWORD	DW#16#0	must be connected with P_to_SLi from SL_Org_ASDU - Module
First_Source_Pos	IN	POINTER	-	First Position of .... e.g. I0.0 - (without default)
First_IEC_Info_Adr	IN	DINT	L#1	First Information object address. Default = 1
Src_Struct_Type :	IN	BYTE	B#16#0	0: (Default) 1 Bit 1: 2 Bit 2: IEC_Info 1 Byte, ... others -> + TimeTag...
Val_Type :	IN	BYTE	B#16#1	1: (Default) SP, 3: DP others are not allowed
No_of_Infos	IN	INT	1	Allowed range: 1-128 - Default = 1
Tx_Prio	IN	INT	1	Priority of transmission: 0 (highest), 15 (lowest) / Default = 1
Time_3_7	IN	BOOL	TRUE	0: -> Time_3 -> TI2 / 1: -> Time_7 -> TI30 / Default = 1
Time_Stamp_spo	IN	BOOL	TRUE	spontaneous Transmission without (0) or with (1) Timestamp / Default = 1
Time_Stamp_cyc :	IN	BOOL	FALSE	cyclic Transmission without (0) or with (1) Timestamp / Default = 0
Time_Stamp_req :	IN	BOOL	FALSE	requested Transmission without (0) or with (1) Timestamp / Default = 0
Set_NT :	IN	BOOL	FALSE	Reset[0], Set[1] all Infos Not Topical
Inro_QOI	IN	BYTE	B#16#14	0: No Reaction; 20 (=14hex,default) General interrogation; 21-36 -> Group 1-16
Send_Cyc_Interval	IN	INT	0	0: without; 1-x [sec] Time interval for cyc transm, works sync to absolute time
Phase_Offset_Cyc_Interv	IN	INT	0	0: without(def), +/-x [sec]: offset for cyc transm related to absolute time
IEC_IOA_FeedBack	IN	DINT	L#0	This IEC-Address will be transmitted with COT 11'Retrem' instead of spo
FB_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: counter of transmitted infos + state

Parameter and function details:

Comp_ID	Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as a byte variable. e.g. B#16#1
P_SLi	P_SLi is a pointer to a common data range from block SL_ORG_ASDU. Therefore it has to be interconnected with its output P_SLi via a DWord variable. The ORG block transfers values such as 'ASDU address', current time stamp, send buffer, and processes e.g. GIs via this pointer or the data range behind it.
First_Source_Pos	Bit pointer to the first information to be collected. e.g. P#E0.0 or P#DB1.DBX0.0 With non bit-oriented image structures byte limits have to be observed (p#Ex.**0**) otherwise there may be 'access errors'.
First_IEC_Info_Adr	Address value of the first collected information object Permissible range with IOA address length 1 is 1 to 255 Permissible range with IOA address length 2 is 1 to 65535. Permissible range with IOA address length 3 is 1 to 16777215 If the information object address is available in structured form (3 octets), it must be converted into a decimal number: IOA dec = octet1 + 256*octet2 + 256*256*octet3 Further information objects are addressed consecutively/in ascending order. Individual address allocation via address parameter-DBs is possible. Detailed description see chapter 6.1.4
Src_Struct_Type	0 → one-pole bit field 1 → two-pole bit field 2 → byte field with IEC assignment can be combined with time stamp capture → For more details, refer to detailed description.
Val_Type	Type identification 1: single point information (preset) Type identification 3: double point information (No others permitted)
No_of_Infos	Number of information to be collected Range of values 1 to 128 Values outside this range result in an output of fault at the FB_RetVal and no further processing is carried out.
Tx_Prio	Used for prioritization of the information transfer. from V1.3: 15: lowest priority 0: highest priority. In case of specifying a priority which is less than the available send buffer, the information is entered automatically into the send buffer with the lowest available priority.
Time_3_7	The time stamp format can be set on input 'Time_3_7': FALSE -> 3-byte length = short or partial time stamp CP24Time2a TRUE -> 7-byte length = long or full time stamp CP56Time2a
Time_Stamp_...	The transmission of the messages with or without time stamp can be parameterized separately for:
Time_Stamp_spo	spontaneous transmission with 'Time_Stamp_spo'
Time_Stamp_cyc	cyclical transmission with 'Time_Stamp_cyc'
Time_Stamp_req	interrogated transmission with 'Time_Stamp_req' FALSE: no time stamp TRUE: with time stamp
Set_NT	The block input 'Set_NT' applies to information processed by this block and works as follows: Set_NT coming -> spontaneous transmission of information with last status (from image) and status NT (not topical), image updating remains locked, GI or cyclical transmission may be carried out from 'old image'. Set_NT going -> Image update and spontaneous transmission of information with the current value

 ©SIEMENS AG 2017

Inro_QOI	Interrogation process: programmable interrogation identification in accordance with IEC standard: - 0: No response - 20 (B#16#14 hex) -> General interrogation - 21 (B#16#15 hex) to 36: Group interrogation + (always) general interrogation In case the block has to react only to group interrogations and **not** to general interrogations, the group number and additional the Bit 2^6 has to be set. Example.: Group interrogation 3 (B#16#43 hex) The actual processing (check that there is actually information of the inquired interrogation group – corresponding reaction = pos/neg confirmation…) is managed by the ORG block. The interrogated data always come from the 'image'
Send_Cyc_Interval	Time interval for cyclical transmission value 0: no cyclical transmission value 1- x sec. time interval in seconds. Information is sent cyclically once within this time period. The timer synchronizes itself to the absolute daily time. From V1.6 messages are transmitted with cause of transmission 2 = background scan.
Phase_Offset_Cyc_Interv	Offset of the time interval for the cyclic transmission compared with the synchronization to the absolute time. Positive as well as negative values are permitted. The unit is seconds.
IEC_IOA_FeedBack	The IEC address given here is sent, if caused by information change, not with transmission cause 3 (spontaneous) but with transmission cause 11 (response on remote command). Can be used for externally programmed response message processing.
FB_RetVal	Positive FB_RetVal values: 0000-0FFFh: Transmit counter (0-4095 decimal) 1xxxh (Bit12) Buffer jam 2xxxh (Bit13) Interrogation active Negative FB_RetVal values: 8101h: Para ‚No_of_Information' < 1 or > 128 8102h P_SLi = 0 or wrong allocated 8104h not supported ‚ValType'

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Src_Struct_Type) in detail

The structure of Src_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			DT	Ext_State	Data_Type	

		Image spacing
Data_Type	Data_Type is used depending on the information type – in this case, as follows: 00 (0): one-pole bit field -> data capture bit by bit 01 (1): two-pole bit field -> double data capture bit by bit 10 (2): byte field with IEC assignment 11 (3): reserved	 1 bit/1 byte* 2 bit/1 byte* 1 byte
Ext_State	With bit set the image is expanded by the status byte 'Ext_State'. As a priority the status byte is used for the optional transfer of time status bits which are not supported by the S7 date and time format.	+ 1 byte
DT	Date and time in S7 format: Normally the CPU time is used. If DT is available, the time stamp submitted is assigned (time status, refer to ext_State).	+ 8 bytes
Add_Dist	Additional image spacing which is taken into account.	+ n bytes
Res	Reserved for future expansions	

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Res	Res	Res	IV	SY	DS	TI

TI: The status bits (DS, SY, and IV) are only transferred if the TI - Bit is set.
This means that Ext_State can also be used as a filler byte (no other function), to establish image structures with even number of bytes.

DS: Daylight Saving Time - summer time:
The bit is transferred directly into SU in IEC messages

SY: synchronized:
The bit is not used in IEC messages

IV: Invalid:
The bit is transferred directly into IV in IEC messages

*) → If additional 'functions' have been activated for the bit field (e.g. DT)

Resultant typical image structures (Examples):

Structure type 0 (B#16#00):

Bit	7	6	5	4	3	2	1	0
Byte 0	7	6	5	4	3	2	1	0
1	15	14	13	12	11	10	9	8
2						...	17	16
...								
15	127							

One-bit data acquisition

Resultant structure: 8 infos per byte

Each bit represents a status 0 (OFF/Going) or 1 (ON/Coming) which you can transfer independently of this in a single point or double point info message.

Structure type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0
Byte 0	Info 3		Info 2		Info 1		Info 0	
1	Info 7		Info 6		Info 5		Info 4	
2			...		9		Info 8	
...								
31	127							

Two-bit data acquisition

Resultant structure: 4 infos per byte

In each case, 2 neighboring bits belong to the information and are transferred directly in double point information messages.
In case of single point information messages the conversion process is as follows
00 -> 0 + status IV (error position 00)
01 -> 0
10 -> 1
11 -> 1 + status IV (error position 11)

 ©SIEMENS AG 2017

Structure type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	
Byte 0	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 0
1	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 1
2	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 2
…									…
127	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 127

1 Byte IEC-data acquisition

Resultant structure: 1 byte per info

In each case, complete bytes are monitored for changes and are transferred unchanged into the message.
Please use the status bits SPI and DPI according to the required transmission type (Val_Typ = SP or DP)

Structure type 18 (B#16#12):

Bit	7	6	5	4	3	2	1	0	
Byte 0	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 0
1	Filler byte								
2	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 1
3	Filler byte								
4	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 2
5	Filler byte								
…									
254	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 127
255	Filler byte								

1 byte IEC data acquisition
+ 1 filler byte (Add_Dist=1)

Resultant structure: 2 bytes per info

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

Structure type 14 (B#16#0E):

Bit	7	6	5	4	3	2	1	0	
Byte 0	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 0
1	Res	Res	Res	Res	IV	SY	DS	TI	
2 … 9	S7 Date and Time								
10	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 1
11	Res	Res	Res	Res	IV	SY	DS	TI	
12 … 15	S7 Date and Time ...								
…									
1270	IV	NT	SB	BL	-	-	- DPI	SPI DPI	Info 127
1271	Res	Res	Res	Res	IV	SY	DS	TI	
1272 … 1279	S7 Date and Time								

1 byte IEC data acquisition
+ 1 byte Ext_State
+ 8 byte time

Resultant structure: 10 bytes per info

6.1.2.2. Step position information - SLi_ST_s8 (FB131)

The block only works in combination with 'SL_ORG_ASDU'.

In the simplest possible case, it monitors a byte field (starting with First_Source_Pos) on changes and telecontrols the contained values as step position messages.

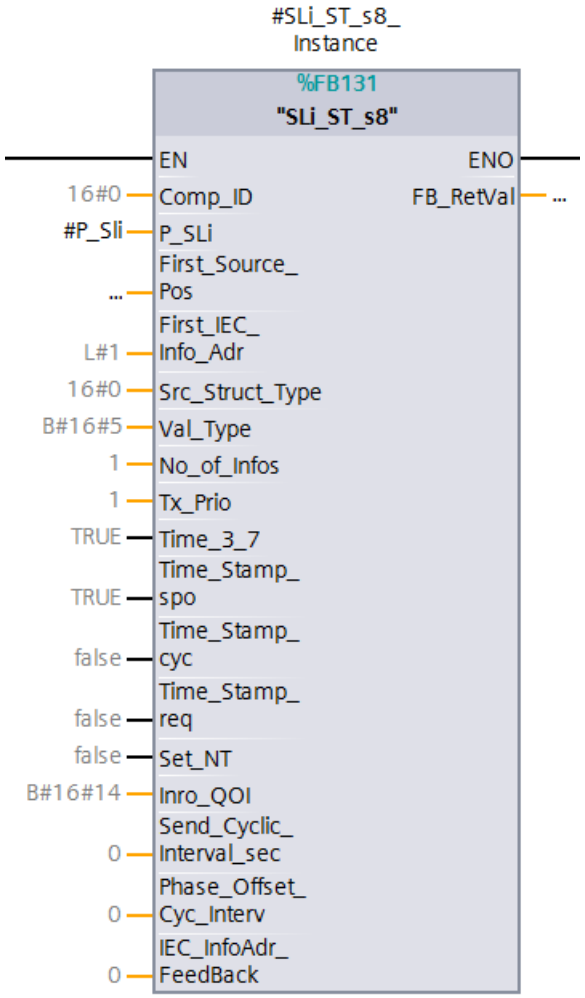


Illustration as FUP block with default values

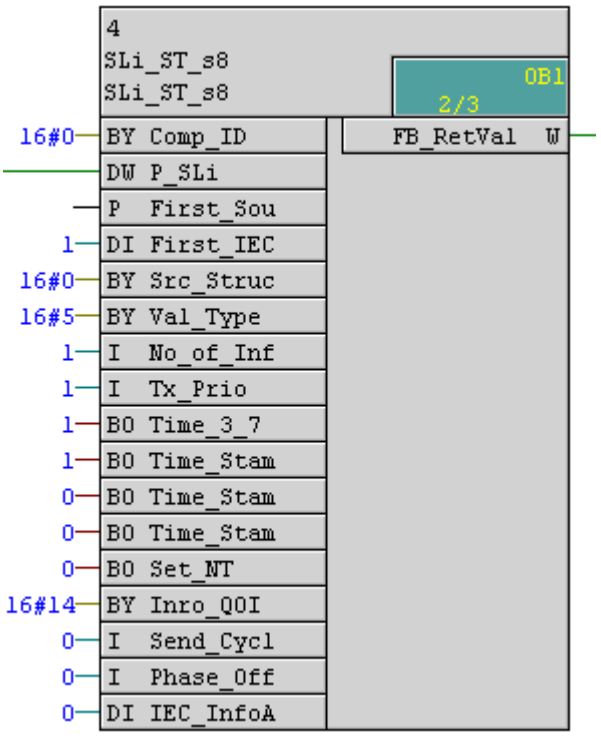


Illustration as CFC block with default values

Important features are:

- With minor effort of parameterization you can telecontrol up to 8 information (No_of_Infos):
  - Transformer position with TI5, 6 or 32 (depending on 'Val_Type' +'Time-Para')
- Default settings can be read in the respective variable comment; on the other hand they require a minimum of mandatory inputs.
  The minimum requirement is the interconnection of P_SLi and FirstSource_Pos.
- The information object addresses (IOA) are automatically calculated in ascending/consecutive order starting with the programmable basic address ('First_IEC_Info_Adr').
- Any type of image sources starting with 'First_Source_Pos; can be addressed (E, M, DB...)
- Numerous data capture structures can be selected with 'Src_Struct_Type':
  - Transformer position 1 byte only
  - IEC format (+status bits)
  - Optional time stamp and/or additional filler bytes

- In addition to spontaneous operation cyclical sending is possible (from V1.6 with cause of transmission 2 = background scan).
  A group interrogation is supported in parallel to the general interrogation (GI).
- Selectable transmission priority (high/low)
- Time stamping can be activated separately for spontaneous, cyclical and interrogated transmission
- A global error input permits the fast identification of information with the status not topical (NT) and simultaneous data capture blocking.
- Input for external realization of feedback processing available (IEC_IOA_Feedback).
- Individual address setting via address parameter-DBs possible
- From Version V1.5 applicable also in time interrupts (see chapter 6.2)

Block parameters with their default assignment and brief comment

FB131		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_SLi	IN	DWORD	DW#16#0	must be connected with P_to_SLi from SL_Org_ASDU - Block
First_Source_Pos	IN	POINTER	-	First Position of .... e.g. I0.0 - (without default)
First_IEC_Info_Adr	IN	DINT	L#1	First Information object address / Default = 1
Src_Struct_Type :	IN	BYTE	B#16#0	0: (Default) only (1) val-byte, 1: IEC_Info 2 Byte, ... others -> TimeTag...
Val_Type :	IN	BYTE	B#16#5	5: (Default) ST / others are not supported
No_of_Infos	IN	INT	1	Allowed range: 1-8 / Default = 1
Tx_Prio	IN	INT	1	Priority of transmission: 0 (highest), 15 (lowest) / Default = 1
Time_3_7	IN	BOOL	TRUE	0: -> Time_3 -> TI2 / 1: -> Time_7 -> TI30 / Default = 1
Time_Stamp_spo	IN	BOOL	TRUE	spontaneous Transmission without (0) or with (1) Timestamp / Default = 1
Time_Stamp_cyc :	IN	BOOL	FALSE	cyclic Transmission without (0) or with (1) Timestamp / Default = 0
Time_Stamp_req :	IN	BOOL	FALSE	requested Transmission without (0) or with (1) Timestamp / Default = 0
Set_NT :	IN	BOOL	FALSE	Reset[0], Set[1] all Info Not Topical
Inro_QOI	IN	BYTE	B#16#14	0: No Reaction; 20 (=14hex,default) General interrogation; 21-36 -> Group 1-16
Send_Cyc_Interval	IN	INT	0	0: without; 1-x [sec] Time interval for cyc transm, works sync to absolute time
Phase_Offset_Cyc_Interv	IN	INT	0	0: without(def), +/-x [sec]: offset for cyc transm related to absolute time
IEC_IOA_FeedBack	IN	DINT	L#0	This IEC-Address will be transmitted with COT 11 'Retrem' instead of spo
FB_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: counter of transmitted infos + state

Parameter and function details:

Comp_ID	Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as a byte variable. e.g. B#16#1
P_SLi	P_SLi is a pointer to a common data range from block SL_ORG_ASDU. Therefore it has to be interconnected with its output P_SLi via a DWord variable. The ORG block transfers values such as 'ASDU address', current time stamp, send buffer, and processes e.g. GIs, via this pointer or the data range behind it.
First_Source_Pos	Bit pointer to the first information to be collected. e.g. P#E0.0 or P#DB1.DBX0.0 With non bit-oriented image structures, byte limits have to be observed (p#Ex.**0**); otherwise there may be 'access errors'.
First_IEC_Info_Adr	Address value of the first collected information object Permissible range with IOA address length 1 is 1 to 255 Permissible range with IOA address length 2 is 1 to 65535. Permissible range with IOA address length 3 is 1 to 16777215 If the information object address is available in structured form (3 octets), it must be converted into a decimal number: IOA dec = octet1 + 256*octet2 + 256*256*octet3 Further information objects are addressed consecutively/in ascending order. Individual address allocation via address parameter-DBs is possible. Detailed description see chapter 6.1.4
Src_Struct_Type	0 -> 1 byte transformer position 1 -> 1 byte transformer position + 1byte status, IEC assignment can be combined with time stamp capture -> For more details, refer to detailed description.
Val_Type	Type identification 5. Step position (preset) (No others permitted)
No_of_Infos	Number of information to be collected Range of values 1 to 8 Values outside this range result in an output of fault at the FB_RetVal and no further processing is carried out.
Tx_Prio	Used for prioritization of the information transfer. from V1.3 15: Lowest priority 0: Highest priority. In case of specifying a priority which is less than the available send buffer, the information is entered automatically into the send buffer with the lowest available priority.
Time_3_7	The time stamp format can be set on input 'Time_3_7': FALSE -> 3-byte length = short or partial time stamp CP24Time2a TRUE -> 7-byte length = long or full time stamp CP56Time2a
Time_Stamp_...	The transmission of the messages with or without time stamp can be parameterized separately for:
Time_Stamp_spo	spontaneous transmission with 'Time_Stamp_spo'
Time_Stamp_cyc	cyclical transmission with 'Time_Stamp_cyc'
Time_Stamp_req	interrogated transmission with 'Time_Stamp_req' FALSE: no time stamp TRUE: with time stamp
Set_NT	The block input 'Set_NT' applies to information processed by this block and works as follows: Set_NT coming -> spontaneous transmission of information with last status (from image) and status NT (not topical), image updating remains locked, GI or cyclical transmission may be carried out from 'old image'. Set_NT going -> Image update and spontaneous transmission of information with the current value

 ©SIEMENS AG 2017

Inro_QOI	Interrogation process: programmable interrogation identification in accordance with IEC standard: - 0: No response - 20 (B#16#14 hex) -> General interrogation - 21 (B#16#15 hex) to 36: Group interrogation + (always) general interrogation In case the block has to react only to group interrogations and not to general interrogations, the group number and additional the Bit 2^6 has to be set. Example.: Group interrogation 3 (B#16#43 hex) The actual processing (check that there is actually information of the inquired interrogation group – corresponding reaction = pos/neg confirmation…) is managed by the ORG block. The interrogated data always come from the 'image'
Send_Cyc_Interval	Time interval for cyclical transmission value 0: no cyclical transmission value 1- x sec. time interval in seconds. Information is sent cyclically once within this time period. The timer synchronizes itself to the absolute daily time. From V1.6 messages are transmitted with cause of transmission 2 = background scan.
Phase_Offset_Cyc_Interv	Offset of the time interval for the cyclic transmission compared with the synchronization to the absolute time. Positive as well as negative values are permitted. The unit is seconds.
IEC_IOA_FeedBack	The IEC address given here is sent, if triggered by information change, not with transmission cause 3 (spontaneous) but with transmission cause 11 (response on remote command). Can be used for externally programmed response message processing.
FB_RetVal	Positive FB_RetVal values: 0000-0FFFh: Transmit counter (0-4095 decimal) 1xxxh (Bit12) Buffer jam 2xxxh (Bit13) Interrogation active Negative FB_RetVal values: 8101h: Para ‚No_of_Infos' < 1 or > 128 8102h P_SLi = 0 or wrong allocated 8104h not supported ‚ValType'

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Src_Struct_Type) in detail

The structure of Src_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			DT	Ext_State	Data_Type	

		Image spacing
Data_Type	Data_Type is used depending on the information type – in this case, as follows: 00 (0): 1 byte transformer position 01 (1): 1 byte transformer position + 1 byte status, IEC assignment 10 (2): reserved 11 (3): reserved	 1 byte 2 bytes
Ext_State	With bit set the image is expanded by the status byte 'Ext_State'. As a priority the status byte is used for the optional transfer of time status bits which are not supported by the S7 date and time format.	+ 1 byte
DT	Date and time in S7 format: Normally the CPU time is used. If DT is available, the time stamp submitted is assigned (time status, refer to ext_State).	+ 8 bytes
Add_Dist	Additional image spacing which is taken into account.	+ n bytes
Res	Reserved for future expansions	

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Res	Res	Res	IV	SY	DS	TI

TI: The status bits (DS, SY, and IV) are only transferred if the TI - Bit is set.
This means that Ext_State can also be used as a filler byte (no other function), to establish image structures with even number of bytes.

DS: Daylight Saving Time - summer time:
The bit is transferred directly into SU in IEC messages

SY: synchronized:
The bit is not used in IEC messages

IV: Invalid:
The bit is transferred directly into IV in IEC messages

Resultant typical image structures (Examples):

Structure type 0 (B#16#00):

Bit	7	6	5	4	3	2	1	0	
Byte 0	Transformer position								Info 1
2	Transformer position								Info 2
3	Transformer position								Info 3
…									
7	Transformer position								Info 8

1 byte transformer position

Resultant structure: 1 byte per info

Each 1 byte represents a transformer position in IEC format which can be transferred as a step position information message.

Structure type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	
Byte 0	Transformer position								Info 1
1	IV	NT	SB	BL	-	-	-	OV	
2	Transformer position								Info 2
3	IV	NT	SB	BL	-	-	-	OV	
…									
14	Transformer position								Info 8
15	IV	NT	SB	BL	-	-	-	OV	

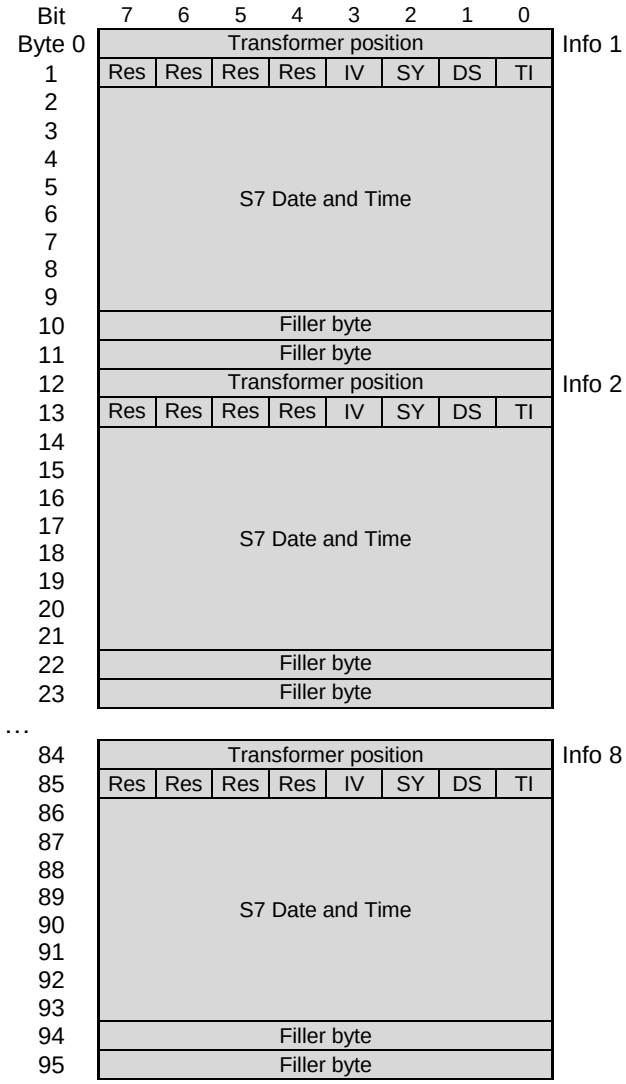
1 byte transformer position + 1 byte status, IEC assignment

Resultant structure: 2 bytes per info

Each 1 byte represents a transformer position in IEC format,
the quality identifier for this transformer position is stored in each second byte in IEC format.

 ©SIEMENS AG 2017

Structure type 44 (B#16#2C):



1 byte transformer position
+ 1 byte Ext_State
+ 8 byte time
+ 2 filler byte

Resultant structure: 12 bytes per info

Each 1 byte represents a transformer position in IEC format.

Followed by
1 byte Ext_State and
8 byte S7 Date and Time

The status bits from Ext_State (DS, SY, IV) are only transferred if the TI - Bit is set.
Even if these status bits are not necessarily being used, the use of the byte Ext_State as an additional filler byte is advisable in this case.
This means that the entry Date and Time start at even-order byte numbers in each case.
This is advantageous for the further processing of Date and Time.

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

The filler bytes in this example provide a data structure with 12 byte grid.

6.1.2.3. Bit pattern messages - SLi_BO_s8 (FB132)

The block only functions in combination with 'SL_ORG_ASDU'.

In the simplest possible case it monitors a field of consecutive DWords (starting with First_Source_Pos) on changes and telecontrols the contained values as a bit pattern message.

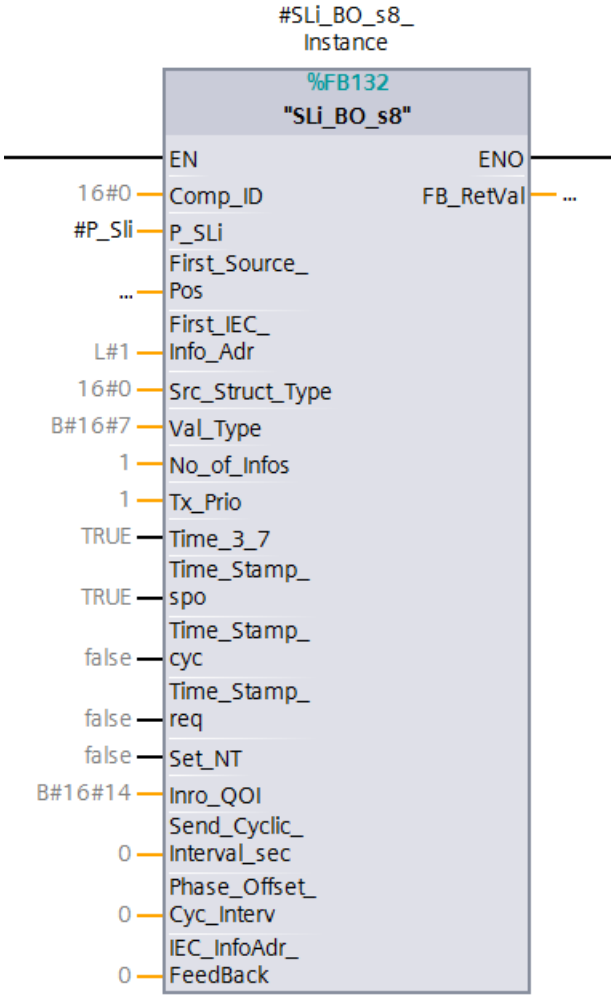


Illustration as FUP block with default values

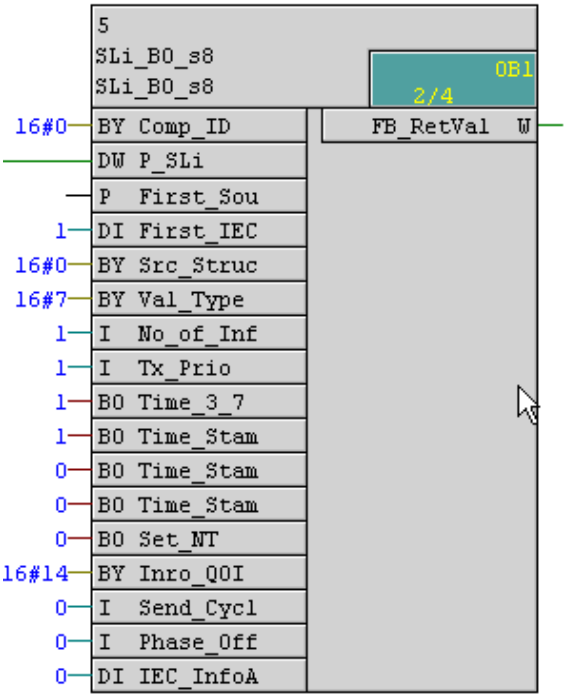


Illustration as CFC block with default values

Important features are:

- With minor effort of parameterization you can telecontrol up to 8 information (No_of_Infos):
  - Bit pattern with TI7, 8, 33 (depending on 'Val_Type' +'Time-Para')
- Default settings can be read in the respective variable comment; on the other hand they require a minimum of mandatory inputs.
  The minimum requirement is the interconnection of P_SLi and FirstSource_Pos.
- The information object addresses (IOA) are automatically calculated in ascending/in consecutive order starting with the programmable basic address ('First_IEC_Info_Adr').
- Any type of image sources starting with 'First_Source_Pos; can be addressed (E, M, DB...)
- Numerous data capture structures can be selected with 'Src_Struct_Type':
  - 4 byte bit pattern
  - IEC format (+ status bits)
  - Optional time stamp and/or additional filler bytes
- In addition to spontaneous operation cyclical sending is possible (from V1.6 with cause of transmission 2 = background scan).
  A group interrogation is supported in parallel to the general interrogation (GI).

- Selectable transmission priority (high/low).
- Time stamping can be activated separately for spontaneous, cyclical and interrogated transmission
- A global error input permits the fast identification of information with the status not topical (NT) and simultaneous data capture blocking.
- Input for external realization of feedback processing available (IEC_IOA_Feedback).
- Individual address setting via address parameter-DBs possible
- From Version V1.5 applicable also in time interrupts (see chapter 6.2)

Block parameters with their default assignment and brief comment

FB132		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_SLi	IN	DWORD	DW#16#0	must be connected with P_to_SLi from SL_Org_ASDU - Block
First_Source_Pos	IN	POINTER	-	First Position of .... e.g. I0.0 - (without default)
First_IEC_Info_Adr	IN	DINT	L#1	First Information object address / Default = 1
Src_Struct_Type :	IN	BYTE	B#16#0	0: (Default) only bit pattern, 1: IEC_Info 5 Byte, ... others -> TimeTag...
Val_Type :	IN	BYTE	B#16#7	7: (Default) BO / others are not supported
No_of_Infos	IN	INT	1	Allowed range: 1-8 / Default = 1
Tx_Prio	IN	INT	1	Priority of transmission: 0 (highest), 15 (lowest) / Default = 1
Time_3_7	IN	BOOL	TRUE	0: -> Time_3 -> TI2 / 1: -> Time_7 -> TI30 / Default = 1
Time_Stamp_spo	IN	BOOL	TRUE	spontaneous Transmission without (0) or with (1) Timestamp / Default = 1
Time_Stamp_cyc :	IN	BOOL	FALSE	cyclic Transmission without (0) or with (1) Timestamp / Default = 0
Time_Stamp_req :	IN	BOOL	FALSE	requested Transmission without (0) or with (1) Timestamp / Default = 0
Set_NT :	IN	BOOL	FALSE	Reset[0], Set[1] all Infos Not Topical
Inro_QOI	IN	BYTE	B#16#14	0: No Reaction; 20 (=14hex,default) General interrogation; 21-36 -> Group 1-16
Send_Cyc_Interval	IN	INT	0	0: without; 1-x [sec] Time interval for cyc transm, works sync to absolute time
Phase_Offset_Cyc_Interv	IN	INT	0	0: without(def), +/-x [sec]: offset for cyc transm related to absolute time
IEC_IOA_FeedBack	IN	DINT	L#0	This IEC-Address will be transmitted with COT 11'Retrem' instead of spo
FB_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: counter of transmitted infos + state

Parameter and function details:

Comp_ID	Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as a byte variable. e.g. B#16#1
P_SLi	P_SLi is a pointer to a common data range from block SL_ORG_ASDU. Therefore it has to be interconnected with its output P_SLi via a DWord variable. The ORG block transfers values such as 'ASDU address', current time stamp, send buffer, and processes e.g. GIs, via this pointer or the data range behind it.
First_Source_Pos	Bit pointer to the first information to be collected. e.g. P#E0.0 or P#DB1.DBX0.0 With non bit-oriented image structures, byte limits have to be observed (p#Ex.**0**); otherwise there may be 'access errors'.
First_IEC_Info_Adr	Address value of the first collected information object Permissible range with IOA address length 1 is 1 to 255 Permissible range with IOA address length 2 is 1 to 65535. Permissible range with IOA address length 3 is 1 to 16777215 If the information object address is available in structured form (3 octets), it must be converted into a decimal number: IOA dec = octet1 + 256*octet2 + 256*256*octet3 Further information objects are addressed consecutively/in ascending order. Individual address allocation via address parameter-DBs is possible. Detailed description see chapter 6.1.4
Src_Struct_Type	0 → 4 byte bit pattern 1 → 4 byte bit pattern + 1byte status, IEC assignment can be combined with time stamp capture → For more details, refer to detailed description.
Val_Type	Type identification 7. bit string of 32 bit (preset) (No others permitted)
No_of_Infos	Number of information to be collected Range of values 1 to 8 Values outside this range result in an output of fault at the FB_RetVal and no further processing is carried out.
Tx_Prio_L_H	Used for prioritization of the information transfer. FALSE: low priority TRUE: high priority. In case of specifying a priority which is less than the available send buffer, the information is entered automatically into the send buffer with the lowest available priority.
Time_3_7	The time stamp format can be set on input 'Time_3_7': FALSE -> 3-byte length = short or partial time stamp CP24Time2a TRUE -> 7-byte length = long or full time stamp CP56Time2a
Time_Stamp_... **Time_Stamp_spo** **Time_Stamp_cyc** **Time_Stamp_req**	The transmission of the messages with or without time stamp can be parameterized separately for: spontaneous transmission with 'Time_Stamp_spo' cyclical transmission with 'Time_Stamp_cyc' interrogated transmission with 'Time_Stamp_req' FALSE: no time stamp TRUE: with time stamp
Set_NT	The block input 'Set_NT' applies to information processed by this block and works as follows: Set_NT coming -> spontaneous transmission of information with last status (from image) and status NT (not topical), image updating remains locked, GI or cyclical transmission may be carried out from 'old image'. Set_NT going -> Image update and spontaneous transmission of information with the current value

 ©SIEMENS AG 2017

Inro_QOI	Interrogation process: programmable interrogation identification in accordance with IEC standard: - 0: No response - 20 (B#16#14 hex) -> General interrogation - 21 (B#16#15 hex) to 36: Group interrogation + (always) general interrogation In case the block has to react only to group interrogations and not to general interrogations, the group number and additional the Bit 2^6 has to be set. Example.: Group interrogation 3 (B#16#43 hex) The actual processing (check that there is actually information of the inquired interrogation group – corresponding reaction = pos/neg confirmation…) is managed by the ORG block. The interrogated data always come from the 'image'
Send_Cyc_Interval	Time interval for cyclical transmission value 0: no cyclical transmission value 1- x sec. time interval in seconds. Information is sent cyclically once within this time period. The timer synchronizes itself to the absolute daily time. From V1.6 messages are transmitted with cause of transmission 2 = background scan.
Phase_Offset_Cyc_Interv	Offset of the time interval for the cyclic transmission compared with the synchronization to the absolute time. Positive as well as negative values are permitted. The unit is seconds.
IEC_IOA_FeedBack	The IEC address given here is sent, if triggered by information change, not with transmission cause 3 (spontaneous) but with transmission cause 11 (response on remote command). Can be used for externally programmed response message processing.
FB_RetVal	Positive FB_RetVal values: 0000-0FFFh: Transmit counter (0-4095 decimal) 1xxxh (Bit12) Buffer jam 2xxxh (Bit13) Interrogation active Negative FB_RetVal values: 8101h: Para ‚No_of_Infos' < 1 or > 128 8102h P_SLi = 0 or wrong allocated 8104h not supported ‚ValType'

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Src_Struct_Type) in detail

The structure of Src_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			DT	Ext_State	Data_Type	

		Image spacing
Data_Type	Data_Type is used depending on the information type – in this case, as follows: 00 (0): 4 byte bit pattern 01 (1): 4 byte bit pattern + 1 byte status, IEC assignment 10 (2): reserved 11 (3): reserved	 4 bytes 5 bytes
Ext_State	With bit set the image is expanded by the status byte 'Ext_State'. As a priority the status byte is used for the optional transfer of time status bits which are not supported by the S7 date and time format.	+ 1 byte
DT	Date and time in S7 format: Normally the CPU time is used. If DT is available, the time stamp submitted is assigned (time status, refer to ext_State).	+ 8 bytes
Add_Dist	Additional image spacing which is taken into account.	+ n bytes
Res	Reserved for future expansions	

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Res	Res	Res	IV	SY	DS	TI

TI: The status bits (DS, SY, and IV) are only transferred if the TI - Bit is set.
This means that Ext_State can also be used as a filler byte (no other function), to establish image structures with even number of bytes.

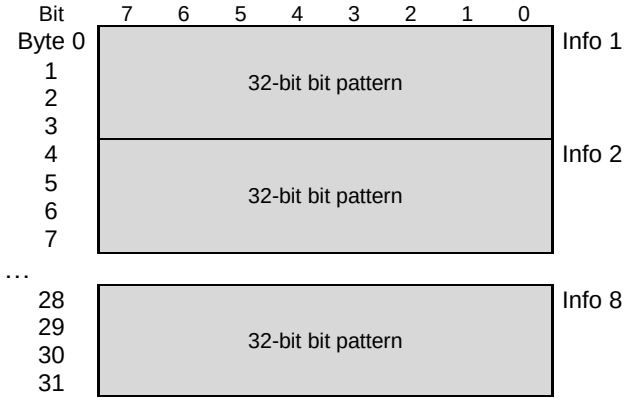
DS: Daylight Saving Time - summer time:
The bit is transferred directly into SU in IEC messages

SY: synchronized:
The bit is not used in IEC messages

IV: Invalid:
The bit is transferred directly into IV in IEC messages

Resultant typical image structures (Examples):

Structure type 0 (B#16#00):



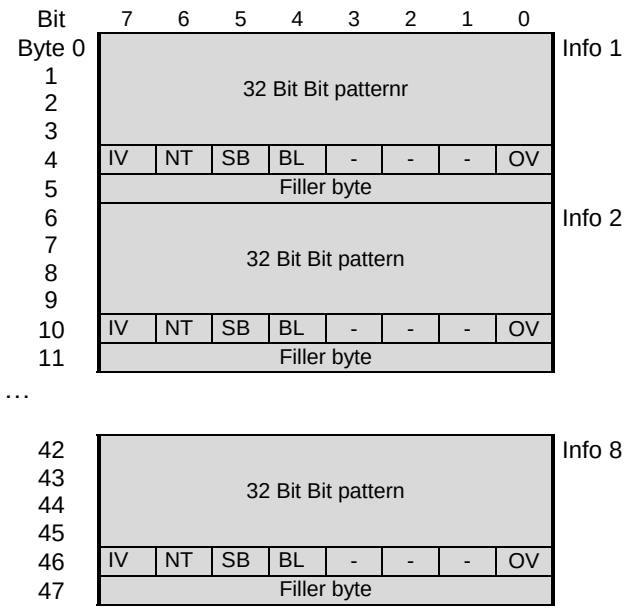
4 byte bit pattern

Resultant structure: 4 bytes per info

Each 4 bytes represent a 32-bit bit pattern which can be transferred as a bit pattern message.

Struktur type 17 (B#16#11): 4 Byte bit pattern

+ 1Byte Status, IEC- assignment
+ 1 Filler byte



Resultant structure: 6 bytes per info

The first 4 bytes each represent a 32-bit bit pattern.

In the fifth byte respectively the quality identifier is stored for this bit pattern in the IEC format.
It follows one filler byte each.

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

Structure type 45 (B#16#2D):

Bit	7	6	5	4	3	2	1	0	
Byte 0	32-bit bit pattern								Info 1
1									
2									
3									
4	IV	NT	SB	BL	-	-	-	OV	
5	Res	Res	Res	Res	IV	SY	DS	TI	
6	S7 Date and Time								
7									
8									
9									
10									
11									
12									
13									
14	Filler byte								
15	Filler byte								
16	32-bit bit pattern								Info 2
17									
18									
19									
20	IV	NT	SB	BL	-	-	-	OV	
21	Res	Res	Res	Res	IV	SY	DS	TI	
22	S7 Date and Time								
23									
24									
25									
26									
27									
28									
29									
30	Filler byte								
31	Filler byte								
…									
112	32-bit bit pattern								Info 8
113									
114									
115									
116	IV	NT	SB	BL	-	-	-	OV	
117	Res	Res	Res	Res	IV	SY	DS	TI	
118	S7 Date and Time								
119									
120									
121									
122									
123									
124									
125									
126	Filler byte								
127	Filler byte								

4 byte bit patterns
+ 1 byte status, IEC assignment
+ 1 byte Ext_State
+ 8 byte time
+ 2 filler byte

Resultant structure: 16 bytes per info

The first 4 bytes represent a 32-bit bit pattern, in each fifth byte the quality identification for this bit pattern is stored in IEC format.

Followed by
1 byte Ext_State
8 byte time and date
2 filler bytes

The status bits from Ext_State (DS, SY, IV) are only transferred if the TI - Bit is set.
Even if these status bits are not necessarily used, the use of the byte Ext_State as an additional filler byte is advisable in this case.
This means that the entry Date and Time start at even-order byte numbers in each case.
This is advantageous for the further processing of Date and Time.

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

The filler bytes in this example provide a data structure with 16 byte grid.

 ©SIEMENS AG 2017

6.1.2.4. Measured values - SLi_ME_ABC_s32 (FB133)

The block only works in combination with 'SL_ORG_ASDU'.

In the simplest possible case it monitors a consecutive integer field (starting with First_Source_Pos) on changes and telecontrols the contained values as measured value in normalized, scaled or floating point format.

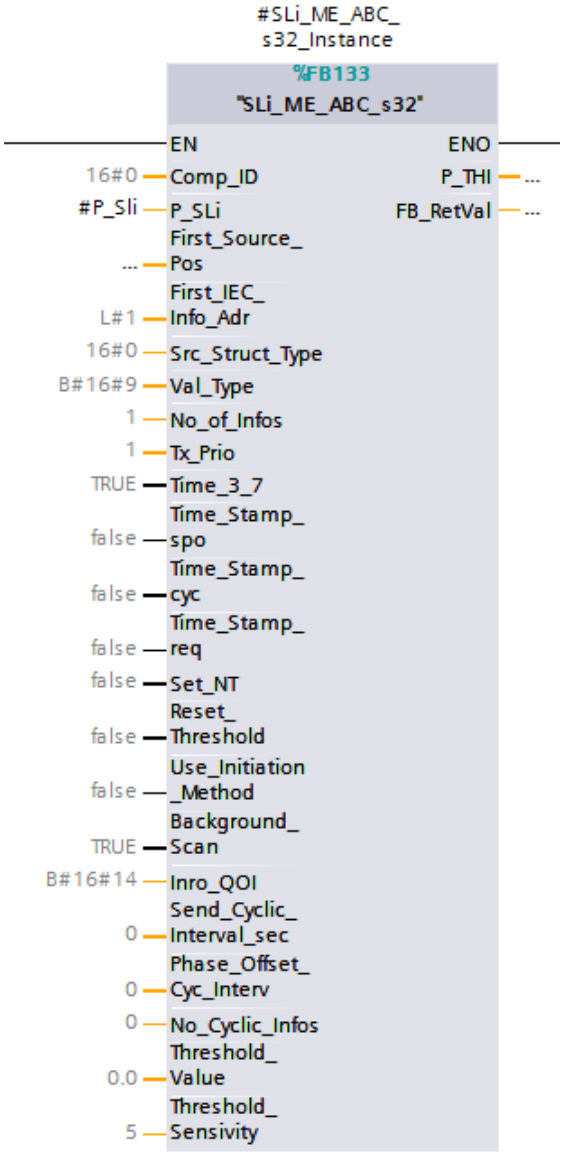


Illustration as FUP block with default values

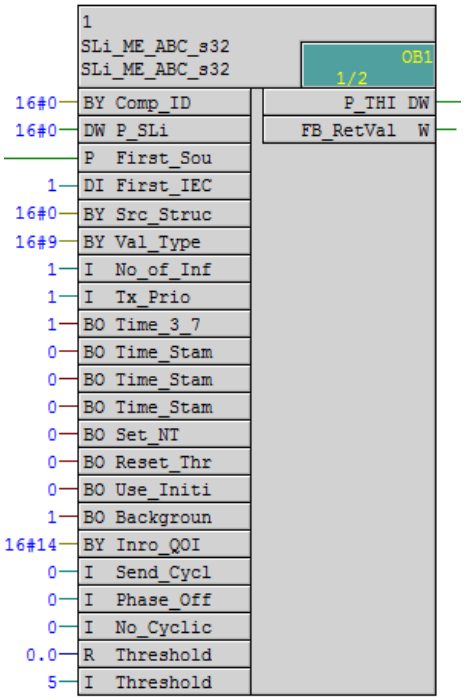


Illustration as CFC block with default values

Important features are:

With minor effort of parameterization you can telecontrol up to 32 information (No_of_Infos):

- as measured value normalized value with TI9, 10, 34
  (depending on 'Val_Type' +'Time-Para'),
- as measured value with scaled value with TI11, 12, 35
  (depending on 'Val_Type' +'Time-Para'),
- as measured value with short floating point number with TI13, 14, 36
  (depending on 'Val_Type' +'Time-Para')

➢ Default settings can be read in the respective variable comment; on the other hand they require a minimum of mandatory inputs.
The minimum requirement is the interconnection of P_SLi and FirstSource_Pos.

- The information object addresses (IOA) are automatically calculated in ascending/consecutive order starting with the programmable basic address ('First_IEC_Info_Adr').
- Individual address setting via address parameter-DBs possible
- Any type of image sources starting with 'First_Source_Pos'; can be addressed (E, M, DB...)
- Numerous data capture structures can be selected with 'Src_Struct_Type':
  - 2 byte integer
  - 4 byte short real
  - IEC format (+ status bits)
  - Optional time stamp and/or additional filler bytes
- Integrated reassuring method according to the additive threshold value method.
- From V1.6 an AddOn block for individual adaption of the threshold of individual measured values is available.
- In addition to spontaneous operation cyclical sending is possible From V1.6 the cause of transmission is switchable between 'cyclic' and 'background scan'.
  A group interrogation is supported in parallel to the general interrogation (GI).
- A global error input permits the fast identification of information with the status not topical (NT) and simultaneous data capture blocking.
- From Version V1.5 applicable also in time interrupts (see chapter 6.2)

Block parameter with their default assignment and brief comment

FB133		Type	Default	Block-Comment/brief description.
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_SLi	IN	DWORD	DW#16#0	must be connected with P_to_SLi from SL_Org_ASDU – Module
First_Source_Pos	IN	POINTER	-	First Position of …. e.g. I0.0 – (without default)
First_IEC_Info_Adr	IN	DINT	L#1	First Information object address / Default = 1
Src_Struct_Type :	IN	BYTE	B#16#0	0:(Default) int; 1:IEC int+state; 2:real; 3:IEC real+state; ...TimeTag...
Val_Type :	IN	BYTE	B#16#9	16#9:(Default) ME_NA (9,norm.); 16#B: ME_NB (11,scaled); 16#D: ME_NC (13,flaot)
No_of_Infos	IN	INT	1	Allowed range: 1-32 / Default = 1
Tx_Prio	IN	INT	1	Priority of transmission: 0 (highest), 15 (lowest) / Default = 1
Time_3_7	IN	BOOL	TRUE	0: -> Time_3 -> TI2 / 1: -> Time_7 -> TI30 / Default = 1
Time_Stamp_spo	IN	BOOL	FALSE	spontaneous Transmission without (0) or with (1) Timestamp / Default = 0
Time_Stamp_cyc :	IN	BOOL	FALSE	cyclic Transmission without (0) or with (1) Timestamp / Default = 0
Time_Stamp_req :	IN	BOOL	FALSE	requested Transmission without (0) or with (1) Timestamp / Default = 0
Set_NT :	IN	BOOL	FALSE	Reset[0], Set[1] all Infos Not Topical
Reset_Threshold	IN	BOOL	FALSE	Resets the actual internal calculated threshold value to 0.
Use_Initiation_Method	IN	BOOL	FALSE	0 (def):new entry; 1:existing measured values in buffer will be updated;
Background_Scan	IN	BOOL	TRUE	0: without; 1 (def): cyclic transmission works as background scan
Inro_QOI	IN	BYTE	B#16#14	0: No Reaction; 20 (=14hex,default) General interrogation; 21-36 -> Group 1-16
Send_Cyc_Interval_sec	IN	INT	0	0: without; 1-x [sec] Time interval for cyc transm, works sync to absolute time
Phase_Offset_Cyc_Interv	IN	INT	0	0: without(def), +/-x [sec]: offset for cyc transm related to absolute time
No_Cyclic_Infos	IN	INT	0	0: all (def); 1…n limits the number of meas. values per cycl. activation
Threshold_Value	IN	REAL	0.00000e+000	0: Dynamic (Default); pos. Value -> Valid Value

Threshold_Sensivity	IN	INT	5	0 (without) – 100 [% Percent] of Threshold_Value ….Default: 5 %
FB_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: counter of transmitted infos + state

Parameter and function details:

Comp_ID
Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as a byte variable.
e.g. B#16#1

P_SLi
P_SLi is a pointer to a common data range from block SL_ORG_ASDU. Therefore it has to be interconnected with its output P_SLi via a DWord variable.

The ORG block transfers values such as 'ASDU address', current time stamp, send buffer, and processes e.g. GIs, via this pointer or the data range behind it.

First_Source_Pos
Bit pointer to the first information to be collected.
e.g. P#E0.0 or P#DB1.DBX0.0
With non bit-oriented image structures, byte limits have to be observed (p#Ex.**0**); otherwise there may be 'access errors'.

First_IEC_Info_Adr
Address value of the first collected information object
Permissible range with IOA address length 1 is 1 to 255
Permissible range with IOA address length 2 is 1 to 65535.
Permissible range with IOA address length 3 is 1 to 16777215
If the information object address is available in structured form (3 octets), it must be converted into a decimal number:
IOA dec = octet1 + 256*octet2 + 256*256*octet3
Further information objects are addressed consecutively/in ascending order.
Individual address allocation via address parameter-DBs is possible.
Detailed description see chapter 6.1.4.

Src_Struct_Type
0 → 2 byte measured value (Integer)
1 → 2 byte measured value (Integer) + 1 byte status, IEC assignment
2 → 4 byte measured value (Short Real)
3 → 4 byte measured value (Short Real) + 1 byte status, IEC assignment
can be combined with time stamp capture
→ For more details, refer to detailed description.

Val_Type
Type identification 9: Measured value, normalized value (preset)
Type identification 11: Measured value, scaled value
Type identification 13: Measured value, short floating point number
(No others permitted)

No_of_Infos
Number of information to be collected
Range of values 1 to 32
Values outside this range result in an output of fault at the FB_RetVal and no further processing is carried out.

Tx_Prio
Used for prioritization of the information transfer.
from V1.3
15: lowest priority
0: highest priority.
In case of specifying a priority which is less than the available send buffer, the information is entered automatically into the send buffer with the lowest available priority.

Time_3_7
The time stamp format can be set on input 'Time_3_7':
FALSE → 3-byte length = short or partial time stamp CP24Time2a
TRUE → 7-byte length = long or full time stamp CP56Time2a

Time_Stamp_...
The transmission of the messages with or without time stamp can be parameterized separately for:

Time_Stamp_spo
spontaneous transmission with 'Time_Stamp_spo'

Time_Stamp_cyc
cyclical transmission with 'Time_Stamp_cyc'

Time_Stamp_req
interrogated transmission with 'Time_Stamp_req'

FALSE: no time stamp
TRUE: with time stamp

Set_NT
The block input 'Set_NT' applies to information processed by this block and works as follows:
Set_NT coming → spontaneous transmission of information with last status (from image) and status NT (not topical), image updating remains locked, GI or cyclical transmission may be carried out from 'old image'.
Set_NT going → Image update and spontaneous transmission of information with the current value

Reset_Threshold

With rising edge the actual calculated threshold value is reset.
In normal operation a new threshold value is taken over only if the last threshold has been reached. This may take a long time in case of adverse circumstances.
With external wiring of ‚Reset_Threshold' the dynamical calculation of the threshold value (see ‚Threshold_Value') is started new.

Use_Initiation_Method

From V1.3: The activation of the function ‚Initiation method' is carried out via the additional parameter input ‚Use_Initiation_Method'.
-0 (Default): Without initiation method. Measured values are entered new into the send buffer.
-1: With initiation method. Before entering the measured value into the send buffer it is checked whether the value (with this IOA) already exists in the send buffer. In this case the existing entry is overwritten resp. refreshed

Background_Scan

-1: (Default): From V1.6 measured values initiated by timer (Send_Cyc_Interval) are transmitted with cause of transmission 2 (background scan) if this parameter is activated. With recognized communication fault the cyclic transmission is stopped.
-0: measured values initiated by timer are transmitted with cause of transmission 1 (cyclic), also during communication fault.

Inro_QOI

Interrogation process: programmable interrogation identification in accordance with IEC standard:
- 0: No response
- 20 (B#16#14 hex) -> General interrogation
- 21 (B#16#15 hex) to 36: Group interrogation + (always) general interrogation

In case the block has to react only to group interrogations and not to general interrogations, the group number and additional the Bit 2^6 has to be set.
Example.: Group interrogation 3 (B#16#43 hex)

The actual processing (check that there is actually information of the inquired interrogation group – corresponding reaction = pos/neg confirmation,…) is managed by the ORG block.
The interrogated data are transmitted from V1.1 with actual value.

Send_Cyc_Interval

Time interval for cyclical transmission
value 0: no cyclical transmission
value 1- x sec. time interval in seconds. Information is sent cyclically once within this time period.
The timer synchronizes itself to the absolute daily time.
See as well parameter ‘Background_Scan'.

Phase_Offset_Cyc_Interv

Offset of the time interval for the cyclic transmission compared with the synchronization to the absolute time. Positive as well as negative values are permitted. The unit is seconds.

No_Cyclic_Infos

With the parameter ‚No_Cyclic_Infos' (1…n) the number of sent measured values per cyclic initiation is limited.
0 (Default) = send all measured values

Threshold_Value

Threshold value for spontaneous transmission of the measured value.
0: dynamic calculation of the threshold value taking into account the parameterized percentage value in 'Threshold_Sensitivity'.

The dynamically calculated threshold value is the
max. occurring measured value multiplied by parameterized percentage value.
>0 (e.g. 20000.0): Calculation of the threshold value without taking into account 'Threshold_Sensitivity'

This value is taken directly as a threshold to be reached arithmetically.
Direct threshold exceedance is checked in every OB1-cycle.
Therefore the response to direct threshold exceedance is fast.

The threshold value processing is integrative – in an interval of 1 sec the measured value difference to the last transmitted value is added. If the threshold value is reached arithmetically the measurement value will be sent spontaneously.
The threshold can either be calculated automatically (default) or set as an absolute value.

 ©SIEMENS AG 2017

Threshold_Sensitivity

Percentage value for the dynamic calculation of the threshold value.
The dynamically calculated threshold value is the max. occurring measured value multiplied by parameterized percentage value.
Value range 0 to 100 % (preset: 5 %)
0: no threshold value processing, each change of the measurement values is transmitted spontaneously
otherwise: Percentage value for the dynamic calculation of the threshold value.
The spontaneous transmission in the adjustment cyclic transmission can be prevented by entering the value 32767.
Only relevant if no value is entered in 'Threshold_Value'.

P_THI

Output for interconnection with the optional AddOn block S7_IEC_Set_Threshold_1 (FC125)

FB_RetVal

Positive FB_RetVal values:

0000-0FFFh:	Transmit counter (0-4095 decimal)
1xxxh (Bit12)	Buffer jam
2xxxh (Bit13)	Interrogation active

Negative FB_RetVal values:

8101h:	Para ‚No_of_Infos' < 1 or > 128
8102h	P_SLi = 0 or wrong allocated
8104h	not supported ‚ValType'

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Src_Struct_Type) in detail

The structure of Src_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			DT	Ext_State	Data_Type	

		Image-spacing
Data_Type	Data_Type is used depending on the information type – in this case, as follows: 00 (0): 2 byte measured value (integer) 01 (1): 2 byte measured value (integer) + 1 byte status, IEC assignment 10 (2): 4 byte measured value (short real) 11 (3): 4 byte measured value (short real) + 1 byte status, IEC assignment	 2 byte 3 byte 4 bytes 5 bytes
Ext_State	With bit set the image is expanded by the status byte 'Ext_State'. As a priority the status byte is used for the optional transfer of time status bits which are not supported by the S7 date and time format.	+ 1 byte

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Res	Res	Res	IV	SY	DS	TI

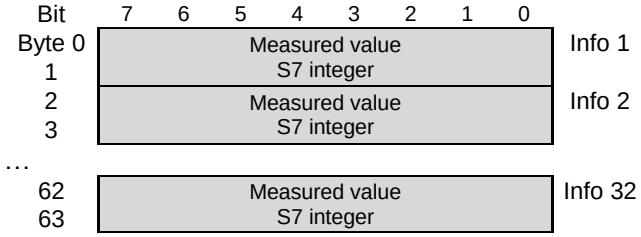
TI: The status bits (DS, SY, and IV) are only transferred if the TI - Bit is set.
This means that Ext_State can also be used as a filler byte (no other function), to establish image structures with even number of bytes.
DS: Daylight Saving Time - summer time:
The bit is transferred directly into SU in IEC messages
SY: synchronized:
The bit is not used in IEC messages
IV: Invalid:
The bit is transferred directly into IV in IEC messages

DT	Date and time in S7 format: Normally the CPU time is used. If DT is available, the time stamp submitted is assigned (time status, refer to ext_State).	+ 8 bytes
Add_Dist	Additional image spacing which is taken into account.	+ n bytes

Res Reserved for future expansions

Resultant typical image structures (Examples):

Structure type 0 (B#16#00):

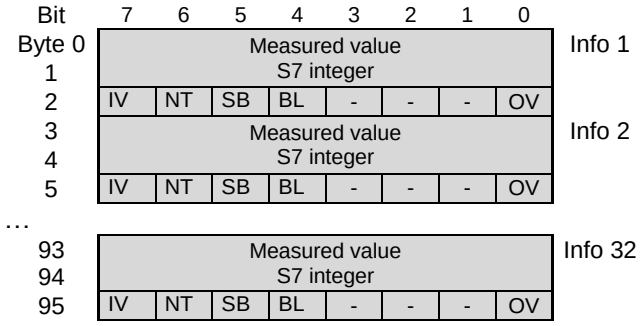


2 byte measured value (integer)

Resultant structure: 2 bytes per info

Each 2 bytes (1 word) represent a measured value in S7 integer format which you can either transfer as a measured value message in normalized or scaled format.

Structure type 1 (B#16#01):

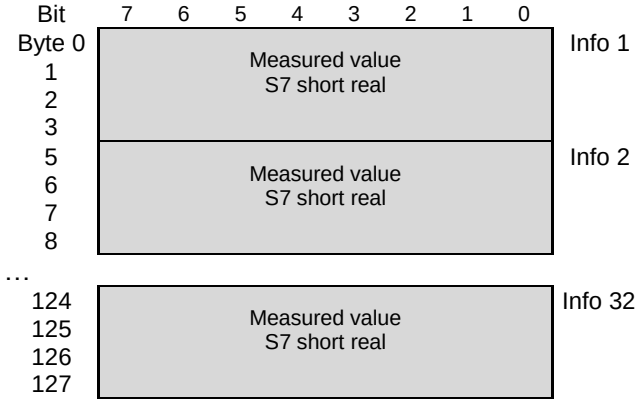


2 byte measured value (integer) + 1 byte status, IEC assignment

Resultant structure: 3 bytes per info

The first 2 bytes respectively (1 word) represent one measured value in S7 integer format, in each third byte the quality identifier is stored for this measured value in the IEC format.

Structure type 2 (B#16#02):



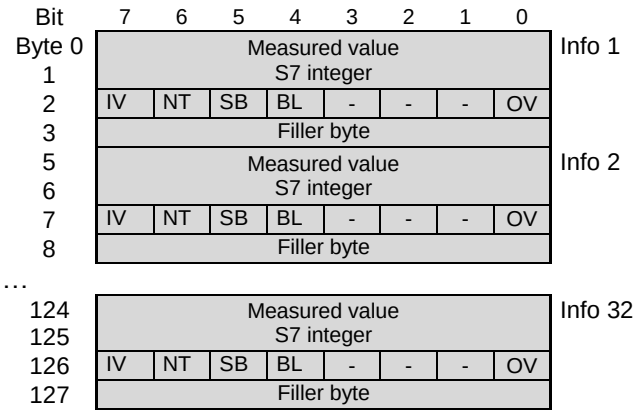
4 byte measured value (short real)

Resultant structure: 4 bytes per info

Each 4 bytes (1 DWord) represent a measured value in S7 short real format which you can transfer as a measured value message with short floating point number.

Structure type 17 (B#16#11):

2 byte measured value (integer)
+ 1 byte status, IEC assignment
+ 1 filler byte (Add_Dist=1)

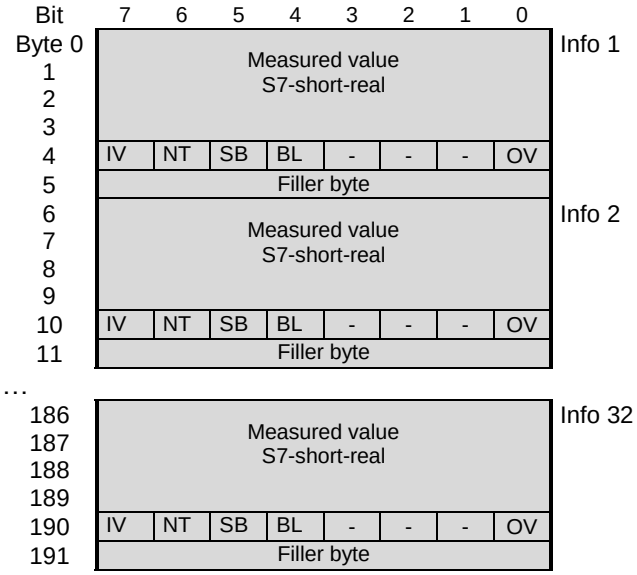


Resultant structure: 4 bytes per info

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

Structure type 19 (B#16#13):

4 byte measured value (short real)
+ 1 byte status, IEC-assignment
+ 1 filler byte



Resultant structur: 6 bytes per information

The first 4 bytes respectively (1DWord) represent a measured value in S7 short real format. In the fifth byte respectively the quality identifier is stored for this measured value in the IEC format.

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

Structure type 47 (B#16#2F):

Bit	7	6	5	4	3	2	1	0	
Byte 0	Measured value S7 short real								Info 1
1									
2									
3									
4	IV	NT	SB	BL	-	-	-	OV	
5	Res	Res	Res	Res	IV	SY	DS	TI	
6	S7 Date and Time								
7									
8									
9									
10									
11									
12									
13									
14	Filler byte								
15	Filler byte								
16	Measured value S7 short real								Info 2
17									
18									
19									
20	IV	NT	SB	BL	-	-	-	OV	
21	Res	Res	Res	Res	IV	SY	DS	TI	
22	S7 Date and Time								
23									
24									
25									
26									
27									
28									
29									
30	Filler byte								
31	Filler byte								
…									
496	Measured value S7 short real								Info 32
497									
498									
499									
500	IV	NT	SB	BL	-	-	-	OV	
501	Res	Res	Res	Res	IV	SY	DS	TI	
502	S7 Date and Time								
503									
504									
505									
506									
507									
508									
509									
510	Filler byte								
511	Filler byte								

4 byte measured value (short real)
+ 1 byte status, IEC assignment
+ 1 byte Ext_State
+ 8 byte time
+ 2 filler byte

Resultant structure: 16 bytes per info

Each first 4 bytes (1 DWord) represent a measured value in S7 short real format, in each fifth byte the quality identifier for this measured value is stored in IEC format.

Followed by
1 byte Ext_State and
8 byte S7 Date and Time

The two filler bytes in this example provide a data structure with 16 byte grid.

6.1.2.5. Integrated totals - SLi_IT_s8 (FB134)

The block only functions in combination with 'SL_ORG_ASDU'.

It accepts or interprets consecutive DWord content (starting with First_Source_Pos) and telecontrols the contained values as integrated totals.

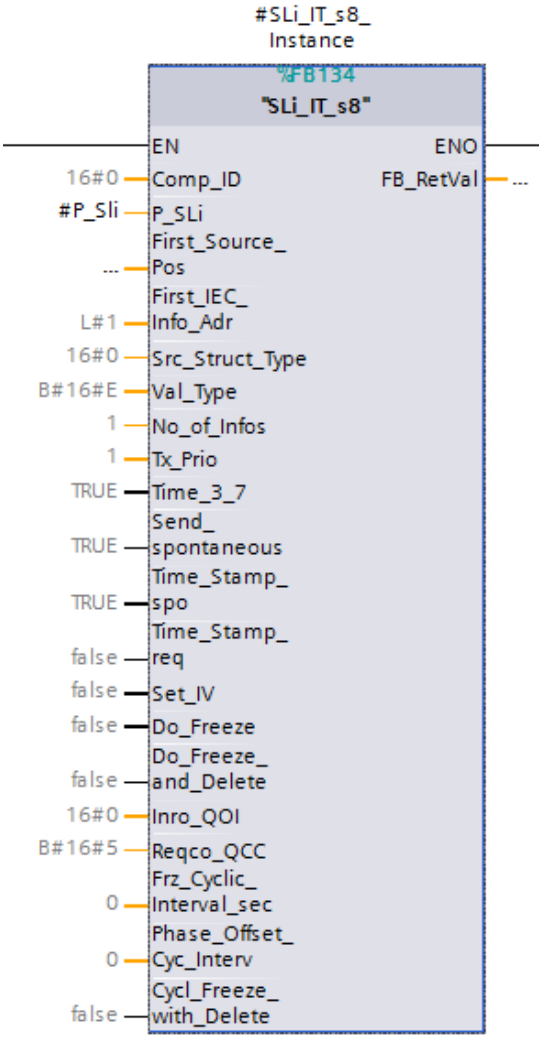


Illustration as FUP block with default values

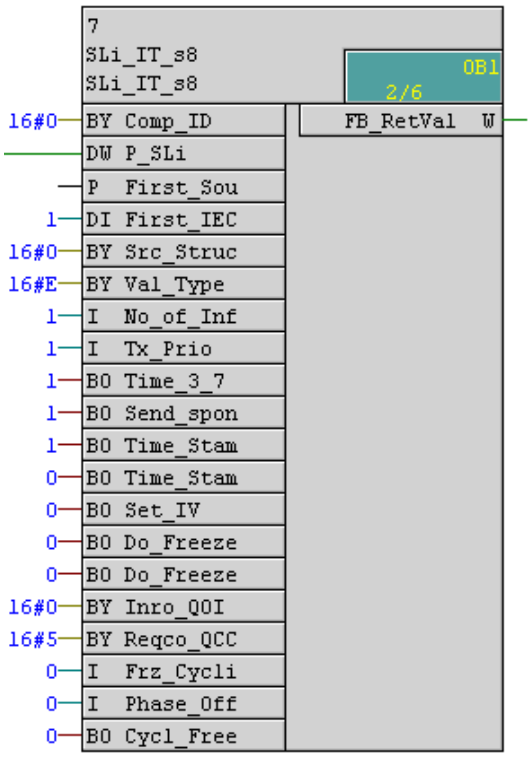


Illustration as CFC block with default values

Important features are:

- With minor effort of parameterization you can telecontrol up to 8 information (No_of_Infos):
  - as integrated totals with TI15, 16, 37 (depending on 'Val_Type' +'Time-Para').
- Default settings can be read in the respective variable comment; on the other hand they require a minimum of mandatory inputs.
  The minimum requirement is the interconnection of P_SLi and FirstSource_Pos.
- The information object addresses (IOA) are automatically calculated in ascending/in consecutive order starting with the programmable basic address ('First_IEC_Info_Adr').
- Any type of image sources starting with 'First_Source_Pos; can be addressed (E, M, DB...)
- Numerous data capture structures can be selected with 'Src_Struct_Type':
  - 4 byte integrated totals
  - IEC format (integrated totals + status byte)
  - optional time stamp and/or additional filler bytes
- Various restore versions

- In addition to spontaneous operation, all interrogation types (TI100, TI101) are supported.
- Selectable transmission priority (high/low)
- Time stamping can be activated separately for spontaneous, cyclical and interrogated transmission
- A global error input permits the fast identification of information with the status Invalid (IV) with the next restore
- Inputs for restore, restore with deletion and counter interrogation are available.
- Individual address setting via address parameter-DBs possible
- From Version V1.5 applicable also in time interrupts (see chapter 6.2)

Block parameter with their default assignment and brief comment

FB134		Type	Default	**Block comment/brief description**
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_SLi	IN	DWORD	DW#16#0	must be connected with P_to_SLi from SL_Org_ASDU - Block
First_Source_Pos	IN	POINTER	-	First Position of .... e.g. DB1.DBX0.0 - (without default)
First_IEC_Info_Adr	IN	DINT	L#1	First Information object address / Default = 1
Src_Struct_Type :	IN	BYTE	B#16#0	0: (Default) only counter 4 Byte, 1: IEC_Info 5 Byte, ... others -> TimeTag...
Val_Type :	IN	BYTE	B#16#F	15: (=B#16#F = Default) IT / others are not allowed
No_of_Infos	IN	INT	1	Allowed range: 1-8 / Default = 1
Tx_Prio	IN	INT	1	Priority of transmission: 0 (highest), 15 (lowest) / Default = 1
Time_3_7	IN	BOOL	TRUE	0: -> Time_3 -> TI2 / 1: -> Time_7 -> TI30 / Default = 1
Send_spontaneous	IN	BOOL	TRUE	If true (default), the ITs will be transmitted spontaneous after freezing
Time_Stamp_spo	IN	BOOL	TRUE	spontaneous Transmission without (0) or with (1) Timestamp / Default = 1
Time_Stamp_req :	IN	BOOL	FALSE	requested Transmission without (0) or with (1) Timestamp / Default = 0
Set_IV :	IN	BOOL	FALSE	Reset[0], Set[1] all Infos to IV with next freeze
Do_Freeze	IN	BOOL	FALSE	Rising edge -> freeze (all values)
Do_Freeze_and_Delete	IN	BOOL	FALSE	Rising edge -> freeze and delete (all values)
Inro_QOI	IN	BYTE	B#16#0	0: No reaction (default); 20 (14hex): General interrogation; 21-36 -> Group 1-16
Reqco_QCC	IN	BYTE	B#16#5	0: No reaction; 1-4: specified Counter Group; 5(default):general request
Frz_Cyclic_Interval_sec	IN	INT	0	0: without; 1-x [sec] Time interval for cyc. freeze, works sync to absolute time
Phase_Offset_Cyc_Interv	IN	INT	0	0: without(def), +/-x [sec]: offset for cyc. Transm. related to absolute time
Cycl_Freeze_with_Delete	IN	BOOL	FALSE	Default = false
FB_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: counter of transmitted infos + state

Parameter and function details:

Comp_ID	Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as a byte variable. e.g. B#16#1
P_SLi	P_SLi is a pointer to a common data range from block SL_ORG_ASDU. Therefore it has to be interconnected with its output P_SLi via a DWord variable. The ORG block transfers values such as 'ASDU address', current time stamp, send buffer, and processes e.g. GIs, via this pointer or the data range behind it.
First_Source_Pos	Bit pointer to the first information to be collected. e.g. P#E0.0 or P#DB1.DBX0.0 With non bit-oriented image structures, byte limits have to be observed (p#Ex.**0**); otherwise there may be 'access errors'.
First_IEC_Info_Adr	Address value of the first collected information object Permissible range with IOA address length 1 is 1 to 255 Permissible range with IOA address length 2 is 1 to 65535. Permissible range with IOA address length 3 is 1 to 16777215 If the information object address is available in structured form (3 octets), it must be converted into a decimal number: IOA dec = octet1 + 256*octet2 + 256*256*octet3 Further information objects are addressed consecutively/in ascending order. Individual address allocation via address parameter-DBs is possible. Detailed description see chapter 6.1.4.
Src_Struct_Type	0 → 4 byte count value 1 → 4 byte count value + 1 byte status, IEC assignment 2 → reserved 3 → reserved can be combined with time stamp capture → For more details, refer to detailed description.
Val_Type	Type identification 14. integrated totals (preset) (No others permitted)
No_of_Infos	Number of information to be collected Value range 1 to 8 Values outside this range result in an output of fault at the FB_RetVal and no further processing is carried out.
Tx_Prio	Used for prioritization of the information transfer. from V1.3: 15: lowest priority 0: highest priority. In case of specifying a priority which is less than the available send buffer, the information is entered automatically into the send buffer with the lowest available priority.
Time_3_7	The time stamp format can be set on input 'Time_3_7': FALSE → 3-byte length = short or partial time stamp CP24Time2a TRUE → 7-byte length = long or full time stamp CP56Time2a
Send spontaneous	Spontaneous transmission of integrated totals: FALSE -> No spontaneous transmission of the integrated totals after the restore TRUE -> After restore the integrated totals are transferred spontaneously.
Time_Stamp_... **Time_Stamp_spo** **Time_Stamp_req**	The transmission of the messages with or without time stamp can be parameterized separately for: spontaneous transmission with 'Time_Stamp_spo' interrogated transmission with 'Time_Stamp_req' FALSE: no time stamp TRUE: with time stamp
Set_IV	The block input 'Set_IV' applies to information processed by this block. Its status is transferred directly into the status bit IV of the integrated total with the next restore.
Do_Freeze	The block input 'Do_Freeze' applies to information processed by this block. With a rising edge all interrogated totals are restored from the data capture image into the integrated totals image (frozen) and transferred spontaneously depending on the parameter 'Send_spontaneous'.
Do_Freeze_and _Delete	The block input 'Do_Freeze_and_Delete' applies to information processed by this block.

	With a rising edge all integrated totals are restored from the data capture image into the integrated total image (frozen) and the data capture image is then deleted (set to 0). With the next restore the status bit CA (Counter adjusted) is shown once by the block.
Inro_QOI	Interrogation process: programmable interrogation identification in accordance with IEC standard: - 0: No response - 20 (B#16#14 hex) -> General interrogation - 21 (B#16#15 hex) to 36: Group interrogation + (always) general interrogation In case the block has to react only to group interrogations and not to general interrogations, the group number and additional the Bit 2^6 has to be set. Example.: Group interrogation 3 (B#16#43 hex) The actual processing (check that there are actually information of the inquired interrogation group – corresponding reaction = pos/neg confirmation,…) is managed by the ORG block. The interrogated data always come from the 'image'
Reqco_QCC	Programmable identifier in accordance with the IEC standard to assign the captured integrated totals to a counter group which applies to counter interrogation, restore and delete commands (by TI101). - 0: No group/response - 1 (B#16#01 hex) to 4: counter interrogation group 1-4 + (always) global counter interrogation - 5 (B#16#05 hex) -> Global counter interrogation The actual processing (check that there are actually information of the inquired interrogation group – corresponding reaction = pos/neg confirmation…) is managed by the ORG block. The interrogated data always come from the 'image'
Frz_Cyclic_Interval_sec	Time interval for cyclical restore of the integrated totals Value 0: No cyclical restore Value 1- x sec. time interval in seconds. Information is restored cyclically once within this time period. The timer synchronizes itself to the absolute daily time.
Phase_Offset_Cyc_Interv	Offset of the time interval for the cyclic transmission compared with the synchronization to the absolute time. Positive as well as negative values are permitted. The unit is seconds.
Cycl_Freeze_with_delete	0/False: the count memories are not deleted after restoring 1/True: the count memories are deleted after restoring
FB_RetVal	Positive FB_RetVal values: 0000-0FFFh: Transmit counter (0-4095 decimal) 1xxxh (Bit12) Buffer jam 2xxxh (Bit13) Interrogation active Negative FB_RetVal values: 8101h: Para ‚No_of_Infos' < 1 or > 128 8102h P_SLi = 0 or wrong allocated 8104h not supported ‚ValType'

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Src_Struct_Type) in detail

The structure of Src_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			DT	Ext_State	Data_Type	

		Image spacing
Data_Type	Data_Type is used depending on the information type – in this case, as follows: 00 (0): 4 byte integrated total (DINT) 01 (1): 4 byte integrated total (DINT) + 1 byte status, IEC assignment 10 (2): reserved 11 (3): reserved	 4 byte 5 byte
Ext_State	With bit set the image is expanded by the status byte 'Ext_State'. As a priority the status byte is used for the optional transfer of time status bits which are not supported by the S7 date and time format. TI: The status bits (DS, SY, and IV) are only transferred if the TI - Bit is set. This means that Ext_State can also be used as a filler byte (no other function), to establish image structures with even number of bytes. DS: Daylight Saving Time - summer time: The bit is transferred directly into SU in IEC messages SY: synchronized: The bit is not used in IEC messages IV: Invalid: The bit is transferred directly into IV in IEC messages	+ 1 byte
DT	Date and time in S7 format: Normally the CPU time is used. If DT is available, the time stamp submitted is assigned (time status, refer to ext_State).	+ 8 byte
Add_Dist	Additional image spacing which is taken into account.	+ n byte
Res	Reserved for future expansions	

Allocation of the status byte according to IEC

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IV	CA	CY	SQ				

SQ Sequence number
With each restoring (from block) the sequence number is increased by 1. With it a secure change of the combination integrated totals/status is achieved, also if the integrated total itself remains unchanged.
A sequence number changed by the user is taken over with the next restoring.

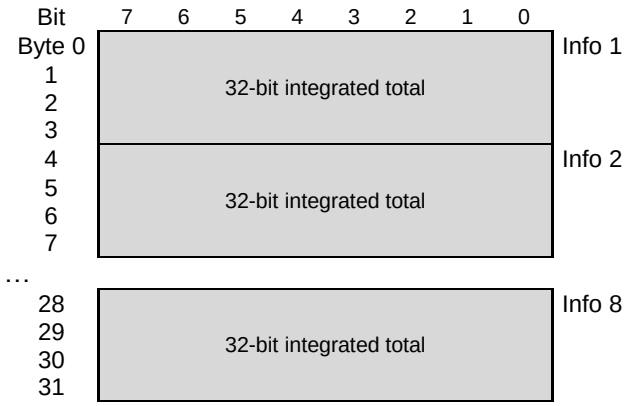
CY Carry
Marks an overflow of the integrated total.
The bit is reset automatically with each restoring/erasing. If required it thus has to be set from the user only once (Set command).

CA Counter adjusted
Marks the integrated total as pre-adjusted value. This is e.g. always the case after a new start because the initialization value (=0) has been pre-adjusted.
It is also the case if the restoring has been triggered with erasing.

IV Invalid
The integrated total is marked as invalid.

Resultant typical image structures (examples):

Structure type 0 (B#16#00):

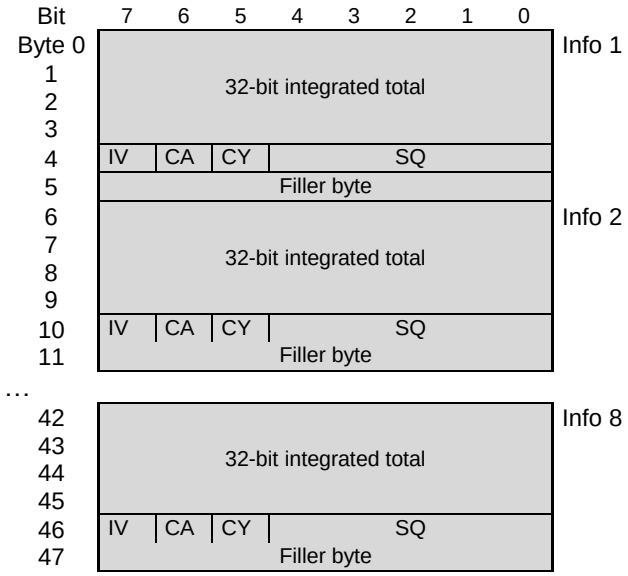


4 byte integrated total

Resultant structure: 4 bytes per information

Each 4 bytes represent a 32 bit integrated total which you can transfer as an integrated total message.

Structure type 17 (B#16#11):



4 byte integrated total
+ 1 byte status, IEC assignment
+ 1 filler byte

Resultant structure: 6 bytes per information

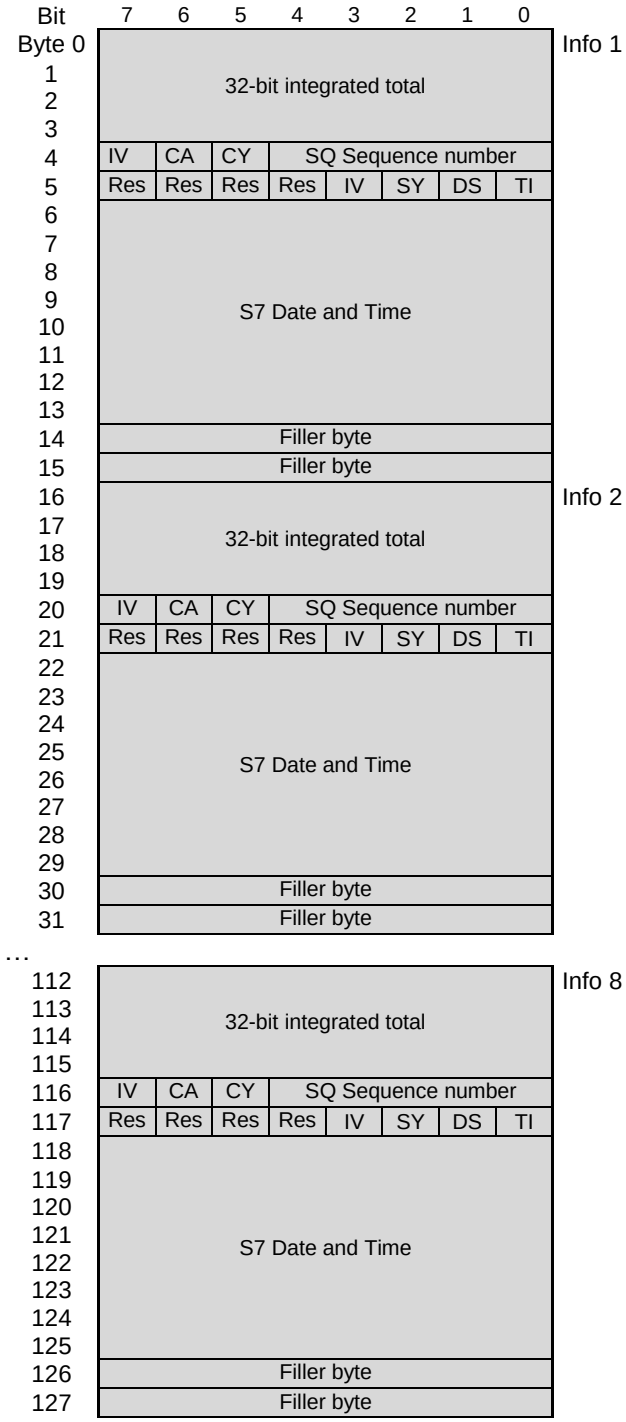
Each first 4 bytes represent a 32 bit integrated total,
in each fifth byte the quality identifier for this integrated total is stored in IEC format.

It follows one filler byte each.

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be arbitrary used by other parts of the program or are simply created just for structuring reasons.

Structure type 57 (B#16#39): 4 byte integrated total

+ 1 byte status, IEC assignment
+ 1 byte Ext_State
+ 8 byte time
+ 2 filler byte



Resultant structure: 16 bytes per information

Every first 4 bytes represent a 32 bit integrated total,
in each fifth byte the quality identifier for this integrated total is stored in IEC format.

Followed by
1 byte Ext_State
8 byte time and
2 filler bytes

The status bits from Ext_State (DS, SY, IV) are only transferred if the TI - Bit is set.
Even if these status bits are not necessarily being used, the use of the byte Ext_State as an additional filler byte is advisable in this case.
This means that the entry Date and Time start at even-order byte numbers in each case.
This is advantageous for the further processing of Date and Time.

The filler bytes remain ignored with regard to image data acquisition and are simply skipped. They may be used as required by other parts of the program or are simply created just for structuring reasons.

The filler bytes in this example provide a data structure with 16 byte grid format.

6.1.3. Blocks for the process output SLo

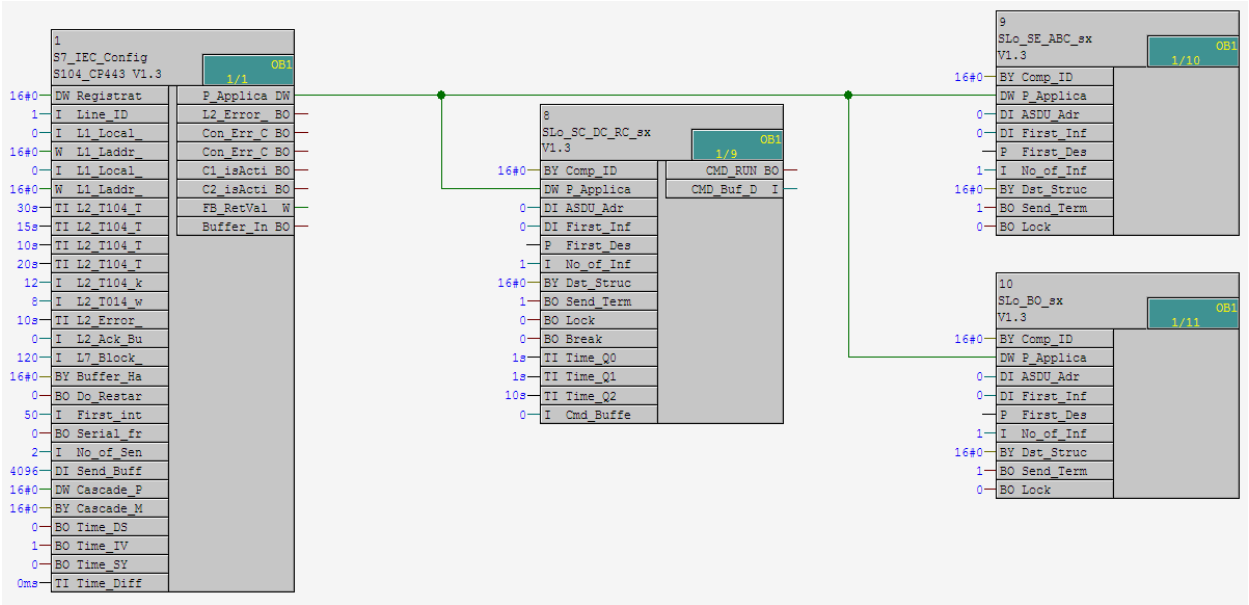
The symbol name encrypts the most important block properties as follows:

SLo_SC_DC_RC_sx	**SLo** is the abbreviation for slave-blocks output, also known as output blocks
SLo_**SC_DC_RC**_sx	Information types supported by the block with the brief designation **SC** = Single Command according to IEC standard:
SLo_SC_DC_RC_**s**x	**S** is the abbreviation for sequential processing, also known as continuous output in the target area and IEC address starting with a base position, start address in each case. **P** would be the abbreviation for programmable allocation of data acquisition point and IEC address.
SLo_SC_DC_ RC_s**x**	x is the (maximum) number of administrable information points of the block. x is used for the fact that the (maximum) number of administrable information points is not limited by the block. Theoretically the block can manage any number of commands. In practice the number is limited by the available information addresses or maximum DB size (depending on the CPU used). The number of information points can be increased by multiple call-ups (instances).

The available SLo functional blocks can be found in the overview below.

Block	Block No.	Function
SLo_SC_DC_RC_sx	FB135	For processing single (SC), double (DC) or regulating step commands RC
SLo_SE_ABC_sx	FB136	For processing analogue setpoint commands, normalized (SE_A), scaled (SE_B) or floating point (SE_C)
SLo_BO_sx	FB137	For processing bit pattern commands (BO) (digital setpoint commands)

Illustration of the necessary interconnection - here in a CFC plan as an example:



6.1.3.1. Single, double and regulating step commands - SLo_SC_DC_RC_sx (FB135)

The block receives single, double or regulating step commands from the telecontrol partner and writes them in the simplest case to a bit field (starting with First_Source_Pos).

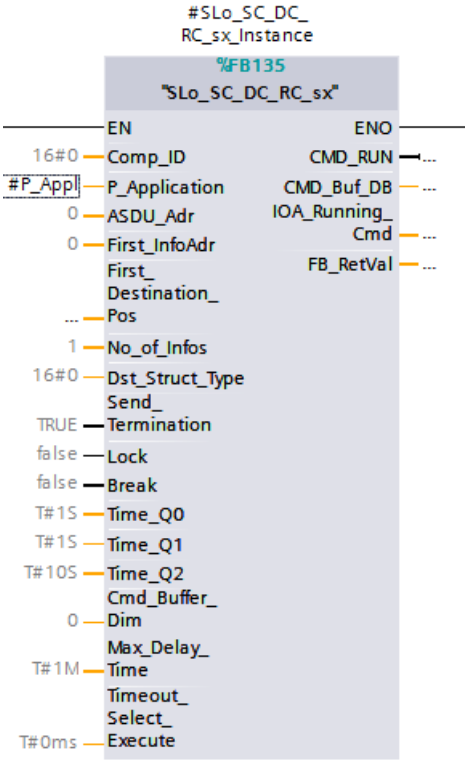


Illustration as FUP block with default values

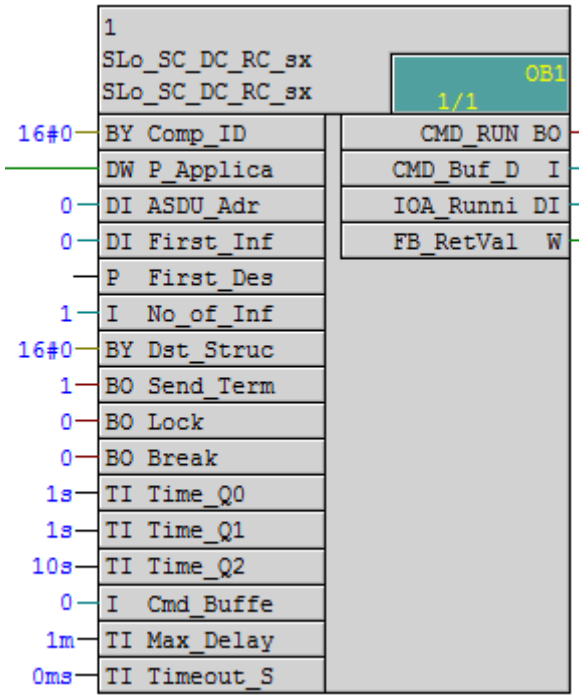


Illustration as CFC block with default values

Important features are:

- With a minor amount of parameterization you can process many commands (No_of_Infos):
  - Single commands with TI45, 58
  - Double commands with TI46, 59
  - Regulating step commands with TI47, 60
- Default settings can be read in the relevant variable comment; on the other hand they require only a minimum of mandatory inputs.
  The minimum requirement is the interconnection of P_Application and First_Destination_Pos.
- The information object addresses (IOA) and output position are automatically calculated in ascending/consecutive order starting with the programmable base address ('First_InfoAdr').
- Any output targets starting with 'First_Destination_Pos' (A, M, DB...) can be addressed automatically.
- Numerous output structures can be selected with 'Dst_Struct_Type':
  - One-pole or two-pole bit field
  - Byte formats (information + status bits)
  - Additional filler bytes
- A global blocking input permits the simple locking of all commands.
- Different output times for commands with Q0, Q1 or Q2 can be parameterized separately
- Output for external evaluation of an active impulse command present. From V1.5 additionally a Return_Value and feedback output of the information object address (IOA) of the active command available.
- Global and selective command release possible (Input ‚Break' or Br bit with byte output)
- Selective blocking of individual commands possible (LK bit with byte output)

- Function (output) independent of received TI
- Individual address setting via address parameter-DBs possible
- Command storage and serial output of the buffered commands possible (number according to ,Cmd_Buffer_Dim)
- Entire support of command select and command execution (select before operate)
- Entire support of aging monitoring of commands with time stamp

Block parameters with their default assignment and brief comment

FB135		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	DW#16#0	Pointer which must be connected to 'S7_IEC_Config' – Module
ASDU_Adr	IN	DINT	L#0	ASDU-Address which will be expected by this module (Default: 0)
First_InfoAdr	IN	DINT	L#0	First Information object address - Default = 0
First_Destination_Pos	IN	POINTER	-	First Position of .... e.g. A0.0 or DB900.DBX5.0
No_of_Infos	IN	INT	1	No of (sequential ascending) IOA which will supported (Default = 1)
Dest_Struct_Type	IN	BYTE	B#16#0	0:(Def.) 1 Bit (SPI);1:2 Bit (DPI);2:1 Byte (Stat+SPI);3:1 Byte (Stat+DPI)
Send_Termination	IN	BOOL	TRUE	0: without; 1 (default): with Termination of activation
Lock	IN	BOOL	FALSE	If set, new cmd will not executed, they will locked and neg. con
Break	IN	BOOL	FALSE	With rising edge a running (pulse) command will be terminated/cancelled.
Time Q0:	IN	TIME	T#1s	Pulse Time for incoming commands with Q0: Default: T#1S
Time Q1:	IN	TIME	T#1s	Pulse Time for incoming commands with Q1: Default: T#1S
Time Q2:	IN	TIME	T#10s	Pulse Time for incoming commands with Q2: Default: T#10S
Cmd_Buffer_Dim	IN	INT	0	0: without (default); >0 max. number of buffered commands
Max_Delay_Time	IN	Time	T#1M	0: without; >0 max. delay time for command with time tag; Default: t#60s
Timeout_Select_Execute	IN	Time	T#0MS	0: without sel/exe (def); >0 max. time between select and execute command.
CMD_Buf_DB	INOUT	INT	0	Shows the data block number of command buffer (which is created at runtime)
CMD_RUN	OUT	BOOL	FALSE	Signals a running (pulse-) command
IOA_Running_Cmd	OUT	DInt		IOA of an active command, otherwise 0; pulse cmd: if active; per.cmd: one cycle
FB_RetVal	OUT	Word		Neg. values: Error-Codes; o: ok

 ©SIEMENS AG 2017

Parameter and function details:

Comp_ID	Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as byte variable. e.g. B#16#1
P_Application	P_Application is a pointer to a common data range of block S7_IEC_Config. Therefore it has to be interconnected to its output P_Application via a DWord variable. The Config block receives user data such as 'ASDU address', receiver buffer, etc. via this pointer or the data range behind it.
ASDU_Adr	Address value of the application service data unit receiving commands of this block. Permissible range with ASDU address length 1 is 1 to 254 Permissible range with ASDU address length 2 is 1 to 65534. The broadcast addresses 255 (FFh with length 1) resp. 65535 (FFFFh with length 2) are always accepted.
First_InfoAdr	Address value of the first information object edited Permissible range with IOA address length 1 is 1 to 255 Permissible range with IOA address length 2 is 1 to 65535. Permissible range with IOA address length 3 is 1 to 16777215 If the information object address exists in structured form (3 octets), it must be converted into a decimal number: IOA dec = octet1 + 256*octet2 + 256*256*octet3 Further information objects are addressed consecutively/in ascending order. Individual address allocation via address parameter-DBs is possible. Detailed description see chapter 6.1.4.
First_Destination_Pos	Bit pointer to the first target position which is receiving the information. e.g. P#A0.0 or P#DB1.DBX0.0 With non bit-oriented output structures, the byte limits must be observed (p#Ax.**0**); otherwise 'access errors' may occur.
No_of_Infos	Number of information to be output Value range 1 to 255.
Dst_Struct_Type	Bit commands can be output regardless of the received TI as 1-bit or 2-bit commands: 0 -> one-pole bit field 1 -> two-pole bit field 2 -> byte field with IEC assignment (one-pole output) 3 -> byte field with IEC assignment (two-pole output) -> For more details, refer to the description.
Send_Termination	Selection whether the command has to be completed with or without sending a 'Termination' of activation. FALSE: no 'Termination of activation' TRUE: with 'Termination of activation' (default setting)
Lock	Blocking input for all commands processed by this block. FALSE: New commands are executed TRUE: New commands are not executed and negatively acknowledged.
Break	With rising edge a current impulse command will be finished or cancelled. It will be positive terminated to the control center.
Time_Q0	Output time for received commands with qualifier Q0.
Time_Q1	Output time for received commands with qualifier Q1.
Time_Q2	Output time for received commands with qualifier Q2.
Cmd_Buffer_Dim	The parameter defines the maximum number of buffered commands 0 without command buffer (like V1.1) >0 maximum number of buffered commands In this case a command buffer DB will be created during the runtime An entry into the command buffer DB happens only if during arrival from an impulse command another impulse command already runs. **Using this FB with S7-1500, ET 200SP and Software Controller please pay attention to the Notice after this parameter description.**

Max_Delay_Time	0 (t#0ms) deactivates the aging monitoring of commands with time stamp. > 0: Aging monitoring for commands with time stamp active. The command must not be older than the parameterized period of time (Default t#60s). Otherwise the command will be neglected. Further (fixed) basic conditions are: The time in the system and command must be valid. The time in the command may be max. 1 sec (1 min from V1.6) in the future. The parameter is available from V1.5.
Timeout_Select_Execute	0 (t#0ms = default): Both select and execute commands will be always accepted Select before execute is activated. Executing commands are only accepted, if they are received within the parameterized period of time after receive of the select command. The parameter is available from V1.5.
CMD_RUN	Command output (impulse command) running The output is set for the duration of the command output, if an impulse command is executed via this block.
CMD_BUF_DB	Indicates the number of the command buffer DB generated at startup. The size of the command buffer DB will be determined with ‚Cmd_Buffer_Dim'. **Using this FB with S7-1500, ET 200SP and Software Controller please pay attention to the Notice after this parameter description.**
IOA_Running_Cmd	Output of the information address (IOA) of a currently running command. In case of persistent command output for one cycle only. The output is available from V1.5
FB_RetVal	Signals the reason why a command could NOT be executed. You will find a detailed list in chapter 6.1.3.4

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

NOTICE

Particularities by usage of the command buffer with S7-1500, ET 200SP and Software Controller:

An activated command buffer (Cmd_Buffer_Dim > 0) causes creation of a data block during runtime. In this DB the commands will be buffered.

To avoid a re-creation of another data block at every CPU-restart there are two different ways:

1. Place the instance data block of the command block in the retentivity memory. Therefore open the instance DB and activate the option ‘Retentivity’.
2. Alternative (uses less retentive memory):
   As from V1.6 Upd4 the INOUT variable ‘CMD_Buf_DB’ which contains the number of the created DB can be assigned to a variable (e.g. marker). This variable should be retentive so that its content is kept through a restart of the CPU.

During the generation of the DB (resp. deletion of an existing one) the block remains in the init routine indicating 7FFF at output variable ‘RetVal’

The structure types (Dst_Struct_Type) in detail

The structure of Dst_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			Res		Data_Type	

		Image spacing
Data_Type	Data_Type used depending on information type – in this case, as follows: 00 (0): one-pole bit field -> single bit output (SCS) 01 (1): two-pole bit field -> double bit output (DCS) 10 (2): byte field with status bits (Stat + SCS) 11 (3): byte field with status bits (Stat + DCS)	 1 bit 2 bit 1 byte 1 byte
Add_Dist	Additional image distance which is considered in case of sequential writing	+ n Byte
Res	Reserved for future expansions	

Resultant typical image structures:

Structure type 0 (B#16#00):

One-bit output

Bit	7	6	5	4	3	2	1	0
Byte 0	7	6	5	4	3	2	1	0
1	15	14	13	12	11	10	9	8
2						...	17	16
...								

Resultant structure: 8 commands per byte

Each bit represents one command. The bit assigned to the relevant impulse command is set for the duration of the command output.
Continuous commands (Q3) with switching direction ON set the bit.
Continuous commands (Q3) with switching direction OFF delete the bit.

Structure type 1 (B#16#01):

Two-bit output

Bit	7 6	5 4	3 2	1 0
Byte 0	DCS 3 E/H | A/T	DCS 2 E/H | A/T	DCS 1 E/H | A/T	DCS 0 E/H | A/T
1	DCS 7 E/H | A/T	DCS 6 E/H | A/T	DCS 5 E/H | A/T	DCS 4 E/H | A/T
2		...	9	DCS 8 E/H | A/T
...				
	DCS n E/H | A/T			

Resultant structure: 4 commands per byte

Two neighbouring bits belong to one command (ON/OFF; LOWER/HIGHER)

The cmd. bits (DCS) are set as follows:

No cmd. output active->	00
Cmd. with switching direction OFF (A)->	01
Cmd with switching direction ON (E)->	10
Regulating step cmd. LOWER (T)->	01
Regulating step cmd. HIGHER (H)->	10

Structure type 2 (B#16#02):

One-bit output + status bits

Bit	7	6	5	4	3	2	1	0	
Byte 0	SE	New Cmd	Lk	Br	-	-	-	SCS	Info 0
1	SE	New Cmd	Lk	Br	-	-	-	SCS	Info 1
2	SE	New Cmd	Lk	Br	-	-	-	SCS	Info 2
...								...	...
	SE	New Cmd	Lk	Br	-	-		SCS	Info n

Resultant structure: 1 byte per command

Each bit represents one command (SCS) including associated status bits. The bit assigned to the relevant pulse command is set for the duration of the command output.
Duration commands (Q3) with switching direction ON set the bit.
Duration commands (Q3) with switching direction OFF delete the bit.

 ©SIEMENS AG 2017

Structure type 3 (B#16#03):

Bit	7	6	5	4	3	2	1	0	
Byte 0	SE	New Cmd	Lk	Br	-	-	DCS E/H	DCS A/T	Info 0
1	SE	New Cmd	Lk	Br	-	-	DCS E/H	DCS A/T	Info 1
2	SE	New Cmd	Lk	Br	-	-	DCS E/H	DCS A/T	Info 2
...								...	...
	SE	New Cmd	Lk	Br	-	-	DCS E/H	DCS A/T	Info n

Two-bit output + status bits

Resultant structure: 1 byte per command

Each byte represents a command including the associated status bits.

The command bits (DCS) are set as follows:

No cmd. output active->	00
Cmd. with switching direction OFF (A)->	01
Cmd. with switching direction ON (E)->	10
Regulating step cmd. LOWER (T)->	01
Regulating step cmd. HIGHER (H)->	10

Structure type 50 (B#16#32):

Bit	7	6	5	4	3	2	1	0	
Byte 0	SE	New Cmd	Lk	Br	-	-	-	SCS	Info 0
1	Filler byte								
2	Filler byte								
3	Filler byte								
4	SE	New Cmd	Lk	Br	-	-	-	SCS	Info 1
5	Filler byte								
6	Filler byte								
7	Filler byte								
8	SE	New Cmd	Lk	Br	-	-	-	SCS	Info 2
9	Filler byte								
10	Filler byte								
11	Filler byte								
...								...	...

One-Bit-Output + Status bits
+ 3 Filler byte (Add_Dist=3)

Resulting Structure: 4 Byte per command

The filler bytes remain regarding command output unconsidered and will be leaped. They may be arbitrary used from other parts of the program or are created only for structuring purposes.

Byte assignment when using Dst_Struct_Type = 2 or 3 in detail

SE
Select/Execute
Select bit from IEC message

NewCmd
New Command,
is set with each output of a command.
Can be reset by the user.

Lk
Locked
Locking bit: If this bit is set before output of a new command, the command is blocked.
Instead of that it is 'negatively' acknowledged to the control centre.

Br
Break – terminate
A running '(pulse) command' can be terminated/cancelled using this command. It is terminated positively to the control centre.
The bit is reset with the output of a new command.

SCS
One bit output
Set with running pulse command (Q0, Q1, Q2) with switching direction ON.
Duration command (Q3) with switching direction ON sets SCS
Duration command (Q3) with switching direction OFF resets SCS

DCS
Two-bit output
Bit 0 is set with running OFF or lower command.
Bit 1 is set with running ON or higher command.

Additional notes for processing commands

- Continuous commands are carried out at every time
  (also during a running switching command)
- If necessary the IEC originator is mirrored and sent in the response messages as well
- Test bits and set P/N bit in the cause of transmission (COT) result in 'non-handling' – the message is not processed.
- Command cancellation via cancel command is supported:
  If a switching command with the InfoAddr of the cancel command is running, this is terminated and the cancel command positively acknowledged.
- In principle a command from SLo-block is only processed (and with it acknowledged) if these are commands which are directed to the block
  -> In V1.0 version, no response to commands with unknown ASDU address, information address,...
- From V1.1 negative command confirmations are sent ‚substitutional' from SL_ORG_ASDU if no command block (SLo_...) has processed (confirmed) the message within the next OB1-cycle after receiving a command message (with COT=6 or COT8).
  The confirmation is carried out with:
  - COT 44: unknown TI, if there is no output block programmed for the received TI.
  - COT 45: unknown COT, if the received COT is not supported from the responsible SLo-block (it is the one with the correct IOA)
  - COT 47: unknown IOA, if there is no output block programmed for the received IOA.

  For further notes please refer to chapter 6.1.1. As well the modified behavior from V1.5 is described there.
- Select/execute is entered in the status bit SE when using Dst_Struct_Type 2 and 3, but not evaluated for the command output.
- The command is output regardless of S/E
  From V1.5 the Select/Execute handling is carried out according to standard. See as well block parameter “Timeout_Select_Execute”
- Command messages with time stamp (TI58, 59, 60) are accepted and output.
  Up to Version V1.4 the time stamp is not evaluated.
  From Version V1.5 see block parameter “Max_Delay_Time”.
- The switching commands are subject to a 1out of n control.
  The 1 out of n control is always active if one switching command (impulse command) is executed and another one has to be processed.

From V1.2 impulse commands can be cached in a command buffer DB in case an impulse command is already active.

With it fast ‘non supervised’ command sequences are supported.

Received commands pointed to the block (ASDU-Addr, IOA-Addr…), are processed with the following rules:

- The buffered commands are executed serial, i.e. the next command is executed after the preceding command has been executed.
  The commands are processed in the sequence of its arrival.
- Continuous commands are executed always immediately (like before, finished in one cycle)
- Cancellation commands are executed if the command which has to be stopped is just active. Otherwise a negative acknowledgement is carried out.
- Cancellation commands have no effect to commands in the buffer.

- **Without cache** (‚Cmd_Buffer_Dim' = 0, or Buffer-DB could not be created) is the behavior like in V1.1. Further switching commands arriving during an active switching command (impulse command) are refused (negative confirmed).
- **With cache** switching commands are immediately executed, if no command is active.
- **With cache** switching commands are buffered if

 ©SIEMENS AG 2017

- a (switching) command is active/executed
- already minimum one command is in the buffer

➢ **With cache** switching commands are refused, if the buffer is full (only relevant with MO and IEC 103-variant). FB165 and FB173 are node blocks. Persistant commands are treated as pulse commands, that means transmitted and controlled.Therefore there is also necessary the buffering of this commands.

➢ (only relevant with IEC 103-variant) In principle, FB173 does not support deactivate commands – they are therefore always rejected and acknowled as negative.

 ©SIEMENS AG 2017

6.1.3.2. Setpoint commands - SLo_SE_ABC_sx (FB136)

The block receives setpoint commands in normalized, scaled or floating point format from the telecontrol partner. In the simplest case it writes them as integer or short-real values to a data field (starting with First_Source_Pos).

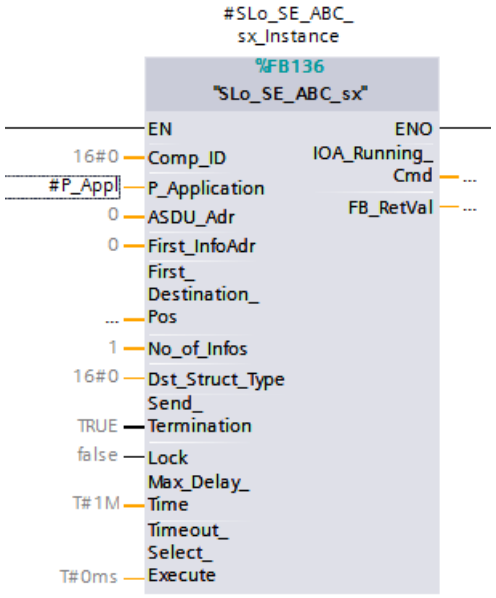


Illustration as FUP block with default values

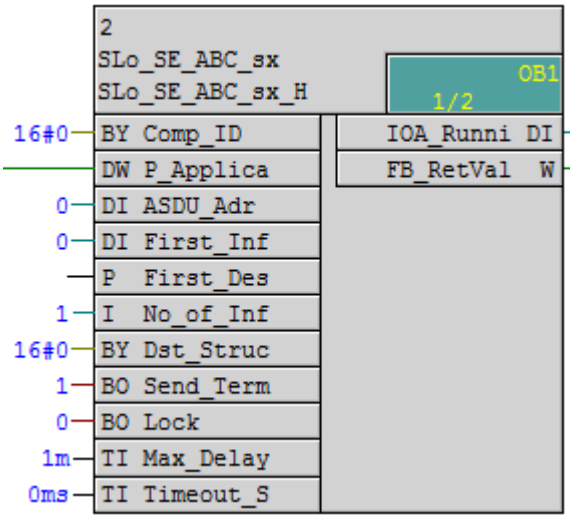


Illustration as CFC block with default values

Important features are:

- With a minor amount of parameterization, you can process many setpoint commands (No_of_Infos):
  - Setpoint commands with normalized value with TI48, 61
  - Setpoint commands with scaled value with TI49, 62
  - Setpoint commands with short floating point number with TI50, 63
- Default settings can be read in the relevant variable comment, on the other hand they require only a minimum of mandatory inputs.
  The minimum requirement is the interconnection of 'P_Application' and 'First_Destination_Pos'.
- The information object addresses (IOA) and output position are automatically calculated in ascending/consecutive order starting with the programmable base address ('First_InfoAdr').
- Any output targets starting with 'First_Destination_Pos' (A,M,DB,...) can be addressed automatically.
- Numerous output structures can be selected with 'Dst_Struct_Type':
  - 2 byte integer
  - 4 byte short real (information + status bits)
  - IEC format (+ status bits)
  - additional filler bytes
- A global blocking input permits the simple locking of all setpoint commands.
- Selective blocking of individual setpoint commands possible (Lk bit in the command byte)
- Individual address setting via address parameter-DBs possible
- Entire support of command select and command execution (select before operate)
- Entire support of aging monitoring of commands with time stamp
- Return_Value and feedback output of the information object address (IOA) of the executed command

Block parameters with their default assignment and brief comment

FB136		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	DW#16#0	Pointer which must be connected to 'S7_IEC_Config' – Module
ASDU_Adr	IN	DINT	L#0	ASDU-Address which will be expected by this module (Default: 0)
First_InfoAdr	IN	DINT	L#0	First Information object address (Default = 0)
First_Destination_Pos	IN	POINTER	-	First Position of .... e.g. A0.0 or DB900.DBX5.0
No_of_Infos	IN	INT	1	No of (sequential ascending) IOA which will supported (Default = 1)
Dst_Struct_Type	IN	BYTE	B#16#0	0: (Def) Int (2 Byte);1:3 Byte (Int+Stat);2:4 Byte (Real);3:5 Byte (Real+Stat)
Send_Termination	IN	BOOL	TRUE	0: without; 1 (default): with Termination of activation
Lock	IN	BOOL	FALSE	If set, new cmd will be not executed, they will be locked and neg. con
Max_Delay_Time	IN	Time	T#1M	0: without; >0 max. delay time for command with time tag; Default: t#60s
Timeout_Select_Execute	IN	Time	T#0MS	0: without sel/exe (def); >0 max. time between select and execute command.
IOA_Running_Cmd	OUT	DInt		IOA of an active command, otherwise 0; pulse cmd: if active; per.cmd: one cycle
FB_RetVal	OUT	Word		Neg. values: Error-Codes; o: ok

Parameter and function details:

Comp_ID

Block identifier for (currently only internal) diagnostics functions. Assign different numbers within a block type as byte variable.
e.g. B#16#1

P_Application

P_Application is a pointer to a common data range of block S7_IEC_Config. Therefore it has to be interconnected to its output P_Application via a DWord variable.
The Config block receives user data such as 'ASDU address', receiver buffer, etc. via this pointer or the data range behind it.

ASDU_Adr

Address value of the application service data unit receiving commands of this block.
Permissible range with ASDU address length 1 is 1 to 254
Permissible range with ASDU address length 2 is 1 to 65534.
The broadcast addresses 255 (FFh with length 1) resp. 65535 (FFFFh with length 2) are always accepted.

First_InfoAdr

Address value of the first information object edited
Permissible range with IOA address length 1 is 1 to 255
Permissible range with IOA address length 2 is 1 to 65535.
Permissible range with IOA address length 3 is 1 to 16777215
If the information object address exists in structured form (3 octets), it must be converted into a decimal number:
IOA dec = octet1 + 256*octet2 + 256*256*octet3
Other information objects are addressed consecutively/in ascending order.
Individual address allocation via address parameter-DB's is possible.
Detailed description see chapter 0.

First_Destination_Pos

Bit pointer to the first target position which is receiving the information.
e.g. P#A0.0 or P#DB1.DBX0.0
With non bit-oriented output structures, the byte limits must be observed (p#Ax.**0**), otherwise 'access errors' may occur.

No_of_Infos

Number of information to be output
Value range 1 to 255.

Dst_Struct_Type	Setpoint commands can be output in the following formats: 0 -> 2 byte setpoint value (integer) 1 -> 2 byte setpoint value (integer) + 1 byte status 2 -> 4 byte setpoint value (short real) 3 -> 4 byte setpoint value (short real) + 1 byte status -> For more details, refer to the description.
Send_Termination	Selection whether the command has to be completed with or without sending a 'Termination' of activation. FALSE: no 'Termination of activation' TRUE: with 'Termination of activation' (default setting)
Lock	Blocking input for all setpoint commands processed by this block. FALSE: New setpoint commands are executed TRUE: New setpoint commands are not executed and negatively acknowledged.
Max_Delay_Time	0 (t#0ms) deactivates the aging monitoring of commands with time stamp. >0: Aging monitoring for commands with time stamp active. The command must not be older than the parameterized period of time (Default t#60s). Otherwise the command will be neglected. Further (fixed) basic conditions are: The time in the system and command must be valid. The time in the command may be max. 1 sec (1 min from V1.6) in the future. The parameter is available from V1.5.
Timeout_Select_Execute	0 (t#0ms = default): Both select and execute commands will be always accepted > 0: Select before execute is activated. Executing commands are only accepted, if they are received within the parameterized period of time after receipt of the select command. The parameter is available from V1.5.
IOA_Running_Cmd	Output of the information address (IOA) of a currently running command The output is available from V1.5
FB_RetVal	Signals the reason why a command could NOT be executed. You will find a detailed list in chapter 6.1.3.4

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Dst_Struct_Type) in detail

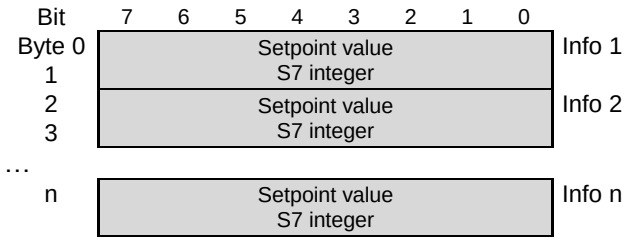
The structure of Dst_Struct_Type

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			Res		Data_Type	

		Image spacing
Data_Type	Data_Type used depending on information type – in this case, as follows: 00 (0): 2 byte setpoint value (integer) 01 (1): 2 byte setpoint value (integer) + 1 byte status 10 (2): 4 byte setpoint value (short real) 11 (3): 4 byte setpoint value (short real) + 1 byte status	 2 byte 3 byte 4 bytes 5 bytes
Add_Dist	Additional image distance which is considered in case of sequential writing	+ n Byte
Res	Reserved for future expansions	

Resultant typical image structures:

Structure type 0 (B#16#00):

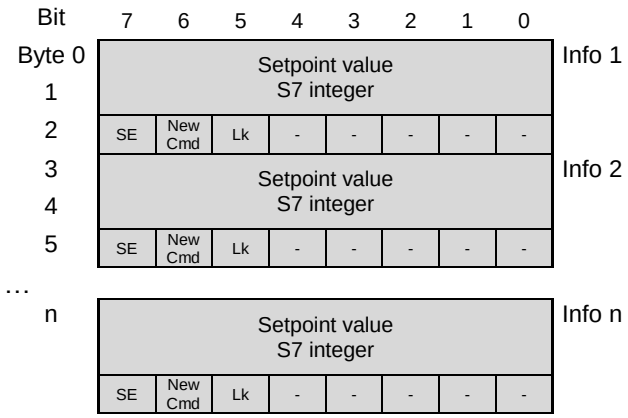


2 byte setpoint value (integer)

Resultant structure: 2 bytes per command

Each 2 bytes (1 word) represent a setpoint value in S7 integer format.

Structure type 1 (B#16#01):

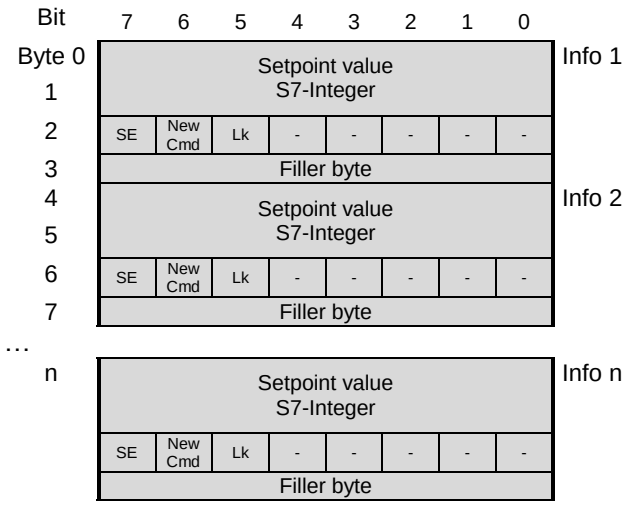


2 byte setpoint value (integer) + 1 byte status, IEC assignment

Resultant structure: 3 bytes per info

Each first 2 bytes (1 word) represent one setpoint value in S7 integer format, in each third byte, status bits for this setpoint value are stored.

Structure type 17 (B#16#11):

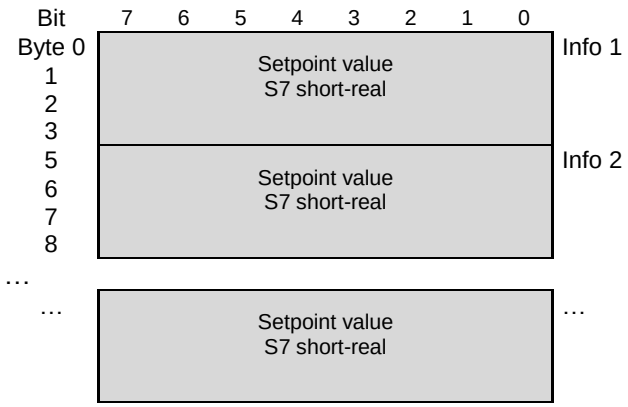


2 Byte setpoint value (Integer)
+ 1 Byte Status, IEC-assignment
+ 1 Filler byte (Add_Dist=1)

Resulting Structure: 4 Byte per information

The filler bytes remain regarding setpoint value output unconsidered and will be leaped. They may be arbitrary used from other parts of the program or are created only for structuring purposes.

Structure type 2 (B#16#02):



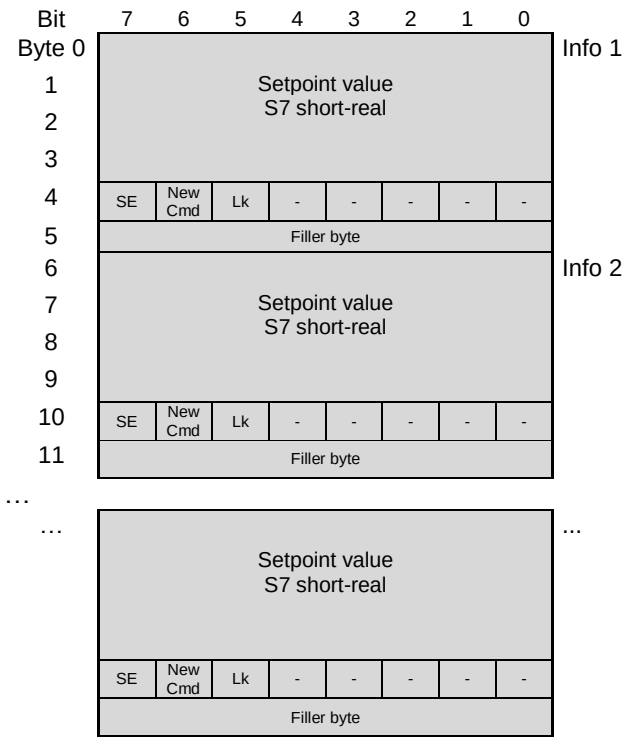
4 byte setpoint value (short real)

Resultant structure: 4 bytes per info

Each 4 bytes (1 DWord) represent a setpoint value in S7 short-real format.

Structure type 19 (B#16#13):

4 byte setpoint value (short real)
+ 1 byte status, IEC assignment
+ 1 filler byte (Add_Dist = 1)



Resultant structure: 6 bytes per info

Each first 4 bytes (1 DWord) represent a setpoint value in S7 short-real format, in each fifth byte; status bits for this setpoint value are stored.

The filler bytes remain ignored with regard to setpoint value output and are simply skipped. They may be arbitrary used by other parts of the program or are simply created just for structuring reasons.

Byte assignment when using Dst Struct Type = 1 or 3 in detail

SE	Select/Execute Select bit from IEC message
NewCmd	New setpoint command, is set with each output of a setpoint command. Can be reset by the user.
Lk	Locked Locking bit: If this bit is set before a new setpoint command is output, the command is blocked. Instead of that it is 'negatively' acknowledged to the control centre.

Additional notes on processing setpoint value adjusting commands

- The value of a received setpoint command is output permanently to the target area. Receipt of a new setpoint value overwrites the value in the target area.
- The IEC originator is mirrored and sent in the response messages as well if necessary
- Test bits and set P/N bit in the cause of transmission (COT) result in 'non-handling' – the message is not processed.
- Command cancellation via cancel command is always acknowledged negatively because setpoint value adjusting commands are terminated immediately.
- In principle the setpoint command from SLo-block is only processed (and therefore acknowledged) if it is a setpoint command which is directed to the block
  → In V1.0 version, no response to setpoint commands with unknown ASDU address, information address,...
- From V1.1 negative command confirmations are sent ‚substitutional' from SL_ORG_ASDU if no setpoint adjusting command block (SLo_...) has processed (confirmed) the message within the next OB1-cycle after receiving a setpoint adjusting command message (with COT6 or COT8). The confirmation is carried out with:

- COT 44: unknown TI, if there is no output block programmed for the received TI.
- COT 45: unknown COT, if the received COT is not supported from the responsible SLo-block (this is the one with the correct IOA)
- COT 47: unknown IOA, if there is no output block programmed for the received IOA.

For further notes please refer to chapter 6.1.1. As well the modified behavior from V1.5 is described there.

- Select/Execute is entered into the status bit SE when using Dst_Struct_Type 2 and 3, but not evaluated for the output of the setpoint command.
  The command is output regardless of S/E
  From V1.5 the Select/Execute handling is carried out according to standard. See as well block parameter “Timeout_Select_Execute”
- Setpoint command messages with time stamp (TI61, 62, 63) are accepted and output.
  Up to Version V1.4 the time stamp is not evaluated.
  From Version V1.5 see block parameter “Max_Delay_Time”

 ©SIEMENS AG 2017

6.1.3.3. Bit pattern commands - SLo_BO_sx (FB137)

The block receives digital setpoint commands as 32-bit bit patterns from the telecontrol partner. In the simplest case, it writes them as double word values to a data field (starting with First_Source_Pos).

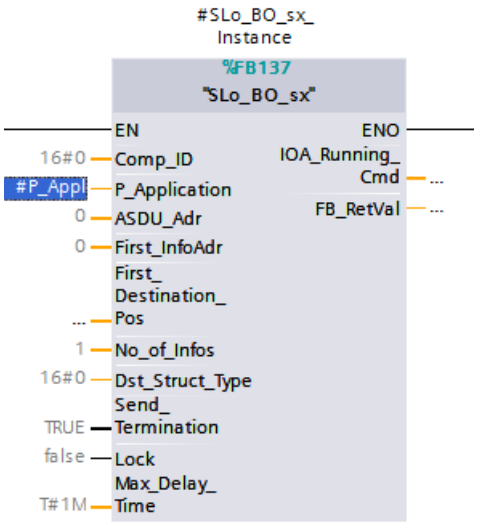


Illustration as FUP block with default values

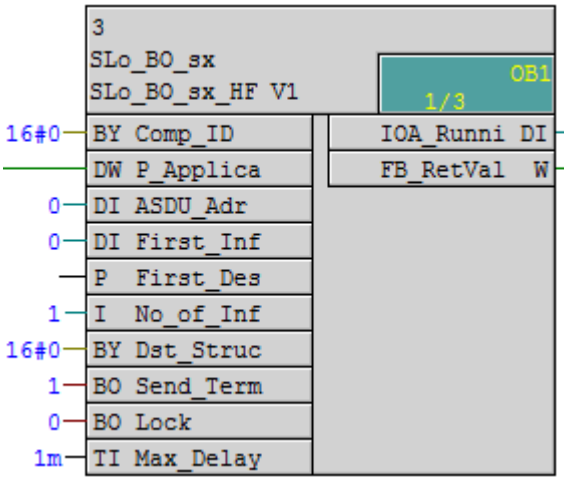


Illustration as CFC block with default values

Important features are:

- With a minor amount of parameterization, you can process many digital setpoint commands (No_of_Infos):
  - Digital setpoint commands with 32-bit bit pattern with TI51, 64
- Default settings can be read in the relevant variable comment, on the other hand they require only a minimum of mandatory inputs.
  The minimum requirement is the interconnection of P_Application and First_Destination_Pos.
- The information object addresses (IOA) and output position are automatically calculated in ascending/consecutive order starting with the programmable base address ('First_InfoAdr').
- Any output targets starting with 'First_Destination_Pos' (A,M,DB,...) can be automatically addressed
- Numerous output structures can be selected with 'Dst_Struct_Type':
  - 4 bytes
  - IEC format (+ status bits)
  - additional filler bytes
- A global blocking input permits the simple locking of all digital setpoint commands.
- Selective blocking of individual digital setpoint commands possible (Lk bit in the command byte)
- Individual address setting via address parameter-DB's possible
- Entire support of aging monitoring of commands with time stamp
- Return_Value and feedback output of the information object address (IOA) of the executed command

Block parameters with their default assignment and brief comment

FB137		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	DW#16#0	Pointer which must be connected to 'S7_IEC_Config' – Module
ASDU_Adr	IN	DINT	L#0	ASDU-Address which will be expected by this module (Default: 0)
First_InfoAdr	IN	DINT	L#0	First Information object address / Default = 0
First_Destination_Pos	IN	POINTER	-	First Position of .... e.g. A0.0 or DB900.DBX5.0
No_of_Infos	IN	INT	1	No of (sequential ascending) IOA which will supported (Default = 1)
Dst_Struct_Type :	IN	BYTE	B#16#0	0: (Def) 4 Byte (BO); 1: 5 Byte (BO Stat)
Send_Termination	IN	BOOL	TRUE	0: without; 1 (default): with Termination of activation
Lock	IN	BOOL	FALSE	If set, new cmd will be not executed, they will be locked and neg. con
Max_Delay_Time	IN	Time	T#1M	0: without; >0 max. delay time for command with time tag; Default: t#60s
IOA_Running_Cmd	OUT	DInt		IOA of an active command, otherwise 0; pulse cmd: if active; per.cmd: one cycle
FB_RetVal	OUT	Word		Neg. values: Error-Codes; o: ok

Parameter and function details:

Comp_ID

Block identifier for (currently only internal) diagnostics functions. Assign different numbers within a block type as byte variable.
e.g. B#16#1

P_Application

P_Application is a pointer to a common data range of block S7_IEC_Config. Therefore it has to be interconnected to its output P_Application via a DWord variable.
The Config block receives user data such as 'ASDU address', receiver buffer, etc. via this pointer or the data range behind it.

ASDU_Adr

Address value of the application service data unit receiving commands of this block.
Permissible range with ASDU address length 1 is 1 to 254
Permissible range with ASDU address length 2 is 1 to 65534.
The broadcast addresses 255 (FFh with length 1) resp. 65535 (FFFFh with length 2) are always accepted.

First_InfoAdr

Address value of the first information object edited
Permissible range with IOA address length 1 is 1 to 255
Permissible range with IOA address length 2 is 1 to 65535.
Permissible range with IOA address length 3 is 1 to 16777215
If the information object address exists in structured form (3 octets), it must be converted into a decimal number:
IOA dec = octet1 + 256*octet2 + 256*256*octet3
Other information objects are addressed consecutively/in ascending order.
Individual address allocation via address parameter-DB's is possible.
Detailed description see chapter 6.1.4.

First_Destination_Pos

Bit pointer to the first target position which is receiving the information.
e.g. P#A0.0 or P#DB1.DBX0.0
With non bit-oriented output structures, the byte limits must be observed (p#Ax.**0**), otherwise 'access errors' may occur.

No_of_Infos

Number of information to be output
Value range 1 to 255

Dst_Struct_Type	Digital setpoint commands can be output in the following formats: 0 -> 4 byte bit pattern (digital setpoint value) 1 -> 4 byte bit pattern (digital setpoint value) + 1 byte status 2 -> Res. 3 -> Res. -> For more details, refer to the description.
Send_Termination	Selection whether the command has to be completed with or without sending a 'Termination' of activation. FALSE: no 'Termination of activation' TRUE: with 'Termination of activation' (default setting)
Lock	Blocking input for all digital setpoint commands processed by this block. FALSE: New digital setpoint commands are executed TRUE: New digital setpoint commands are not executed and negatively acknowledged.
Max_Delay_Time	0 (t#0ms) deactivates the aging monitoring of commands with time stamp. >0: Aging monitoring for commands with time stamp active. The command must not be older than the parameterized period of time (Default t#60s). Otherwise the command will be neglected. Further (fixed) basic conditions are: The time in the system and command must be valid. The time in the command may be max. 1 sec (1 min from V1.6) in the future. The parameter is available from V1.5.
IOA_Running_Cmd	Output of the information address (IOA) of a currently running command The output is available from V1.5
FB_RetVal	Signals the reason why a command could NOT be executed. You will find a detailed list in chapter 6.1.3.4

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

The structure types (Dst_Struct_Type) in detail

The structure of Dst_Struct_Type

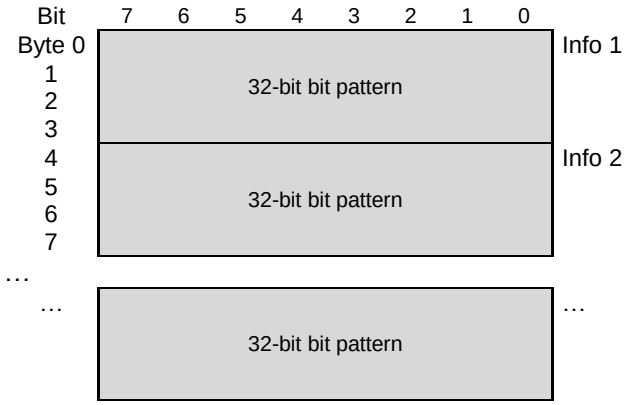
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Add_Dist			Res		Data_Type	

		Image-spacing
Data_Type	Data_Type used depending on information type – in this case, as follows: 00 (0): 4 byte bit pattern (digital setpoint value) 01 (1): 4 byte bit pattern (digital setpoint value) + 1 byte status 10 (2): Res. 11 (3): Res.	 4 byte 5 byte
Add_Dist	Additional image distance which is considered in case of sequential writing	+ n Byte
Res	Reserved for future expansions	

Resultant typical image structures:

Structure type 0 (B#16#00):

4 byte bit pattern (digital setpoint value)

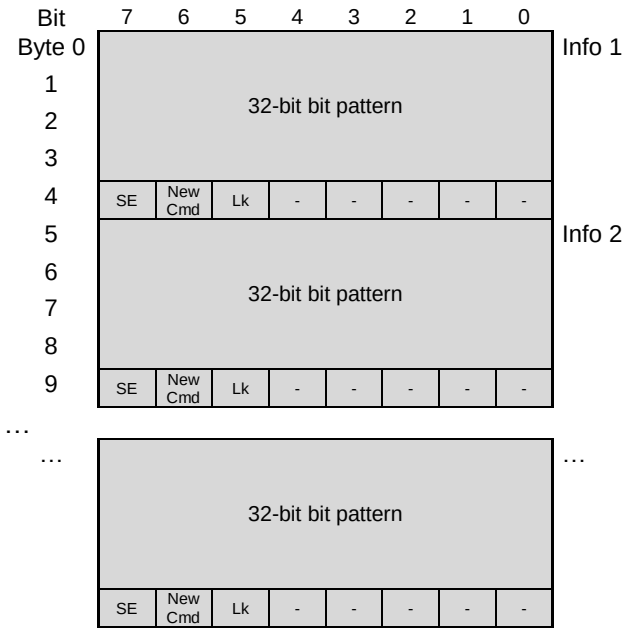


Resultant structure: 4 bytes per command

Each 4 bytes represent a 32-bit bit pattern which can be interpreted as 32 bit setpoint value.

Structure type 1 (B#16#01):

4 byte bit pattern (digital setpoint value)
+ 1 byte status, IEC assignment

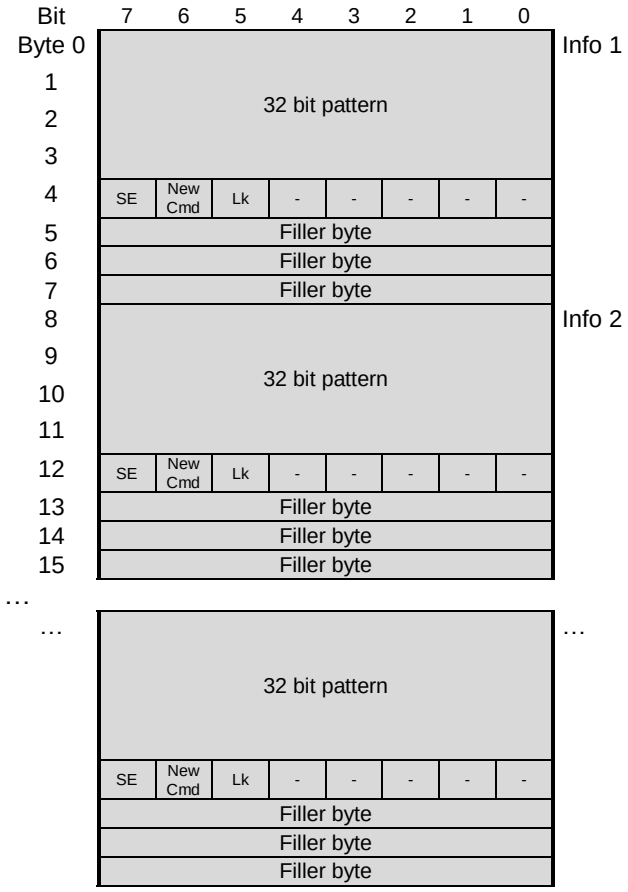


Resultant structure: 5 bytes per command

In each case the first 4 bytes represent a 32-bit bit pattern (digital setpoint value),
in each fifth byte the quality identifier is stored for this bit pattern.

Structure type 49 (B#16#31):

4 byte bit pattern (digital setpoint value)
+ 1 byte status, IEC-assignment
+ 3 filler byte (Add_Dist=3)



Resulting Structure: 8 Byte per information

The filler bytes remain regarding bit pattern output unconsidered and will be leaped. They may be arbitrary used from other parts of the program or are created only for structuring purposes.

Assignment of the status byte when using Dst Struct Type = 1 or 3 in detail

SE	Select/Execute Select bit from IEC message
NewCmd	New command - New digital setpoint command. Is set with each output of a digital setpoint command. Can be reset by the user.
Lk	Locked - blocked Blocking bit: If this bit is set before a new digital setpoint command is emitted, the setpoint command is not emitted. Instead of that it is 'negatively' acknowledged to the control center.

Additional notes on processing digital setpoint value commands

- The value of a received digital setpoint value command is output permanently to the target area. Receipt of a new digital setpoint value overwrites the value in the target area.
- The IEC originator is mirrored and sent in the response messages as well if necessary
- Test bits and set P/N bit in the cause of transmission (COT) result in 'non-handling' – the message is not processed.
- Command cancellation via cancel command is always acknowledged negatively because setpoint value commands are terminated immediately.
- In principle, the digital setpoint command from SLo-block is only processed (and therefore acknowledged) if it is a digital setpoint command which is directed to the block

-> In V1.0 version, no response to digital setpoint commands with unknown ASDU address, info address,...

- From V1.1 negative command confirmations are sent ‚substitutional' from SL_ORG_ASDU if no setpoint adjusting command block (SLo_...) has processed (confirmed) the message within the next OB1-cycle after receiving a digital setpoint adjusting command message (with COT6 or COT8).
  The confirmation is carried out with:
  - COT 44: unknown TI, if there is no output block programmed for the received TI.
  - COT 45: unknown COT, if the received COT is not supported from the responsible SLo-Block (it is the one with the correct IOA)
  - COT 47: unknown IOA, if there is no output block programmed for the received IOA.

  For further notes please refer to chapter 6.1.1. As well the modified behavior from V1.5 is described there.
- Digital setpoint command messages with time stamp (TI64) are accepted and output.
  Up to the Version V1.4 the time stamp is not evaluated
  From Version V1.5 see block parameter „Max_Delay_Time".

 ©SIEMENS AG 2017

6.1.3.4. Return_Values of the Slo-Blocks

FB_RetVal	Reason	Action	FB135	FB136	FB137
			SC..	SE..	BO..
16#8103	P_Application faulty		X	X	X
16#8401	Invalid status DCS (11 or 00)	Refuse command (negative confirmation COT 7)	X	-	-
16#8402	Non supported Qualifier	Refuse command (negative confirmation COT 7)	X	-	-
16#8410	Global command lock set	Refuse command (negative confirmation COT 7)	X	X	X
16#8411	Command specific lock set	Refuse command (negative confirmation COT 7)	X	X	X
16#8412	Command running (1 out of n)	Refuse command (negative confirmation COT 7)	X	-	-
16#8413	Command buffer full	Refuse command (negative confirmation COT 7)	X	-	-
16#8420	Select command received during select runs	Refuse command (negative confirmation COT 7)	X	X	-
16#8421	Select command received during command runs	Refuse command (negative confirmation COT 7)	X	-	-
16#8422	Execute command received without active select	Refuse command (negative confirmation COT 7)	X	X	-
16#8423	Execute command does not fit to active select	Refuse command (negative confirmation COT 7)	X	X	-
16#8431	Break command but no command active, or wrong IOA	Refuse command (negative confirmation COT 7)	X	X	X
16#8501	Time stamp in the command invalid	Neglect command	X	X	X
16#8502	System time invalid	Neglect command	X	X	X
16#8503	Fault at time difference calculation (FC SB_DT_DT)	Neglect command	X	X	X
16#8504	Time stamp in the command is in the future (> 1sec)	Neglect command	X	X	X
16#8505	Aging monitoring: Command to old	Neglect command	X	X	X

6.1.4. Individual IEC-Addressing

With the input parameter ‚First_IEC_Info_Adr' of the SLi/SLo-Blocks is defined whether the address allocation is carried out block by block, from an assignable start address or via address parameter data blocks.

If the addressing has to be carried out via an address parameter DB an offset from 100000000 (eight zeros) has to be entered there.

If the addresses have to be taken from the DB1301 for example, the value L#100001301 has to be entered at ‚First_IEC_Info_Adr'.

In the parameter DB the addresses then have to be entered in the sequence of the collected information.

The single information can be deactivated by entering the value 0 for address (Input L#0) in the parameter DB.

The structure of the address parameter DB's is the same for all collection blocks (SLi_...) and output blocks (SLo_...) and has the following basically structure:

Group/ Parameter	Addr. rel.	Addr. Abs.	Type	Initial value	Comment
DB_Manag	0	0	STRUCT		
DB_Manag. **Pos_of_DB**	+0.0	0.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **No_of_all_DBs**	+1.0	1.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Act**	+2.0	2.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Prev**	+4.0	4.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Next**	+6.0	6.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **Reserved**	+8.0	8.0	WORD	W#16#0	reserved for future - do not change!
	+10.0	10.0	END_STRUCT		END_STRUCT (10 BYTE)
Info_Manag			STRUCT		
Info_Manag. **P_Byte_First_Info**	+0.0	10.0	INT	24	(Byte-) Position of first Para-Block - do not change!
Info_Manag. **No_of_Infos**	+2.0	12.0	INT	0	0:unspecified (DB is filled up to the end); otherwise size n of ARRAY [1..n]
Info_Manag. **Len_Info**	+4.0	14.0	BYTE	B#16#4	Difference in [byte] between two Para-Blocks - do not change!
Info_Manag. **Len_Sort_Key**	+5.0	15.0	BYTE	B#16#0	0: without sorting; > 0: Data sets are sorted ascending with x Bytes
DB_Manag. **Re_internal_usage**	+6.0	16.0	DWORD	DW#16#0	
	+10.0	20.0	END_STRUCT		END_STRUCT (10 BYTE)
Global			STRUCT		
Global. **Para_DB_Type**	0.0	20.0	WORD	W#16#101	Internal identifier for this Data Block - do not change
Global. **Int_W22**	2.0	22.0	WORD	W#16#0	Reserved for internal use
	4.0	24.0	END_STRUCT		END_STRUCT (4 BYTE)
IEC_Adr			ARRAY [1..n] STRUCT		n = Number Parameter inputs
IEC_Adr **IOA_x**	+0.0	24.0	DINT	L#0	IOA Information object address
	4.0	28.0	END_STRUCT		END_STRUCT (4 BYTE)

	These values are pre-adjusted in the sample DB and need not to be changed. A sample DB is included in the corresponding block libraries. DB139 = ‚Para_DB_IOA_Demo'.
	!!! These parameters you have to set !!!
	Note: Always make changes in the 'Data view', not in the 'Declaration view' (except dimensioning of the array)

DB_Management	Parameter blocks which include an enumeration are created so, that more than one data block can be used if required. The following information is required for management. Currently the DB management is not used – for this reason all parameters can be left at their default settings (0)
Pos_of_DB	Number (1 – n) of the current data block for the parameter block 0: not used
No_of_all_DBs	Total number (n) of data blocks for the parameter block 0: not used
DB_No_Act	DB number of the current data block 0: not used
DB_No_Prev	DB number of the previous data block, if one exists. Otherwise: 0
DB_No_Next	DB number of the subsequent data block, if there is another one. Otherwise: 0
Info_Management	Parameter blocks containing an enumeration have one information block 'Info management'. It contains the following information The values of the information management parameters are already preset and therefore do not need to be modified.
P_Byte_First_Info	Byte position of the first data record in this DB, may not be changed
No_of_Infos	Number of data records contained in this DB 0: unspecified -> the data block end marks the last data record
Len_Info	Length of the data records: In this case 4: may not be changed
Len_Sort_Key	0: no sorting (default setting) Other sort lengths are not supported when using as parameter DB for Individual IEC addressing.
Global Parameter	
Para_DB_Type	Internal used: Identifier for this Parameter-DB – don't change!
IEC-Addresses	
IEC_Adr	The rating of the array is carried out in the declaration view of the data block. It has to be selected min. as large as the parameter No_of_Infos from the related SLi-/SLo blocks
IOA	Address value of the captured information objects Permissible range for length of the IOA-address 1 is 1 up to 255 Permissible range for length of the IOA-address 2 is 1 up to 65535. Permissible range for length of the IOA-address 3 is 1 up to 16777215 Is the information object address available in structured form (3 Octets), it has to be converted into a decimal number: IOA dec = Octet1 + 256*Octet2 + 256*256*Octet3 The value 0 (entered as initial value in the DB) deactivates the corresponding information. In this case no telecontrol specific processing is carried out

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

For each call of a capture or output block an own address parameter DB is needed.

6.2. Calling SLi-Blocks from time interrupts (Cyclic Interrupts)

From Version V1.5 **exclusively** SLi-Blocks (FB130-FB134) may be embedded in a time interrupt (e.g. OB35) **additionally** to the standard sequence (OB1). This is realized automatically from the blocks and the inputs are scanned only in the time interrupt.

In case of slow processing (e.g. 500 ms for measured values) a release of the standard cycle and therefore a faster communication can be achieved.

Faster time interrupt cycles (e.g. 10 ms) are suitable for gathering indication with high demand of timing. Caution: This is permitted only for a restricted number otherwise there is a possibility for cyclic time faults. The standard sequence is extended as well.

Time stamp and SL_Org_ASDU

Normally the actual time is generated in the Communication-FB. This is used as time stamp for the gathered information of the subsequent called SLi-blocks.

In order to achieve a higher time stamp accurateness for the SLi-blocks in time interrupts (in particular fast alarms) it is allowed to call the SL-Org_ASDU block belonging to the SLi-block as well in the time interrupt (as well without parameter supply). In this case it actualizes the time for all SLi-blocks running afterwards.

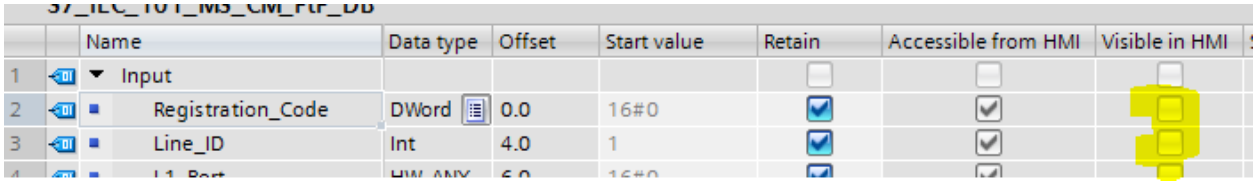
Note:

Please note that every FB in both calls (OB1 and time interrupt) is using the same instance DB.

The parameter supply (interconnection of the inputs and outputs, incl. e.g. P_SLi) can be renounced in the time interrupt. With it you not only save the global pointer connection but also the necessary multiple matching in the case of parameter changes.

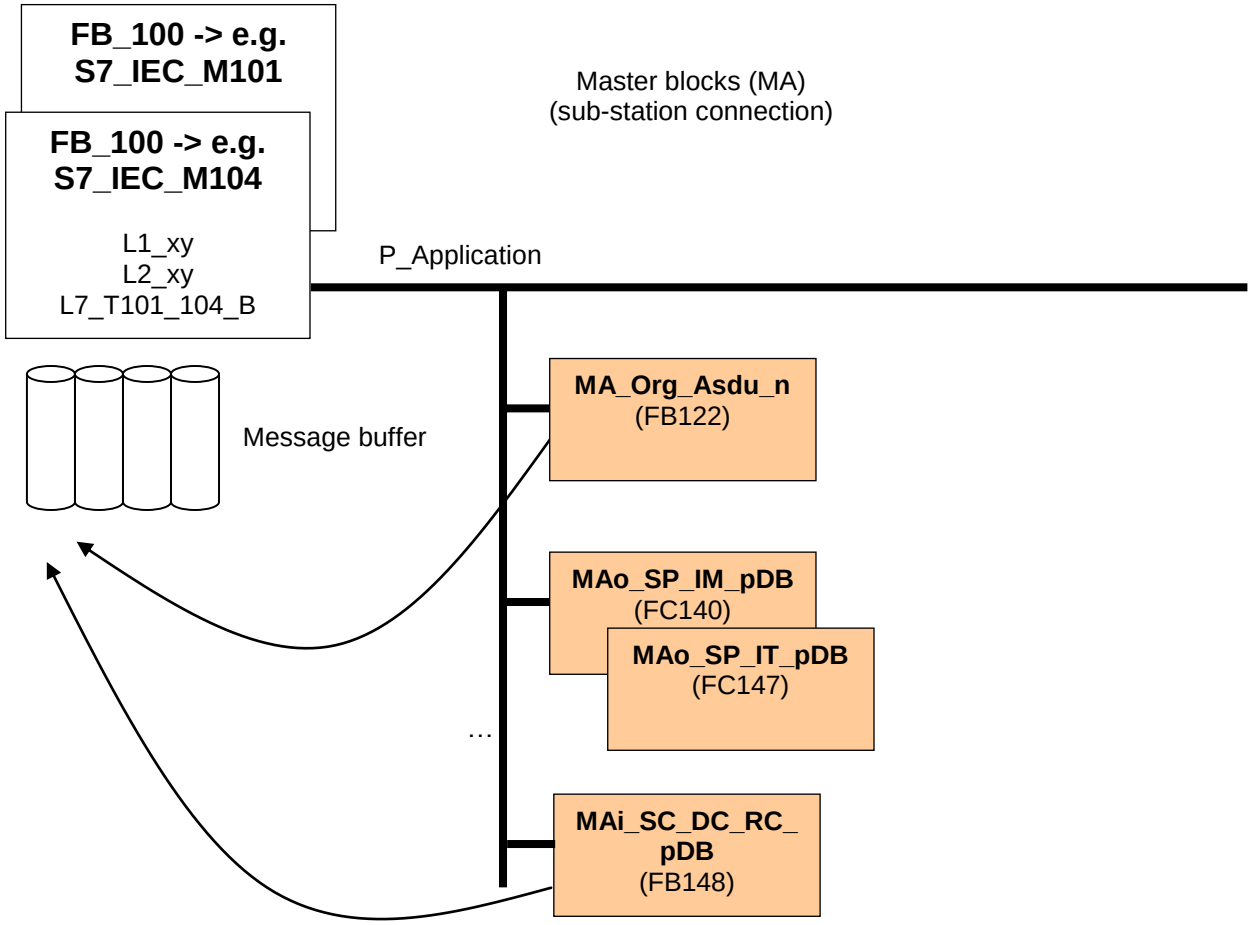
6.3. Symbol access HMI

The attributes "Visible in HMI" have been deactivated in all SIPLUS RIC blocks (FB and DB). This avoids the unnecessary provision for HMI of non-relevant information points.



		Name	Data type	Offset	Start value	Retain	Accessible from HMI	Visible in HMI
1		Input				☐	☐	☐
2		Registration_Code	DWord	0.0	16#0	☑	☑	
3		Line_ID	Int	4.0	1	☑	☑	

6.4. Master Blocks MA



All blocks are also multiple applicable

The Master (MA) application blocks are connected via the pointer 'P_Application' to the protocol version supplied, i.e. the functional block 'MA_Org_Asdu_1' and the blocks for the outputs in monitoring direction (MAo) as well as the inputs in control direction (MAi) are coupled directly to the IEC interface connection.

The following are available:

- **MA_Org block** for organizational tasks per ASDU address, such as send time, GI after link error.
- **MAi blocks** for the activation and monitoring of commands and setpoint values.
  One block for all commands and setpoint values.
- **MAo blocks** for the output of messages, measured values and count values.
  One block per information type.

The ending included in the block symbol names 'pDB' (MAi_xyz_**pDB**, MAo_xyz_**pDB**) refers to the 'parameterization' using data blocks. It contains 'global' parameters as well as 'assignment lists' and is described in detail in the subsequent chapters.

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

6.4.1. Organization block - MA_Org_ASDU_n (FB122)

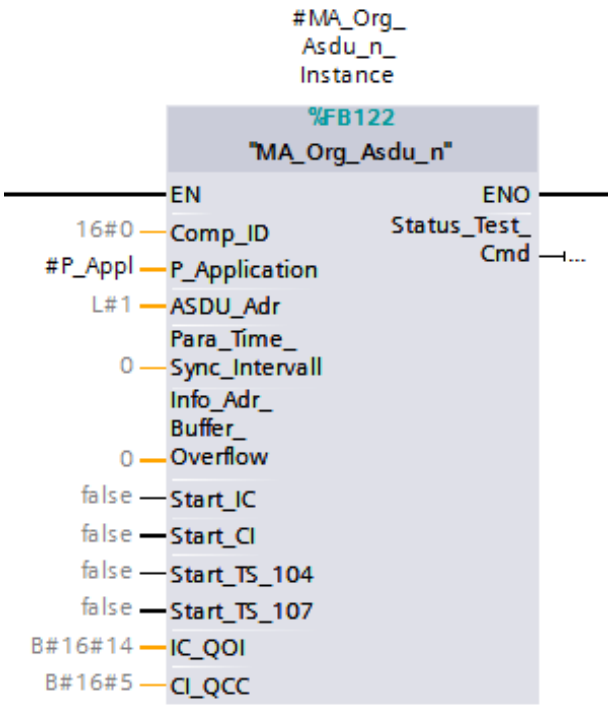


Illustration as FUP block with default values

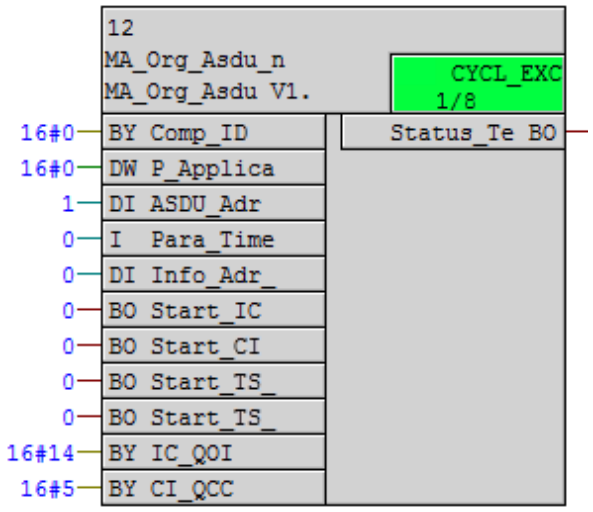


Illustration as CFC block with default values

The block has been created as a function block (FB), and therefore requires an instance DB (e.g. Call FB122,DB122) or can be used within a higher level FB as a 'multi-instance'.

The task of MA_ORG_ASDU_n is the station monitoring for the parameterized ASDU address concerning:

- Detected link error (from link layer L2)
- Receipt of start-up message TI70
- Receipt of a buffer overrun message (single info TI1 with parameterized address)

From V1.2 the MA_Org_ASDU_n is Multi-ASDU capable and supports the processing of several ASDU addresses per station/device.

Depending on the events the following actions are executed:

- GI request
  - After outgoing link error
  - After receipt of TI70
  - After receipt of incoming buffer overrun message
- Clock adjustment request (TI103), if activated
  - After receipt of TI70
  - In cyclical intervals according to parameterization

The block can also execute interrogation commands 'manually' by activating the relevant input and setting the associated identifier. With appropriate control cyclical/time-controlled (list) interrogations thus can be realized.

- (General) interrogation TI100
- Counter interrogation TI101

From version V1.6 the block can send test commands (TI104) resp. test commands with time stamp (TI107). The initialization is carried out by inputs. An output is available for monitoring.

Number of connectable sub-stations:

From V1.2 also polling lines with IEC 101 connections and several stations per interface are supported.

Block parameters with their default assignment and brief comment

FB122		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	0	Pointer which must be connected to 'S7_IEC_Config' – Module
ASDU_Adr	IN	DINT	L#1	ASDU-Address which will be controlled by this module Default L#1
Para_Time_Sync_Intervall	IN	INT	0	Send Clock synchronization command cyclic [min] or never (0 = default)
Info_Adr_Buffer_Overflow	IN	DINT	L#0	Incoming single point information with this address will generate a GI-Request
Start_IC	IN	Bool	FALSE	a rising edge will start an interrogation command with parameterized IC_QOI
Start_CI	IN	Bool	FALSE	a rising edge will start an counter interrogation with parameterized CI_QCC
Start_TS_104	IN	Bool	FALSE	a rising edge will start a test command C_TS_NA_1 (TI104)
Start_TS_107	IN	Bool	FALSE	a rising edge will start a test command C_TS_TA_1 (TI107)
IC_QOI	IN	BYTE	B#16#14	IEC-Qualifier of manual started IC (Default: B#16#14= global Station interro.)
CI_QCC	IN	BYTE	B#16#5	IEC-Qualifier of manual started CI (Default: B#16#05=general request counter)
Status_Test_Cmd	OUT	Bool	FALSE	1: with starting a test command (TI104,107); 0: with receiving the confirmation

Parameter and functional details:

Comp_ID

Block identifier for (currently only internal) diagnostic functions. Assign different numbers within a block type as byte variable.
e.g. B#16#1

P_Application

P_Application is a pointer to a common data range of block S7_IEC_Config. Therefore it has to be interconnected to its output P_Application via a DWord variable.
The Config block receives user data such as 'ASDU address', receiver buffer, etc. via this pointer or the data range behind it.

ASDU_Adr

Address value of the application service data unit receiving commands of this block.
Permissible range with ASDU address length 1 is 1 to 254
Permissible range with ASDU address length 2 is 1 to 65534.
The broadcast address 255 (FFh at length 1) resp. 65535 (FFFFh at length 2) is valid as well and may be given, if the remote station in a point-to-point connection transmits the information with several ASDU addresses.
In stations with several defined ASDU addresses or with several stations at one polling line, the number of an ASDU address parameter DB has to be given here

For more details refer to chapter 6.4.2.

Para_Time_Sync_Intervall

Time interval for sending a time synchronization message (TI103)
Permissible range at 0 to 65535.
Value 0: none
Value 1-x: Interval in min. In this interval time adjusting messages TI103 are sent to the partner.

Info_Adr_Buffer_Overflow

Spontaneously received single point information with the information object address parameterized here and status 'KOM' start a general interrogation.

Permissible range with IOA address length 1 is 1 to 255
Permissible range with IOA address length 2 is 1 to 65535.
Permissible range with IOA address length 3 is 1 to 16777215
If the information object address is available in structured form (3 octets), it must be converted into a decimal number:
IOA dec = octet1 + 256*octet2 + 256*256*octet3

Start_IC	With a rising edge an interrogation command (TI100) is started with interrogation identification according to parameter 'IC_QOI'. If a defined ASDU address is parameterized at the input ‚ASDU_Adr' this ASDU address will be used. When using an ASDU address parameter DB the broadcast address 255 (FFh at length 1) resp. 65535 (FFFFh at length 2) is used. Note: To avoid entries into the send buffer when the link layer has not been created, the trigger should be linked to LINK_ERR.
Start_CI	With a rising edge, a counter interrogation command (TI101) is started with interrogation identification according to parameter 'CI_QCC'. If a defined ASDU address is parameterized at the input ‚ASDU_Adr' this ASDU address will be used. When using an ASDU address parameter DB the broadcast address 255 (FFh at length 1) resp. 65535 (FFFFh at length 2) is used. Note: To avoid entries into the send buffer when the link layer has not been created, the trigger should be linked to LINK_ERR.
Start_TS_104 **Start_TS_107**	A rising edge initializes a test command C_TS_NA_1 (TI104) resp. C_TS_TA_1 (TI107). It will be only transmitted if there is no actual Link_Error. see also output: ‚Status_Test_Cmd'
IC_QOI	Interrogation identification 'QOI' according to the standard, for transmitting a manual started interrogation command. Permissible values: 0 to 255 Common values: 20 = B#16#14 (global station interrogation, GI) 21 = B#16#15 (group 1 interrogation) … 36 = B#16#24 (group 16 interrogation) The interrogation is started with a rising edge at 'Start_IC'.
CI_QCC	Interrogation identification 'QCC' according to standard, with which the counter interrogation command of a manually started counter interrogation is sent. Permissible values: see description QCC Common values: B#16#05 (general counter interrogation, no re-save, no reset) B#16#45 (general counter interrogation, with re-save, no reset) B#16#85 (general counter interrogation, with re-save, with reset) Interrogation is started with a rising edge at 'Start_CI'. -> For more details, refer to the description QCC.
Status_Test_Cmd	The output is set with transmission of the test command TI104 or TI107. The output is reset in case of: • startup • active Link_Error • with receive of a TI104 or TI107 message with COT = 7 (onfirmation) at the own ASDU address.

The counter interrogation identification (QCC) in detail

The structure of QCC according to the standard

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FRZ		RQT					

RQT Interrogation
(0): no counter interrogation (not used)
(1): counter interrogation group 1
(2): counter interrogation group 2
(3): counter interrogation group 3
(4): counter interrogation group 4
(5): general counter interrogation
(6..31): reserved (compatible range)
(32..63): reserved (private range)

FRZ Restore:
00 (0): interrogation (no restore or reset)
01 (1): counter restore no reset
10 (2): counter restore with reset
11 (3): counter reset
The action specified using FRZ is only effective for the group specified using RQT.

 ©SIEMENS AG 2017

6.4.2. ASDU-Address Parameter-DB

From V1.2 the FB ‚MA_Org_ASDU_1' is Multi-ASDU capable and supports the processing of several ASDU addresses per device.

In case of processing information with different ASDU addresses in one device (defined via the link address) the use of an ASDU address parameter DB is provided for this.

For realization from a polling line with several stations at one line the use of the ASDU address parameter DB is mandatory. Furthermore the use of the ASDU address parameter DB can be useful as well if only one station (point-to-point operating) is connected, in order to achieve a structuring of the information to be transmitted.

In addition to the parameterization of the ASDU addresses the assignment to the link address is carried out in this data block too. The number of this data block has to be indicated at the input parameter ‚ASDU_Adr' of the MA_Org_Asdu block.

If the address setting has to be carried out via an ASDU address parameter DB, an offset of 100000000 (eight zeros!) has to be entered there.

If the addresses are taken from the DB112 for example, the value of L#100000112 has to be entered in the ‚ASDU_Adr'.

In the parameter DB the ASDU addresses and the assignment to the corresponding station (link address) have to be indicated. The path for reaching the respective ASDU is defined via this assignment.

System related, the maximum number of the link and ASDU addresses is only limited by the DB length. Up to 8 link and ASDU addresses have been tested.

Design and values of the parameter data block:

The parameter-DB has the following basic design:

Group/ Parameter	Addr. rel.	Addr. Abs	Type	Initial value	Comment
DB_Manag	0	0	STRUCT		
DB_Manag. **Pos_of_DB**	+0.0	0.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **No_of_all_DBs**	+1.0	1.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Act**	+2.0	2.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Prev**	+4.0	4.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Next**	+6.0	6.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **Reserved**	+8.0	8.0	WORD	W#16#0	reserved for future - do not change!
	+10.0	10.0	END_STRUCT		END_STRUCT (10 BYTE)
Info_Manag			STRUCT		
Info_Manag. **P_Byte_First_Info**	+0.0	10.0	INT	34	(Byte-) Position of first Para-Block - do not change!
Info_Manag. **No_of_Infos**	+2.0	12.0	INT	0	0:unspecified (DB is filled up to the end); otherwise size n of ARRAY [1..n]
Info_Manag. **Len_Info**	+4.0	14.0	BYTE	B#16#4	Difference in [byte] between two Para-Blocks - do not change!
Info_Manag. **Len_Sort_Key**	+5.0	15.0	BYTE	B#16#0	0: without sorting; > 0: Data sets are sorted ascending with x Bytes
DB_Manag. **Re_internal_usage**	+6.0	16.0	DWORD	DW#16#0	
	+10.0	20.0	END_STRUCT		END_STRUCT (10 BYTE)
Global			STRUCT		
Global. **Glob_R1**	0.0	20.0	INT	0	
Global. **Glob_R2**	2.0	22.0	WORD	W#16#0	
	4.0	24.0	END_STRUCT		END_STRUCT (4 BYTE)

ASDU_Adr_Para			ARRAY [0..n] STRUCT		n = number-1 of parameter entries
ASDU_Adr_Para **ASDU_Address**	+0.0	24.0	DINT	L#0	Parameter: ASDU address
ASDU_Adr_Para **Link_Para_DB**	+4.0	28.0	INT	0	Parameter: associated link db no. 0 -> direct (no link db available)
ASDU_Adr_Para **Link_Para_Pos**	+6.0	30.0	INT	0	Parameter: parablock no. in link db
ASDU_Adr_Para **Error**	+8.0	32.0	BOOL	FALSE	internal
ASDU_Adr_Para **Res_1**	+9.0	33.0	BYTE	B#16#0	internal
ASDU_Adr_Para **Res_2**	+10.0	34.0	DINT	L#0	internal
ASDU_Adr_Para **Res_3**	+14.0	38.0	INT	0	internal
	16.0	40.0	END_STRUCT		END_STRUCT (16 BYTE)

	These values are preset in the sample DB and must not be changed. A sample DB included in the relevant block libraries. DB112 = ‚P_ASDUAdr_n'.
	!!! These parameters you have to set !!!
	Note: Always make changes in the 'Data view', not in the 'Declaration view' (except dimensioning of the array)

DB_Management

Parameter blocks which include an enumeration are created so that more than one data block can be used if required. The following information is required for management.
Currently the DB management is not used – for this reason all parameters can be left at their default settings (0)

Pos_of_DB
Number (1 – n) of the current data block for the parameter block
0: not used

No_of_all_DBs
Total number (n) of data blocks for the parameter block
0: not used

DB_No_Act
DB number of the current data block
0: not used

DB_No_Prev
DB number of the previous data block, if one exists.
Otherwise: 0

DB_No_Next
DB number of the subsequent data block, if there is another one.
Otherwise: 0

Info_Management

Parameter blocks containing an enumeration have one information block 'Info management'. It contains the following information
The values of the information management parameters are already preset and therefore do not need to be modified.

P_Byte_First_Info
Byte position of the first data record in this DB, may not be changed

No_of_Infos
Number of data records contained in this DB
0: unspecified -> the data block end marks the last data record

Len_Info
Length of the data records:
In this case 4: may not be changed

Len_Sort_Key
0: no sorting (default setting)
Other sorting lengths are not supported when using as parameter DB for optional IEC-addressing.

Global Parameter

Glob_R1	Reserve, not used
Glob_R2	Reserve, not used

ASDU_Adr_Para

The dimensioning of the array is carried out in the declaration view of the data block. It has to be done according to the number of the requested ASDU addresses.

There are n+1 ASDU-Addresses requested.

ASDU_Adress	Address value of the Application-Service –Data-Unit, receiving the commands from the MA_Org_ASDU block. Permissible range for length of the ASDU-address 1 is 1 up to 254 Permissible range for length of the ASDU-address 2 is 1 up to 65534 The broadcast addresses 255 (FFh at length 1) resp. 65535 (FFFFh at length 2) are automatically used from the block if necessary (e.g. for the clock synchronization with TI103)
Link_Para_DB	Refers to the assigned link address parameter DB 1 - n: Number of the assigned link address parameter DB 0: no link address parameter DB available (default setting) This setting has to be selected, when only one station with several ASDU addresses has to be requested.
Link_Para_Pos	Refers to a parameter block in the link address parameter DB and defines the link address (station/device) the ASDU belongs to. 0 – n Number of the assigned parameter block in the link address parameter DB
Error	Indicates whether the connection to the ASDU address is disturbed: FALSE: ASDU is not disturbed TRUE: ASDU is disturbed

NOTICE

Please note the permissible valuation of the block parameters. They are not mandatory checked for plausibility. Incorrect inputs can cause unpredictable responses and incorrect functions.

6.4.3. Output blocks - MAo_xyz_pDB

For the different information types in the monitoring direction, there are output blocks (FCs) available which have to be 'interconnected' to the relevant protocol version supplied, via the pointer 'P_Application'. The blocks then respond to receive messages with the relevant type identification. They take from this messages values and qualifiers which are written into an image (IM = Image; in data blocks or PAA).

The 'Allocation' of the IEC information objects to the required image position(s), as well as the transfer of basic settings/parameters is carried out in an assigned parameter data block.

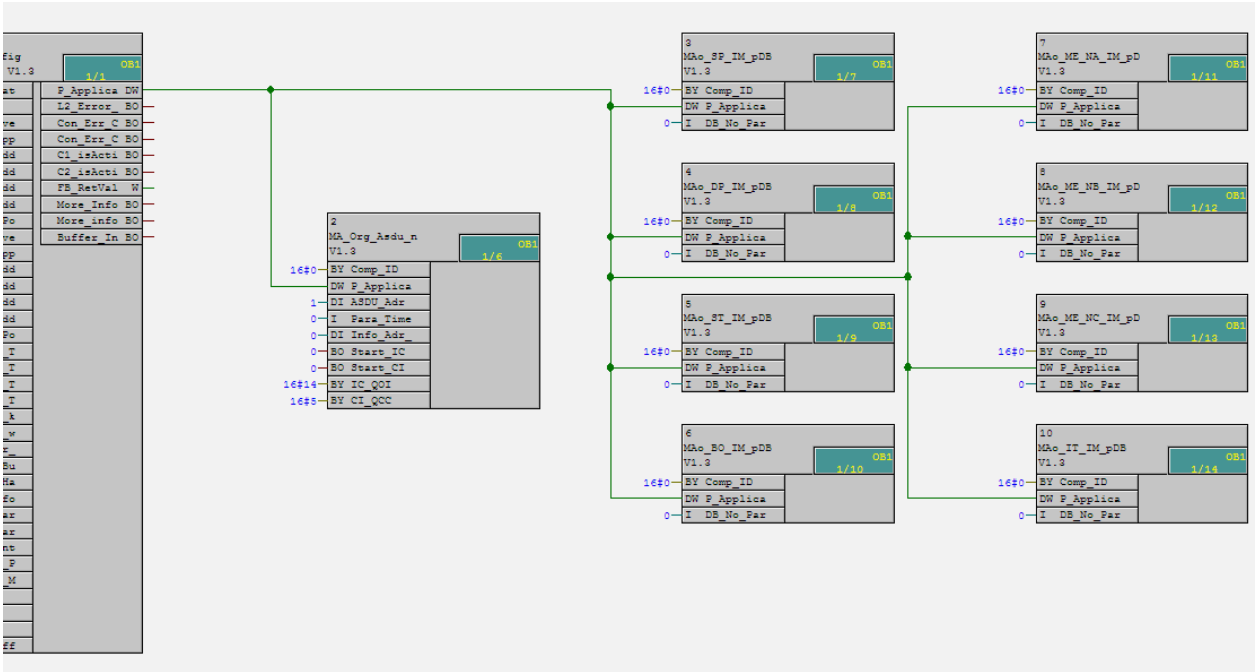
The symbol name encrypts the most important block properties as follows:

MAo_SP_IM_pDB	**MAo** is the abbreviation for master block output. The block outputs the information contained in the message (in this case to the PAA or a DB).
MAo_**SP**_IM_pDB	Information types supported by the block with the brief designation according to IEC standard: **SP** = Single Point information
MAo_SP_**IM**_pDB	**IM** is the abbreviation for image, i.e. the information is written from the block into an image (PAA or DB)
MAo_SP_IM_**p**DB	**p** is the abbreviation for programmable allocation of IEC address and output point.
MAo_SP_IM_p**DB**	The parameters for the individual information are stored in a **DB**.

The available MAo functions can be found in the following overview.

Block	Block. no.	Para DB no. (Default)	Assignment from … … into image (DB) or to PAA.
MAo_SP_IM_pDB	FC140	DB140	Single point information (SP; TI1, 2, 30) …
MAo_DP_IM_pDB	FC141	DB141	Double point information (DP; TI3, 4, 31) …
MAo_ST_IM_pDB	FC142	DB142	Step position information (ST; TI5, 6, 32) …
MAo_BO_IM_pDB	FC143	DB143	Bit pattern messages (BO; TI7, 8, 33) …
MAo_ME_NA_IM_pDB	FC144	DB144	Normalized measured values (ME_NA: TI9, 10, 34)
MAo_ME_NB_IM_pDB	FC145	DB145	Scaled measured values (ME_NB; TI11, 12, 35) …
MAo_ME_NC_IM_pDB	FC146	DB146	Measured value with short floating point number (ME_NC; TI13, 14, 36)
MAo_IT_IM_pDB	FC147	DB147	Integrated totals (IT; TI15, 16, 37) …

Illustration of the necessary interconnection - here in a CFC plan as an example:



The parameter DBs have the following basic structure (using the example of the parameter DB for single information):

Group/ Parameter	Addr. rel	Addr. abs.	Type	Start value	Comment
DB_Manag	0	0	STRUCT		
DB_Manag. **Pos_of_DB**	+0.0	0.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **No_of_all_DBs**	+1.0	1.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Act**	+2.0	2.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Prev**	+4.0	4.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Next**	+6.0	6.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **Reserved**	+8.0	8.0	WORD	W#16#0	reserved for future - do not change!
	+10.0	10.0	END_STRUCT		END_STRUCT (10 BYTE)
Info_Manag			STRUCT		
Info_Manag. **P_Byte_First_Info**	+0.0	10.0	INT	34	(Byte-) Position of first Para-Block - do not change!
Info_Manag. **No_of_Infos**	+2.0	12.0	INT	0	0:unspecified (DB is filled up to the end); otherwise size n of ARRAY [1..n]
Info_Manag. **Len_Info**	+4.0	14.0	BYTE	B#16#10	Difference in [byte] between two Para-Blocks - do not change!
Info_Manag. **Len_Sort_Key**	+5.0	15.0	BYTE	B#16#0	0: without sorting; > 0: Data sets are sorted ascending with x Bytes
DB_Manag. **Re_internal_usage**	+6.0	16.0	DWORD	DW#16#0	
	+10.0	20.0	END_STRUCT		END_STRUCT (10 BYTE)
Global			STRUCT		
Global. **Para_DB_Type**	0.0	20.0	WORD	W#16#101	Internal identifier for this Data Block - do not change!
Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1 Bit-Image, 2 IEC-Image, 3 IEC+Ext_State+Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#1	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors?
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated). For SP only 0 and 1 possible
Global. **Internal_10**	10.0	30.0	WORD	W#16#0	Internal use: Last found parablock – do not change!
Global. **Internal_12**	12.0	32.0	WORD	W#16#0	Internal use: Last processed info - do not change!
	14.0	34.0	END_STRUCT		END_STRUCT (14 BYTE)
Para_SP			ARRAY [1..n] STRUCT		n = Number Parameter inputs
Para_[x]. **S_Line**	+0.0	34.0	INT	1	Source: No. of Line (1..x)-actually do not change from 1
Para_[x]. **S_ASDU**	+2.0	36.0	INT	0	Source: Address of ASDU (ASDU-Address)
Para_[x]. **S_Info**	+4.0	38.0	DINT	L#0	Source: Address of Info object (IOA)
Para_[x]. **D_DB_No**	+8.0	42.0	INT	0	Dest. No. of Data Block, 0 -> PIQ (PAA)
Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest.: Bit position
Para_[x]. **P_No_of_fol_Infos**	+14.0	48.0	INT	0	Parameter: Number of following sequ. info elements
	16.0	50.0	END_STRUCT		END_STRUCT (16 BYTE)

	These values are preset in the default DBs and must not be changed. The default DBs are included in the relevant block libraries.
	!!! These parameters you have to set !!!
	Note: Always make changes in the 'Data view', not in the 'Declaration view' (except dimensioning of the array).

DB_Management	Parameter blocks which include an enumeration are created so, that more than one data block can be used if required. The following information is required for management. Currently the DB management is not used – for this reason all parameters can be left at their default settings (0)
Pos_of_DB	Number (1 – n) of the current data block for the parameter block 0: not used
No_of_all_DBs	Total number (n) of data blocks for the parameter block 0: not used
DB_No_Act	DB number of the current data block 0: not used
DB_No_Prev	DB number of the previous data block, if one exists. Otherwise: 0
DB_No_Next	DB number of the subsequent data block, if there is another one. Otherwise: 0
Info_Management	Parameter blocks containing an enumeration have one information block 'Info management'. It contains the following information The values of the information management parameters are already preset and therefore do not need to be modified.
P_Byte_First_Info	Byte position of the first data record in this DB
No_of_Infos	Number of data records contained in this DB 0: unspecified -> the data block end marks the last data record
Len_Info	Length of the data records: 0: not permitted 1 - 254: data record length 255: no standard length, length specifications in the byte format by sort criterion
Len_Sort_Key	0: no sorting (default setting) 8: the first 8 bytes of the information-specific parameters (see parameter group Para_SP, for example) form the sort criterion. Other sort lengths are not permitted when using as parameter DB for MAo blocks.
	If the DB is being used as parameter DB for MAo blocks and a large number of parameter blocks is required, it is advisable to create the parameter blocks sorted in ascending order by 'S_Line, S_ASDU and S_Info' in the parameter DB. A data block created in this way is then effectively searched in combination with 'Len_Sort_Key' = 8 (binary search).

General (global) parameters	These parameters are type-specific. For this reason their functions are only partially described here. Supported values are found in the relevant chapters in the block description.
Para_DB_Type	Used internally: Identifier for this parameter DB – do not change!
Image_Type	Output form of the information (raw values, IEC format… IEC-Format+ Ext_State+ Time)
Image_Len	Image spacing in bit – relevant with 'blocked' parameterization – results from Image_Type. 0: automatic calculation. The calculated length is written from MAo-block into the DB <> 0: fixed space of the process image in bit. From V1.1 ‚Image_Len' is pre-allocated with 0. Adaption is only necessary if a specific other (larger) image length shall be created.
Subst_on_Error	Use substitute value in case of error: TRUE: Use substitute value All affected information is set to a defined status/value. - With detected link error to the (IEC) partner, the substitute value is transferred into the image for all info points with the affected ASDU address. Outgoing link errors do not affect the value directly. This normally occurs via a general interrogation. This is running, with the resultant actual values. - If received information contains the set status bit 'NT' or 'IV', the substitute value is only entered in the image for this information. After receiving the relevant information with the status bits 'NT' and 'IV' not set, the actual value is entered into the image again. FALSE: Do not use substitute value Last collected status/value will be kept even in the event of an error. The use of substitute values is primarily useful if the image does not contain any status information. The substitute value is specified with the following parameter 'Subst_Value'.
Subst_Value	Substitute value – only relevant if 'Subst_on_Error' is set (TRUE).
Internal_10	Used internally: Last parameter block found – do not change!
Internal_12	Used internally: Last info processed – do not change!

Note on error processing

Regardless of the use of substitute values (Parameter 'Global.Subst_on_Error'), the NT bit in IEC images (Parameter 'Global.Image_Type') is always affected with a detected error.

The NT bit in the image is set for the relevant affected information:

- in case of link error detected with (IEC) partner
- in case of NT bit set (not topical) in the received message
- in case of IV bit set (invalid) in the received message

The updating of the information is carried out in the same way as described when using substitute values.

Para_SP ***Para per info (group)***	Create any number of parameter blocks by setting the array value 'n' accordingly in the declaration view. In the data view the parameters can then be modified individually. **refer in particular also to parameter 'P_No_of_fol_Infos'**
S_Line	Reserved for future applications. Must currently be set/remain on 1
S_ASDU	ASDU address of the (IEC/source) information being processed. Created as an INT value for easier input. If values > 32767 have to be entered, you either need to convert the data type to WORD (and enter hexadecimal) or calculate a corresponding negative value.
S_Info	Information object address (IOA) of the (IEC/source) information being processed
D_DB_No	Target position: This is the data block number for the image. The value 0 causes the output to PAA (process image of the outputs) instead of output to the data block.
D_Address	Target position: The image in the data block or PAA starts at this bit position. The following are permitted depending on Image_Type: Image_Type 1: Any values Image_Type 2: Multiple of 8 (byte limits)
P_No_of_fol_Infos	This parameter simplifies the parameterization work in case of linear structure. It also shortens program runtimes because in the ideal situation information can be allocated with a few, or possibly just one parameter block. Values not equal to 0 result in a sequential expansion of the above source and target parameters, in other words in a range starting with S_Info or D_Address. Starting with the base Info_address the following info addresses are processed in accordance with the number given here. The image position is calculated using 'D_Address' and the information position multiplied by 'Image_Len'.

Info on data throughput

The IEC communication-FBs each provides max. 1 user information per (OB1) cycle. In the same OB1 cycle this information has to be processed/output by the process blocks.

This prevents on the one hand multiple outputs from one and the same information and thus the loss of indication changes (if the images are also evaluated in the OB1). On the other hand, this produces a direct dependency of the maximum data throughput with the CPU cycle time, which is relevant in particular with high CPU load. In this case it can also result in backlogs in the partner station.

With programming a suitable loop using the S7_IEC_Config Parameter ‚More_Info_available' and ‚Next_Info' an increasing of the data flow-rate per OB1 cycle can be achieved with only slight increased cycle time. See also chapter 4.5.

6.4.3.1. Single point information - MAo_SP_IM_pDB (FC140)

MAo_SP_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain single point information and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

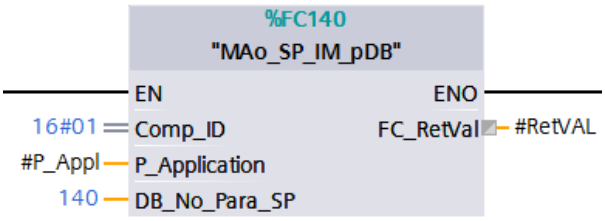


Illustration as FUP block with default values

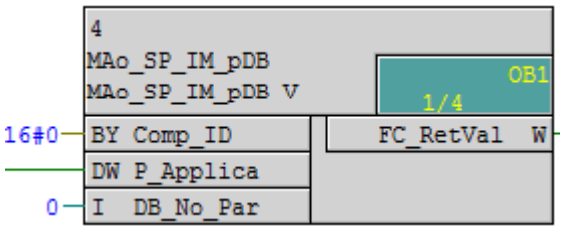


Illustration as CFC block with default values

Important features are:

- The block processes single point information messages with TI1, 2 or 30.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of info points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value (bit format)
  - IEC- Format
  - IEC- Format and time stamp
  - For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC140		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_SP	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No of processed info

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_SP

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:

0:	ok
1-n:	number of information processed in the current cycle
7001h	partner disturbance coming (Error_Link)

Negative FC_RetVal values:

8112h:	P_Application - fault
8302h	unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block can be found in the main chapter.

Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1 Bit-Image, 2 IEC-Image, 3 IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#1	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors ?
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated). For SP only 0 and 1 possible

Image_Type/ Image_Len

In the current version the block supports 3 versions of the message output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value/Bit output	1 (B#16#1)	X
2 (B#16#2)	IEC- Output	8 (B#16#8)	
3 (B#16#3)	IEC-Format+ Ext_State +Time	80 (B#16#50)	

From V1.1 ‚Image_Len' is pre-allocated with 0 (automatic calculation in dependence of ‚Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
In combination with single point information, only the value range 0 - 1 is appropriate for Subst_Value.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address

Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type the following are permitted:

Image_Type 1: Any values (bit position)
Image_Type 2: Multiple of 8 (byte limits have to be kept)
Image_Type 3: Multiple of 8 (byte limits have to be kept)

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	7	6	5	4	3	2	1	0	0 - 7
1	15	14	13	12	11	10	9	8	8 – 15
2						...	17	16	16 ...
...									...
16	127								...127
…									...

Bit output:

Resultant structure:
8 information per byte

Only the status of the EM (SPI = 1 Bit) contained in the IEC message is read and written to the relevant bit position (Parameter'D_Address').

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 1 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No').

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	IV	NT	SB	BL	-	-	-	SPI	0
1	IV	NT	SB	BL	-	-	-	SPI	8
2	IV	NT	SB	BL	-	-	-	SPI	16
…									…
127	IV	NT	SB	BL	-	-	-	SPI	1016
...									…

1 byte IEC output:

Resultant structure:
1 byte per info

In this mode, the output is carried out by the direct transfer of the entire information 'SIQ' from the IEC message.
The image therefore has the structure shown opposite.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the status bit SPI is also affected depending on the parameter 'Subst_on_Error'.

 ©SIEMENS AG 2017

Image_Type 3 (B#16#03):

1 Byte IEC-output
+ 1 Byte Ext_State
+ 8 Byte time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	IV	NT	SB	BL	-	-	-	SPI	0
1	Res	Res	Res	LO	IV	SY	DS	TI	
2	S7 Date and Time								
3									
4									
5									
6									
7									
8									
9									
10	IV	NT	SB	BL	-	-	-	SPI	80
11	Res	Res	Res	LO	IV	SY	DS	TI	
12	S7 Date and Time								
13									
14									
15									
16									
17									
18									
19									
20	IV	NT	SB	BL	-	-	-	SPI	160
…									…
1270	IV	NT	SB	BL	-	-	-	SPI	10160
...									…

Resulting structure:
10 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚SIQ' from the IEC message.
Subsequently
1 byte Ext_State and
8 byte S7 Date and Time

The image has so the adjoining structure.

In the IEC output format the block sets ‚substitutional' the contained NT-Bit in case of appearing connection disturbances. As an option also the status bit SPI is affected depending on parameter ‚Subst_on_Error'.

Compared with ‚Image_Type' 2 the image is extended with the status byte ‚Ext_State' and ‚Date and Time' in S7 format.

‚Ext_State' and ‚S7 Date and Time' in detail

Ext_State

The status byte serves prior for handing over the time status bits which are not supported from the S7-Date and Time format.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Res	Res	Res	LO	IV	SY	DS	TI

TI: The TI - Bit is always set from MAo-Block.
From SLi-Blocks the status bits (DS, SY, and IV) are only taken over if the TI - Bit is set.

DS: Daylight Saving Time
Direct transfer from IEC-Message with full time stamp, otherwise
Transfer from time status, carried in the CPU (Communication-FB-Parameter Time_DS).

SY: Synchronized
Transfer from time status, carried in the CPU Communication-FB-Parameter Time_DS).
The bit is not used in IEC-Messages and serves only for information.

IV: Invalid

IV is built as follows

Receive message	IV
With full time stamp	Direct transfer from IEC-Message
With part time stamp	Set, if the IV-bit in the IEC-Message or in the time status carried in the CPU is set (Communication-FB Parameter Time_IV).
without time stamp	Transfer from time status, carried in the CPU (Communication-FB -Parameter Time_IV)

LO: Local time is set if the Communication-FB-Parameter Time_Diff <> 0.
Transfer from time status, carried in the CPU.
The bit is not used in IEC-Messages and serves only for information.

S7 Date and Time

Date and Time in S7-Format

The time is built as follows

Receive message	S7 Date and Time
With full time stamp	Transfer of the time from IEC-Message.
With part time stamp	The part time from IEC-Message is completed to full time with the time carried in the CPU
without time stamp	Transfer of the time carried in the CPU

The images are initialized with the actual time of the CPU

 ©SIEMENS AG 2017

6.4.3.2. Double point information - MAo_DP_IM_pDB (FC141)

MAo_DP_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain double point information and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

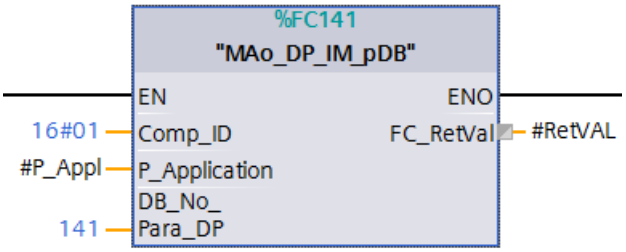


Illustration as FUP block with default values

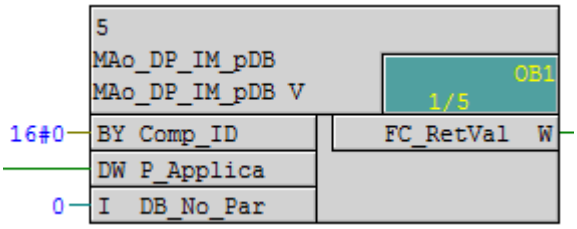


Illustration as CFC block with default values

Important features are:

- The block processes double point information messages with TI3, 4 or 31.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of info points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value (bit format)
  - IEC- Format
  - IEC- Format and time stamp
  - For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC141		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_DP	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed info

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_DP

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:
0: ok
1-n: number of information processed in the current cycle
7001h partner disturbance coming (Error_Link)

Negative FC_RetVal values:
8112h: P_Application - fault
8302h unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block can be found in the main chapter.

Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1 Bit-Image, 2 IEC-Image, 3 IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#2	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors ?
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated). For DP only 0, 1, 2 or 3 possible

Image_Type/ Image_Len

In the current version the block supports 2 versions of the message output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value/Bit output	2 (B#16#2)	X
2 (B#16#2)	IEC- output	8 (B#16#8)	
3 (B#16#3)	IEC-Format+ Ext_State + Time	80 (B#16#50)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
In combination with double point information, only the value range 0 - 3 is appropriate for Subst_Value.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address

Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type the following are permitted:

Image_Type 1: Any values (bit position)
Image_Type 2: Multiple of 8 (byte limits) have to be kept)
Image_Type 3: Multiple of 8 (byte limits have to be kept)

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Info 3		Info 2		Info 1		Info 0		0, 2, ...
1	Info 7		Info 6		Info 5		Info 4		8, 10, ...
2			...		9		Info 8		16, 18 ...
...									...
16	63								... 126
…									...

Bit output:

Resultant structure:
4 infos per byte

Only the status of the DM contained in the IEC message (DPI = 2 bits) is read and written to the relevant bit position (Parameter 'D_Address').

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 2 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No').

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	IV	NT	SB	BL	-	-	DPI		0
1	IV	NT	SB	BL	-	-	DPI		8
2	IV	NT	SB	BL	-	-	DPI		16
…							DPI		…
127	IV	NT	SB	BL	-	-	DPI		1016
...									…

1 byte IEC output:

Resultant structure:
1 byte per info

In this mode the output is carried out by the direct transfer of the entire information 'DIQ' from the IEC message.
The image therefore has the structure shown opposite.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the status bit DPI is also affected depending on the parameter 'Subst_on_Error'.

Image_Type 3 (B#16#03):

1 Byte IEC-output
+ 1 Byte Ext_State
+ 8 Byte time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	IV	NT	SB	BL	-	-	DPI		0
1	Res	Res	Res	LO	IV	SY	DS	TI	
2	S7 Date and Time								
3									
4									
5									
6									
7									
8									
9									
10	IV	NT	SB	BL	-	-	DPI		80
11	Res	Res	Res	LO	IV	SY	DS	TI	
12	S7 Date and Time								
13									
14									
15									
16									
17									
18									
19									
20	IV	NT	SB	BL	-	-	DPI		160
…									…
1270	IV	NT	SB	BL	-	-	DPI		10160
...									…

Resulting structure:
10 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚DIQ' from the IEC message.
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image has so the adjoining structure.

In the IEC output format the block sets 'substitutional' the contained NT-Bit in case of appearing connection disturbances. As an option also the status bit DPI is affected depending on parameter ‚Subst_on_Error'.

Compared with, Image_Type' 2 the image is extended with the status byte 'Ext_State' and 'Date and Time' in S7 format.

Detailed information to 'Ext_State' and 'S7 Date and Time' see chapter 6.4.3.1.

6.4.3.3. Step position information - MAo_ST_IM_pDB (FC142)

MAo_ST_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain step position information and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

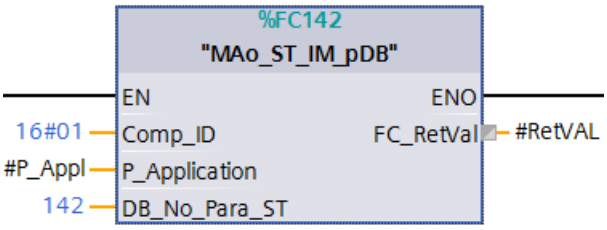


Illustration as FUP block with default values

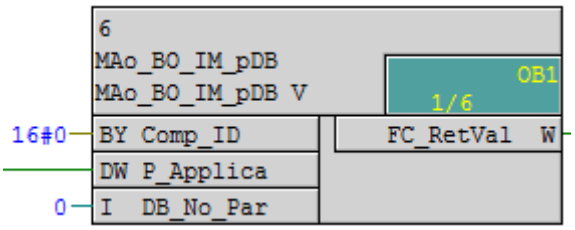


Illustration as CFC block with default values

Important features are:

- The block processes step position information with TI5, 6 or 32.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of information points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value (bit format)
  - IEC- Format
  - IEC- Format and time stamp
- For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).

    An EXCEL based help tool for a fast production of parameter-DBs (AWL sources) particularly for larger number of data points and integration of external data sources (Excel lists) is available on inquiry.
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC142		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_ST	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed info

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_ST

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:

0:	ok
1-n:	number of information processed in the current cycle
7001h	partner disturbance coming (Error_Link)

Negative FC_RetVal values:

8112h:	P_Application - fault
8302h	unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block can be found in the main chapter.

Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1: 1 Byte excl.T, 2: 1 Byte incl. T, 3: IEC 2 Bytes; 4: IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#8	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors.
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated)

Image_Type/ Image_Len

In the current version the block supports 3 versions of the message output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value/Byte output without intermediate position display (T)	8 (B#16#8)	X
2 (B#16#2)	Raw value/Byte output with Intermediate position display (T)	8 (B#16#8)	
3 (B#16#3)	IEC- Output	16 (B#16#10)	
4 (B#16#4)	IEC-Format+ Ext_State + Time	88 (B#16#58)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
Permissible values are 0 to 255.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address Target position: The image in the data block or PAA starts at this bit position. Regardless of Image_Type, 'D_Address' has to be a multiple of 8 (byte limits have to be observed).

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	-	Info 1							0
1	-	Info 2							8
2	-	Info 3							16
...	-	…							...
16	-	Info 17							128
…	-								...

1 Byte Byte output:

Resultant structure:
1 byte per information

Only the value (value = 7 Bit) of the level setting contained in the IEC message is read and written to the relevant bit position (Parameter 'D_Address').

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 8 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No')

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	T	Info 1							0
1	T	Info 2							8
2	T	Info 3							16
...	T	...							...
16	T	Info 17							128
…	T								...

1 Byte Byte output:

Resultant structure:
1 byte per info

The value (value = 7 Bit) of the level setting contained in the IEC message is read and written to the relevant bit position (Parameter 'D_Address').
In addition, with 'T' .

T=0: Equipment is not in intermediate state
T=1: Equipment is in intermediate state

Image_Type 3 (B#16#03):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	T	Info 1							0
Byte 1	IV	NT	SB	BL	-	-	-	OV	
Byte 2	T	Info 2							16
Byte 3	IV	NT	SB	BL	-	-	-	OV	
Byte 4	T	Info 3							32
Byte 5	IV	NT	SB	BL	-	-	-	OV	
…	T	…							...
…	IV	NT	SB	BL	-	-	-	OV	
Byte16	T	Info 9							128
…	IV	NT	SB	BL	-	-	-	OV	
…	T	…							136
…	IV	NT	SB	BL	-	-	-	OV	

2 Byte IEC output:

Resultant structure:
2 bytes per info

In this mode the output is carried out by the direct transfer of the entire information 'VTI' and 'QDS' from the IEC message.
The image therefore has the structure shown opposite.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the value of the information is also affected depending on the parameter 'Subst_on_Error'.

 ©SIEMENS AG 2017

Image_Type 4 (B#16#04):

2 Byte IEC-Output
+ 1 Byte Ext_State
+ 8 Byte Time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	T	Info 1							0
1	IV	NT	SB	BL	-	-	-	OV	
2	Res	Res	Res	LO	IV	SY	DS	TI	
3	S7 Date and Time								
4									
5									
6									
7									
8									
9									
10									
11	T	Info 2							88
12	IV	NT	SB	BL	-	-	-	OV	
13	Res	Res	Res	LO	IV	SY	DS	TI	
14	S7 Date and Time								
15									
16									
17									
18									
19									
20									
21									
22	T	Info 3							176
…	IV	NT	SB	BL	-	-	-	OV	
…									
…									
77	T	Info 9							616
…	IV	NT	SB	BL	-	-	-	OV	
…									
…									

Resulting structure:
11 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚VTI' and ‚QDS' from the IEC message.
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image has so the adjoining structure.

In the IEC output format the block sets 'substitutional' the contained NT-Bit in case of appearing connection disturbances. As an option also the value of the information is affected depending on parameter ‚Subst_on_Error'.

Compared with 'Image_Type' 3 the image is extended with the status byte 'Ext_State' and 'Date and Time' in S7 format.

Detailed information to 'Ext_State' and 'S7 Date and Time' see chapter 6.4.3.1.

6.4.3.4. Bit pattern messages - MAo_BO_IM_pDB (FC143)

MAo_BO_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain bit pattern messages and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

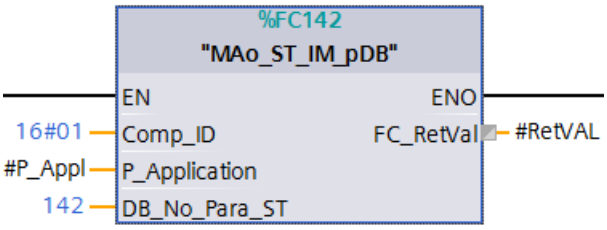


Illustration as FUP block with default values

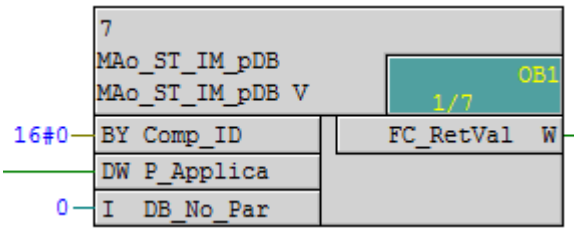


Illustration as CFC block with default values

Important features are:

- The block processes bit pattern messages with TI7, 8 or 33.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of info points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value
  - IEC- Format
  - IEC- Format and time stamp
- For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).

    An EXCEL based help tool for a fast production of parameter-DBs (AWL sources) particularly for larger number of data points and integration of external data sources (Excel lists) is available on inquiry.
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC143		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_SP	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed information

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_SP

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:
0: ok
1-n: number of information processed in the current cycle
7001h partner disturbance coming (Error_Link)

Negative FC_RetVal values:
8112h: P_Application - fault
8302h unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block can be found in the main chapter. Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#4	1-4: 1 - 4 Byte Bit pattern , 5: IEC 5 Bytes; 6: IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#20	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors.
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated)

Image_Type/ Image_Len

In the current version the block supports 5 versions of the message output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value output 1 Byte bit pattern length	8 (B#16#8)	
2 (B#16#2)	Raw value output 2 Byte bit pattern length	16 (B#16#10)	
3 (B#16#3)	Raw value output 3 Byte bit pattern length	24 (B#16#18)	
4 (B#16#4)	Raw value output 4 Byte bit pattern length	32 (B#16#20)	X
5 (B#16#5)	IEC- Output	40 (B#16#28)	
6 (B#16#6)	IEC-Format + Ext_State + Time	112 (B#16#70)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
Permissible values are 0 to 255 in each case in every byte.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type, 'D_Address' has to be a multiple of 8 (byte limits have to be observed).

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	8-bit bit pattern (Info 1)								0
1	8-bit bit pattern (Info 2)								8
2	8-bit bit pattern (Info 3)								16
...	...								...
16	8-bit bit pattern (Info 17)								128
…									...

1 byte output:

Resultant structure:
1 byte per information

Only the first byte of the bit pattern (8 bit) contained in the IEC message is read and written to the relevant bit position (Parameter 'D_Address').

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 8 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No').

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	16-bit bit pattern (Info 1)								0
1									
2	16-bit bit pattern (Info 2)								16
3									
4	16-bit bit pattern (Info 3)								32
5									
…	…								...
…									
32	16-bit bit pattern (Info 17)								256
33									
…	…								...
…									

2-byte output:

Resultant structure:
2 bytes per information

Only the first 2 bytes of the bit pattern (16 bit) contained in the IEC message are read and written to the relevant bit position (Parameter 'D_Address').

Image_Type 3 (B#16#03):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	24-bit bit pattern (Info 1)								0
1									
2									
3	24-bit bit pattern (Info 2)								24
4									
5									
6	24-bit bit pattern (Info 3)								48
7									
8									
…	…								...
…									
...									
48	24-bit bit pattern (Info 17)								384
49									
...									
…	…								...
…	...								
...									

3-byte output:

Resultant structure:
3 bytes per information

Only the first 3 bytes of the bit pattern (24 bit) contained in the IEC message are read and written to the relevant bit position (Parameter 'D_Address').

Image_Type 4 (B#16#04):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	32-bit bit pattern (Info 1)								0
1									
2									
3									
4	32-bit bit pattern (Info 2)								32
5									
6									
7									
8	32-bit bit pattern (Info 3)								64
9									
10									
11									
…	…								...
…									
…									
…									
64	32-bit bit pattern (Info 17)								512
65									
66									
67									
…	…								...
…									
…									
…									

4-byte output:

Resultant structure:
4 bytes per information

All 4 bytes of the bit pattern (32 bit) contained in the IEC message are read and written to the relevant bit position (Parameter 'D_Address').

Image_Type 5 (B#16#05):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	32-bit bit pattern (Info 1)								0
1									
2									
3									
4	IV	NT	SB	BL	-	-	-	OV	
5	32-bit bit pattern (Info 2)								40
6									
7									
8									
9	IV	NT	SB	BL	-	-	-	OV	
10	32-bit bit pattern (Info 3)								80
11									
12									
13									
14	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…									
…									
…	IV	NT	SB	BL	-	-	-	OV	
80	32-bit bit pattern (Info 17)								640
81									
82									
83									
84	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…									
…									
…	IV	NT	SB	BL	-	-	-	OV	

5 Byte IEC output:

Resultant structure:
5 bytes per information

In this mode, the output is carried out by the direct transfer of the entire information 'BSI' and 'QDS' from the IEC message.
The image therefore has the structure shown opposite.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the value of the information is also affected depending on parameter 'Subst_on_Error'.

 ©SIEMENS AG 2017

Image_Type 6 (B#16#06):

5 Byte IEC-Output
+ 1 Byte Ext_State
+ 8 Byte Time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	32 Bit Bit pattern (Info 1)								0
1									
2									
3									
4	IV	NT	SB	BL	-	-	-	OV	
5	Res	Res	Res	LO	IV	SY	DS	TI	
6	S7 Date and Time								
7									
8									
9									
10									
11									
12									
13									
14	32 Bit Bit pattern (Info 2)								112
15									
16									
17									
18	IV	NT	SB	BL	-	-	-	OV	
19	Res	Res	Res	LO	IV	SY	DS	TI	
20	S7 Date and Time								
21									
22									
23									
24									
25									
26									
27									
28	32 Bit Bit pattern (Info …)								224
…									
…									
…									
…	IV	NT	SB	BL	-	-	-	OV	
…	Res	Res	Res	LO	IV	SY	DS	TI	
…	S7 Date and Time								
…									
…									
…									
…									
…									
…									

Resulting structure:
14 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚BSI' and ‚QDS' from the IEC message.
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image has so the adjoining structure.

In the IEC output format the block sets ‘substitutional’ the contained NT-Bit in case of appearing connection disturbances. As an option also the value of the information is affected depending on parameter ‘Subst_on_Error’.

Compared with ‘Image_Type’ 5 the image is extended with the status byte ‘Ext_State’ and ‘Date and Time’ in S7 format.

Detailed information to ‘Ext_State’ and ‘S7 Date and Time’ see chapter 6.4.3.1.

6.4.3.5. Measured values - MAo_ME_NA_IM_pDB (FC144)

MAo_ME_NA_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain measured values in the normalized format and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

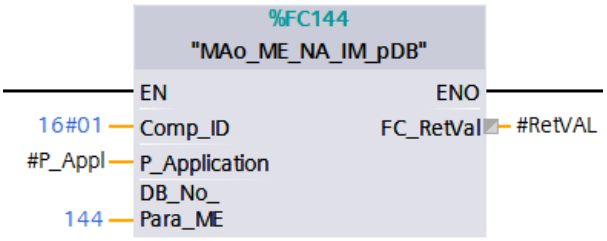


Illustration as FUP block with default values

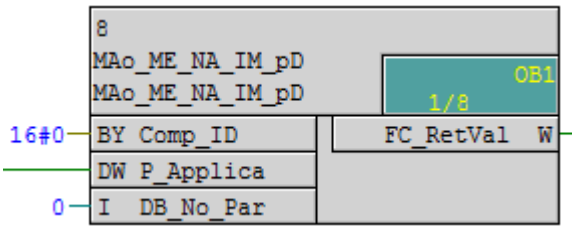


Illustration as CFC block with default values

Important features are:

- The block processes measured value messages with normalized value with TI9, 10 or 34.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of info points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value
  - IEC- Format
  - IEC- Format and time stamp
- For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).

    An EXCEL based help tool for a fast production of parameter-DBs (AWL sources) particularly for larger number of data points and integration of external data sources (Excel lists) is available on inquiry.
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC144		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_ME	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed info

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_ME

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:
0: ok
1-n: number of information processed in the current cycle
7001h partner disturbance coming (Error_Link)

Negative FC_RetVal values:
8112h: P_Application - fault
8302h unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block, can be found in the main chapter. Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1: Measured value 2 Bytes, 2: IEC, 3 IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#10	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors.
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated)

Image_Type/ Image_Len

In the current version, the block supports 2 versions of the measured value output with the following dependencies of 'Image_Len' to 'Image_Type':

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value output 2 Byte measurand (Integer)	16 (B#16#10)	X
2 (B#16#2)	IEC- Output 2 Byte measurand (Integer) + 1 Byte Status	24 (B#16#18)	
3 (B#16#3)	IEC-Format + Ext_State + Time	96 (B#16#60)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
Permissible values are -32767 to +32767.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type, 'D_Address' must be a multiple of 8 (byte limits have to be observed).

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measured value (Info 1) S7 integer								0
1									
2	Measured value (Info 2) S7 integer								16
3									
4	Measured value (Info 3) S7 integer								32
5									
…	…								...
…									
32	Measured value (Info 17) S7 integer								256
33									
…	…								...
…									

2-byte output:

Resultant structure:
2 bytes per information

Only the value of the normalized measured values contained in the IEC message (15 bit + VZ) are read and written in S7 integer format to the relevant bit position (Parameter 'D_Address').

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 16 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No').

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measured value (Info 1) S7 integer								0
1									
2	IV	NT	SB	BL	-	-	-	OV	
3	Measured value (Info 2) S7 integer								24
4									
5	IV	NT	SB	BL	-	-	-	OV	
6	Measured value (Info 3) S7 integer								48
7									
8	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…	IV	NT	SB	BL	-	-	-	OV	
48	Measured value (Info 17) S7 integer								384
49									
50	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…	IV	NT	SB	BL	-	-	-	OV	

3 Byte IEC output:

Resultant structure:
3 bytes per information

In this mode, the output is carried out by the direct transfer of the entire information 'NVA' and 'QDS' from the IEC message.
The image therefore has the structure shown opposite.

The value of the normalized measured values contained in the IEC message (15 bit + VZ) is stored in the image in S7 integer format.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the value of the information is also affected depending on the parameter 'Subst_on_Error'.

Image_Type 3 (B#16#03):

3 Byte IEC-Output
+ 1 Byte Ext_State
+ 8 Byte Time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measurand (Info 1) S7-Integer								0
1									
2	IV	NT	SB	BL	-	-	-	OV	
3	Res	Res	Res	LO	IV	SY	DS	TI	
4	S7 Date and Time								
5									
6									
7									
8									
9									
10									
11									
12	Measurand (Info 2) S7-Integer								144
13									
14	IV	NT	SB	BL	-	-	-	OV	
15	Res	Res	Res	LO	IV	SY	DS	TI	
16	S7 Date and Time								
17									
18									
19									
20									
21									
22									
23									
24	Measurand (Info 3) S7-Integer								288
…									
…	IV	NT	SB	BL	-	-	-	OV	
…	Res	Res	Res	LO	IV	SY	DS	TI	
…	S7 Date and Time								
…									
…									
…									
…									
…									
…									

Resulting structure:
12 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚NVA' and ‚QDS' from the IEC message.
The value of the standardized measurand (15 Bit + VZ) from the IEC message is stored in the image in S7-Integer Format
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image has so the adjoining structure.

In the IEC output format the block sets ‘substitutional’ the contained NT-Bit in case of appearing connection disturbances. As an option also the value of the information is affected depending on parameter ‘Subst_on_Error’.

Compared with ‘Image_Type’ 2 the image is extended with the status byte ‘Ext_State’ and ‘Date and Time’ in S7 format.

Detailed information to ‘Ext_State’ and ‘S7 Date and Time’ see chapter 6.4.3.1.

6.4.3.6. Measured values - MAo_ME_NB_IM_pDB (FC145)

MAo_ME_NB_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain measured values in the scaled format and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

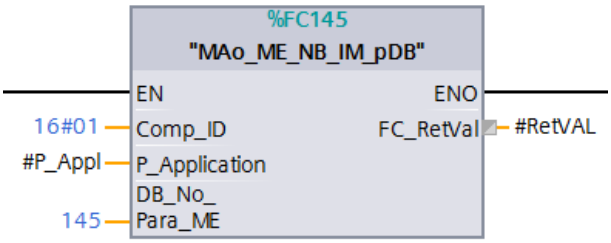


Illustration as FUP block with default values

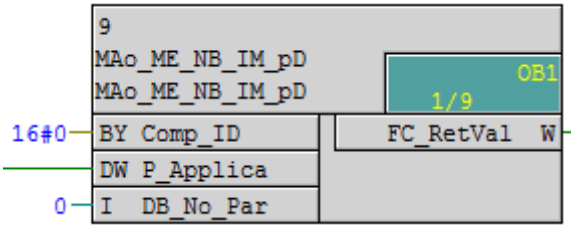


Illustration as CFC block with default values

Important features are:

- The block processes measured value messages with scaled value with TI11, 12 or 35.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of information points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value
  - IEC- Format
  - IEC- Format and time stamp
- For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC145		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_ME	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed information

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_ME

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:

0:	ok
1-n:	number of information processed in the current cycle
7001h	partner disturbance coming (Error_Link)

Negative FC_RetVal values:

8112h:	P_Application - fault
8302h	unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block, can be found in the main chapter. Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1: Measured value 2 Bytes, 2: IEC, 3 IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#10	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors.
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated)

Image_Type/ Image_Len

In the current version, the block supports 2 versions of the measured value output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value output 2 Byte Measurand (Integer)	16 (B#16#10)	X
2 (B#16#2)	IEC- Output 2 Byte Measurand (Integer) + 1 Byte Status	24 (B#16#18)	
3 (B#16#3)	IEC-Format + Ext_State + Time	96 (B#16#60)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
Permissible values are -32767 to +32767.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type, 'D_Address' must be a multiple of 8 (byte limits have to be observed).

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measured value (Info 1) S7 integer								0
1									
2	Measured value (Info 2) S7 integer								16
3									
4	Measured value (Info 3) S7 integer								32
5									
…	…								...
…									
32	Measured value (Info 17) S7 integer								256
33									
…	…								...
…									

2-byte output:

Resultant structure:
2 bytes per information

Only the value of the scaled measured values contained in the IEC message (15 bit + VZ) are read and written to the relevant bit position (Parameter 'D_Address') in S7 integer format.

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 16 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No').

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measured value (Info 1) S7 integer								0
1									
2	IV	NT	SB	BL	-	-	-	OV	
3	Measured value (Info 2) S7 integer								24
4									
5	IV	NT	SB	BL	-	-	-	OV	
6	Measured value (Info 3) S7 integer								48
7									
8	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…	IV	NT	SB	BL	-	-	-	OV	
48	Measured value (Info 17) S7 integer								384
49									
50	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…	IV	NT	SB	BL	-	-	-	OV	

3 Byte IEC output:

Resultant structure:
3 bytes per information

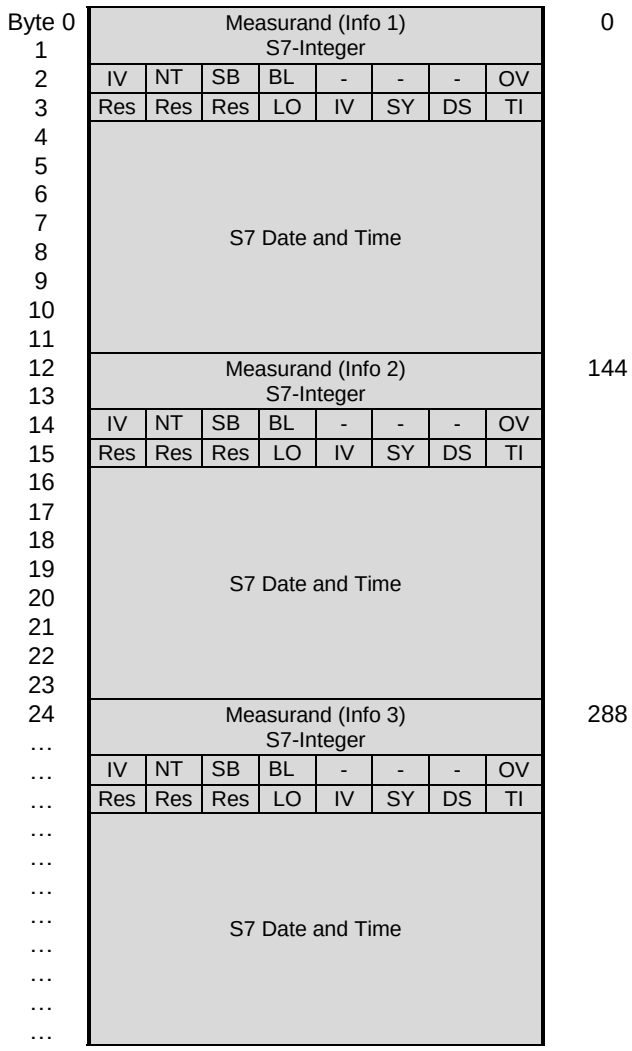
In this mode, the output is carried out by the direct transfer of the entire information 'SVA' and 'QDS' from the IEC message.
The image therefore has the structure shown opposite.

The value of the scaled measured values contained in the IEC message (15 bit + VZ) is stored in the image in S7 integer format.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the value of the information is also affected depending on the parameter 'Subst_on_Error'.

Image_Type 3 (B#16#03):

3 Byte IEC-Output
+ 1 Byte Ext_State
+ 8 Byte Time



Resulting structure:
12 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚SVA' and ‚QDS' from the IEC message.
The value of the scaled measurand (15 Bit + VZ) from the IEC message is stored in the image in S7-Integer Format
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image structure is shown in the adjoining figure.

In the IEC output format the block sets 'substitutional' the contained NT-Bit in case of appearing connection disturbances. As an option also the value of the information is affected depending on parameter ‚Subst_on_Error'.

Compared with 'Image_Type' 2 the image is extended with the status byte 'Ext_State' and 'Date and Time' in S7 format.

Detailed information to 'Ext_State' and 'S7 Date and Time' see chapter 6.4.3.1.

6.4.3.7. Measured values - MAo_ME_NC_IM_pDB (FC146)

MAo_ME_NC_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain measured values in short floating point number format and transfers the useful data into images (DB and PAA) according to the assigned parameter data block.

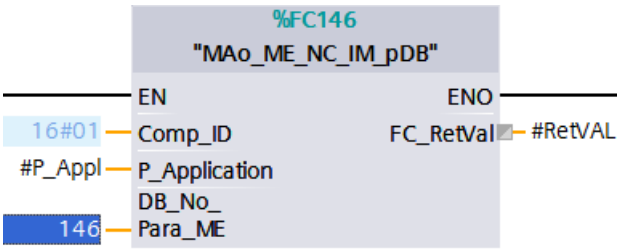


Illustration as FUP block with default values

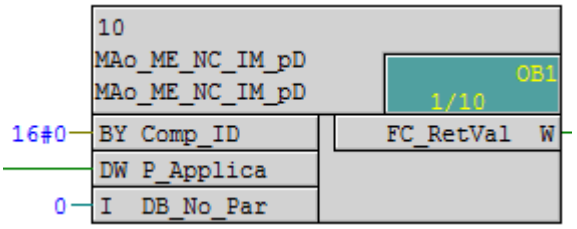


Illustration as CFC block with default values

Important features are:

- The block processes measured value messages with short floating point number with TI13, 14 or 36.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of info points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value
  - IEC- Format
  - IEC- Format and time stamp
  - For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the NT bit
  - in case of link failure to the IEC partner
  - in case of NT bit set (not topical) in received message
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC146		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_SP	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed info

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_SP

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:
0: ok
1-n: number of information processed in the current cycle
7001h partner disturbance coming (Error_Link)

Negative FC_RetVal values:
8112h: P_Application - fault
8302h unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block, can be found in the main chapter. Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1: Measured value 4 Bytes, 2: IEC 5 Bytes, 3 IEC+Ext_State+Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#20	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors?
Global. **Subst_Value**	6.0	26.0	REAL	0.0e+0	Substitute value (if activated)

Image_Type/ Image_Len

In the current version, the block supports 2 versions of the measured value output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value output 4 Byte Meas. (Short Real)	32 (B#16#20)	X
2 (B#16#2)	IEC- Output 4 Byte Meas. (Short Real) + 1 Byte Status	40 (B#16#28)	
3 (B#16#3)	IEC-Format + Ext_State + Time	112 (B#16#70)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

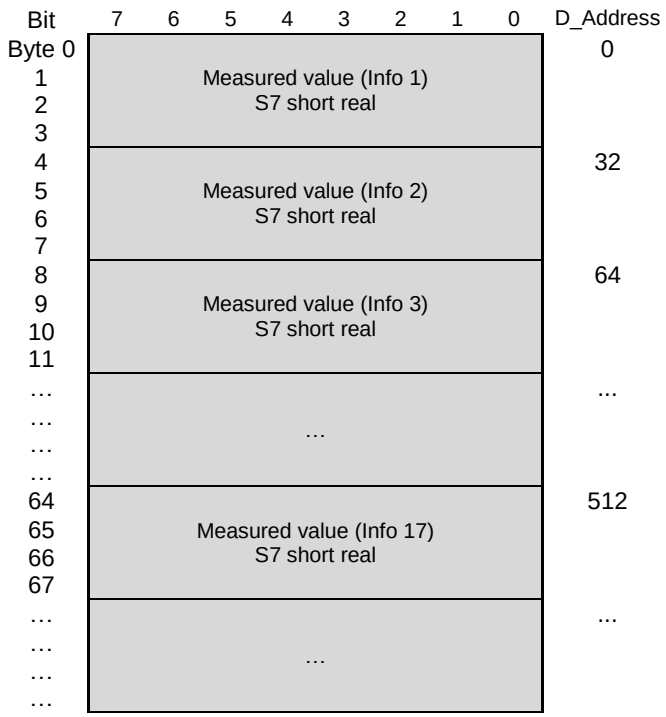
The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
Permissible values are any floating point values in the format x.y.
(e.g.: 0.0, -1234.5678, 9.876543 …)

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type, 'D_Address' must be a multiple of 8 (byte limits have to be observed).

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):



4-byte output:

Resultant structure:
4 bytes per info

Only the value of the measured values contained in the IEC message (4 byte, short floating point number) are read and written in S7 short real format to the relevant bit position (Parameter 'D_Address').

In combination with sequential use (Parameter 'P_No_of_fol_Infos') and 16 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No').

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measured value (Info 1) S7 short real								0
1									
2									
3									
4	IV	NT	SB	BL	-	-	-	OV	
5	Measured value (Info 2) S7 short real								40
6									
7									
8									
9	IV	NT	SB	BL	-	-	-	OV	
10	Measured value (Info 3) S7 short real								80
11									
12									
13									
14	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…									
…									
…	IV	NT	SB	BL	-	-	-	OV	
80	Measured value (Info 17) S7 short real								640
81									
82									
83									
84	IV	NT	SB	BL	-	-	-	OV	
…	…								...
…									
…									
…									
…	IV	NT	SB	BL	-	-	-	OV	

5 Byte IEC output:

Resultant structure:
5 bytes per information

In this mode, the output is carried out by the direct transfer of the entire information 'IEEE STD 754' and 'QDS' from the IEC message.
The image therefore has the structure shown opposite.

The value of the measured values contained in the IEC message (4 byte, short floating point number) is stored in the image in S7 short real format.

In IEC output format, the block sets the contained NT bit 'representative' in the case of incoming link errors. As an option, the value of the information is also affected depending on the parameter 'Subst_on_Error'.

Image_Type 3 (B#16#03): 5 Byte IEC-Output

+ 1 Byte Ext_State
+ 8 Byte Time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	Measurand (Info 1) S7-Short-Real								0
1									
2									
3									
4	IV	NT	SB	BL	-	-	-	OV	
5	Res	Res	Res	LO	IV	SY	DS	TI	
6	S7 Date and Time								
7									
8									
9									
10									
11									
12									
13									
14	Measurand (Info 2) S7-Short-Real								40
15									
16									
17									
18	IV	NT	SB	BL	-	-	-	OV	
19	Res	Res	Res	LO	IV	SY	DS	TI	
20	S7 Date and Time								
21									
22									
23									
24									
25									
26									
27									
28	Measurand (Info 3) S7-Short-Real								80
…									
…									
…									
…	IV	NT	SB	BL	-	-	-	OV	
…	Res	Res	Res	LO	IV	SY	DS	TI	
…	S7 Date and Time								
…									
…									
…									
…									
…									
…									

Resulting structure:
14 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚IEEE STD 754' and ‚QDS' from the IEC message.
The value of the measurand (4 Byte shortened floating point) from the IEC message is stored in the image in S7-Short-Real Format
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image has so the adjoining structure.

In the IEC output format the block sets 'substitutional' the contained NT-Bit in case of appearing connection disturbances. As an option also the value of the information is affected depending on parameter 'Subst_on_Error'.

Compared with 'Image_Type' 2 the image is extended with the status byte 'Ext_State' and 'Date and Time' in S7 format.

Detailed information to 'Ext_State' and 'S7 Date and Time' see chapter 6.4.3.1.

6.4.3.8. Integrated totals - MAo_IT_IM_pDB (FC147)

MAo_IT_IM_pDB is a function (FC) which monitors receive messages from the IEC link whether they contain integrated totals and transfers the useful data into images (DB or PAA) according to the assigned parameter data block.

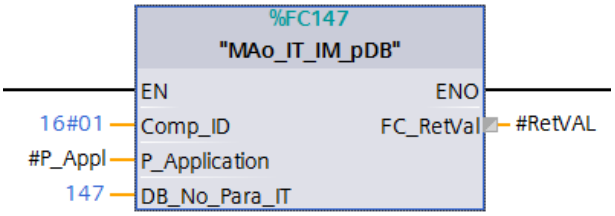


Illustration as FUP block with default values

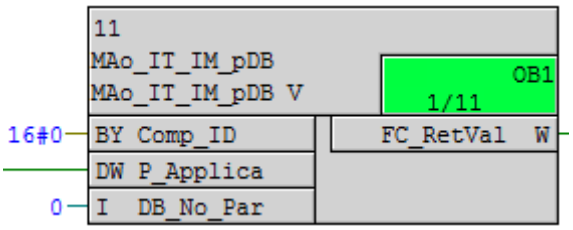


Illustration as CFC block with default values

Important features are:

- The block processes count value messages with TI15, 16 or 37.
- The FC can be used in CFC plans as well as in KOP/FUP/AWL with minimal interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The routing of the info points to the required image positions is carried out in a data block and is…
  - exceptionally effective with sequential IEC address ranges for a large number of information points each with just one parameter entry
  - individually possible for each IEC address
- Multiple output - image versions can be parameterized (in parameter data block):
  - Raw value
  - IEC- Format
  - IEC- Format and time stamp
- For details see subsequent description
- Output direct to PAA or into data block
- Integrated error processing with programmable substitute value and influence on the IV bit
  - in case of link failure to the IEC partner
  - in case of IV bit (invalid) in received message
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FC is supplied as well in the block library and can be modified immediately (Simatic Manager).
- From V1.5 the block only accepts the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal exit. FC_RetVal also shows how many information objects have been processed from the block in the current call.

Block variables of the FC with their default assignment and brief comment

FC147		Type		Block comment/brief description
Comp_ID	IN	BYTE		Identifier for Diagnostic-Functions
P_Application	IN	DWORD		Pointer which must be connected to 'S7_IEC_Config' – Function Block
DB_No_Para_IT	IN	INT		Number of Data Block (DB) which contains the parameters for this function
FC_RetVal	OUT	WORD		Neg. values: Error-Codes; Pos. values: No. of processed info

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output of the block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

DB_No_Para_IT

DB no. of the associated parameter data block.

FC_RetVal

Positive FC_RetVal values:
0: ok
1-n: number of information processed in the current cycle
7001h partner disturbance coming (Error_Link)

Negative FC_RetVal values:
8112h: P_Application - fault
8302h unknown cause of transmission (COT) received

Block-specific values in the parameter data block:

The entire structure, as well as the basic description of the parameter data block, can be found in the main chapter. Only the type-specific details are considered at this point:

Global. **Image_Type**	2.0	22.0	BYTE	B#16#1	1: Counter Value (4 Bytes), 2: IEC, 3: IEC + Ext_State + Time
Global. **Image_Len**	3.0	23.0	BYTE	B#16#20	0: auto (standard); <> 0: fixed distance [Bit] of process images
Global. **Subst_on_Error**	4.0	24.0	BOOL	TRUE	Substitute values in case of connection errors?
Global. **Subst_Value**	6.0	26.0	DWORD	DW#16#0	Substitute value (if activated)

Image_Type/ Image_Len

In the current version the block supports 2 versions of the message output with the following dependencies of Image_Len to Image_Type:

Image_Type	Signification	Image_Len	Default
1 (B#16#1)	Raw value output 4 Byte integrated total	32 (B#16#20)	X
2 (B#16#2)	IEC- Output	40 (B#16#28)	
3 (B#16#3)	IEC-Format + Ext_State + Time	112 (B#16#70)	

From V1.1 'Image_Len' is pre-allocated with 0 (automatic calculation in dependence of 'Image_Type'). Adaption is only necessary if a specific other (larger) image length shall be created.
For more details, refer to the pages below.

Subst_on_Error/ Subst_Value

The substitute value (Subst_Value) in case of a link error is only relevant if 'Subst_on_Error' is set.
Permissible values are 0 to 2^{32}-1.

Para_[x]. **D_Address**	+10.0	44.0	DINT	L#0	Dest. (target): Bit position

D_Address Target position: The image in the data block or PAA starts at this bit position. Depending on Image_Type, 'D_Address' must be a multiple of 8 (byte limits have to be observed).

The image versions (Image_Type) in detail

Image_Type 1 (B#16#01):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0									0
1	32-bit integrated total (Info 1)								
2									
3									
4									32
5	32-bit integrated total (Info 2)								
6									
7									
8									64
9	32-bit integrated total (Info 3)								
10									
11									
…									...
…	…								
…									
…									
64									512
65	32-bit integrated total (Info 17)								
66									
67									
…									...
…	…								
…									
…									

4-byte output:

Resultant structure:
4 bytes per information

The integrated total contained in the IEC message (32 bit) is read and written to the relevant bit position (Parameter 'D_Address').

In combination with the sequential use (Parameter 'P_No_of_fol_Infos') and 32 bit spacing (Image_Len), connected ranges are produced as shown in the diagram. The output range can be in the PAA as well as in data block(s) (Parameter 'D_DB_No')

Image_Type 2 (B#16#02):

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	32-bit integrated total (Info 1)								0
1									
2									
3									
4	IV	CA	CY	Sequence number					
5	32-bit integrated total (Info 2)								40
6									
7									
8									
9	IV	CA	CY	Sequence number					
10	32-bit integrated total (Info 3)								80
11									
12									
13									
14	IV	CA	CY	Sequence number					
…	…								...
…									
…									
…									
…	IV	CA	CY	Sequence number					
80	32-bit integrated total (Info 17)								640
81									
82									
83									
84	IV	CA	CY	Sequence number					
…	…								...
…									
…									
…									
…	IV	CA	CY	Sequence number					

5 Byte IEC output:

Resultant structure:
5 bytes per information

In this mode, the output is carried out by the direct transfer of the entire information 'BCR' from the IEC message.
The image therefore has the structure shown adjoining.

Image_Type 3 (B#16#03): 5 Byte IEC-Output

+ 1 Byte Ext_State
+ 8 Byte Time

Bit	7	6	5	4	3	2	1	0	D_Address
Byte 0	32 Bit integrated totals (Info 1)								0
1									
2									
3									
4	IV	CA	CY	Sequence number					
5	Res	Res	Res	LO	IV	SY	DS	TI	
6	S7 Date and Time								
7									
8									
9									
10									
11									
12									
13									
14	32 Bit integrated totals (Info 2)								40
15									
16									
17									
18	IV	CA	CY	Sequence number					
19	Res	Res	Res	LO	IV	SY	DS	TI	
20	S7 Date and Time								
21									
22									
23									
24									
25									
26									
27									
28	32 Bit integrated totals (Info 3)								80
…									
…									
…									
…	IV	CA	CY	Sequence number					
…	Res	Res	Res	LO	IV	SY	DS	TI	
…	S7 Date and Time								
…									
…									
…									
…									
…									
…									
…									

Resulting structure:
14 byte per information

In this mode the output is carried out with direct transfer of the complete information ‚BCR' from the IEC message.
Subsequently
1 byte Ext_State and
8 byte S7 Date and time

The image has the structure shown adjoining.

Compared with 'Image_Type' 2 the image is extended with the status byte 'Ext_State' and 'Date and Time' in S7 format.

Detailed information to 'Ext_State' and 'S7 Date and Time' see chapter 6.4.3.1.

6.4.4. Input blocks - MAi_xyz_pDB

For the various information types in the control direction, input blocks (FBs) are available which have to be 'interconnected' via the pointer 'P_Application' with the relevant protocol version supplied. The blocks monitor the control bits assigned to them and generate the relevant IEC command messages.

The 'routing' from input point to the resultant IEC command message, as well as the transfer of basic settings/parameters, is carried out (if parameterization is possible) in an allocated parameter data block.

The symbol name encrypts the most important block properties as follows:

MAi_SC_DC_RC_pDB	**MAi** is the abbreviation for master blocks input. The block evaluates the entered information (from the PAE or DB) and generates the respective command messages.
MAi_**SC_DC_RC**_pDB	Information types supported by the block with the brief designation according to IEC standard: **SC** = Single Command **DC** = Double Command **RC** = Regulation Command
MAi_SC_DC_RC_**p**DB	**p** is the abbreviation for programmable allocation of input point and IEC command address.
MAi_SC_DC_RC_p**DB**	The number of administered information points of the block and the parameters for the individual information are stored in a **DB**. An existing number shows the maximum number of information points of the block which can be administered.

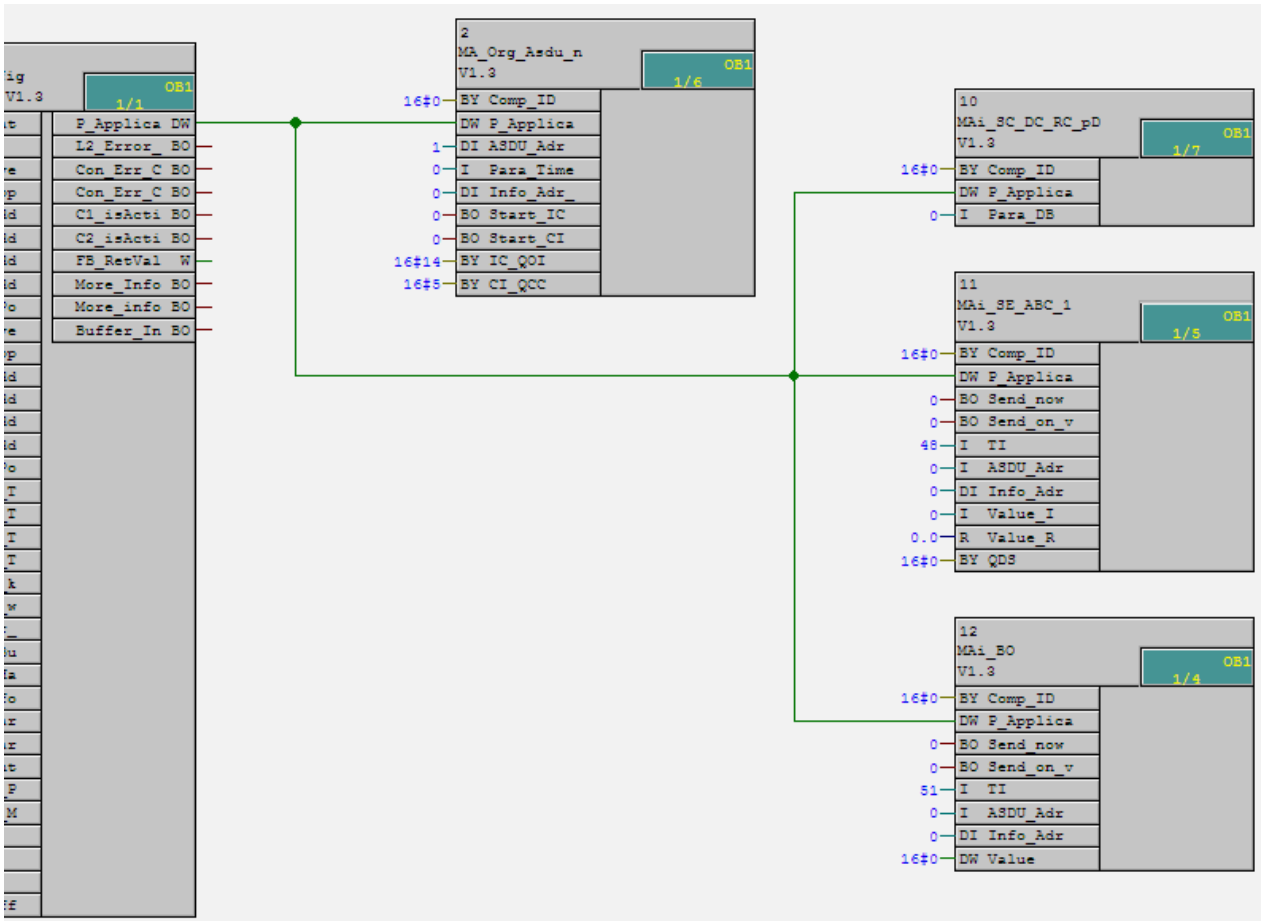
The available MAi function blocks can be found in the overview below.

Block	Block No.	Para DB no. (Default)	Generation of …
MAi_SC_DC_RC_pDB	FB148	DB148	… Single commands (SC; TI45, 58) … Double commands (DC; TI46, 59) … Regulation commands (RC; TI47, 60)
MAi_SE_ABC_1	FB149	n. v.	… Analogue setpoint value commands with normalized values (SE_A; TI48, 61) scaled values (SE_B; TI49, 62) floating point values (SE_C; TI50, 63)
MAi_BO	FB150	n. v.	… digital setpoint values, bit pattern (BO; TI51, 64)

The block MAi_SC_DC_RC_pDB is currently the only MAi block which is parameterized via a DB. The entire description of this parameter DB can be found in the corresponding block specific chapter.

The basic structure of the parameter DB largely corresponds to the DB for the MAo blocks.

Overview of the available MAi blocks – for a better overview in a CFC plan:



6.4.4.1. Single, double and regulating step commands - MAi_SC_DC_RC_pDB (FB148)

MAi_SC_DC_RC_pDB is a block (FB) which monitors control bits (in a DB) for a rising edge according to the allocated parameter data block. Depending on this control bits IEC commands are generated.

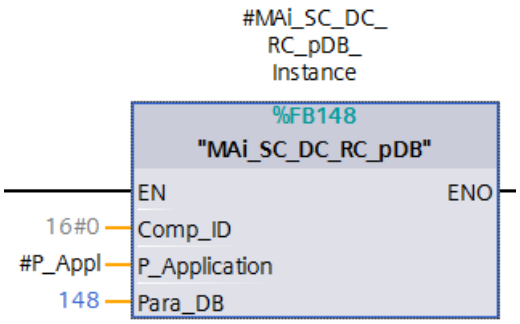


Illustration as FUP block with default values

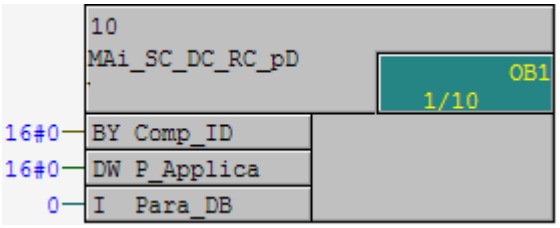


Illustration as CFC block with default values

Important features are:

- With a rising edge of the control bit, the block generates commands with
  TI45, 46 or 47 (no time mark)
  TI58, 59 or 60 (with time mark)
- The FB can be used in CFC plans as well as in KOP/FUP/AWL with a small amount of interconnection work. The allocation of parameter DB and pointer to IEC link (P_Application) are sufficient.
- The specification and allocation of the control bits to the required IEC commands is carried out in a data block
- The properties of the commands are defined individually for each command in the parameter data block
- Simple handling of the parameter data blocks:
  - A default DB with the same number as FB is supplied in the block library and can be modified directly (Simatic Manager).
- -Sequential processing of the commands
  the block issues only one command at the same time (1 out of n check across all commands assigned to the block).

Block variables of the FB with their default assignment and brief comment

FB148		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	B#16#0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	DW#16#0	Pointer which must be connected to 'S7_IEC_Config' – Function Block
Para_DB	IN	INT	0	Number of Data Block (DB) which contains the parameters for this function

Meaning of the block variables:

Comp_ID — Block identifier for (currently only internal) diagnostic functions. Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application — Pointer which **has to be** connected to the 'P_Application' output on block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

Para_DB — DB number of the associated parameter data block.

Structure and values of the parameter data block:

The parameter DB has the following structure:

Group/ Parameter	Addr. Rel.	Addr. sect.	Type	Start value	Comment
DB_Manag	0	0	STRUCT		
DB_Manag. **Pos_of_DB**	+0.0	0.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **No_of_all_DBs**	+1.0	1.0	BYTE	B#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Act**	+2.0	2.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Prev**	+4.0	4.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **DB_No_Next**	+6.0	6.0	WORD	W#16#0	reserved for future - do not change!
DB_Manag. **Reserved**	+8.0	8.0	WORD	W#16#0	reserved for future - do not change!
	+10.0	10.0	END_STRUCT		END_STRUCT (10 BYTE)
Info_Manag			STRUCT		
Info_Manag. **P_Byte_First_Info**	+0.0	10.0	INT	20	(Byte-) Position of first Para-Block - do not change!
Info_Manag. **No_of_Infos**	+2.0	12.0	INT	0	0:unspecified (DB is filled up to the end); otherwise size n of ARRAY [1..n]
Info_Manag. **Len_Info**	+4.0	14.0	BYTE	B#16#18	Difference in [byte] between two Para-Blocks - do not change!
Info_Manag. **Len_Sort_Key**	+5.0	15.0	BYTE	B#16#0	0: without sorting; > 0: Data sets are sorted ascending with x Bytes
DB_Manag. **Re_internal_usage**	+6.0	16.0	DWORD	DW#16#0	
	+10.0	20.0	END_STRUCT		END_STRUCT (10 BYTE)
Para_CMD			ARRAY [1..n] STRUCT		n = Number Parameter inputs
Para_[x]. **S_DB_No**	+0.0	20.0	Word	W#16#0	Source: No. of DB
Para_[x]. **S_Adress**	+2.0	22.0	DWORD	DW#16#0	Source: Bit position
Para_[x]. **D_Line**	+6.0	26.0	BYTE	B#16#1	Dest.: No. of Line (1..x)-actually do not change from 1
Para_[x]. **D_ASDU**	+8.0	28.0	WORD	W#16#0	Dest.: Address of ASDU (ASDU-Address)
Para_[x]. **D_Info**	+10.0	30.0	DWORD	DW#16#0	Dest.: Address of Infoobject (IOA)
Para_[x]. **Timeout**	+14.0	34.0	BYTE	B#16#14	Timeout [sec] beginning with entry into send buffer...
Para_[x]. **TI**	+15.0	35.0	BYTE	B#16#2D	Type identification 45(2Dhex=SC); 46(2E=DC); 47(2F=RC);

 ©SIEMENS AG 2017

					58, 59, 60 ->with time stamp
Para_[x]. **Value**	+16.0	36.0	DWORD	DW#16#1	SCO/DCO/RCO-Byte, including Value-Bits (SCS/DCS/RCS), Qualifier (QU) and S/E-Bit
Para_[x]. **Termination_Type**	+20.0	40.0	BYTE	B#16#1	how a command will terminated (0:with sending;1:with recv.conf, 2:with recv.term
Para_[x]. **CMD_Info_Count**	+21.0	41.0	BYTE	B#16#0	Only for reading: will incremented with each started command
Para_[x]. **CMD_Info_State**	+22.0	42.0	WORD	B#16#0	Only for reading: shows details of the last command process
	24.0	44.0	END_STRUCT		END_STRUCT (24 BYTE)

	These values are preset in the default DBs and must not be changed. The default DBs are included in the relevant block libraries.
	!!! These parameters you have to set !!!
	Note: Always make changes in the 'Data view', not in the 'Declaration view' (except dimensioning of the array)

DB_Management

Parameter blocks which include an enumeration are created so, that more than one data block can be used if required. The following information is required for management.
Currently the DB management is not used – for this reason all parameters can be left at their default settings (0)

Pos_of_DB
Number (1 – n) of the current data block for the parameter block
0: not used

No_of_all_DBs
Total number (n) of data blocks for the parameter block
0: not used

DB_No_Act
DB number of the current data block
0: not used

DB_No_Prev
DB number of the previous data block, if one exists.
Otherwise: 0

DB_No_Next
DB number of the subsequent data block, if there is another one.
Otherwise: 0

Info_Management

Parameter blocks containing an enumeration have one information block 'Info management'. It contains the following information
The values of the information management parameters are already preset and therefore do not need to be modified.

P_Byte_First_Info
Byte position of the first data record in this DB

No_of_Infos
Number of data records contained in this DB
0: unspecified -> the data block end marks the last data record

Len_Info
Length of the data records:
0: not permitted
1-254: data record length
255: no standard length, length specifications in the byte format by sort criterion

Len_Sort_Key
0: no sorting (default setting)
Other sort lengths are not useful here because the parameter is currently only used by the MAo blocks.

Para_CMD
Para per info (group)

Create any number of parameter blocks by setting the array value 'n' accordingly in the declaration view. In the data view the parameters can then be modified individually.
One parameter block should be created here for each required command.
The allocation of control bit to command is carried out statically, as well as the complete description of the relevant command with all its properties.

S_DB_No

Number of the source DB containing the control bit required for this command.
It is essential that the control bits are present in a DB as they are reset by the command block after the command is completed.
Use of markers, PAE, etc. is not permissible.

S_Adress

Source position of the control bit:
Bit position of the control bit in the data block.
A rising edge on the control bit activates the command.

D_Line

Target number of the communication line:
Reserved for future applications.
Must currently be set/remain on 1

D_ASDU

ASDU address used for the command output.

D_Info

Information object address (IOA) of the generated command message.

Timeout

Monitoring time in seconds.
The time runs from the entry in the send buffer.
After the monitoring time elapses, the command is completed in every case.

TI

Type identification of the command message being generated.
Permissible values are:
45 (B#16#2D) for single commands (SC) without time stamp
46 (B#16#2E) for double commands (DC) without time stamp
47 (B#16#2F) for regulating step commands (RC) without time stamp
58 (B#16#3A) for single commands (SC) with time stamp
59 (B#16#3B) for double commands (DC) with time stamp
60 (B#16#3C) for regulating step commands (RC) with time stamp

Value

Command byte:
Depending on 'TI', the required command byte (SCO/DCO/RCO) is given here, including command bits (SCS/DCS/RCS), command identifier (QU) and S/E bit

Bit	7	6	5	4	3	2	1	0	TI
Byte	S/E	QU					0	SCS	45/58
Byte	S/E	QU					DCS		46/59
Byte	S/E	QU					RCS		47/60

Structure of the command byte depending on 'TI':

The value stored here is transferred directly into the command byte.

Termination_Type

Command termination:
0: with sending
1: with receipt of the confirmation of the activation
2: with receipt of the termination of the activation

CMD_Info_Count

Command counter (for diagnostics):
The command counter is increased by 1 with each command started.
→ read access only

CMD_Info_State

Command status (for diagnostics):
This value provides information on the status of the last processed command.
- > read access only

The individual command status entries are added.
With the start of the command (entry in the message buffer), the status is set to 1 and updated after the command is terminated. The passed status conditions from the command are then available for evaluation.

Overview of possible values (with decimal illustration):
> 127: The command is terminated successfully (according to 'Termination_Type')
< 0 (bit 2^{15} set = 8xxx hex): Error during command output

Detailed information can be found in the list below.

Bit 2^0	Command in the message memory
Bit 2^1	Not used
Bit 2^2	Not used
Bit 2^3	Positive confirmation of the activation received
Bit 2^4	Positive termination of the activation received
Bit 2^5	Not used
Bit 2^6	Not used
Bit 2^7	Accumulative-Bit: BFE positive: The command has been terminated according ‘Termination_Type‘, successfully
Bit 2^8	Transmission fault. Possible reasons are: - fault in the subordinated Station - Number of repetitions at Timeout reached (Link layer)
Bit 2^9	Negative confirmation of the activation received
Bit 2^{10}	Negative termination of the activation received
Bit 2^{11}	Timeout
Bit 2^{12}	Not used
Bit 2^{13}	Not used
Bit 2^{14}	Not used
Bit 2^{15}	Accumulative-Bit: BFE negative: Command output fault occurred.

Command prioritization

The block executes the present command jobs sequential.
In case of multiple command trigger bits are set at the same time, these are processed in the sequence as written in the parameter DB.
This permits the prioritization of the commands.
Higher priority commands have to be in front of lower priority commands in the parameter DB.

Command termination

The respectively next command is only started after a running command is terminated.
The termination of the running command occurs
in the normal situation as specified for 'Termination_Type', or
in the event of an error, at the latest after expiry of the time span according to 'Timeout'

Command processing with link error

After a link error to the (IEC) partner disappears, the send buffer of the IEC master is cleared. This prevents that commands which have been started during a detected link error are output delayed.
If applicable the relevant commands have to be started again.

6.4.4.2. Setpoint commands - MAi_SE_ABC_1 (FB149)

MAi_SE_ABC_1 is a block (FB) which generates IEC setpoint commands.

The command is started due to a rising edge on the allocated trigger bit (PAE, DB; M) and/or due to a change in the setpoint value.

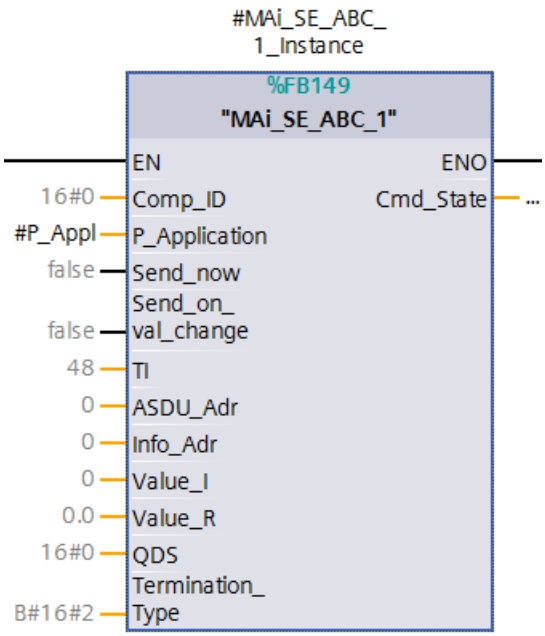


Illustration as FUP block with default values

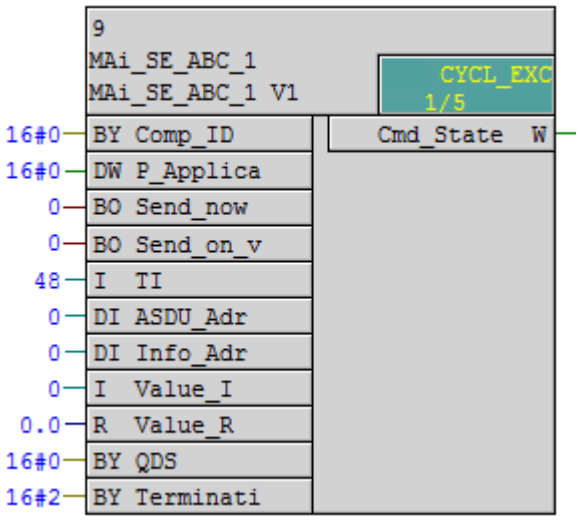


Illustration as CFC block with default values

Important features are:

- The block generates setpoint commands with
  TI48, 49 or 50 (without time mark)
  TI61, 62 or 63 (with time mark)
- The FB can be used in CFC plans as well as in KOP/FUP/AWL with a small amount of interconnection work. The parameterization is carried out directly via interconnection of the block inputs.
- The properties of the setpoint command are defined individually for each command
- Simple handling of the block:
  Complete parameterization of the setpoint commands is carried out directly at the block inputs.

Block variables of the FB with their default assignment and brief comment

FB149		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	0	Pointer which must be connected to 'S7_IEC_Config' – Function Block
Send_now	IN	BOOL	0	A rising edge on this input starts sending a set point command message
Send_on_val_change	IN	BOOL	0	If this input is true, set point cmd will send autom. through each val-change
TI	IN	INT	0	Supported TIs are 48 (SE_NA), 49 (SE_NB), 50 (SE_NC) and with time stamp 61, 62, 63
ASDU_Adr	IN	INT	0	Common Address of ASDU of the set point command
Info_Adr	IN	DINT	0	Information object address (IOA) of the set point command
Value_I	IN	INT	0	Set point value as integer - not used if 0 (default)
Value_R	IN	REAL	0	Set point value as real - not used if 0 (default)
QDS	IN	BYTE	0	QDS (default 0) - Qualifier of setpoint command
Termination_Type	IN	BYTE	B#16#2	how a command will be terminated (0:with sending;1:with recv.conf,2:with recv.term
CMD_State	OUT	WORD	W#16#0	shows details of the last command process

Meaning of the block variables:

Comp_ID

Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application

Pointer which **has to be** connected to the 'P_Application' output on block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

Send_now

Control bit/acceptance bit
A rising edge on this input starts the sending of the setpoint command.
Permissible sources are PAE, DB, marker, and timer.

Send_on_val_change

Send after change
If this input is 'TRUE', the sending of the setpoint command is triggered automatically after the setpoint value is changed ('Value_I' or 'Value_R').

TI

Type identification of the setpoint command being generated.
Permissible values are:
48 for setpoint commands with normalized value (SE_NA) without time stamp
49 for setpoint commands with scaled value (SE_NB) without time stamp
50 for setpoint commands with short floating point number (SE_NC) without time stamp
61 for setpoint commands with normalized value (SE_NA) with time stamp
62 for setpoint commands with scaled value (SE_NB) with time stamp
63 for setpoint commands with short. floating point number (SE_NC) with time stamp

ASDU_Adr

ASDU address used for the command output.

Info_Adr

Information object address (IOA) of the generated command message.

Value_I

Setpoint value (Format: 2 byte, S7 integer):
This value is relevant with 'TI' = 48, 49, 61 or 62. The value entered here is transferred to the setpoint command message.
With 'TI' = 50 or 63, this input is not relevant and has to be set to 0.

Value_R

Setpoint value (Format: 4 byte, S7 short real):
This value is relevant with 'TI' = 50 or 63. The value entered here is transferred into the setpoint command message.
With 'TI' = 48, 49, 61 or 62, this input is not relevant and has to be set to 0.

 ©SIEMENS AG 2017

QDS

Identifier for the setpoint command (QOS):
Enter the required identifier here.

Structure of the identifier for the setpoint command:

Bit	7	6	5	4	3	2	1	0	
Byte	S/E	QL							QOS

Termination_Type

The value stored here is transferred directly into the command message.
Defines when the command is considered as terminated from the view of this block:
0: with the transmission
1: with receive of the activation confirmation
2: with receive of the activation termination
The progress of the command can be read at output ‚CMD_State'.

CMD_State

Command status (for diagnostics)
This value informs about the status of the last processed command.
-> read only access.
For details see chapter 6.4.4.1 Single, double and regulating step commands - MAi_SC_DC_RC_pDB (FB148)

Generation of several different setpoint commands

The block manages precisely the setpoint command defined on its inputs per call-up.
If you require more than one setpoint command for your application, the block needs to be called up multiple times (once for each setpoint command) and the calls of the FB149 are locked against each other.

Setpoint command termination

The respectively next setpoint command is only started after the running setpoint command is terminated.
The termination of the running setpoint command occurs
in the normal situation with the receipt of the 'confirmation of activation', or
in the event of an error, at the latest after 5 seconds (fixed timeout set).

 ©SIEMENS AG 2017

6.4.4.3. Bit pattern commands - MAi_BO (FB150)

MAi_BO is a block (FB) which generates IEC bit pattern commands.

The command is started due to a rising edge on the allocated trigger bit (PAE, DB; M) and/or due to a change in the bit pattern.

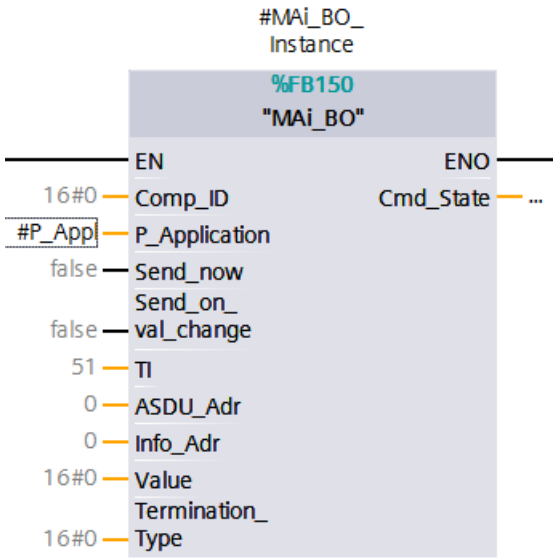


Illustration as FUP block with default values

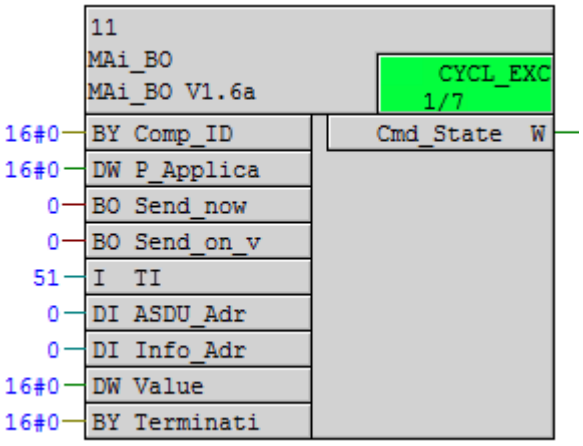


Illustration as CFC block with default values

Important features are:

- The block generates bit pattern commands with
  TI51 (no time marker) or
  TI64 (with time marker)
- The FB can be used in CFC plans as well as in KOP/FUP/AWL with a small amount of interconnection work. The parameterization is carried out directly via interconnection of the block inputs.
- The properties of the bit pattern command are defined individually for each command
- Simple handling of the block:
  Complete parameterization of the bit pattern command is carried out directly on the block inputs.
- Project specific special-purpose solutions can be cost-efficient realized.

Block variables of the FB with their default assignment and brief comment

FB150		Type	Default	Block comment/brief description
Comp_ID	IN	BYTE	0	Identifier for Diagnostic-Functions
P_Application	IN	DWORD	0	Pointer which must be connected to 'S7_IEC_Config' – Function Block
Send_now	IN	BOOL	0	A rising edge on this input starts sending a set point command message
Send_on_val_change	IN	BOOL	0	If this input is true, set point cmd will send autom. through each val-change
TI	IN	INT	0	Supported TIs are 51 (Default) and with time stamp 64
ASDU_Adr	IN	INT	0	Common Address of ASDU of the set point command
Info_Adr	IN	DINT	0	Information object address (IOA) of the set point command
Value	IN	DWORD	0	BSI = binary state information 32 Bit
Termination_Type	IN	BYTE	B#16#0	how a command will be terminated (0:with sending;1:with recv.conf,2:with recv.term
CMD_State	OUT	WORD	W#16#0	shows details of the last command process

Meaning of the block variables:

Comp_ID
Block identifier for (currently only internal) diagnostic functions.
Recommendation: Number each use of this block consecutively starting with 1 (B#16#1).

P_Application
Pointer which **has to be** connected to the 'P_Application' output on block 'S7_IEC_Config' (Communication-FB). It refers to all data required internally such as send and receive buffer, etc.

Send_now
Control bit/acceptance bit
A rising edge on this input starts the sending of the setpoint command.
Permissible sources are PAE, DB, marker, and timer.

Send_on_val_change
Send after change
If this input is 'TRUE', the sending of the bit pattern command message is triggered automatically after the bit pattern ('Value') is changed.

TI
Type identification of the bit pattern command message being generated.
Permissible values are:
51 for bit pattern commands (BO) without time stamp
64 for bit pattern commands (BO) with time stamp

ASDU_Adr
ASDU address used for the command output.

Info_Adr
Information object address (IOA) of the generated command message.

Value
Bit pattern (BSI = 32-bit bit pattern):
The value of the bit pattern being transferred is given here (digital setpoint value).
The value entered here is transferred directly into the bit pattern command message.

Termination_Type
Defines when the command is considered as terminated from the view of this block:
0: with the transmission
1: with receive of the activation confirmation
2: with receive of the activation termination
The progress of the command can be read at output ‚CMD_State'.

CMD_State
Command status (for diagnostics)
This value informs about the status of the last processed command.
-> read only access.
For details see chapter 6.4.4.1 Single, double and regulating step commands - MAi_SC_DC_RC_pDB (FB148)

Generation of several different bit pattern commands

The block manages precisely the bit pattern command defined on its inputs per call-up.
If you require more than one bit pattern command for your application, the block needs to be called up multiple times (once for each bit pattern command) and the calls of the FB150 are locked against each other.

Command termination

The respectively next bit pattern command is only started after the running bit pattern command is terminated.
The termination of the running bit pattern command occurs
in the normal situation with the receipt of the 'confirmation of activation', or
in the event of an error, at the latest after 5 seconds (fixed timeout set).

 ©SIEMENS AG 2017

7. AddOn blocks ‚misc'

There are a number of common auxiliary blocks existing, whose functions are described subsequently.

Block	Short description	Reference
S7_IEC_Buf_Manager FC100	Enables the use of non remanent memory areas for the message storage as well as individual adjustments of the DB length. Not for CPU15xx.	Chapter 8
H_Calc_DST_UTC_Europe FC121	The block monitors the time which is controlled from the CPU, and determines whether daylight saving time or winter season is active.	Chapter 7.2
S7_IEC_Set_Threshold_1 FC125	The block enables the threshold adaption of individual measured values.	Chapter 7.1
DP_Supp_Interm_States FB21	A simple block for intermediate state information suppression in case of double indication.	
S7_IEC_Appl_Manager FB125	Enables the more rarely call up of the time-consuming recording blocks. Thus the (OB1-) cycle time decreases, the Communication-FB is called up more often and the communication is accelerated. The block should be used only project specific and after consultation. Please use preferred the possibility to process SLi blocks in cyclic interrupts which is available from V1.5 (see chapter 6.2).	
S7_IEC_REM_MEM FB126	Only CPU15xx. Prevents the repeated generating of DBs if the instance DB of FB10x is operated in the non remanent area.	Chapter 7.2.4

7.1. AddOn-block S7_IEC_Set_Threshold_1 (FC125)

The block enables the threshold adaption of several measured values. A block output informs about the status of the respective measured value.

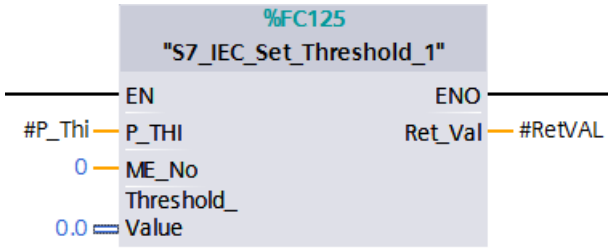


Illustration as FUP block with default values

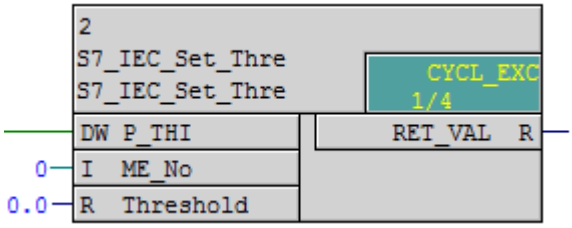


Illustration as CFC block with default values

Block parameters with their default assignment and brief comment

FC125 S7_IEC_Set_Threshold_1		Type	Default	Block comment/brief description
P_THI	IN	DWORD		Pointer which be must connected to Output P_THI of 'SLi_ME' - Function Block
ME_No	IN	INT		0..x: Number of the measured value, whose threshold value shall be changed
Threshold_Value	IN	REAL		Threshold_Value [digits] for the selected measured value
Ret_Val	OUT	REAL		Error: -9999.0; otherwise actual integrated difference

Parameter- and function details:

P_THI
P_THI (**P**ointer **Thr**es**h**old **I**nterface) is a pointer to the data area of the threshold values of the SLi_ME block. Therefore it has to be interconnected with its output P_THI via a DWord variable.

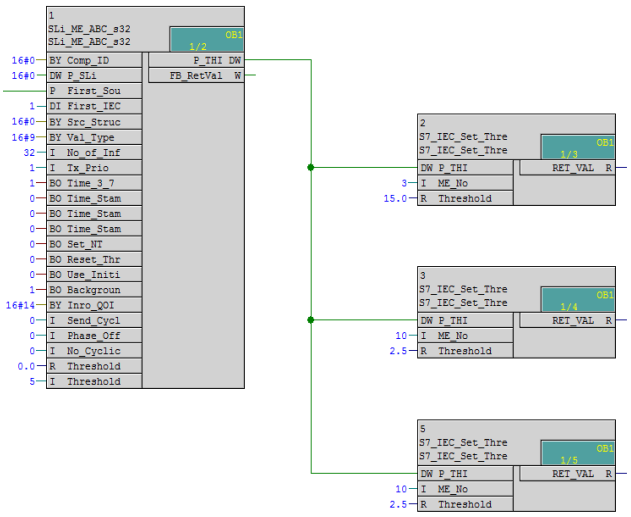
ME_No
Number of the measured value which threshold has to be individual adapted. The counting starts from 0 (first recorded measured value).

Threshold_Value
Desired threshold. A change of the measured value (compared with the last transmission) about this threshold leads to an immediate spontaneous transmission. Smaller changes will be added together (integrated) and lead as well to the measured value transmission if the added value achieves the threshold.

Ret_Val
An invalid pointer P_THI (0) or incorrect measured value number lead to the output -9999.0.
Otherwise the actual added value is output (see parameter Threshold_Value.

Use:

The block is ideally called immediately after the measured values recording block SLi_ME…as often as thresholds have to be adapted.



7.2. Clock synchronization

CPU-clocks can be set by using the following methods.
- IEC-Protocol
- NTP-Protocol
- SIMATIC Mode

7.2.1. Clock synchronization via the IEC-Protocol

The IEC-Protokoll sets the clock to local time via time setting telegram (TK103) inclusive summer time bit and the time_IV will be set, hence the time stamp is valid, so the FC121 is not necessary.

7.2.2. Clock synchronization via the NTP-Protocol and SIMATIC Mode

The NTP-Porotcol and SIMATIC Mode can set the clock to UTC-time.

The FC121 calculates the current time shift from UTC to local time and summer time according to the European specifications
- Start of the summer time: last sunday in march 2:00 MEZ
- End oft the summertime: last sunday in october 3:00 MESZ

The time IV will not be reset, that menas that the time stamp is invalid and the time IV input of the communication FB will be reset. This is also necessary for locations with local time.

```
CALL  "H_Calc_DST_UTC_Europe"
     Diff_LocalTime :=T#1h
     Diff_Dst       :=T#1h
     Time_Diff      :=#time_diff
     DST            :=#dst
SET
R    #T_IV

CALL  "S7_IEC_101_MS_CM_PtP", "S7_IEC_101_MS_CM_PtP_DB"
     Time_DS                  :=#dst
     Time_IV                  :=#T_IV
     Time_SY                  :=
     Time_Diff                :=#time_diff
```

FC121 H_Calc_DST_UTC_Europe		Typ	Default	Block comment / brief description
Diff_LocalTime	IN	Time		Differenc between UTC and Localtime = time zone (e.g. UTC+1h Berlin)
Diff_DST	IN	Time		Differenc between winter time to summer time (normally +1h)
Time_Diff	OUT	Time		Please connect to input Time_Diff of S7_IEC_Config function block
DST	OUT	Bool		Signals daylight saving time (DST European summer time) Please connect to input Time_DS of S7_IEC_Config function block

Parameter- und Funktionsdetails:

Diff_LocalTime	Diff_LocalTime is the difference between UTC time and local time (e.g. UTC+1h Berlin)
Diff_DST	Difference between winter time and summer time (normal +1h)
Time_Diff	Current time difference between UTC and local time. Please connect to the input time_diff of the communication FB.
DST	Displays the summer time. Please connect with the inpuit time DS of the Communication-FB.

7.2.3. Time in the S7-1500

In addition to the system time, the S7-1500 runs the local time with summer time

- „Use_CPU_Local_Time“ = False
  The function described in chapter 7.2.1 is used
- „Use_CPU_Local_Time“ = True
- The local time works worldwide.

Notes:

The IEC protocol represents the local time, including summer time, and the time_IV will be reset. The time stamp is valid, so the **FC121** is not necessary and the call **must be deleted.**

The default is **not** compatible with the previous versions.

7.3. Auxiliary block S7_IEC_REM_MEM (FB126) for S7-15xx

The AddOn block 'S7_IEC_REM_MEM' (FB126) should be called with its own instance DB (less memory consumption and automatically always remanent) always immediately after the communication-FB. Connected via P_Application the FB takes over the storage of the book-keeping of the instance DBs of the communication FBs.

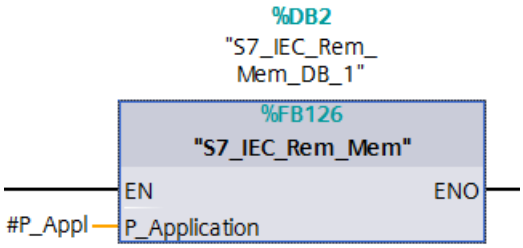


Illustration as FUP block with default values

Important features are:

- connected via P_Application, the FB takes over the storage of the bookkeeping of the instance DBs of the communication FBs.
- should always be called directly after the communication FB with its own instance DB

Block parameters with their default assignment and short comment

FB126		Typ	Default	Block comment / brief description
P_Application	IN	DWORD	DW#16#0	Pointer which must be connected to 'S7_IEC_Config' – Module

Meaning of the block variables:

P_Application Pointer, which **must be** connected to the output, P_Application 'of the block' S7_IEC_Config '(communication FB). It refers to all internally necessary data, such as send and receive buffers, etc.

8. Using the NON_Retain Memory only S7-300

With the introduction of the NON_Retain memory the user has the possibility to gain additional data memory by using the (none buffered) working storage of the CPU. This NON_Retain memory can be used for buffering of data in SIPLUS RIC IEConS7.

Until now data blocks are processed in the remanence storage of the CPU. This applies for offline-DBs as well as for DBs generated by system functional block.

Notes:

Non remanent DBs are supported from **S7-300 CPUs from V2.6** as well as from **ET 200S CPUs.**

In principle also instance DBs can be set to non remanent except the instance DB of the Communication-FB. Otherwise new buffer data blocks will be created after each voltage drop.

In the technical data of the S7-300 and ET 200S CPUs the working storage integrated and the size of the remanent memory is mentioned. The remanent memory is always a subset of the working storage.

Note:

S7-400 CPUs and WinAC differentiate only between working storage code and working storage data (both are battery buffered/remanent in the S7-400). This means data blocks are generated resp. can be stored only in the associated memory range.

Up to now send buffer DBs are generated exclusively during run time (online, in the remanent data memory of the CPU).

In order to use additional non remanent working storage of the S7-300 resp. ET 200S CPUs in case of high demand of buffer please generate first of all data blocks with defined size and type of array „NON-Retain“ offline in your project.

The procedure is as follows:

- Generate a data block (with New Project Insertion/Data Block) in the project
- Open the data block and declare the desired length. Example: Array [1…10000] Word. The amount of DB length is then 20000 Bytes. A structuring or initialization is not necessary.
- Open the DB’s object properties (context menu/Object Properties) and activate (in the index card General – Part 2) the checkbox ‚NON-Retain’

For embedding of the so generated ‘NON-Retain DB’ the FC100 block is required which is working as S7_IEC_Buffer_Manager. For further details see chapter 8.1.1. Extend your program after the call of the Communication-FB with a call of the FC100 and interconnect it with P_Application from Communication-FB.

Note:

For embedding of several buffer DBs for each NON_Retain DB the call of the FC100 is required!

The inputs of the FC100 are used as follows:

- P_Application: Interconnection with Communication-FB
- DB_Type: The NON_Retain DB has to be assigned to the desired DB type specified here (see also FC100 block comment).
- DB_No: DB-Number of the offline generated NON-Retain DB
- DB_Len: The length will be assigned automatically. In this case 0 has to be entered here.

8.1.1. AddOn-block S7_IEC_Buf_Manager (FC100)

This block is available with version V1.3, from V1.5 as AddOn-Block.

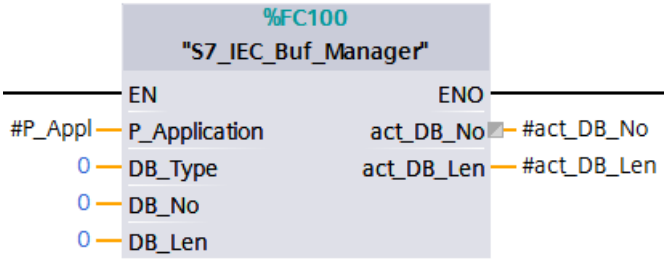


Illustration as FUP block with default values

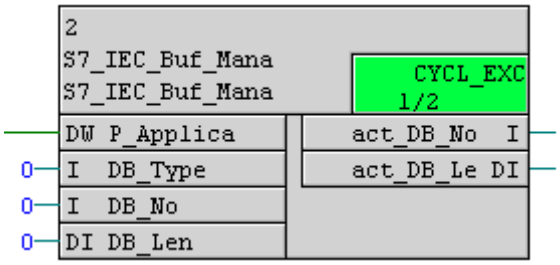


Illustration as CFC block with default values

The FC100 block can be used for the following tasks:

- Use of an offline generated DB (also NON-Retain) as buffer-DB (see chapter)
- Adaption of the DB-length from automatically generated DBs
- Readout of DB number and DB length from internal buffer DBs

Adaption of the DB-length from automatically generated DBs:

Up to version V1.2 the number of send buffers was definitely specified, and the change of size only restrained. With V1.3 or higher the user can specify number and size via Communication-FB parameter. In this case the indication of size always refers to all send buffers which have to be generated with the exception of the always generated DBs for diagnostics and organizational messages.

In order to generate several of the automatically created DBs with individual length, the FC100 (S7_IEC_Buf_Manager) now can be used as well. For this purpose extend your program after the call of the Communication-FB with a call of the FC100 and interconnect it with P_Application from Communication-FB and the following parameters:

- DB_Type: according to the desired DB-Type
- DB_No: Has to be 0 here
- DB_Len: Desired DB length
- interconnect act_DB_No and act_DB_Len with arbitrary (Temp-) variables.

Note:

In case of embedding the block or changing parameters during operation the values are taken over directly and a restart of the IEC application is carried out. During start up the previous DB is deleted and created new. This may also be with a new DB number.

Read out of DB-Number and DB_Length of internal Buffer-DB

With FC100 (S7_IEC_Buf_Manager) the actual DB-numbers and DB-length from an internal buffer DB can be read. For this purpose extend your program after the call of the Communication-FB with a call of the FC100 and interconnect it with P_Application from Communication-FB and the following parameters:

- DB_Type: according to the desired DB-Type
- DB_No.: Has to be 0 in case of ‚read only'
- DB_Len: Has to ne 0 in case of ‚read only
- interconnect act_DB_No and act_DB_Len with arbitrary (Temp-) variables. This now contains the number and length of the requested DB. The length is only valid if the DB number is <> 0. The response of the DB number as well is only correct if the requested buffer is created or existing (depending on the parameter 'No_of_Send_Buffer').

Block parameter with its block comments

FC100		Type	Block - Comment
P_Application	IN	DWORD	Pointer which must be connected to 'S7_IEC_Config'
DB_Type	IN	INT	0..15: Transmission buffer Prio 0..15, 100: Diag_Buffer, 101: Org_Buffer
DB_No	IN	INT	<> 0: number of an offline DB to be assigned; 0: DB will be created auto.
DB_Len	IN	DINT	<> 0: individual (instead of default) length for autom. created DB at runtime
act_DB_No	OUT	INT	
act_DB_Len	OUT	DINT	

Parameter and functional details:

P_Application

Pointer which **must be** connected to the 'P_Application' output of the 'S7_IEC_Config' (Communication-FB) block. It refers to all internal required data like send and receive buffer etc.

DB_Type

according to the desired DB-Type:
0…15: Send buffer Priority 0…15;
100: Diagnostic buffer
101: Buffer for organizational messages
104: Assignment of the T104 acknowledgment buffer from V1.6 Upd4
When the function is used, the FB10x input must be L2_Ack_Buf_Dim '0.

DB_No

DB number if an offline DB has to be assigned, otherwise 0

DB_Len

0: If an offline DB has to be assigned (the length is automatically calculated)
<>0: Length of the automatically during run time created DB type specified from user

act_DB_No

The number of the requested DB is written here. The response of the DB number as well is only correct if the requested buffer is created or existing (depending on the parameter 'No_of_Send_Buffer').

act_DB_Len

The number of the requested DB is written here. The length is only valid if the DB number is <> 0.

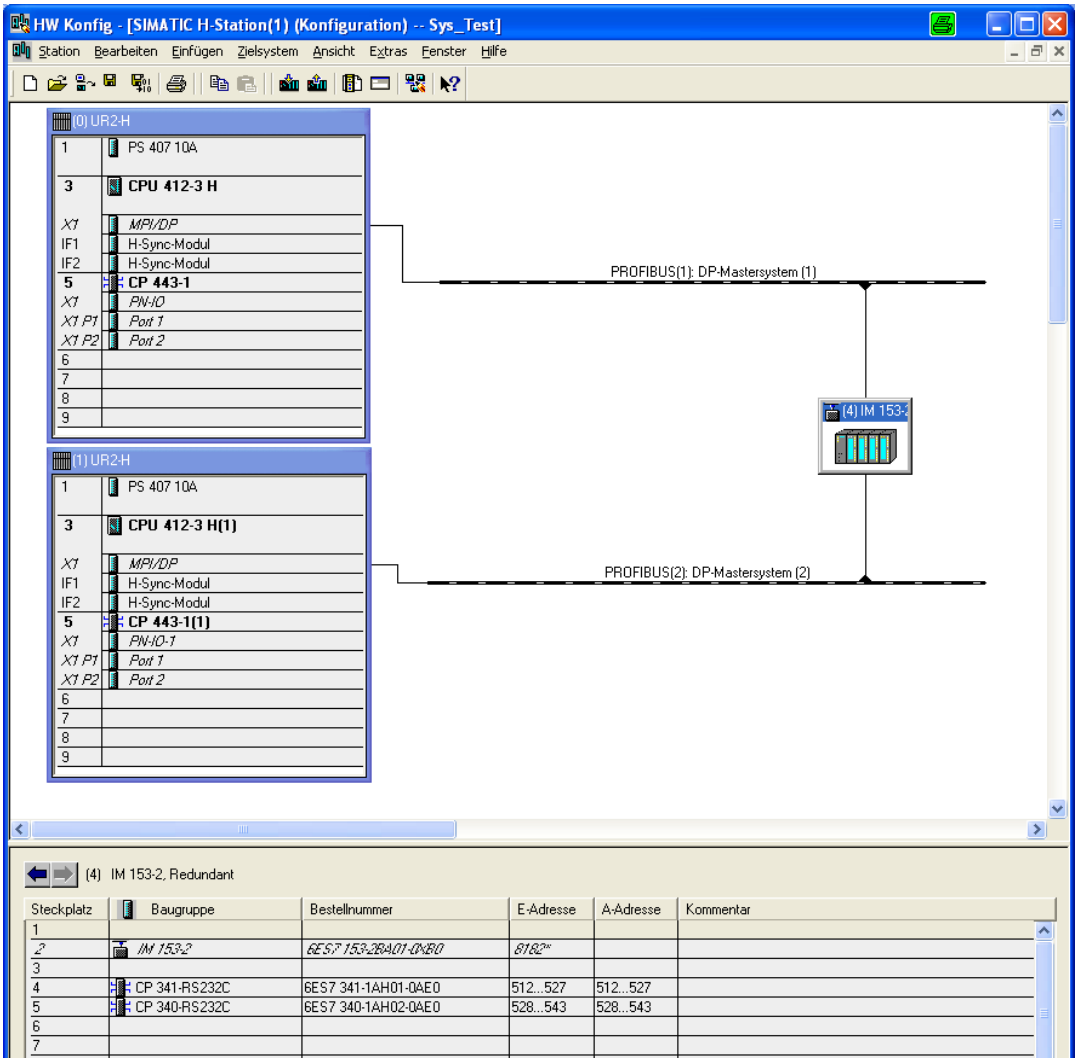
 ©SIEMENS AG 2017

9. SIPLUS RIC IEConS7 in SIMATIC H-Systems

The blocks from SIPLUS RIC IEConS7 can be inserted directly in SIMATIC H-Systems.

Subsequently we introduce some typical configuration examples for the telecontrol protocols IEC 60870-5-101 (serial) resp. IEC 60870-5-104 (TCP/IP).

9.1. Configurations



The figure shows a possible hardware configuration. In this configuration the assembly of the following IEC communication links would be possible:

- Protocol IEC 101 Master or Slave, via CP340 in ET 200M- Rack
- Protocol IEC 101 Master or Slave, via CP341 in ET 200M- Rack
- Protocol IEC 104 Master or Slave, via the two CP443-1 in the S7-400H Racks

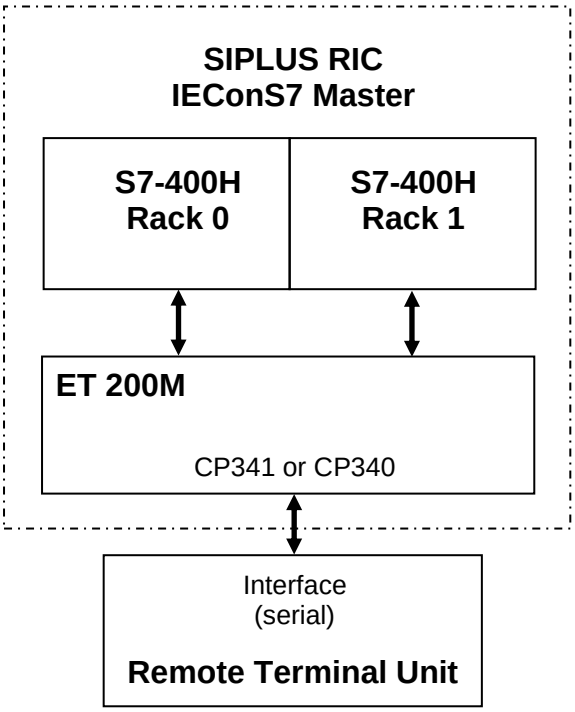
Depending on the protocol variant which has to be carried out, only a part of the hardware is required.

9.1.1. IEC 60870-5-101 Master

Configuration:
SIPLUS RIC IEConS7 is working as Master.
The connection of a RTU (Remote Terminal Unit) is carried out serial via a CP340 or CP341 in ET 200M-Rack.

Normal operation:
The process leading CPU (Rack) in the H-System communicates via the CP340/341 with the RTU.

Breakdown in the process leading Rack:
The CPU in the reserve rack takes over the data traffic via the CP340/341.
The RTU can't and has not to discover the switch over. No loss of data occurs.



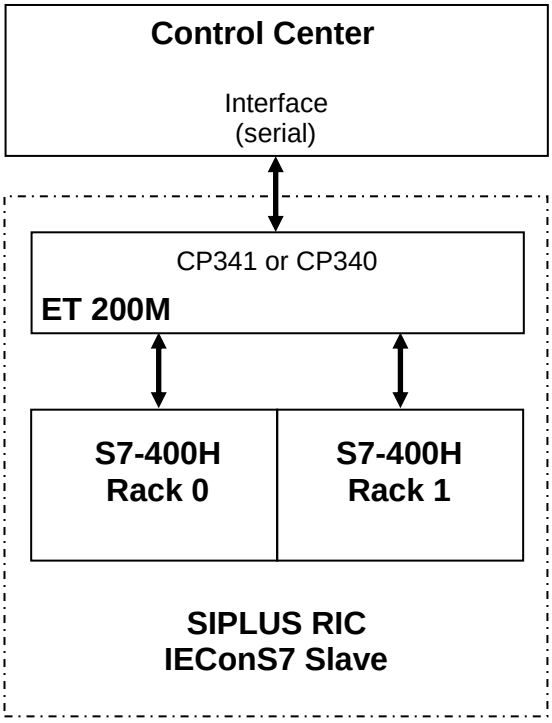
The projection of the CP340/341 is carried out like described in chapter 3.2

9.1.2. IEC 60870-5-101 Slave

Configuration:
SIPLUS RIC IEConS7 is working as Slave.
The connection to the control center is carried out serial via a CP340 or CP341 in ET 200M-Rack.

Normal operation:
The control center communicates via the CP340/341 with the process leading CPU (rack) in the H-System.

Breakdown in the process leading Rack:
The CPU in the reserve rack takes over the data traffic via the CP340/341. The control center can't discover the switch over. The breakdown of the rack can be reported via indications to the control center.



The projection of the CP340/341 is carried out like described in chapter 3.2

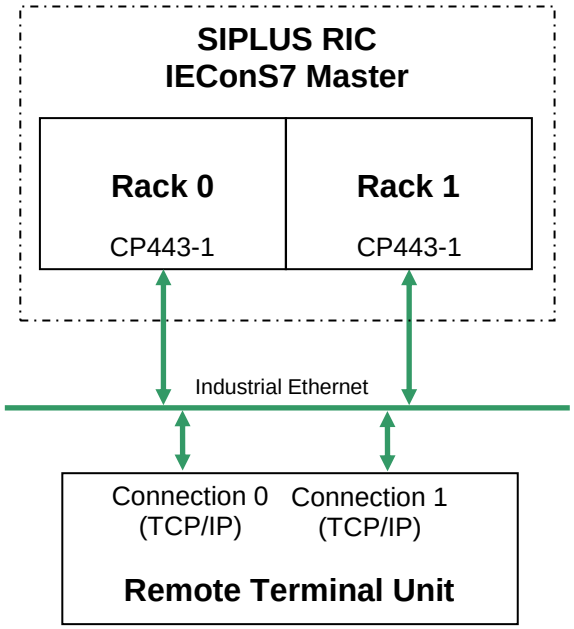
9.1.3. IEC 60870-5-104 Master

Configuration:
SIPLUS RIC IEConS7 is working as Master.
The connection of the RTU via a network is carried out redundant via respectively one CP443-1 in Rack 0 and Rack 1.

Normal operation:
The process leading CPU (Rack) in the H-System establishes via the two CPs respectively one TCP/IP connection to the RTU. It starts the IEC-Protocol via one of the two connections and monitors the second connection using test frames.
This is called in the norm redundancy group with 2 connections.

Breakdown of the active connection:
SIPLUS RIC IEConS7 discovers the up to this time active connection as disturbed and starts the protocol via the other connection.

If the behavior of the RTU is conforming to standards the non confirmed messages are repeated and no loss of data occurs.



The projection of the CP443-1 is carried out like described in chapter 4.2.

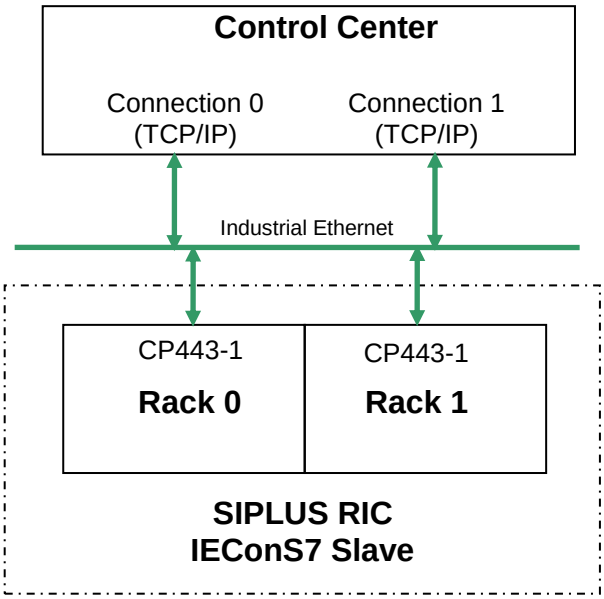
Alternatively the two connections can be established also via two separate LANs.

9.1.4. IEC 60870-5-104 Slave

Configuration:
The control center has a network connection.
SIPLUS RIC IEConS7 is working as Slave.
The connection to the network is carried out redundant via respectively one CP443-1 in Rack 0 and Rack 1

Normal operation:
The control center establishes respectively one TCP/IP connection to the two CPs and with it to the process leading CPU.
It starts the IEC-Protocol via one of the two connections and monitors the second connection using test frames.
This is called in the norm redundancy group with 2 connections.

Breakdown of the active connection:
The control center discovers the up to this time active connection as disturbed and starts the protocol via the other connection.
SIPLUS RIC IEConS7 repeats first the non confirmed messages, so that no loss of data occurs in the control center.
Subsequently possible buffered messages are sent.



The projection of the CP443-1 is carried out like described in chapter 4.2.

Alternatively the two connections can be established also via two separate networks.

9.1.5. H-CPUs V6 – Use of the integrated PN-Interface

From V1.4 SIPLUS RIC supports the use of the PN-Interfaces integrated on the V6-H-CPUs for IEC 60870-104 communication.

The applicable variants are:

- **S7IEC_S104_41xPN**
- **S7IEC_M104_41xPN**

In this case the first connection (C1) communicates always via the PN-Interface of the CPU in RACK 0 resp. solo systems.

For the second connection (C2) it is possible to switch over to the PN-Interface of the CPU in RACK 1 by parameterization (Parameter „L1_C2_over_CPU_Rack_1“).

You can carry out special configurations (e.g. several connections via the CPU in RACK 1 or mixed operation with CP-Communication) by using the possibilities of the multiple channel/redundancy (ref. to chapter 5).

Note:

Please read up on the SIMATIC documentation about possible limitations when using simultaneously the PN-Interfaces for IEC-Communication (free TCP-Communication) and coupling to local peripherals.

9.2. Licensing in the H-System

Up to 2012 available CPU for H-Systems were equipped with Firmware version V4.

For its licensing the serial number of the memory card can't be used. Instead of that the serial number of the SIMATIC H-CPUs must be taken for the generating of the registration codes. Please activate this by parameter input 'Serial_from_CPU'=True

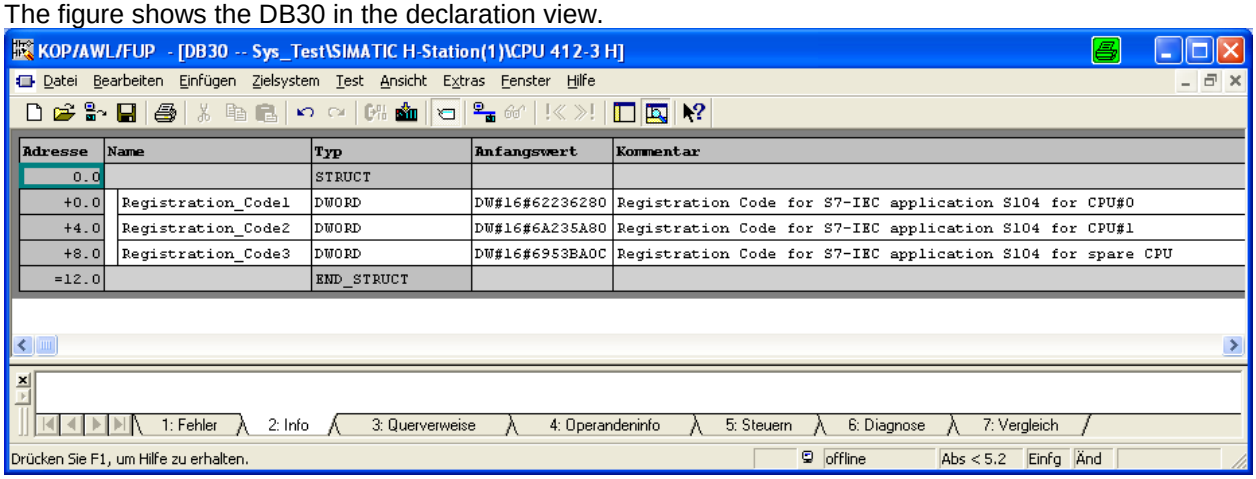
CPUs for H-Systems with the release V5 serial numbers of the memory card can also be used. Please activate this by means of parameter input, serial from CPU'=False.

Each SIMATIC H-System consists of 2 identical sub systems with respectively own SIMATIC H-CPU (Master/Reserve). For both SIMATIC H-CPU's an own activating code is necessary.

This registration codes have to be entered in a data block whose number has to be entered at the Communication-FB input 'Registration_Code'. For that the DB number has to be converted into a hexadecimal syntax.

Example with DB30 as data block containing the registration codes:

The figure shows the DB30 in the declaration view.



Call of the Communication-FB (S7_IEC_Config):

```
CALL  "S7_IEC_Config" , "S7_IEC"
 Registration_Code       :=DW#16#1E    //Reg.-Codes in DB30 (1E Hex)
 Line_ID                 :=
```

NOTICE

Exchange of H-CPU!

H-Systems are normally used in equipment where high availability is requested. In order to allow a change of module without PG in case of service (exchange of H-CPU) it is strongly recommended to get licensing also for a possible existing spare part CPU.

The necessary serial number for that you'll find in case of the H-CPU also marked on the front panel (SVP….). It is not mandatory to read out the serial number in the normal operation (refer also to chapter 12.2).

10. SIPLUS RIC under TIA Step7

10.1. Library concept

Under TIA Step7 the SIPLUS RIC functionality is available as 'Family-Library'. Per device family (ET 200S, S7-300, S7-400, WinAC, S7-15xx) the available protocols (T101, T103, T104) are bundled for all supported interfaces in each case in one library. The following libraries are available:

- IEConS7 ET 200S_Vx_y
- IEConS7 S7-300_Vx_y
- IEConS7 S7-400_Vx_y
- IEConWinAC_Vx_y
- IEConS7-S7-15xx_Vx_y (S7-1500 and ET 200SP)

Compared to the single libraries in the SIMATIC manager where the Communication-FB (S7_IEC_Config) is always the same it has now for each variant an explicit FB-number and symbolic name. This is assembled out of protocol and interface e.g. FB210 = S7IEC_S104_41xPN. With it in one library all SIMATIC blocks are unique.

Within the library the SIMATIC blocks are functional subdivided in groups. As user you have to take over either all (high amount of resources) or only the necessary groups into your project.

10.1.1. Procedure for using the libraries in TIA

The libraries are available unpacked in each case and in archived form. After opening as global library you'll find the block groups under copy templates:

Group	Contents	Action
S7IEC_M… S7IEC_S…	Contain the basic block for the desired protocol (earlier Communication-FB, S7_IEC_Config) and in each case a corresponding instance data block.	Depending on the required protocol (if needed several too) select the group(s) and transfer/copy it into your project. -> (Program blocks)
S7IEC_internal	Contains (auxiliary) blocks which are always required	Always has to be transferred into your project.
S7IEC_Driver…	Contains the blocks for communication via several interfaces	Please select according to the subsequent matrixes
S7IEC_Appl_M…	Contains the typical Master application blocks MAi.., MAo,,,,	Copy only if required
S7IEC_Appl_S…	Contains the typical Slave application blocks SLi…, SLo…	Copy only if required
S7IEC_AddOn…	Contains special AddOn blocks	Copy only if required
VAT_S7_IEC…	Available once per basic block (resp. its instance-DB)	Transfer it analogically to the basic block(s) into your project. -> (Watch and force tables)

 ©SIEMENS AG 2017

Required groups for S7-1500/ET 200SP

	S7IEC_101_MS_CM_PtP	S7IEC_103_M_CM_PtP	S7IEC_104_MS_CPU_PN
S7IEC_Internal	X	X	X
S7IEC_Appl_M101_104	X1		X1
S7IEC_Appl_M103		X	
S7IEC_Appl_S101_104	X1		X1
S7IEC_Driver_PN-interface			X
S7IEC_Driver_PtP-Interface	X	X	

X1: depending on traffic mode select Master or Slave

Required groups for ET 200S

	S7IEC_M101_CP1SI	S7IEC_M103_CP1SI	S7IEC_M104_151PN	S7IEC_S101_CP1SI	S7IEC_S104_151PN
S7IEC_Internal	X	X	X	X	X
S7IEC_Appl_M_101_104	X		X		
S7IEC_Appl_M_103		X			
S7IEC_Appl_S_101_104				X	X
S7IEC_Driver_PN-Interface			X		X
S7IEC_Driver_1SI	X	X		X	

Required groups for S7-300

	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M104_31xPN	S7IEC_M104_CP343	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S104_31xPN	S7IEC_S104_CP343
S7IEC_Internal	X	X	X	X	X	X	X	X	X	X
S7IEC_Appl_M_101_104	X	X			X	X				
S7IEC_Appl_M_103			X	X						
S7IEC_Appl_S_101_104							X	X	X	X
S7IEC_Driver_PN-Interface					X				X	
S7IEC_Driver_CP343						X				X
S7IEC_Driver_CP340	X		X				X			
S7IEC_Driver_CP341		X		X				X		

 ©SIEMENS AG 2017

Required groups for S7-400

	S7IEC_M101_CP441	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M103_CP441	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M104_41xPN	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_S101_CP441	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S104_41xPN	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp
S7IEC_Internal	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
S7IEC_Appl_M_101_104	X	X	X				X	X	X						
S7IEC_Appl_M_103				X	X	X									
S7IEC_Appl_S_101_104										X	X	X	X	X	X
S7IEC_Driver_PN-Interface							X						X		
S7IEC_Driver_CP443								X	X					X	X
S7IEC_Driver_CP441	X			X						X					
S7IEC_Driver_CP340		X			X						X				
S7IEC_Driver_CP341			X			X						X			

Required groups for WinAC

	S7IEC_M101_CP1SI	S7IEC_M103_CP1SI	S7IEC_M104_151PN	S7IEC_S101_CP1SI	S7IEC_S104_151PN
S7IEC_Internal	X	X	X	X	X
S7IEC_Appl_M_101_104	X		X		
S7IEC_Appl_M_103		X			
S7IEC_Appl_S_101_104				X	X
S7IEC_Driver_PN-Interface_WinAC			X		X
S7IEC_Driver_1SI_WinAC	X	X		X	

10.1.2. Memory improvement

In case of memory congestion you can try to gain memory by cleanup of not necessary blocks in your project. This is unconsidered possible:

For all not required (not used by your) application blocks of the groups S7IEC_Appl_M… and S7IEC_Appl_S…

Non used AddOn blocks

Data blocks which are provided in the libraries but not used in your project

10.1.3. First Steps

1. Transfer the required groups/blocks from the library into your project (minimum basic block, the appropriate driver group S7IEC_Driver_xy and S7IEC_internal, the suitable variables table)
2. Call up the basic block from process level OB1 and assign the instance-DB enclosed as well (an automatically produced instance-DB would be possible as well)
3. Supply the block with the most important parameters

   connection adjustments (IDs, IPs, port numbers….)

   protocol adjustments

   at T104 protocol e.g. as Slave via PN-interfaces a passive connection set-up to port 2404 operates without any adjustment

The basically telecontrol communication is already now available (with suitable hardware loaded)
By means of a protocol test equipment or another remote station a communication should be possible. Check this with the attached table of variables. The purpose is 'L2_Error_Link = FALSE. FB_RetVal' should be "0" or the still running demo time should show max. 900 sec = 15 min.

The basic block receives already application messages (except M103-communication). In case of messages with confirmation responsibility in control direction (e.g. general interrogation, switching command,..) the message is send back mirrored with cause of transmission (44, 45, 46 or 47).

4. Establish in each case a temporary DWORD variable named P_Appl and P_SLi (global variables in data blocks or markers are possible as well) and assign P_Appl to the output P_Application.

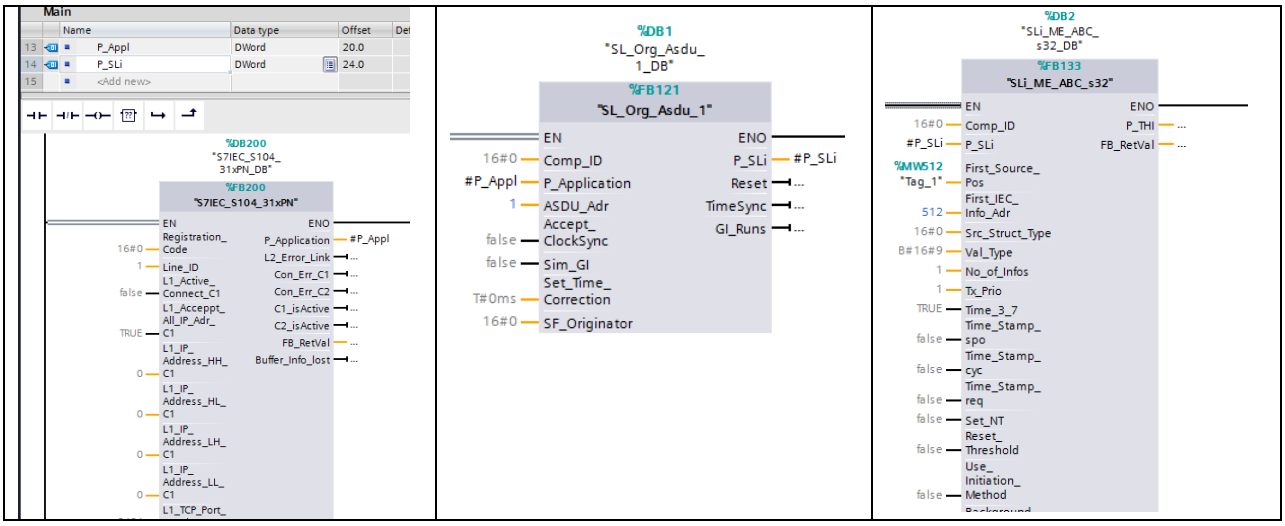
The further process depends on the desired functionality. Subsequently in this abstract is described how a measured value is recorded which is located in MW512.

5. In OB1 call up the block 'SL_Org_ASDU_1' from the group 'S7IEC_Appl_S_101_104' (create the instance-DB assign). Minimum an ASDU-address has to be indicated at the block, the input P_Application has to be interconnected with the variable P_Appl and the provided variable P_SLi has to be interconnected with the output P_SLi.
6. Call up the block 'SLi_ME_ABC_s32' from the group 'S7IEC_Appl_S_101_104' and finish (minimum) the following adjustments:
   Interconnection input 'P_SLi' with variable P_SLi
   Input 'First_Source_Pos' = MW512 (there the measured value to be transmitted is located)
   Input 'First_IEC_Info_Adr' = 512 (example) -> IEC-information object address of the measured value.

The measured value should be now executable and transmitted in general interrogation as well as spontaneously in case of changes.

Now further SL-blocks can be connected at P_SLi. At P_Application e.g. blocks for command output (SLo…) or with Master functionality (MAo…).

Without activating/registration please observe that the protocol runs in demo mode. This means after each CPU start for 15 min.



Example projects

Please observe as well the available example projects

Restrictions in TIA Step7

- Re-wiring (change of the number of protected blocks) is (still) not currently supported
- Test of know-how protected blocks in PLC SIM is not possible

11. SIPLUS RIC IEConS7 in S7-1500- and ET 200SP-systems

11.1. Available Libraries

From V1.6 SIPLUS RIC supports the S7-1500 and ET 200SP CPU. Both platforms can be used with the subsequent library:

IEConS7-S7-15xx_Vx_y

The library is constructed according to the new concept for TIA (see chapter 10) and contains the protocols:

- T104-protocol as Master and Slave via the PN-interface(s) of the CPU, as well as CP1543
- S7IEC_1500_104_MS_CPU_PN (FB104).
- T101and T103-protocol via the communication modules CM PtP (S7-1500 / ET 200SP)
- S7IEC_1500_101_MS_CM_PtP (FB101)
- S7IEC_1500_103_M_CM_PtP (FB103)
- AddOn functions

A corresponding Instance-DB each with identical name which is extended with '_DB' is included as well. The table of variables deposited in the respective library is also specifically adapted and refers to the Instance-DB.

11.2. Details and notes

11.2.1. Development and project planning environment

Creation of the V1.6 libraries and example projects was carried out with TIA STEP7 V13 SP1. This version is minimum required for use.

11.2.2. Manual

Large parts of this manual are still designed on the SIPLUS RIC blocks for S7 classic systems created under SIMATIC manager Step7. With regard to contents however they are also correct for the S7-15xx blocks created under TIA Step7. A few data type adaptions carried out for easier data input and better readability are not explicit documented.

11.2.3. Use of several protocols at the same time

This is supported by the SIPLUS RIC library concept for TIA. See chapter 10.

11.2.4. Activation

The product-IDs of the protocols are generated at runtime depending on the hardware platform (S7-1500 or ET 200SP).

The activating codes for permanent activation are generated from product-ID and the (unique) serial number of SMC. Therefore they are depending on platform.

11.2.5. PN-Interfaces

From V1.6 all Profinet interfaces of the CPUs as well as CP1543 are supported for T104 communication also mixed within a redundancy group thus enabling network redundancy. The selection is carried out via the new parameter 'L1_InterfaceID (_C1,_C2)'

 ©SIEMENS AG 2017

11.2.6. Project planning CM PtP

The following adjustments have to be carried out at serial communication (protocol T101/103)

- Protocol: Freeport
- Port parameters:
  - Parity: Even
  - Data bits: 8
  - Stop bits: 1
- Data flow control: None, or with RTS – Handling.
- Configuration of the message transfer:
  - Send idle line: active
  - Duration of the idle line: at least 33 Bit times
- Configuration of the message receipt:
  - Frame start detection: Start on any character
  - End of frame detection:
  - After character delay times elapses; at least 33 Bit times.
  - After receipt of a maximum number of characters: 300 bytes
- Receive buffer
  - Buffered received frames: 2
  - Clear receive buffer on startup: yes
  - Prevent overwriting: no

11.2.7. PUT/GET – Access protection

At the CP15xx by default the access via PUT/GET communication is not allowed (parameter: “Protection/Connecting mechanism”)

However this function is needed in case of profoundly diagnosis with special tools. If you have no doubt about security you should allow the access via PUT/GET.

11.2.8. Notice for loading

We recommend to load the CPU-SW always completely and to run a new start. For that use the menue <Online/PLC-Program load into device and reset>.

11.2.9. Data blocks created during run time

In the S7-1500 data blocks created during run time are exclusively supported from DB number 60000. The parameter ‘First_internal_DB_No‘ is not applicable. Please use the online presentation to watch which DBs are created from SIPLUS RIC at run time and displayed in the online display as 'online only'.

The generated data blocks are not remanent.

- They do not need a remanence memory
- They are no longer present after voltage interruption

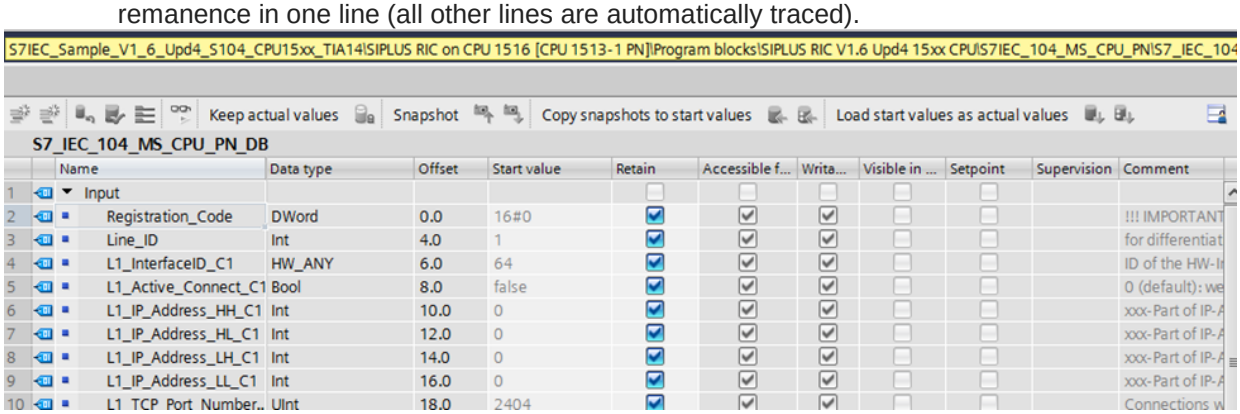
S7-1500 has a slightly different system behavior (compared to S7 classic). This has not been fully compensated and could lead to misbehavior, e. g. after voltage failure and / or when several FB10x instances are used

The data blocks are generated asynchronously, over several OB cycles. This means that subsequent blocks can only access with a delay.

- The bookkeeping of the data blocks takes place in the instance data blocks of e. g. FB10x. If this one is not remanent (default after automatically generation), in S7-1500 will be reset the content already with CPU-stop – thus clearing the bookkeeping system via data blocks, which were already created (and still existing) and therefore re-generating them at the next start. The DBs, which are only visible online, are becoming more and more.

Solutions are:

- The instance data block for FB10x supplied in the libraries has already been created remanent. Instead of creating a new DB, you can assign it (at least for one call). The enclosed monitoring table "VAT_S7_IEC _..." is already compatible to this.
- Alternatively, you can also create a new DB and set it remanent by opening and activating the remanence in one line (all other lines are automatically traced).



- In the S7-1500 remanence must be explicitly activated compared to the classic systems. SIPLUS RIC needs remanent data management only at a few positions. Basically this is the corresponding Instance-DB for the communication-FB (FB101-FB104). Here the bookkeeping of (buffer-) data blocks is created during run time.
- Without remanence new DBs are created with each restart of the CPU. In order to prevent this instance-DB has to be generated remanent.
  You achieve this by assigning the instance-DB included in each library with active remanence or you open the concerned (automatically created, non remanent) DB afterwards and insert a value into the column ‘remanence‘. With it the complete block will be remanent

Notice:

Without remanence, new DBs are generated with each restart of the CPU. To prevent this, the instance DB must be created remanently.

- If very much instances of the communication-FBs are needed the available memory range for remanent data can be used up because of the size of the instance DBs. At the latest we recommend the use of the AddOn block ‘S7_IEC_REM_MEM‘ (FB126). For further information we refer you to chapter 7.3.
- The behavior (non activatable remanence) applies to the command blocks as well. It is needed there to notice the number of the buffer-DB for temporary storage of commands. If you use the function Cmd_Buffer_Dim you should activate the remanence of the instance DB.
- The T103-‘Gateway’ resp. nodal blocks with automatically created image are concerned as well. However in this case the behavior plays a subordinated role, because the mechanism of the automatic update is only provided for the commissioning phase resp. for test purpose. Create parameter-DBs with remanent properties and assign this tightly.
- A further option is the use of the auxiliary blocks S7_IEC_REM_MEM (FB126)
- The data blocks must have an even number of useful data (e.g., array [0..4095] of bytes, always the case with Word array) and have a reasonable minimum length (e.g., 1000 bytes).

12.Software protection and demo mode

12.1. General activation of SIPLUS RIC IEConS7

For the relevant versions, libraries with protected blocks (Know How Protect) are supplied. As a general principle these can be freely copied and used.

To ensure that only the number of purchased licenses can be used, the SIMATIC Memory Card SMC (S7-1500 und ET 200SP), Micro Memory card MMC (ET 200S and S7-300), MC (S7-400) or rather CPU V4.x and CPU410 or the license number of the Software Controller is used in normal case as a dongle.

These must be plugged into the CPU to operate the application.

Depending on the serial number of the memory card, you will receive from Siemens an 8-digit registration code which you need to enter as an input parameter (Registration_Code) when calling up the Communication-FB ('S7_IEC_Config').

Example:

```
|   CALL  "S7IEC_S101_CP1SI", "S7IEC_S101_CP1SI_DB"
        Registration_Code          :=DW#16#1D49CCDC
        Line_ID                     :=
        L1_Laddr_HW                :=256
```

From V1.1 the registration code can be entered alternatively also in the DB.

The value entered at the input “Registration_Code" is interpreted as DB number if it is more than 0 and less or equal 2000 (decimal!).

In this DB also several registration codes can be entered (using the format DWORD, DW#16#...). The corresponding DB will be opened and searched for the suitable registration code.

12.2. Activation of SIPLUS RIC IEConS7 on H-systems and other systems

For the use in SIMATIC H-Systems and reasonable exceptions the possibility was created to use the CPU as a dongle. This is parameterized at the input 'Serial_from_CPU' of the Communication-FB (S7_IEC_Config).

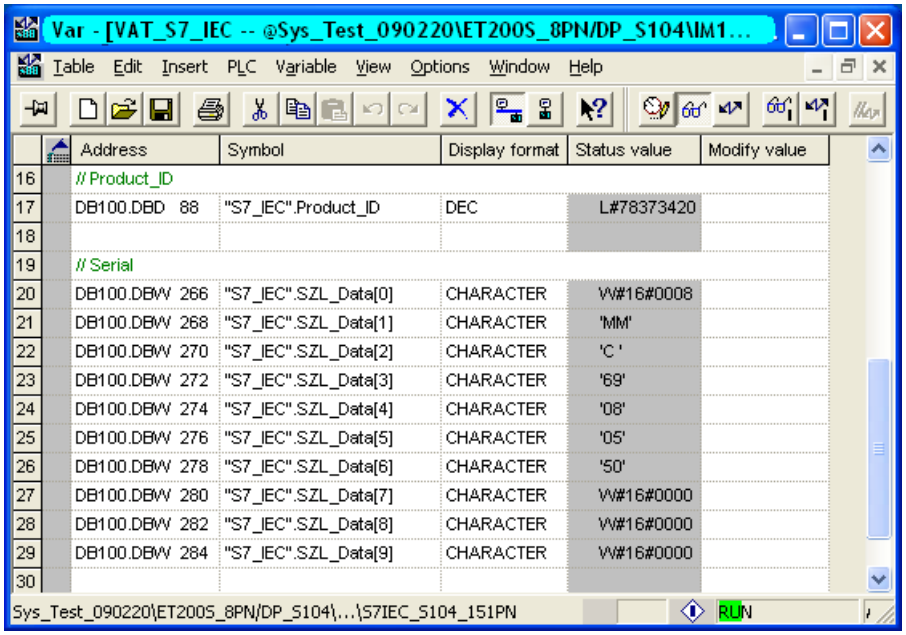
For the generation of the registration code, the following information is required

- order number from the supplier, to be found on the label of the delivered data carrier
- the reference number(s) of the application(s) to be activated (6AG6003-xxxxx-xxxx)
- the serial number readout from the memory card

Requirements for the use in SIMATIC H-systems

- the serial number of the two used SIMATIC H-CPUs and
- the serial number of a spare CPU.

Each of the available block libraries includes a table of variables 'VAT_S7_IEC'. This table of variables can be used to readout the serial number of the memory card (//Serial).



The registration code is required for **each** CPU, **per** used application (master, slave, via CPU, via CP…)!

When the valid registration code is entered the telecontrol communication is permanently activated. No entry or an invalid entry will permit unrestricted operation in demo mode for 15 minutes. After this time the telecontrol communication will be stopped, i.e. the CPU continues working but the communication will fail. The corresponding fault outputs of the Communication-FB are set.

For receiving a registration code please call one of the contact persons mentioned under **SIPLUS RIC Support line** (see chapter 13).

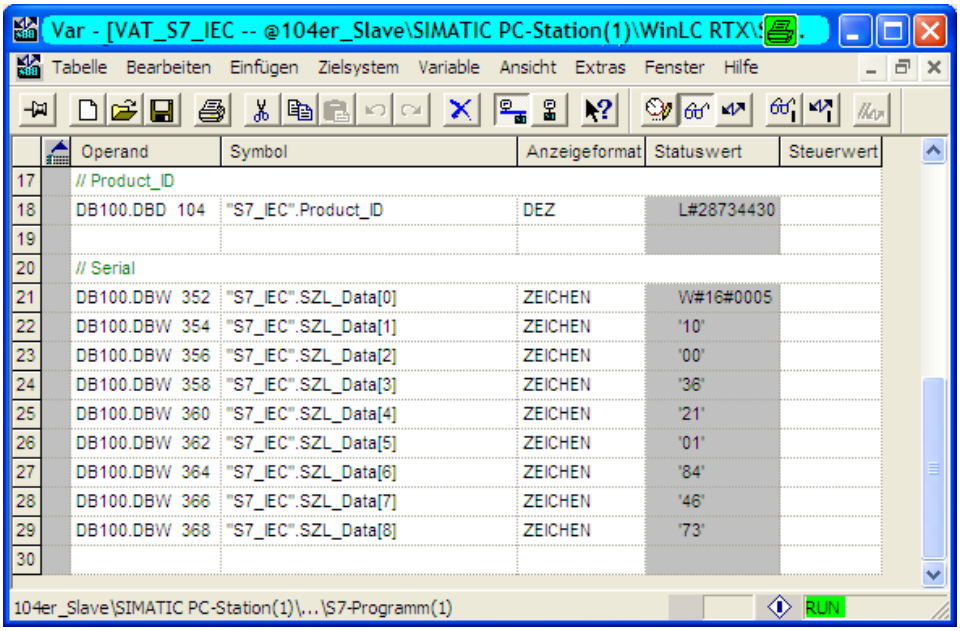
12.3. Activation of SIPLUS RIC IEConWinAC

For activation SIPLUS RIC on WinAC the license number (16 digits, as of the 5th digit) of the software package SIMATIC WinAC RTX20xx is used. If you didn't get a registration code from Siemens, the following steps have to be done **(recommended method)**:

For generating a registration code the following information are needed:

- order number from the supplier, to be found on the label of the delivered data carrier
- the reference number(s) of the application(s) to be activated (6AG6003-xxxxx-xxxx)
- the license number readout from WinAC

Each of the available block libraries includes a table of variables 'VAT_S7_IEC'. This table of variables can be used to readout (a part of) the license number of WinAC.



The registration code is required for **each** WinAC installation, **per** used application (master, slave)!

When the valid registration code is entered the telecontrol communication is permanently activated. No entry or an invalid entry will permit unrestricted operation in demo mode for 15 minutes. After this time the telecontrol communication will be stopped, i.e. the CPU continues working but the communication will fail. The corresponding fault outputs of the Communication-FB are set.

For receiving a registration code please call one of the contact persons mentioned under **SIPLUS RIC Support line** (see chapter 13).

Alternative method for activation of SIPLUS RIC IEConWinAC, if you haven't done any projecting in SIMATIC-Manager yet:

For generating a registration code the following information are required:

- order number from the supplier, to be found on the label of the delivered data carrier
- the reference number(s) of the application(s) to be activated (6AG6003-xxxxx-xxxx)
- the license number of WinAC

  This number you will find
  - on the WinAC software package
  - in the Automation License Manager, if you have already transferred the License Key of WinAC onto your system or
  - via the „WinLC RTX-Controller" → Help → About → „Serial Number:" (the license no. is named as „Serial Number").

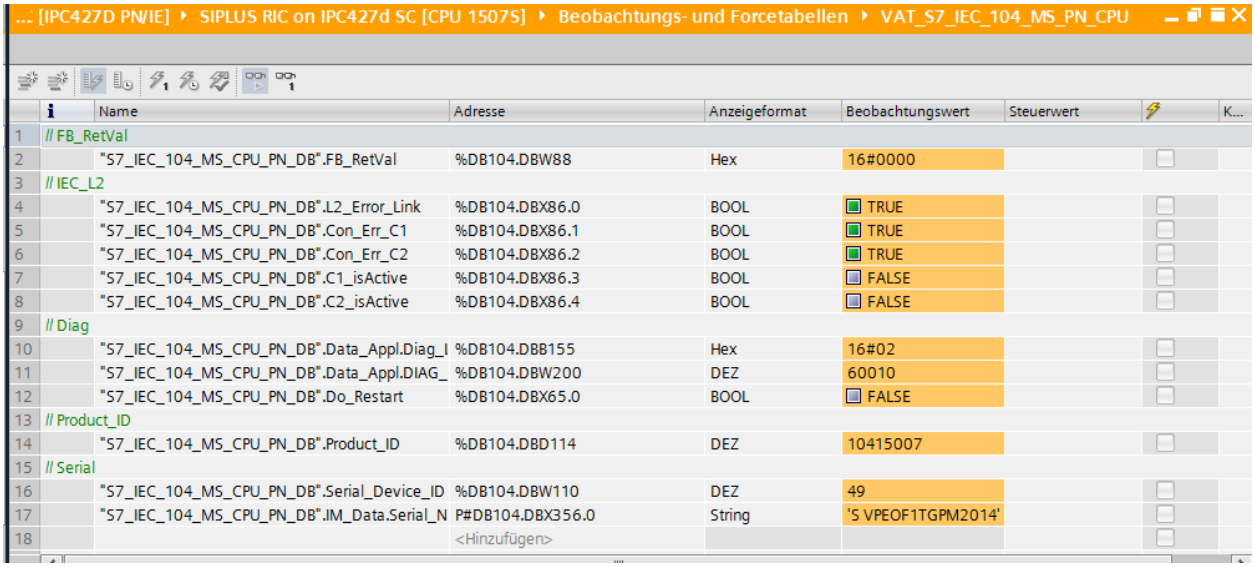
12.4. Activation of SIPLUS RIC IEConS7-1500

For activation SIPLUS RIC of S7-1500 (16 digits) of the software package SIMATIC S7-1500 is used. If you didn't get a registration code from Siemens, the following steps have to be done:

For generating a registration code the following information are needed:

- order number from the supplier, to be found on the label of the delivered data carrier
- the reference number(s) of the application(s) to be activated (6AG6003-xxxxx-xxxx)
- the license number readout from WinAC

Each of the available block libraries includes a table of variables 'VAT_S7_IEC'. This table of variables can be used to readout the license number of Station Controller.



	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert
1	// FB_RetVal				
2	"S7_IEC_104_MS_CPU_PN_DB".FB_RetVal	%DB104.DBW88	Hex	16#0000	
3	// IEC_L2				
4	"S7_IEC_104_MS_CPU_PN_DB".L2_Error_Link	%DB104.DBX86.0	BOOL	TRUE	
5	"S7_IEC_104_MS_CPU_PN_DB".Con_Err_C1	%DB104.DBX86.1	BOOL	TRUE	
6	"S7_IEC_104_MS_CPU_PN_DB".Con_Err_C2	%DB104.DBX86.2	BOOL	TRUE	
7	"S7_IEC_104_MS_CPU_PN_DB".C1_isActive	%DB104.DBX86.3	BOOL	FALSE	
8	"S7_IEC_104_MS_CPU_PN_DB".C2_isActive	%DB104.DBX86.4	BOOL	FALSE	
9	// Diag				
10	"S7_IEC_104_MS_CPU_PN_DB".Data_Appl.Diag_I	%DB104.DBB155	Hex	16#02	
11	"S7_IEC_104_MS_CPU_PN_DB".Data_Appl.DIAG_	%DB104.DBW200	DEZ	60010	
12	"S7_IEC_104_MS_CPU_PN_DB".Do_Restart	%DB104.DBX65.0	BOOL	FALSE	
13	// Product_ID				
14	"S7_IEC_104_MS_CPU_PN_DB".Product_ID	%DB104.DBD114	DEZ	10415007	
15	// Serial				
16	"S7_IEC_104_MS_CPU_PN_DB".Serial_Device_ID	%DB104.DBW110	DEZ	49	
17	"S7_IEC_104_MS_CPU_PN_DB".IM_Data.Serial_N	P#DB104.DBX356.0	String	'S VPEOF1TGPM2014'	
18		<Hinzufügen>			

Serial_Device_ID shows the HW ID of the device from which the serial number is read:

- 49: Libraries for S7-1500 Software Controller -> 'Local CPU'
- 51: Libraries for S7-1500 CPUs -> SMC (Simatic Memory Card)

The registration code is still based on the displayed serial number and the product ID.

The registration code is required for **each** Station Controller installation, **per** used application (master, slave)!

When the valid registration code is entered the telecontrol communication is permanently activated. No entry or an invalid entry will permit unrestricted operation in demo mode for 15 minutes. After this time the telecontrol communication will be stopped, i.e. the CPU continues working but the communication will fail. The corresponding fault outputs of the Communication-FB are set.

For receiving a registration code please call one of the contact persons mentioned under **SIPLUS RIC Support line** (see chapter 13).

13.Contacts/Addresses

In case you have queries concerning the SIPLUS RIC product line or you want advice via phone: we will help.

SIPLUS RIC – Product manager

Dieter Söhlmann
Phone: +49 (0) 911 750-4790
Email: dieter.soehlmann@siemens.com

Based on the existing blocks, enhanced requirements for IEC communication (e.g. other image layouts or information content, transmission causes, etc.…) can be implemented with minimal work and cost-effectively.

Please ask us if your application requires a customized project solution with enhanced functionality. As a service provider in the field of SIMATIC project planning and programming, we look forward to solve your problems together with you.

SIPLUS RIC – Support line
Mail: siplus-ric.automation@siemens.com

You find further information on the SIPLUS Internet pages
Website: http://www.automation.siemens.com/siplus/index_00.htm

Site address
Siemens AG
Digital Factory
Factory Automation
Systems Engineering
DF FA SE - SIPLUS RIC
Breslauer Str. 5
D-90766 Fürth

Postal address
Siemens AG
Digital Factory
Factory Automation
Systems Engineering
DF FA SE - SIPLUS RIC
P.O. Box 2355
D-90713 Fürth

SIPLUS RIC IEConS7 is based on standard Siemens products. You will find information and support from our central service & support:

Service & Support

Industry Customer Carecenter
DF CS SD CCC CRU OP
Phone: +49 (0) 911/895 7222
Fax. +49 (0) 911/895-7223
Inquiries: http://www.siemens.de/automation/support-request
Website: http://www.siemens.de/automation/service&support

14. Appendix

14.1. Resource assignment

Resources required by the application blocks:

Type	Amount or number range	Note
Memory (S7-300 and S7-400)		
Loading memory	64 kB	approx. value, min. requirement
Working memory	64 kB	approx. value, min. requirement
Data	10 kB	approx. value, min. requirement
Memory (S7-1500 and ET 200SP):		
Loading memory	4.5 MB	approx. value, min. requirement if all application blocks have been loaded
Code-Working memory	96 kB	
Data	26 kB	
Remanence memory	7.5 kB	
Blocks:		
FBs	1 – 199 200 -228 (TIA-Portal V13 SP1)	can be re-wired, (with SIMATIC Manager) multi-instance compatible
FCs	1 - 199	can be re-wired
DBs	1 - 199	
Other:		
Markers	none	
Timers	none	
Counters	none	

All functional blocks (FB) are designed in a way so that they can be used in the application as multi-instance functional blocks.

NOTICE

Released CPU versions!

Only the following CPU versions are released:

ET 200S IM151-7 CPU from V2.6
ET 200S IM151-8 PN/DP CPU from V2.7
S7-300 from V2.0 with min 64kB working memory
S7-400 from V4.0
WINAC from version 2010
S7-1500 and ET 200SP from V1.0
Station Controller from V1.8

The tables below show in detail which blocks are included in the individual libraries.

Type	No.	Symbolic Name	S7IEC_M101_CP1SI	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M101_CP441	S7IEC_M101_WinAC	S7IEC_M103_CP1SI	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M103_CP441	S7IEC_M103_WinAC	S7IEC_M104_151PN	S7IEC_M104_31xPN	S7IEC_M104_41xPN	S7IEC_M104_CP343	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_M104_WinAC	S7IEC_S101_CP1SI	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S101_CP441	S7IEC_S101_WinAC	S7IEC_S104_151PN	S7IEC_S104_31xPN	S7IEC_S104_CP343	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp	S7IEC_S104_41xPN	S7IEC_S104_WinAC	S7_IEC_AddOn
DB	80	Para_TCP_WinAC																	x												x	
DB	82	Instance DB for																	x												x	
	..88	Para_TCP_WinAC																														
DB	90	Para_SER_WinAC					x					x												x								
DB	91	Instance DB for					x					x												x								
	.. 96	Para_SER_WinAC																														
DB	100	S7_IEC	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
DB	111	P_LinkAdr_n	x	x	x	x	x	x	x	x	x	x																				
DB	112	P_ASDUAdr_n	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x													
DB	139	Para_DB_IOA_Demo	x	x	x	x	x						x	x	x	x	x		x										x			
DB	140	Para_DB _MAo_SP	x	x	x	x	x						x	x	x	x	x	x	x													
DB	141	Para_DB _MAo_DP	x	x	x	x	x						x	x	x	x	x	x	x													
DB	142	Para_DB _MAo_ST	x	x	x	x	x						x	x	x	x	x	x	x													
DB	143	Para_DB _MAo_BO	x	x	x	x	x						x	x	x	x	x	x	x													
DB	144	Para_DB _MAo_MEa	x	x	x	x	x						x	x	x	x	x	x	x													
DB	145	Para_DB _MAo_MEb	x	x	x	x	x						x	x	x	x	x	x	x													
DB	146	Para_DB _MAo_MEc	x	x	x	x	x						x	x	x	x	x	x	x													
DB	147	Para_DB _MAo_IT	x	x	x	x	x						x	x	x	x	x	x	x													
DB	148	Para_DB _MAi_CMD	x	x	x	x	x						x	x	x	x	x	x	x													
DB	161	Para_DB_MAoSLi_MD																														x
DB	166	Para_DB_SLoMAi_CD																														x
DB	170	S7_IEC_T103						x	x	x	x	x																				
DB	172	P_T103_IND						x	x	x	x	x																				
DB	173	P_T103_ME						x	x	x	x	x																				
DB	174	P_T103_CMD						x	x	x	x	x																				
DB	500	P_T103_MAo_pTI79_IM																														x

 ©SIEMENS AG 2017

Type	No.	Symbolic Name	S7IEC_M101_CP1SI	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M101_CP441	S7IEC_M101_WinAC	S7IEC_M103_CP1SI	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M103_CP441	S7IEC_M103_WinAC	S7IEC_M104_151PN	S7IEC_M104_31xPN	S7IEC_M104_41xPN	S7IEC_M104_CP343	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_M104_WinAC	S7IEC_S101_CP1SI	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S101_CP441	S7IEC_S101_WinAC	S7IEC_S104_151PN	S7IEC_S104_31xPN	S7IEC_S104_CP343	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp	S7IEC_S104_41xPN	S7IEC_S104_WinAC	S7_IEC_AddOn
FB	2	S_RCV	x					x	x											x											
FB	3	S_SEND	x					x	x											x											
FB	2	P_RCV		x								x									x										
FB	3	P_SEND		x								x									x										
FB	7	P_RCV_RK			x					x												x									
FB	8	P_SND_RK			x					x												x									
FB	8	S_V24						x																							
FB	21	DP_Supp_Interm_States																													x
FB	40	FTP_CMD																													x
FB	63	TSEND											x	x	x										x	x				x		
FB	64	TRCV											x	x	x										x	x				x		
FB	65	TCON											x	x	x										x	x				x		
FB	66	TDISCON											x	x	x										x	x				x		
FB	82	TINT_WINAC																	x												x	
FB	83	TSEND_WINAC																	x												x	
FB	84	TRCV_WINAC																	x												x	
FB	85	TCON_WINAC																	x												x	
FB	86	TDISCON_WINAC																	x												x	
FB	87	TUSEND_WINAC																	x												x	
FB	88	TURCV_WINAC																	x												x	
FB	91	P_RECV_WINAC					x					x												x								
FB	92	P_SEND_WINAC					x					x												x								
FB	93	V24_STAT_WINAC					x					x												x								
FB	94	V24_SET_WINAC					x					x												x								
FB	95	COM_INIT_WINAC					x					x												x								
FB	96	P_RESET_WINAC					x					x												x								
FB	100	S7_IEC_Config	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FB	101	L1_CP341			x					x												x										

©SIEMENS AG 2017

Type	No.	Symbolic Name	S7IEC_M101_CP1SI	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M101_CP441	S7IEC_M101_WinAC	S7IEC_M103_CP1SI	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M103_CP441	S7IEC_M103_WinAC	S7IEC_M104_151PN	S7IEC_M104_31xPN	S7IEC_M104_41xPN	S7IEC_M104_CP343	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_M104_WinAC	S7IEC_S101_CP1SI	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S101_CP441	S7IEC_S101_WinAC	S7IEC_S104_151PN	S7IEC_S104_31xPN	S7IEC_S104_CP343	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp	S7IEC_S104_41xPN	S7IEC_S104_WinAC	S7_IEC_AddOn
FB	102	L1_CP441				x					x												x								
FB	103	L1_CP340		x					x												x										
FB	104	L1_1SI	x					x												x											
FB	105	L1_CP343														x											x				
FB	106	L1_CP443															x											x			
FB	107	L1_CPU31x_PN											x	x										x	x						
FB	108	L1_CPU41x_PN													x														x		
FB	109	L1_WinAC_TCP																	x											x	
FB	110	L2_T101_103_bal_unbal	x	x	x	x	x	x	x	x	x	x								x	x	x	x	x							
FB	111	L1_WinAC_Ser					x					x												x							
FB	112	L1_CP443_Speed																x											x		
FB	115	L2_T104_APCI											x	x	x	x	x		x						x	x	x	x		x	x	
FB	116	L2_T104_APCI_S																x											x			
FB	120	L7_T101_103_104_B	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FB	121	SL_Org_Asdu_1																		x	x	x	x	x	x	x	x	x	x	x		
FB	122	MA_org_Asdu_1	x	x	x	x	x						x	x	x	x	x	x	x													
FB	125	S7_IEC_Appl_Manager																														x
FB	130	SLi_SP_DP_s128																		x	x	x	x	x	x	x	x	x	x	x		
FB	131	SLi_ST_s8																		x	x	x	x	x	x	x	x	x	x	x		
FB	132	SLi_BO_s8																		x	x	x	x	x	x	x	x	x	x	x		
FB	133	SLi_ME_ABC_s32																		x	x	x	x	x	x	x	x	x	x	x		
FB	134	SLi_IT_s8																		x	x	x	x	x	x	x	x	x	x	x		
FB	135	SLo_SC_DC_RC_sx																		x	x	x	x	x	x	x	x	x	x	x		
FB	136	SLo_SE_ABC_sx																		x	x	x	x	x	x	x	x	x	x	x		
FB	137	SLo_BO_sx																		x	x	x	x	x	x	x	x	x	x	x		
FB	148	MAi_SC_DC_RC_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FB	149	MAi_SE_ABC_1	x	x	x	x	x						x	x	x	x	x	x	x													
FB	150	MAi_BO	x	x	x	x	x						x	x	x	x	x	x	x													
FB	161	MAoSLi_MD_pDB																														x
FB	166	SLoMAi_CD_pDB																														x

©SIEMENS AG 2017

Type	No.	Symbolic Name	S7IEC_M101_CP1SI	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M101_CP441	S7IEC_M101_WinAC	S7IEC_M103_CP1SI	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M103_CP441	S7IEC_M103_WinAC	S7IEC_M104_151PN	S7IEC_M104_31xPN	S7IEC_M104_41xPN	S7IEC_M104_CP343	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_M104_WinAC	S7IEC_S101_CP1SI	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S101_CP441	S7IEC_S101_WinAC	S7IEC_S104_151PN	S7IEC_S104_31xPN	S7IEC_S104_CP343	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp	S7IEC_S104_41xPN	S7IEC_S104_WinAC	S7_IEC_AddOn
FB	170	T103_S7_IEC_Config						x	x	x	x	x																			
FB	171	T103_MA_ORG_ASDU						x	x	x	x	x																			
FB	172	T103_MAo_IND_SLi_T1014						x	x	x	x	x																			
FB	173	T103_MAo_ME_SLi_T1014						x	x	x	x	x																			
FB	174	T103_MAi_CMD_SLo_T1014						x	x	x	x	x																			
FB	181	Ext_Buffer_CP_4																													x
FB	185	FT_CD_1DB																													x
FB	500	T103_MAo_pTI79_IM																													x
SFB	65001	CREA_COM					x					x							x					x						x	
SFB	65002	EXEC_COM					x					x							x					x						x	

©SIEMENS AG 2017

Type	No.	Symbolic Name	S7IEC_M101_CP1SI	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M101_CP441	S7IEC_M101_WinAC	S7IEC_M103_CP1SI	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M103_CP441	S7IEC_M103_WinAC	S7IEC_M104_151PN	S7IEC_M104_31xPN	S7IEC_M104_41xPN	S7IEC_M104_CP343	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_M104_WinAC	S7IEC_S101_CP1SI	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S101_CP441	S7IEC_S101_WinAC	S7IEC_S104_151PN	S7IEC_S104_31xPN	S7IEC_S104_CP343	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp	S7IEC_S104_41xPN	S7IEC_S104_WinAC	S7_IEC_AddOn
FC	1	AD_DT_TM	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	5	V24_STAT		x	x											x					x	x					x					
FC	6	V24_SET		x	x											x					x	x					x					
FC	10	AG_CNTRL														x	x	x									x	x	x			
FC	34	SB_DT_DT(ab V1.5)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	50	AG_LSEND															x											x				
FC	53	AG_SSEND																x											x			
FC	60	AG_LRECV															x											x				
FC	63	AG_SRECV																x											x			
FC	100	Set_Buf_Para																														x
FC	110	S7_IEC_Buf_Create	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	111	S7_IEC_Buf_Control	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	112	S7_IEC_Job_Info	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	113	S7_IEC_Diag	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	114	S7_IEC_TimeConv	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	115	S7_IEC_MA_Find_IEC_Adr	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	116	S7_IEC_Copy	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
FC	121	H_Calc_DST_UTC_Europe																														x
FC	125	S7_IEC_Set_Threshold_1																														x
FC	140	MAo_SP_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	141	MAo_DP_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	142	MAo_ST_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	143	MAo_BO_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	144	MAo_ME_NA_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	145	MAo_ME_NB_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	146	MAo_ME_NC_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													
FC	147	MAo_IT_IM_pDB	x	x	x	x	x						x	x	x	x	x	x	x													

©SIEMENS AG 2017

Type	No.	Symbolic Name	S7IEC_M101_CP1SI	S7IEC_M101_CP340	S7IEC_M101_CP341	S7IEC_M101_CP441	S7IEC_M101_WinAC	S7IEC_M103_CP1SI	S7IEC_M103_CP340	S7IEC_M103_CP341	S7IEC_M103_CP441	S7IEC_M103_WinAC	S7IEC_M104_151PN	S7IEC_M104_31xPN	S7IEC_M104_41xPN	S7IEC_M104_CP343	S7IEC_M104_CP443	S7IEC_M104_CP443_Sp	S7IEC_M104_WinAC	S7IEC_S101_CP1SI	S7IEC_S101_CP340	S7IEC_S101_CP341	S7IEC_S101_CP441	S7IEC_S101_WinAC	S7IEC_S104_151PN	S7IEC_S104_31xPN	S7IEC_S104_CP343	S7IEC_S104_CP443	S7IEC_S104_CP443_Sp	S7IEC_S104_41xPN	S7IEC_S104_WinAC	S7_IEC_AddOn
UDT	65	TCON_PAR											x	x	x										x	x				x		
UDT	66	TADDR_PAR																	x												x	
VAT		VAT_S7_IEC	x	x	x	x		x	x	x	x		x	x	x	x	x	x		x	x	x	x		x	x	x	x	x	x		

The (green) marked blocks are standard SIMATIC blocks.

All others are application-specific blocks.

If there are already blocks in your project with the same number, you can adapt the numbers according to the requirements of your project using the 'block/re-wiring' function (see chapter 14.2)

The Communication-FB blocks ('S7_IEC_Config') are designed exclusively for the OB1 sequence level. Within this also the application blocks have to be operated!

In TIA portal the Communication-FB blocks have been replaced by an explicit number band (200 – 228).

©SIEMENS AG 2017

14.2. Re-wiring blocks (only SIMATIC Manager)

Particularly if using SIPLUS RIC blocks in existing projects or in conjunction with other libraries crossover of block numbers may occur.

Prior foreign blocks (from the view of SIPLUS RIC) should be renamed. The block numbers of SIPLUS RIC should be kept. This allows a better support by Siemens and an easy change in case of updates.

In case this is not possible the following rules apply:

14.2.1. Data blocks:

There are no fixed assigned data blocks. They are all created as instance DB or are assignable by parameterizing. Therefore they are arbitrary in the numbering band.

Exception:

IEConS7 for WinAC: Here the data blocks are part of the Runtime-Software. If its DB numbers have to be changed, they have to be changed in the Parameter DB as well.

14.2.2. Independent FBs/FCs:

FB/FC which may not be called from other blocks may be re-named directly. These are in IEConS7:

- Communication-FB /FB170 (S7_IEC_Config)
- All SLi, SLo, MAi and MAo-Blocks
- Add_On-Blocks like FC121 (H_Calc_DST_UTC_Europe)

14.2.3. FBs/FCs with dependency

All other FB and FC numbers have to be changed if required using the function ‚re-wiring'. Procedure:

1. Copy the desired block library into a project like independent S7 program
2. Re-name the Communication-FB to a high block number (see 14.2.3.1)
3. For the re-wiring procedure follow the online help in the Simatic manager

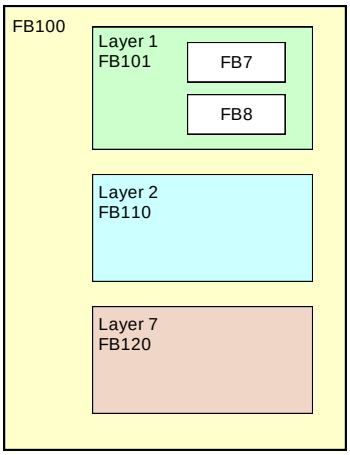
 ©SIEMENS AG 2017

14.2.3.1. Possible problems at re-wiring

Problems may occur by **multilevel multi instance FB** and **its request sequence.** The re-wiring routine of the Simatic manager needs an ascending interlacing sequence.

Example:

In the T101 variant the IEConS7 has the following format:



In case of rewiring the FB7 for example, the rewiring routine of the Simatic manager is not working correctly. The location of FB7 in FB101 and FB101 located in the Communication-FB is no ascending sequence because FB101 is > Communication-FB.

14.2.3.2. Solution/Workaround:

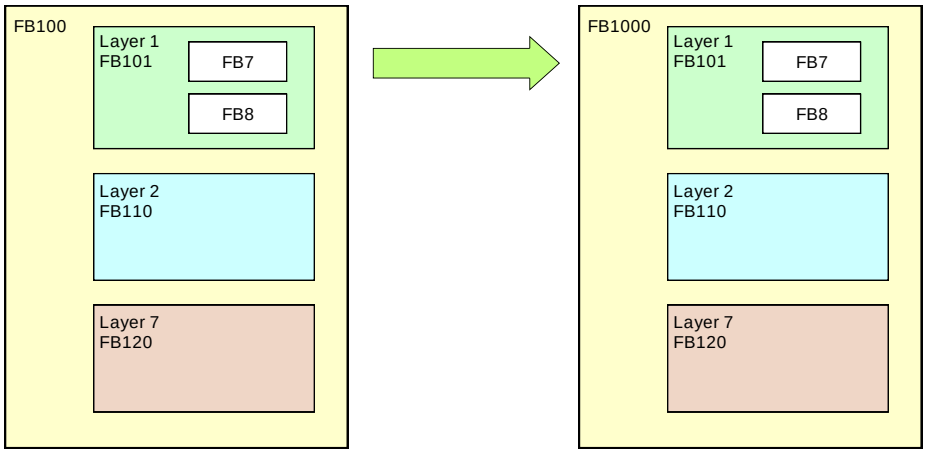
A solution would be the re-naming of Communication-FB into Communication-FB 0 because Communication-FB 0 >FB101 (FB7 located in FB101 located in Communication-FB 0).

Since Communication-FB is a „Start-FB“ at IEConS7 it can/may be re-named just by changing the block number in the Simatic-Manager:

1 Convert the Communication-FB to a high FB-number (e.g. Communication-FB 0) with the function re-name (F2)

2 Carry out the rewiring function according to your demands

3 In the end the re-named Communication-FB /1000 may be arbitrary re-named again.



14.3. Interoperability list for T101/T104

The interoperability list for the protocols IEC 6087-5-101 and IEC 6087-5-104 is included on the CD as a separate PDF file (IEConS7 Interoperability.pdf).

14.4. New Features and Functions

14.4.1. Delivery Version V1.1

Below you find a listing with the fundamental improvements in the delivery version V1.1

Assignment of the blocks in SIMATIC H-Systems

The direct assignment of the blocks in SIMATIC H-Systems is now possible because several registration codes can be entered and the request of the serial number is parameter driven.

see chapter 9

Supporting CP340

CP340 is supported as an additional interface for T101 protocol variant.
Same product-ID like CP341-variant.
Delivery also with this variant.

T101/104: Message buffering - Using message memory

Parameter driven processing of the message memory in case of Link Error.

see chapter 4.4.1

T104: Confirmation buffer

From V1.1 sent T104-messages with user data (I-Formats) can be buffered meeting technical standard and with this can be recalled.

see chapter 4.4.2

T104: Reverse link connection

The TCP link connection can be carried out active as well as passive also with connections via the CPU.

T104: Multiple channel Master (T104-Master-Redundanz)

Now also in the Master two active TCP connections are possible (T104 redundancy group). Both are monitored with test frames. The data control using StartDT is carried out automatically.

see chapter 5.2

T104: Standardized switch off from not used channels

PN and CP-Interface

PN-Interface can be switched off by Port=0

CP-Interface can be switched off by ID=0

T104: Enhanced connection supervisory

I-Formats are not any more confirmed if the connection is non active (S-Format)

Possibility for the communication partner to detect a non active channel and to restart the link connection.

 ©SIEMENS AG 2017

T104: Handling of faults – time delayed

Also with established TCP connection the fault output L2_Link_Error will be set after a parameter driven delay time, if no T104 connection is active.

T104: Further diagnosis bits (active channel,…)

At T104 additional diagnosis bits are available:
connection faults channel specific
signaling of the active channel

SL-Blocks: Individual information object address (IOA)

Individual (none ascending) IEC address allocation is now possible in all SL blocks via address parameter DB's.

see chapter 6.1.4

SLi-Blocks: Interrogation groups can be called without GI

Pure group interrogations can be set by parameter.
(without reaction on GI).

SLi_ME_ABC: Collection of measurands

Interrogated measurands are now actual values.
Reset of the threshold value possible via a block input.
(Up to now a new threshold value was taken over only with achievement of the last threshold)

SLi_IT: Collection of integrated totals:

At collection variants with IEC status the CY-Bit (CY = carry) is reset from V1.1 with each restoring/clearing. If required it has to be set from the user only one time (Set command).

SLo-Blocks: Structure extension with filler bytes

The command output format ‚Dst_Struct_Type' now allows also filler bytes (1-7) like at the SLi blocks.

SL(o)-Blocks: Negative confirmation of commands with unknown TI/IOA

Not executed commands due to unknown TI or IOA are negative confirmed with adapted cause of transmission (COT).

see chapter 6.1.1

SL_Org: Improved adjustment with TI103 and Receive signaling

Improved procedure for time synchronization for synchronization via time adjustment message (TI103).
with programmable transit time correction
Diagnosis output for transfer of the time from TI103

MAo-Blocks: Automatically image distance

Determination of the image length automatically in dependency on the image type at default '*Image_Len*' in the image = 0.

MAo-Blocks: Output format with time stamp

The MAo-Blocks have been extended with an output format with time stamp.

see chapter 6.4.3

Several registration codes

If the specified registration code is not suitable it will be interpreted as DB number if it is > 0 and ≤ 2000 (decimal!). The DB will be opened and scanned for suitable registration codes.
Important for the application in SIMATIC H-Systems.

see chapter 9.2

Licensing via serial number the CPU or MC

Optional selection whether the serial number of the CPU or the memory card has to be taken for licensing.
Important for the application in SIMATIC H-Systems.

see chapter 9.2

Reset-Input for 'quasi' Online-Modifications

Block input for program controlled or manual restart of the IEC application without CPU Stop/Start.
Important because some parameter are not effective if loaded online.

Optional DB-No. for buffer

The lowest DB number created from program is now programmable.

Set diagnosis bits Link_Error etc. at demo time expiration

In case of fault (demo time expired) the disturbance outputs L2_Error_Link, Con_Err_Cx are set and the outputs Cx_isActive are reset.

Performance increasing for Master - Applications

In case of blocked receive messages a repeated (and then reduced) cycle of the Master block can be programmed.

see chapter 4.5

 ©SIEMENS AG 2017

14.4.2. Delivery Version V1.2

Subsequent you find a list with the essential improvements in the delivery version V1.2

Application of the blocks in the ET 200S CPUs

Now also variants for the application in SIMATIC ET 200S CPUs available.

see chapter 3.1 and chapter 4.1

T101 Master, polling line

The variants for the T101 Master now also support the polling mode with several stations on one line.

see chapter 6.4.2

Command buffering

From V1.2 fast successively received switching commands (impulse commands) can be buffered.
With it command sequences are supported.

see chapter 6.1.3.1

T103 Master

The functionality of the SIPLUS RIC IEConS7 application for the IEC 60870-5-103 Master protocol is described in the SIPLUS RIC IEC 103 on S7 manual.

14.4.3. Delivery Version V1.3

Subsequent you find a list with the essential improvements in the delivery version V1.3

IEC on WinAC

With the version V1.3 WinAC based systems are supported from SIPLUS RIC IEConS7 now. On these computers the software package WinAC from RTX 2010 must be installed.

see chapter 3.2.3 and chapter 4.2.3

GI initiation manually

SL_Org_ASDU (FB121) was extended by the boolean input 'Sim_GI'. A positive edge at this input simulates the initiation of a general interrogation (with originator 0). Like at a real interrogation the GI-Confirmation and the GI-End message are transmitted as well.

see chapter 6.1.1

Signaling GI runs

SL_Org_ASDU (FB121) was extended by the boolean output GI_Runs. It serves for the signaling of a received (and from SL_Org_ASDU processed) GI-Request.

see chapter 6.1.1

Support of TI102 (Single request)

The single request with message type TI102 specified in the IEC-Standard is supported now.

see chapter 6.1.1

Processing measured values → Initiation method

Particularly at slow communication lines it may be disturbing to transmit each detected change of a measured value. With the ‚Initiation Method' the problem can be avoided. The activation of the function initiation method is carried out via an additional input 'Use_Initiation_Method'

see chapter 6.1.2.4

Limit of the number of measured values at cyclic transmission

With the new parameter 'No_Cyclic_Infos' the number of transmitted measured values per cyclic initiation is limited.

see chapter 6.1.2.4

Processing measured values → instant reaction at threshold changing

A change of the parameter 'Threshold_Sensivity' at run time is monitored and treated like a reset of the thresholds with parameter 'Reset_Threshold'

see chapter 6.1.2.4

Extension of the send buffer: Number and Dimension

The Communication-FB blocks were extended by two parameters:
No_of_Send_Buffers (1..16): Number of send buffer to be created
Send_Buffer_Dim: Dimension in Byte of send buffer to be created

see chapter 3.3 and 4.3

Priority assignment of the Send buffer

In all SLi-blocks the previous boolean parameter 'Tx_Prio_L_H' was replaced by the integer input 'Tx_Prio'. By means of it the user can give the priority assignment 0 (high) up to 15 (low).

see chapter 6.1.2

Use of the NON_Retain memory

With the introduction of the NON_Retain memory the user has the possibility to gain additional data memory by using the (none buffered) working storage of the CPU. This NON_Retain memory can be used for buffering of data in SIPLUS RIC IEConS7.

see chapter 7

Introduction of the FC100 Block

The FC100 block can be used for the following tasks:

Use of an offline generated DB (also NON-Retain) as buffer-DB; Adaption of the DB-length from automatically generated DBs; Readout of DB number and DB length from internal buffer DBs.

see chapter 8.1.1

Multiple Channel/Redundancy Communication-FB

With the IEConS7 variant V1.3 the cascading of Communication-FB blocks is now possible. Basic concept is the coupling of another (or more) Communication-FB to an existing Communication-FB.

see chapter 5.2

Parameter Line_ID, Comp_ID und Con_ID

The Line_ID at the Communication-FB blocks now is standardized used for the derivation of the internal Connection_ID (Con_ID).

Parameter Block_Len

The parameter 'Block_Len' previous used only internally is now available in all Slave versions. Here the max. length of the ASDU is adjusted.

CP343 und CP443 -receiving advanced

If the number of (still) expected signs is unknown the receiving at CP343 and CP443 is adapted from 1 to 6 signs.

 ©SIEMENS AG 2017

14.4.4. Delivery Version V1.4

Subsequent you find a list with the essential improvements in the delivery version V1.4

AddOn-blocks

Application blocks FB161 and FB166 for nodal functions meeting Master as well as Slave functions	see manual AddOn nodal blocks
FB161 for transfer of information in monitoring direction (indications, measured values etc.) from a subordinated station to the remote station.	
FB166 for transfer of the commands and setpoint values from a control center to the subordinated station.	

SLi-blocks parameter Phase_Offset_Cyc_Interv

For a parameterizable offset of the initiation for cyclic transmission and the restoring interval of the integrated totals.	see chapter 6.1.2

Supporting the polling mode of the ET 200S (1SI)

The RTS lead and follow-up time for ET 200S variants with 1SI can be adjusted at the Communication-FB with the parameters **'L1_Data_Wait_Time'** as well as **'L1_Time_RTS_OFF'.** At other CPs the parameter is located in the hardware configuration.	

Special function SF_Originator

The value denoted here is taken over as originator in messages of the own block and from the SLi-blocks.	see chapter 6.1.1

Support of H-CPUs V6

SIPLUS RIC supports the use of the PN-Interfaces integrated on the V6-H-CPUs for IEC 60870-104 communication on both CPUs.	see chapter 9.1.5

WinAC-Variants

The driver supports now Windows 7 as well

14.4.5. Delivery Version V1.5

Subsequently you will find a list with the essential improvements in the delivery version V1.5

General

Extensions at the internal interfaces require a complete upgrade of existing projects during the actualization. Single V1.5 blocks may not be taken over into older projects.

Excluded from this are all parameter data blocks. These can be continued to use from all previous projects.

KEMA-Certification and associated functional extensions

From 17.03.-22.03.13 the software version S7IEC_S104_31xPN SP1 including blocks from the MA-libraries (for reversed direction) was introduced at the DNV/KEMA. It is a matter of the serial version V1.4 with improvements considered to be necessary in the front-end (->SP1) and corresponding PID V1.0 (Protocol Implementation Documentation).

During certification a few objections (almost exclusively in negative tests) were resolved and additional tested as total version SP1.1 (+PID V1.1). This version has passed the certification. However a marginal note is that the tests of standard direction (pure substation) and reversed direction were not conducted at the same time. The simultaneous functionality cannot be arbitrarily used without further adaption/boundary conditions. This results from the now tight implemented rejection (mirroring) of unknown messages (resp. message contents).

V1.5 is based on version V1.4 and additionally extended by the following functions:

Slave and Master blocks can be used without restrictions and simultaneously. ASDU marked with ‘CON‘ in the standard (confirmed application services) are centrally mirrored and with it generally with the corresponding cause of transmission if the message cannot be ‘executed‘.

The following causes of transmission are possible and executed in the subsequent order:

<44>:= unknown type identification:
If there is no evaluation block for the message type ‚CON‘. This applies generally without interconnection at P_Application.

<46>:= unknown common address of the ASDU:
broadcast addresses are not mirrored.

<45>:= unknown cause of transmission (COT)
In case of receiving unexpected or not supported cause of transmission.
Received messages with one of the cause of transmission mentioned here (44-47) are not mirrored.

<47>:= unknown address of the information object

Further functions:

The standard command blocks SLo (FB135-FB137) now support the function ‘Select before Operate’ completely, i.e. including time monitoring. The function can be activated by parameter.	see chapter 6.1.3
The standard command blocks SLo (FB135-FB137) now not only accept commands with time but also the time stamp is monitored on aging activatable by parameter input.	see chapter 6.1.3
The command blocks SLo (FB135-FB137) have been extended by a return value. It signals the reason if a command could not be executed.	see chapter 6.1.3
The MAo-blocks (FC140 - FC147) only accept the causes of transmission (COT) which are permissible in the list of interoperability. If information is rejected because of 'unknown' COT, this is signaled at the newly established FC_RetVal output.	see chapter 6.4.3

Some FBs need the SIMATIC FC34 (SB_DT_DT) from V1.5. It is now part of all IEC libraries.

Validity of the KEMA Certification
A preliminary result V1.4 SP1.1 has been certified.
SIPLUS RIC V1.5 is not certified. However it can be classified functionally as follows:

- Communication:
  The variant ‚S104_31xPN‘ has been certified. Based on this variant in V1.5 are as well:
  - S7IEC_S104_31xPN
  - S7IEC_S104_41xPN
  - S7IEC_S104_IM151PN
  - S7IEC_S104_WinAC (LAN interface administrated from WinLC, with Port 2404 only one connection possible)
  - S7_IEC_1500_104_MS_CPU_PN
  - S7_IEC_1500_104_MS_SC

- All SLi- (FB130-FB134) and SLo blocks (FB135-FB137) are based on the certified blocks. These blocks are identical in all SIPLUS RIC libraries.

  All MAo- (FC140-FC147) and MAi blocks (FB148-FB150) are based on the certified blocks. These blocks are identical in all SIPLUS RIC libraries.

 ©SIEMENS AG 2017

Performance Improvement ‚internal Loop processing'

At receipt of blocked messages of each cycle information was removed until now and provided to the evaluation blocks (MAo).

From version V1.5 up to 10 information objects are internal provided per cycle and edited from the MAo blocks. This is carried out independently of the use of the blocks, in Master or Slave communication in standard or reversed direction.

The possibility to improve the performance of the Master libraries described in chapter 4.5 is no longer necessary. It is inapplicable including the related parameter.

see chapter 4.5

Use of time interrupts

The SLi-Blocks (FB130-FB134) may be embedded in a time interrupt additionally to the standard sequence (OB1). This is realized automatically from the blocks and the inputs are scanned only in the time interrupt.

In case of slow processing (e.g. 500 ms for measured values) a release of the standard cycle and therefore a faster communication can be achieved.

Fast time interrupt cycles (e.g. 10 ms) are suitable for gathering indications with high demand of timing. Caution: This is permitted only for a restricted number otherwise there is a possibility for cyclic time faults. The standard sequence is extended as well.

see chapter 6.2

Support of TIA-Portal

The current libraries for the ET 200S, S7-300, S7-400 and WinAC are also available as TIA projects. It is a matter of example projects; the library part is directly available in a program group.

Minimum required: TIA V12 SP1

Support of S7-1500

SIPLUS RIC supports the S7-1500 – Hardware:

- T104-Protocol as Master and Slave via the PN-interface of the CPU.
- T101 and T103-Protocol via the communication modules CM PtP.

see chapter 10

14.4.6. Delivery Version V1.6

Subsequently you will find a list with the essential improvements in the delivery version V1.6

General

Software / Development tools - Simatic Manager Step 7 V5.5 + SP4 - TIA Step 7 V13 SP1 Update For the utilization of SIPLUS RIC minimum the versions mentioned above or newer are required. For the application of CPU410 PCS7 is required. The libraries are available for Simatic-Manager Step7 and TIA Step7. An example project respectively as well.	
Libraries and example projects For SIMATIC manager single libraries for the different hardware platforms (S7_Classic), protocol and Master/Slave are available like before. An example project in each case as well. For project planning with TIA step 7 example projects are available as well. The libraries are reduced to the device families ET 200S, S7-300, S7400, WinAC and S7-15xx. They contain in grouped format all blocks.	see chapter 10

Support of new hardware platforms and interfaces

CPU410 / general CPU400 (H): SIPLUS RIC libraries for S7-400 CPUs can be operated on the CPU410 as well. Both PN interfaces (X5, X8) are supported also within a H-system (CPU in rack 0/rack 1). The selection is carried out by new parameter (L1_Local_Device_Id_Cx). Since the CPU410 doesn't support any MemoryCard the activation has to be carried out exclusively based on the CPU serial number. The required switchover parameter 'Serial_from_CPU' is fed out in all library variants for S7-400. Note: The CPU410 may be used maximal 256kB Ram-Speicher for Online generated Datenblocks. The loading and operating of the CPU410 is exclusively possible with PCS7.	see chapter 4.3
CP443 with Speed-Communication: SIPLUS RIC provides a further library for each T104 protocol Master and Slave supporting the speed communication (internal communication mechanism) which is possible with CP443. The data transmission between CPU and CP is thus more effective. The application is recommended if CP443 Adv is used for the communication. Important: Please observe the necessary adjustments in hardware configuration and at the project planning of the connection in NetPro.	see chapter 4.2.7

 ©SIEMENS AG 2017

S7-1500: all CPU-PN-Interfaces and CP1543 V1.6 supports T104 communication via all PN-interfaces which are available on the CPUs and via CP1543. The interface selection is carried out per channel/TCP-connection by handing over the hardware identifier to the parameter “L1_InterfaceID_Cx” (symbolically possible as well).	Chapter 1500/ET 200SP
ET 200SP CPU/CPU1510/CPU1512 V1.6 supports ET 200SP CPUs (CPU1510, CPU1512). T104 communication can be processed via the PN-interface of the CPU. T101 and T103 communication via the communication module ‘CM PtP’. Please observe the resources of the ET 200SP CPU. It should be operated preferably as small device. Multiple protocol and nodal solutions are only restricted possibly.	Chapter 1500/ET 200SP

New functions

T101-Redundancy according to Norwegian User Convention (NUC101) SIPLUS RIC V1.6 supports T101 according to NUC101 in all modes (Master, Slave, balanced mode, unbalanced mode). This is achieved by cascading two or several Communication-FB (S7-1500 FB10x) in mode 3.	see chapter 5
Background Scan From V1.6 the SLi blocks SLi_SP_DP_s128, SLi_ST_s8 and SLi_BO_s8 send with the standard conformal cause of transmission (COT) = 2 (background scan) instead of COT = 1 (cyclic) at activation of the cyclic data transmission. According to standard at measured values blocks SLi_ME_ABC_s32 cyclic data transmission as well as background scan is possible. The selection is carried out via a new parameter input ‘Background_scan’. With a recognized communication fault the background scan is interrupted. The cyclic recording is continued also in this case. The (AddOn-) nodal block MAoSLi_MD_pDB supports the background scan as well.	see chapter 6.1.2
Individual threshold adjustment A new AddOn block (S7IEC_AddOn_misc/S7_IEC_Set_Threshold_1/FC125) allows the threshold adaption of single measured values. A block output informs about the respective measured value status.	see chapter 7.1

Send test command TI104, TI107

Actuated by one input each the MA_Org_Asdu_n (FB122) block can send test commands TI104 or TI107. The status output is set respectively and reset when receiving the confirmation.

see chapter 6.4.1

Special function

All Communication-FB (S7_IEC_Config) resp. FB10x at S7-1500 have been extended by an input double word 'Special_Functions'. Thus diverse special functions can be activated. Especially a liberally handling with the standard. E.g. no filtering on cause of transmission at the output of indication (MAo_SP). According to standard information with cause of transmission 'cyclic' may not be issued. Using the special function Bit 0 it is permitted again.

see chapter 3.3.2/4.3.2

Master – Slave – switchover/libraries

Now all Communication-FB (S7_IEC_Config) have an input for a switchover between Master (controlling) and Slave (controlled) functionality.

At T104 blocks the switchover parameter is 'L2_is_Master' at T101 blocks 'L2_Master_or_Dir_Bit'.

The classical Master and Slave libraries are further existing. At Communication-FB the switchover input is preallocated respectively and the library includes the typical application blocks (Slave -> signal record, Master -> signal receive and output).

The advantages are the fast switchover for test purposes as well as the possibility to use Master and Slave simultaneously without a second Communication-FB.

see chapter 3.3.2/4.3.2

Aging monitoring commands

From V1.6 the time in the command may be up to 1 min in the future (instead of 1 sec like until now)

Optimizing/Remanence in S7_1500

An AddOn block (S7_IEC_Rem_Mem) allows to operate the storage space intensely instance-DBs of the FB10x blocks in the non remanence memory area.

Cascading

From V1.6 CascMode 1 is supported (project specific). An AddOn block serves for buffering.

Cascading combination with mode 3 is now possible.

see chapter 5

 ©SIEMENS AG 2017

Important bug-fixes

F1503:

T101-protocol: Redundancy in Casc-Mode 3 did not work

The Casc-Mode 3 works properly at T101 according to redundancy description of the Norwegian User Convention NUC101

F1507:

In cascade operation Mode 0 with more than one cascade block it occurred sporadically that individual messages were not sent on all channels.

An upgrading is recommended. Alternatively Cascade Mode 2 can be operated.

F1512:

Fault indications like ‚Activation not occurred/Demo time expired' appeared in the CPU15xx at a large number of connections (FB10x).

An upgrade is recommended.

14.4.7. Delivery Version V1.6 Upd4

SIPLUS RIC V1.6 Upd4 mainly contains adaptations and extensions for S7-15xx and is therefore only available for these components.

Software environment / development tools

Libraries and sample projects of the S7-1500 are delivered in TIA Step7 V13 SP1 and TIA Step7 V14.

In addition to the updated libraries, updated sample projects are also available.

In both, a restructuring was made to include another group with the name of the SIPLUS RIC version. This means that the pure SIPLUS RIC library percentage is always summarised in one group.

see chapter 10

You can now copy the entire folder, SIPLUS RIC ... from the library into your project and delete unnecessary subgroups there if necessary, or, as before, transfer only the required groups into your project

S7-1500 Software Controller

Support (new) HW platforms and interfaces
S7-1500 software controller
A new library "SIPLUS RIC IEConS7 15xxOC_V1_6_Upd4" supports the S7-1500 software controller version V1.8

- ET 200SP open Controller (CPU 1505S)
- Software Controller (CPU11507S), tested on IPC427d

The protocols IEC60870-5-101, 103 and 104 can be used in their entirety. Released and tested are:

see chapter 1.4

CP1542

Is also supported and can be used by assigning the corresponding HW ID for T104 communication.

see chapter 4.2

Data blocks S7-15xx generated during runtime

SIPLUS RIC generates some data blocks internally with program start. A somewhat different system behavior of the S7-1500 (compared to S7 classic) was not completely compensated so far and could lead to incorrect behavior, e. g. after voltage failure and / or when several FB10x instances are used.

see chapter 11.2.9

Observation tables

The observing tables "VAT_S7_IEC _...", which are included in the libraries and example projects, display the serial number relevant for the release code directly as a string.

Serial_Device_ID shows the HW ID of the device from which the serial number is read:

- 49: Libraries for S7-1500 Software Controller -> 'Local CPU'
- 51: Libraries for S7-1500 CPUs -> SMC (Simatic Memory Card)

The release code is still based on the displayed serial number and the product ID

Supported basic cycle OBs

In addition to OB1 as a basic cycle, S7-1500 permits further OBs or other OB numbers – also as a basic cycle or its extension.

SIPLUS RIC should continue to be operated in the basic cycle. In exceptional cases (small quantity structure), SIPLUS RIC (all block calls!) can also be operated in a cyclic interrupt – the start-up and related start-up initialization is also detected and carried out in cyclic interrupts.

Symbol access HMI

The attributes "Visible in HMI" have been deactivated in all SIPLUS RIC blocks (FB and DB).
This avoids the unnecessary provision for HMI of non-relevant information points

Function time stamping with local time

- Can be activated with a new parameter (default local time is used)
- In Samples FC121 is not included, but the IV-bit must be reseted.

see chapter 7.2

Fixed bugs

F1625 / FB122 (MA_Org_Asdu_n) does not always send general interrogation.

FB122 in SIPLUS RIC V1.6 does not always send GA-requests to the substations as master after connection interruptions.

 ©SIEMENS AG 2017