Introduction

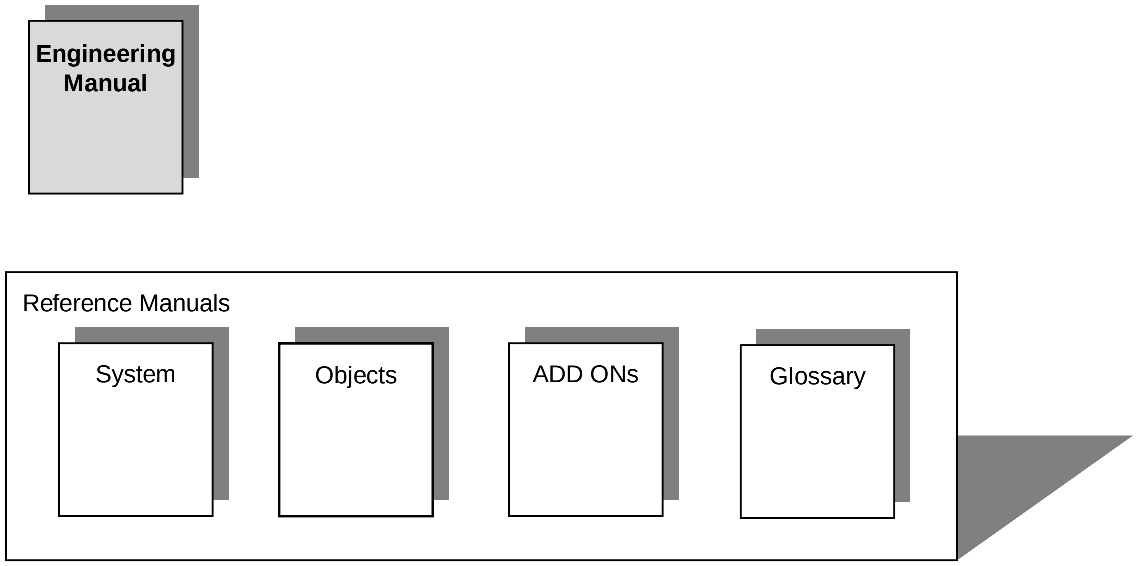
Content

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\01_Introduction_009.doc

General

This is the **Engineering Manual** for CEMAT V8.2. It should support you in performing the work required to configure your plant.

The **Engineering Manual** is part of a comprehensive CEMAT V8.2 documentation which consists of the volumes listed below.



After the installation of CEMAT V8.2 the CEMAT documentation is available as PDL in directory D:\CEMAT_CS\Docu

On the following pages you will find the content of each manual.

Documentation structure

The manuals contain the following chapters:

	Engineering Manual
1	Introduction
2	Preparations
3	Installation of a PCS 7 Project
4	Assignments
5	Engineering Examples
6	PLC Engineering
7	AS-AS Coupling
8	OS Engineering
9.	Engineering Tools
10	Engineering Check List
11	Project Administration
12	free
13	Graphic Templates
14	Tips&Tricks
15	Update Information

	Reference - System
1	Introduction
2	Cemat System Architecture
3	
4	
5	
6	

	Reference - Glossary
1	Introduction
2	Definitions and Instructions

	Reference - Objects
1	Introduction
2	General
3	Unidirectional Drive
4	Bi-Directional Drive
5	Damper
6	Valve
7	Adapt
8	Annunciation Module
9.	Annunciation Module with 7 Alarms
10	Measuring Value
11	Measured Value Integrator
12	Storage
13	Storage Multichamber
14	Silopilot
15	Group Module
16	Route Module
17	Selection
18	Interlock with 8 inputs
19	Interlock with 4 inputs
20	Interlock
21	Interlock 5
22	Object Data Acquisition
23	PID Controller
24	PID Controller with 3 Parameter sets
25	Polygon 3
26	CNT-Module
27	RT-Module
28	Related Objects
29	Analog Value Selection
30	

	Reference – ADD ONs
1	Introduction
2	
3	Adapt to Simocode PRO
4	Adapt to Simocode (advanced)
5	Robicon
6	Sinamics
7	
8	Parameter Management

Guide

The current manual is divided into the topic areas: Installation, Engineering and Project Administration.

The engineering of CEMAT plants is done with the Engineering Tool PCS 7. With PCS 7 the following steps are carried out.

- Configuring the hardware
- Creation of the PLC programs, including logic diagram (CFC) and documentation
- Transferring the PLC/OS Connection data
- Engineering of the Operator Interface in WinCC

Preparations

Content

General

This document contains rules and thoughts you have to consider before you start with the Engineering of a Cemat Project.

Please also consider the check list at the end of the document. You should be able to answer all Questions before you start with the Engineering.

Order Information

When you read this document you probably have ordered Cemat already.

Nevertheless we want to mention that for any doubt you can ask the Competence Center for Order Information.

Based on the Plant Configuration and the number of plant objects the Hardware and Software requirements and the necessary Licenses can be calculated. Configuration Examples will help you in the plant design.

Project Standards

During the installation of Cemat you have to select the correct project key dependent on the desired project standard.

000 = CEMAT Standard
001 = CEMAT Minerals
004 = Holcim
006 = Dyckerhoff
007 = Heidelberger Zement
023 = Vigier
024 = Bushehr
025 = Caima
026 = Alsen
027 = Lafarge
028 = Rossi

The main difference between the project standards are the change of operation mode and the evaluation of the local switch. Make sure that the selected project standard fits to the requirements of the plant (MCC / Plausibility, Local switch, Operation modes).

With Cemat V8 a new project standard was added: **001 = CEMAT Minerals**. The project standard 001 comprises all requirements from Cement and Mining industry.

Caution: The migration from existing project standards to project standard 001 is not possible because the block interfaces are completely different!

With project standard 001 the change of operation mode and the local switch function can be adapted to the requirements of the plant. Via Feature bits and OS Permissions nearly all variants of the existing project standard can be parameterized.

Exception: The functions of the Lafarge Standard (027) can not completely be realized with project standard 001!

The name of the project standard is often derived from the Company or Group for which the project version was designed. In some cases it is just the name of the first plant where it was used.

Important: Please consider that in case of a Project standard for a Group this does not automatically mean that each plant of this group uses this version. Some plants were bought by a group when there were already running Cemat installation. For the sake of compatibility it may be better to continue with the existing Cemat Version.

Caution: In some occasions, additional adaptations may be needed for the evaluation of the signals. These adaptations have to be achieved through a specific logic in the application program.

On the following pages you find a description of the project standards with its main distinguishing characteristics.

If you furthermore have questions regarding project versions, please don't hesitate to ask in the Competence Center.

Main classification of the Project standards

Classification by Operating Mode

Code	Standard	Main Characteristics
000 007 023 024 025 028	“normal” Heidelberger Zement Vigier Bushehr Caima Rossi	Enable for single-start mode and local mode from group faceplate
004 006 026	Holcim Dyckerhoff Alsen	Single start mode and local mode can be enabled individually for each drive
027	Lafarge *)	Only Sequential mode and Local mode. Change of Operation mode individually per Object.

***)** Lafarge has a completely different operating philosophy and different block names and can not be compared with other project standards.

Classification by Local Switch

Code	Standard	Main Characteristics
000 024 028	“normal” Bushehr Rossi	Local/Automatic switch Signal “EVO”
004 006 007 023 025 026	Holcim Dyckerhoff Heidelberger Zement Vigier Caima Alsen	Local mode is enabled through Operator Station (Faceplate buttons)
027	Lafarge *)	Local mode is enabled through Operator station.

***)** Lafarge has a completely different operating philosophy and different block names and can not be compared with other project standards.

Note: With Project Standard 001, all kind of operation mode change and local switch functions can be achieved. An additional Engineering manual exists.
Using Project Standard 001 for Lafarge Plants has not been approved yet.

Project key	Main characteristic and description	Object types involved
000 normal	<u>Operation modes</u> - Automatic mode, Single-start-mode and Local mode can be switched only group-wise - Local/Remote-switch xVO must be "1" in automatic mode and "0" in local mode. - Local Start via xSR, Local Stop via xSP. - In Automatic mode and in Single-start-mode the Local stop signal xSP is by default not effective (can be configured). - In Automatic mode and in Single-start-mode all interlocks are effective. - In Local mode only the protection interlock xSVG is effective.	
001 Minerals	<u>Operation modes</u> - Operation mode can be changed individually or via superordinated group. Can be parameterized via feature bits and OS Permissions - Manual modes (Single-start-mode) with different interlocking grades (for reduced interlocks) - Via feature bits different versions of local switches can be configured. - Additional mode: Out of Service <u>Additional functions</u> - all block parameters are in English - Adaptation to the functions of the APL. <u>Group functions</u> - "Group not empty" indication - Interlocks can be by-passed - Group-Drive communication between PLCs <u>Drive functions</u> - Essential interlocks (effective also in local mode) - Separate block for Process feedback supervision - Contactor feedback supervision for damper and valve possible. - Calibration function for the damper - Limit switch simulation for the valve - Emergency stop via faceplate possible. - Annunciation blocks and measuring values can be assigned directly to the drives. for details see Manuals for CEMAT Minerals	All blocks!

Project key	Main characteristic and description	Object types involved
006 Dyckerhoff	Operating modes - Dyckerhoff Standard has an automatic mode, a single-start-mode and a so-called manual mode. - In the single-start-mode all interlocks are effective and it is used for individual start/stop from the operator faceplate. - The manual mode is a non-interlocked operation and it is mainly use for local operation. In some exceptions it is also used for start/stop via operator faceplate (special rights required). - All drives can be switched individually into the respective modes The group only indicates that at least one drive is switched to single-start mode or to manual mode. - It is also possible to switch all drives group-wise back into automatic mode. Running drives continue to run after switching back to automatic mode. - In local mode Start-Button must remain pressed to run the drive. - Local mode is seen as a test mode (only for commissioning and repair works). Speed monitor must be acknowledged from local - additional acknowledge signal EVQD required Different Parameter names and additional interface flags - ESB=ESB, EBM=MSB, ESD=DWS etc. - The parameter names are used for FB and OS	C_DRV_1D C_DRV_2D C_DAMPER C_VALVE C_ROUTE C_GROUP

Project key	Main characteristic and description	Object types involved
007 HZ	Operating modes - The operating modes Automatic mode, Single-start-mode and Local mode are enabled by the group. Local mode for the drives is by default enabled. If the group signal is not connected the drive is always enabled for the local mode. - Motor, Damper, Valve: Local start with push-button xVT - With EVO = 0-Signal the drives will be stopped. EVO must be "1" in all operating modes. Additional interface ENOT for Emergency Stop - ENOT = 0-Signal will stop the drives in all modes The measured values can be switched into service mode - In the service mode the actual value of the measure is still displayed but the measure will not give any alarms and all Module outputs for limit violation are switched to "0". Additional mode non-interlocked individual start - In this mode the drives can be started/stopped via operator faceplate without evaluation of the interlock conditions for EBVG and EEVG. ESVG is still effective. - The non-interlocked individual mode needs a special release (different rights). - The symbol color turns to orange (orange flashing for running objects).	C_DRV_1D C_DRV_2D C_DAMPER C_VALVE C_MEASUR
026 Alsen	Operating modes - Local mode or Single start mode for Drive, Damper and Valve can be released individually for each object (at the drive faceplate). - But via group faceplate it is also possible to switch all objects of the group into local mode or back to automatic mode. - The group symbol has a summarizing indication that at least one object has been switched to local mode. - Switching from local mode to automatic mode already started drives continue to run as long as all operating conditions and interlocks are fulfilled. Characteristics for Local Switch Signals xVO and xSP - xVO must always have 1-Signal. 0-Signal at xVO will stop the drive in all modes - Local stop xSP is effective also in automatic mode. Holderbank Code - The Holderbank code HAC is not used for function block parameter names. The normal abbreviations ERM, EVO, ESB are used.	C_DRV_1D C_DRV_2D C_DAMPER C_VALVE C_VAL_2D C_GROUP

Project key	Main characteristic and description	Object types involved
004 Holcim	The main differences between Normal Standard and Holcim Version are the Parameter Names for Hardware signals (HAC code is used), the Operating modes and the representation of the plant objects in the pictures. HAC Code - Block parameters for Hardware Signals (MCC and Field) are changed to HAC code Operating modes - Each drive/damper/valve can individually be switched to single or local mode. - Switching to local mode and single mode is only possible if the drive is not running. - Local stop is also active in automatic mode (Alarm LST). - Through a button in the group faceplate all drives of a group can be switched to local or back to AUTO mode simultaneously. With the start command of the group all drives are automatically switched back to AUTO mode. Drives which were previously started in single start mode or in local mode will continue to run. Block icons - In Holcim standard no additional motor symbols are shown beside the equipment. The equipment itself is green if it is running and grey/white if it is not in operation. Next to the equipment there is the Tagname which is the block icon. The faceplates will be opened by clicking on the tagname. The tagname shows red background if there is a fault, yellow background in case of a warning and violet background in case the object is not available. Additional indications beside the tagname show the operating mode (blue for single and white for local) and active simulations or bypass functions (orange) Alarms - The non-essential drives can be configured as “warning drives”. If these drives have a fault, a yellow warning message is given. The group start is not interrupted by warning drives. Interlocks - In Single mode the Start Interlock and the Operating Interlock are not active. - Two Protection Interlocks PINT1 and PINT2, where PINT1 is always active and PINT2 can be by-passed in local mode (Inching mode). Process Feedback - Additional block C_PROFB can be used for supervision of process Feedback (Speed Monitor or Pressure). C_PROFB has to be linked to the drive block. Special Training for Holcim - The Engineering for Holcim Standard is different from the normal Standard. Therefore a separate Training is available. HDRS Engineering Tool - An engineering Tool (Excel Macros) is available. With this tool, based on the HDRS data base from Holcim, Basic CFCs can be created using Process Tag Types (IEA Assistant)	C_DRV_1D C_DRV_2D C_DAMPER C_VALVE C_ANNUNC C_ANNUN8 C_PROFB C_ROUTE C_GROUP C_SILOP C_ADAPT

Project key	Main characteristic and description	Object types involved
024 Bushehr	<u>Different Local switch with only 2 Signals</u> - Instead of using 3 Signals for xVO = Local/Remote Switch, xSR = Local Start and xSP = Local Stop, the same information is built internally in the function block, using only two signals which have to be seen in combination. - The Signal names are K0 and KX (KY) and they must always be seen together. - Signal coding for the local switch of the drive: K0 KX 1 0 Auto 0 0 Local Stop 0 1 Local 1 1 Local Start - Signal coding for the local switch of the damper: K0 KY KX 0 1 1 Auto 0 0 0 Local Stop 1 1 0 Local Start Di2 0 1 0 Di2 1 0 1 Local Start Di1 0 0 1 Di1	C_DRV_1D C_DRV_2D C_D_REV C_DAMPER C_VALVE
025 Caima	<u>Different Local switch with only 2 Signals</u> - Instead of using 3 Signals for xVO = Local/Remote Switch, xSR = Local Start and xSP = Local Stop, the same information is built internally in the function block, using only two signals which have to be seen in combination. - The Signal names are xSR and xVO like in normal Standard but the signals must always be seen together. - Signal coding for the local switch of the drive: EVO ESR 1 0 Auto 0 0 Local Stop 1 1 Local Start - Signal coding for the local switch of the damper: KVO KOP KCL 1 0 0 Auto 0 0 0 Local Stop 1 1 0 Local Start Di2 1 0 1 Local Start Di1 - After the local start the local switch goes back to position Automatic. The drive is then running in local mode. In order to Stop the drive the local stop must be pressed. This position is latched. - Local Stop is also effective in Automatic mode. The drive stops then with Alarm "LST".	C_DRV_1D C_DRV_2D C_DAMPER C_VALVE

<table>
<tr><th>Project key</th><th>Main characteristic and description</th><th>Object types involved</th></tr>
<tr><td>023 Vigier</td><td><u>Different Parameter names:</u>
G = Local Start
S = Local Stop
K = Motor Protection
R = Feedback
etc.
no Signals for sSB and xVO</td><td>C_DRV_1D
C_DRV_2D
C_DAMPER
C_VALVE
C_GROUP</td></tr>
<tr><td>027 Lafarge</td><td><u>Different Module types for all Control functions</u>
- C_M2B (to be used for unidirectional and bi-directional drives)
- C_DAB (to be used for Valves with one output and two limit switches, no local switch)
- C_DABMAB (to be used for valves with two outputs (DAB) and motorized actuators without positioning mode(MAB))
- C_AAB (to be used for damper positioners)
- C_DIB (to be used for annunciations of any binary signal)
- C_AIB (to be used for analog inputs for measuring values)
- C_SSB (to be used for sequences)
- C_SSDB (to be used for Sequential Stop including stop delays)
- C_BPB (to be used for subcontrols or "black boxes")
<u>Different operation concept and control philosophy</u>
- Sequences replace the CEMAT Group function, but group status call and group object list are available as in normal CEMAT standard.
- No route modules used. C_SELECT can be used for selections.
- Plant section (sequence) turns into "sleep mode", if it is not in use.
- Before start of a Sequence, it must be checked for availability through button AV.
- Pressing AV does a wake-up. If all drives are available (all essential interlocks ok), the sequence can be started via RUN.
- Normal Stop (single click on STOP) and Quick-Stop (double-click on STOP) possible. Stop sequence and stop delays are programmed with SSDB block.
- Lafarge Standard has a variety of different interfaces for individual interlock signals.
- No single-start-mode via operator faceplate
<u>Special engineering manual</u>
- Different engineering rules for Installation of a PCS 7 Project, AS Engineering, OS Engineering.
<u>Special Training Course</u>
- As Lafarge Standard is so much different from normal CEMAT, a special Course is available for Lafarge.</td><td>C_M2B
C_DAB
C_DABMAB
C_AAB
C_DIB
C_AIB
C_SSB
C_SSDB
C_BPB</td></tr>
<tr><td>028 Rossi</td><td><u>Waiting time for single start-up-warning</u>
- In single-start-mode, the drives are switched on after the horn is off and the waiting time has elapsed (as it is in the group).
<u>Alarms for xBM and xSB in local mode</u>
- In the normal standard, in local mode no alarm will be generated for xBM and xSB.
In Rossi Standard, alarms will be generated also in local mode.</td><td>C_DRV_1D
C_DRV_2D</td></tr>
</table>

Project language

At the beginning of the engineering you have to decide which language is going to be used for the engineering and which display language(s) are required.

We recommend you to carry out the engineering in the main display language.

The following list shows you, which project standard is available in which language as per standard. At this we distinguish between Display language and language of the documentation:
(In some project versions the display language can be switched into French but the French documentation is not available.)

Project version	**Software (Display Language)**	**Documentation**
000 = CEMAT	German + English (USA) + French + Italian + Spanish (international)	German + English
001 = CEMAT Minerals	English (USA)	English
004 = Holcim	German + English (USA) + French + Italian + Spanish (international)	English
006 = Dyckerhoff	German + English (USA) + French + Italian + Spanish (international)	German + English
007 = Heidelberger Zement	German + English (USA) + French + Italian + Spanish (international)	German + English
023 = Vigier	German + English (USA) + French	German
024 = Bushehr	German + English (USA) + French + Italian + Spanish (international)	English
025 = Caima	German + English (USA) + French + Italian + Spanish (international)	English
026 = Alsen	German + English (USA) + French + Italian + Spanish (international)	German + English
027 = Lafarge	English (USA) + French + Spanish (international)	English
028 = Rossi	German + English (USA) + French + Italian + Spanish (international)	English

Additional languages can be realized as well. In this case the Texts in the Faceplates have to be completed. To simplify this work, a text library with the faceplate text is available. The file Language_V01.xls you find on the Cemat CD under Additional Information\Language. (Please also ask in the Competence Center Cement for the latest version of this file). '
The translations are carried out in the text library and can than be imported into the Faceplates with the Text Distributor.

Beside the Faceplate texts, for additional languages adaptations of the blocks (messages) and in the Config files may be required.
See also under "Text definitions" in Chapter AS Engineering.

Designation rules for Cemat

Naming conventions for PCS 7

Depending on the language and PCS 7 components, only certain characters are permitted in names. Make sure that you don't use illegal characters in your PCS 7 Project (e. g. for Computer names, chart names, block types and instance names, parameter names and comments, global variables……)

The restrictions for the names of the PCS 7 components you find in the Help function of the WinCC Explorer. Use *Help → Contents and Index* and search for "Illegal characters". There you will find a list of characters which are not allowed for the individual components.

Naming conventions for CEMAT

Name of the PC Station

For the computer name follow the restrictions of PCS 7 V7!

In the PCS 7 Manual "PCS 7 PC – Configuration and Authorization" you find the following note: **Use only letters and numbers for the computer names.**

Additional notes you find in the Online help of WinCC. Special characters are not allowed in the PC Name.

The following characters <u>are not allowed</u> in PCS 7 and Cemat:

. , ; : ! ? " ' ^ ´ ` ~ - + = / \ ¦ @ * # $ % & § ° ( ) [ ] { } < > _
blanks
only uppercase relevant

The first character must be a letter!

Example for a valid Designation system:

Station name	Info bus: TCP/IP	Plant bus: MAC	Plant bus: TCP/IP *)
ENG01	192.168.1.10	Address marked on card	192.168.2.10
ENG02	192.168.1.11	Address marked on card	192.168.2.11
SERVER1A	192.168.1.20	Address marked on card	192.168.2.20
SERVER1B	192.168.1.21	Address marked on card	192.168.2.21
SERVER2A	192.168.1.22	Address marked on card	192.168.2.22
SERVER2B	192.168.1.23	Address marked on card	192.168.2.23
.....			
OSC01	192.168.1.101		
OSC02	192.168.1.102		
OSC03	192.168.1.103		
OSC04	192.168.1.104		
....			

*) TCP/IP should be deactivated in case of Softnet CPs.

Tagnames

The tagnames are plant wide unique designations for the plant objects. With this name the plant object can be identified unequivocally.

A meaningful tagname contains the designation of the plant section, the designation of the equipment, the object type and a serial number.

e. g.:	Plant Section:	Clinker transport:	CT01
	Equipment:	Bucket Elevator 3	BE03
	Object type:	Motor	M
	Serial Number :	01	01

In PCS 7 this name is built by the chart name and the instance name (=block name), separated by a Slash (/). Optionally the hierarchy folder names can be used as a prefix. (This is not recommended for Cemat because in this case the tagnames get too long.

If the Plant Section and the Equipment Name is used as chart name and the Object type plus Serial Number as Instance name, the example above will result in the following tagname:

CT01_BE03/M01

The tagname is used in the facaplates and in the message system. The maximum length is 20 characters. As special characters only the underscore should be used!

Note: The name of the motors, dampers, valves, groups, routes, selections and controllers should not be more than **13** characters.
Names for Annunciations and measuring values can reach a length of **20** characters (if a separator and an extension of max. **6** characters is added to the name of the main module).

Example for the Tagname of a Belt drift switch:

	CT01_BC01/M01_DS01	(If the Signal is assigned to the motor)
or	CT01_BC01/DS01	(If the Signal is assigned to the equipment)

Note: The above description is only an example. Of course you have to consider the Tagging rules for particular companies.

Comment

The block comment is used for the description of the plant object. This can be for Example:

Bucket Elevator Main Drive

The block Comment is used in the Faceplates and (by copying) in the message system.
In most of the Cemat Faceplates there are two lines of 20 characters each, which means the maximum length of the comment can be 40 characters.

Note: For the facaplate display an automatic word wrap is used. If possible the system avoids cutting in the middle of a word but in case of long words this may still happen sometimes!

Message texts (Event texts in WinCC)

In most of the Cemat Objects the message texts are standard and they are defined at the function block and in the config files. The maximum text length is 20 characters.

For the Annunciation blocks C_ANNUNC and C_ANNUN8 the Message text must be defined per Instance. For technical reasons the text length must not exceed 16 characters.

Signal names

The maximum length for global symbols in PCS 7 is 24 characters. The signal names for the inputs and outputs of the CEMAT modules should be derived from the module tags.

The maximum length of the signal comments is 80 characters.

Example:

123_BE1/M01_ERM	Bucket Elevator Main Drive Contactor Feedback
<------24 characters---->	<----------------------------------80 characters------------------------------>

Plant view

One of the most important decisions at the beginning of the engineering of a Cemat Project is the structure of the plant view. From the plant view the process control interface is derived.

The hierarchy level which is defined as "OS Area", is relevant for the Selection Buttons in the Overview Range and for the assignation of Access Right.

Example:

The first hierarchy folder was selected as “OS Area” and it contains the Hierarchy folders Crusher, RawMill, Kiln, CementMill and Packing.

In the Overview Area of the Runtime System you will see the buttons: Crusher, RawMill, Kiln, CementMill and Packing for direct Picture calls.

If an Operator has only the Authorization for the areas Crusher and Raw Mill, he is only permitted to watch and to operate these areas.

The Messages in the Alarm line and the acknowledgement function is filtered in the same way via the OS Area.

Caution: Because of the filter functions in the CEMAT Message system the maximum number of Areas buttons is restricted to 30.

Object Types

The definition of object types (e. g. bucket elevator, belt conveyor) with all the associated modules and interlocks will save engineering effort. Approved solutions can be used several times, whenever the function is required.

In the Cemat Library ILS_CEM you find an S7-Programm TYPICALS with some examples of typical Cement Applications. Modify these Examples according to your requirement in order to define them as "Process tag type" and to use them with the Import-/Export Assistant.

The following Examples are available and can be adapted and completed according to your requirements:

BC	Belt conveyor with speed monitor, rope switch, drift switch and current measurement.
BC_IL	Belt conveyor with speed monitor, rope switch, drift switch and current measurement, including Interlock blocks
BC_IL_71	Belt conveyor with speed monitor, rope switch, drift switch and current measurement and Pcs7AnIn, including Intlk02 blocks
BC_REV	Reversible belt conveyor with speed monitor, rope switch, drift switch and current measurement.
BC_REV_IL	Reversible belt conveyor with speed monitor, rope switch, drift switch and current measurement, including interlock block.
BC_REV_IL_71	Reversible belt conveyor with speed monitor, rope switch, drift switch and current measurement and Pcs7AnIn, including intlk02 block.
BE_AUX	Bucket elevator auxiliary drive
BE_MAIN	Bucket elevator main drive with speed monitor, level switch, drift switch and current measurement
BE_MAIN_71	Bucket elevator main drive with speed monitor, level switch, drift switch and current measurement and Pcs7AnIn
CC	Chain conveyor with speed monitor, rope switch and current measurement.
CC_IL	Chain conveyor with speed monitor, rope switch and current measurement, including interlock block
CC_IL_71	Chain conveyor with speed monitor, rope switch and current measurement and Pcs7AnIn, including Intlk02 block
DAMPER	Standard damper block with torque switches
DAMPER_POSI	Damper Positioner Function
DAMPER_POSI_71	Damper Positioner Function
MOT_1D	Standard Unidirectional drive with speed monitor
MOT_1D_71	Standard Unidirectional drive with speed monitor
MOT_1D_MEAS	Standard Unidirectional drive with speed monitor and current meas.
MOT_1D_MEAS_71	Standard Unidirectional drive with speed monitor and current meas.
MOT_2D	Standard Bi-directional drive with speed monitor
MOT_2D_71	Standard Bi-directional drive with speed monitor
MOD_2D_MEAS	Standard Bi-directional drive with speed monitor and current meas.
MOD_2D_MEAS_71	Standard Bi-directional drive with speed monitor and current meas.
VALVE	Standard Valve

Caution: Please consider that these examples were created for normal standard (Project key = 000). If you want to use the examples in a project with a different project version, you have to adapt the examples accordingly.

In chapter "Engineering Examples“ of this manual you find solutions for special tasks.

Engineering Rules

Per plant the responsible engineering department must determine the project definitions. Henceforth these unique definitions are valid for all AS of the plant.

Especially if the engineering is done at different locations and for supplementary plant extensions it has to be insured that the original project definitions are used.

Definitions for the Engineering:

- Name of the PC Stations
- Definition of AS Numbers
- Addressing of the AS and PC Stations (MAC Addresses, TCP/IP Addresses, Profibus)
- Definition of the Project Standard and Version
- The designation system of the plant tags (charts, modules, signals)
- Definition of the message texts (Event)
- Variable names
- Names of the WinCC pictures
- Rules for the generation of scripts and actions
- Engineering rules (Customer standards, Teamwork)
- Definitions for the documentation of a project

Definitions for the Runtime-Project:

- The operation surface (Division of the screen, Font style and –size, Language in Runtime, Representation of the objects)
- The operation concept (Picture hierarchy, operating philosophy, user rights, permitted keys)
- The color definition for the annunciations, Limit values, Status, Font etc.
- The communication (Communication type, Actualization cycles)
- The volume of project data (Quantity of alarms, Archive values, Curves, Clients etc.)
- Message and archiving procedure (Archive size and Export Concept)
- Archive names per Area

Verification of the received documents

The received documents for the project must be examined, verified and if necessary completed. These are:

- Object list (List of drives, measures, groups, etc.)
- Signal list (with or without absolute addresses)
- Plant Topology (Building, Location)
- Process view (Plant, Unit, Function)
- Electrical view (MCC, Panel, Feeder, Bimetal, Fuse Rating)
- Panel Structure (I/O-Panels, Rack, Slot, Card, Address)
- Group/ Route/ Drive assignments

Project Structure

Assignment of the Objects to the AS

- Sensible (technological) division of the objects
- Communicate as little data as possible between the individual AS
- If possible, do not separate groups

Determine the number of AS required

- Maximum number of objects per AS
- Maximum number of signals per AS
- Make allowance for reserves
- See table for module data in the reference manual, objects, chapter 1, General.

Structure of the plant from different views

- Plant topology (Locations)
- Process view (Technological structure)
- Control system view (Control systems, AS, I/O-Panels etc.)
- Electrical View (MCC)

Planning the Process Control Interface

Configuration Guidelines

Summary of the steps required to create the process control interface:

	Description of the action	Executor
1	Determine which plant sections of the process are to appear on the screen.	Customer / project management
2	Divide the complete plant into sub-processes that are to appear together in a diagram.	Customer / project management
3	Determine the general rules for the process diagram creation.	Project management
4	Process diagram assignment for the menu entries.	Refer to the next pages
5	Determination of the general representation rules for the plant.	Customer / project management
6	Determination of the color assignment, blinking, font size and form.	Customer / project management refer to the default picture
7	Determination of the operation philosophy.	Customer / project management CEMAT standard
8	Determination of the representation of the block icons	Customer / project management CEMAT standard
9	Determination of the representation for typical process-relevant elements.	Customer / project management CEMAT standard
10	Design the process diagram layout on paper or drawing program or directly using Visual Basic.	Customer / project management
11	Determination of the directories / file names for process diagrams, key assignments, graphical modules, etc.	CEMAT standard

What is shown WHERE and HOW?

Complete plant overview

Representation:	Schematic representation of the complete plant
Content:	Overview of the production plant with global status display from the running plant sections
Operation:	Selection of a production section / a sub-plant

System monitoring, plant configuration diagrams

Representation:	Schematic representation of the complete control system
Content:	Overview of the network connection of the individual control system devices with global status display of the individual components.
Operation:	Selection of an individual component for detailed analysis

Plant sections:

Representation:	Material flow of a complete production plant without secondary aggregates and details
Content:	Display of all physical variables relevant for the process control.
Operation:	Start and stop of the plant Specification of set points Assignment of recipes Invoke overviews for flow diagrams, measured values, closed-loop controllers, curves Tabular measured value listing

Process Pictures:

Representation:	Representation of the plant sections with all aggregates
Content:	Display of the operational modes, measured values, states, curves, filling levels
Operation:	Selection, operation of individual aggregates and diagnosis, information and object parameter invocations

Detail Pictures:

Representation:	Representation of the individual aggregates
Content:	Display of the operational modes, measured values, states
Operation:	Selection, operation of individual aggregates and diagnosis, information and object parameter invocations

Help:

Representation:	User’s guide for the CEMAT system User’s guide for the plant User key assignment
Content:	Explanations and examples of the system handling.
Operation:	Complete table of contents and search key control

Check list

Step	Process	Status
0	**Verifying the received documents**	
0.1	Object list	
0.2	Signal list (with or without absolute addresses)	
0.3	Plant Topology (Building, Location)	
0.4	Process Overview (Plant, Unit, Function)	
0.5	Electrical view (MCC, Panel, Feeder, Bimetal, Fuse Rating)	
0.6	MCC Types	
0.7	Group-, Route-, Drive Assignment	
0.8		
1	**Definition of the Project structure**	
1.1	Structuring the plant based on technological functions	
1.2	Definition of plant sections	
1.3	Assigning the plant objects to the plant sections	
1.4	Locations (AS, OS, MCC, Periphery)	
1.5	Assigning the plant objects to the locations	
1.6		
2	**Calculation of the required Hardware and Software**	
2.1	Variable calculation	
2.2	Definition of the ES	
2.3	Definition of the AS (CPU Type)	
2.4	Definition of the Periphery	
2.5	Definition of the OS (Server, Clients, CAS, WebServer)	
2.6	Definition of the Bus Components	
2.7		
3	**Ordering Hardware and Software, including Licenses**	
3.1	Ordering the Hardware	
3.2	Ordering the Software for PCS 7	
3.3	Ordering the Software for Cemat	
3.4	Miscellaneous	
3.5		
4	**Names and Addresses in the System**	
4.1	Definition of the MAC Addresses	
4.2	Definition of the TCP/IP Addresses	
4.3	Definition of the Profibus Addresses	
4.4	Definition of the AS Numbers	
4.5	Designation of the PC Stations	
4.6	Designation of the AS	
4.7	Designation of the Periphery	

Step	Process	Status

5	**Decision for the Project standard**	
5.1	Evaluation for the Cemat Project Version to be used in the Project.	
5.2	Additional adaptations required in the User Program to adapt the Program to the MCC/Local Switch Signal?	
5.3	Are additional messages required (e. g. SIMOCODE, DRIVE_ES or Subcontrols)?	

6	**Project language**	
6.1	Definition of the Project language	
6.2	Definition of the Display language	
6.3		

7	**Tagging System**	
7.1	Tagging system available from the customer?	
7.2	Any special demands for Tagnames from the Machinery supplier?	
7.3	Plant Update or Migration: How can the existing Tagging System be converted into PCS 7 Tagnames?	
7.4	Are the Standard fault texts OK or is a modification of the texts required?	
7.5	Definition of the fault texts for C_ANNUNC and C_ANNUN8	
7.6	Definition names for the WinCC Pictures	

8	**Project specific modifications of the Standard**	
8.1	Adaptations of the Blocks (e. g. fort he message system)	
8.2	Adaptations in the Config Files	
8.3	Adaptations of the Faceplate (e. g. for an additional language)	
8.4	All Modifications must be saved and restored in case of a Cemat update!!!	

9	**Definition of the Process Control Interface**	
9.1	Definition of the plant sections "Areas" for the Overview	
9.2	Determine which plant sections of the process are to appear on the screen.	
9.3	How many levels should be used?	
9.4	Definition of the Process pictures and Detail Pictures.	
9.5	Assigning the process pictures to the area	
9.6	Design of the process pictures.	
9.7	Definition of the Operating Philosophy	
9.8	Definition of the block icons (how shall the plant objects be displayed)?	
9.9	Definitions, how to display typical technological element.	

Step	Process	Status

10	**Structuring the PCS 7 Project**	
10.1	Creating a Guideline with engineering rules (especially for bigger projects).	
10.2	Technological Hierarchy (Plant View)	
10.2.1	Definition of the Hierarchy structure (all levels)	
10.2.2	Definition of the Hierarchy folder names	
10.3	AS	
10.3.1	Definition of the plant names	
10.3.2	Definition of the Instance names (block names)	
10.3.3	Definition of the signal names	
10.3.4	Definition of Object types (Process Tag Types)	
10.3.5	Definition of rules for additional Texts (C_ANNUNC and C_ANNUN8)	
10.4	OS	
10.4.1	General rules for the process picture design.	
10.4.2	General rules for the representation of the plant	
10.4.3	Definition of colors, blinking, font size and shape	

11	**Creating a PCS 7 MultiProject according to the Plant Configuration**	
11.1	Creating a MultiProject with partial projects	
11.2	Creating an ES	
11.3	Creating a Project Library	
11.4	Definition of Software Types	
11.4.1	Creating Process Tag Types for the IEA	
11.5	Creating the Template Pictures for the automatic generation of block icon.	
11.6	Creating the AS Stations	
11.7	Creating the OS Stations - (Server, Clients, CAS)	

12	**Project backup**	
12.1	Define backup medium	
12.2	Define backup directory	
12.3	Name of the backup file(s)	
12.3	Define Backup cycles	

Installation of a PCS 7 Project

Content

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\03_PCS7_Project_009.doc

General

This chapter describes the installation of a PCS 7 project on the Engineering Station (ES) and covers all the required settings for CEMAT based on PCS 7 V8.2.

For general questions regarding PCS 7 Engineering we refer to the Engineering Manual for PCS 7 V8.2. These items are not included in the Engineering Manual for CEMAT.

As the project structure will be different from Project to Project (depending on project size, number of objects, number of AS, Servers, OS Clients etc) this description can only be an example for one of many possible solutions.

The description was made based on an Example-Project with the following configuration:

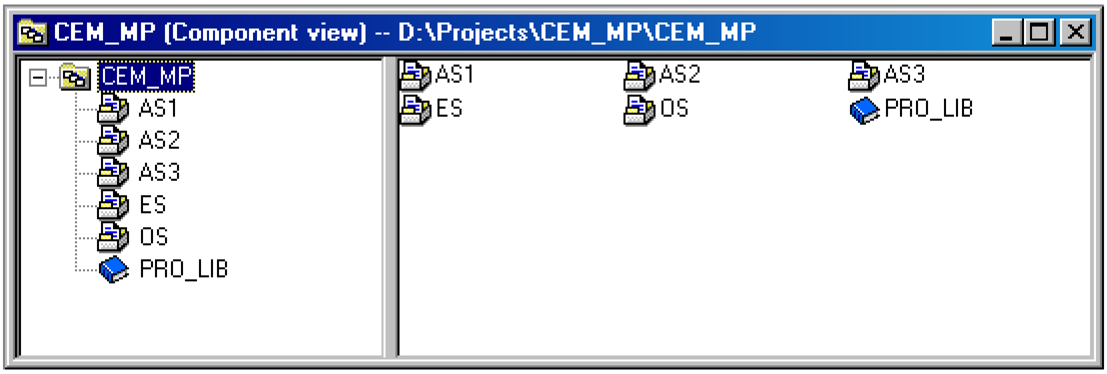
3 AS
1 Engineering Station
1 redundant Server-Pair
2 OS Clients.

The following example describes the engineering of a Multiproject **CEM_MP** with 6 Projects and a Library:

To enable independent Engineering for the different AS, individual PCS 7 -Projects have been created for each AS.

AS1 for first AS
AS2 for second AS
AS3 for third AS
ES for the Engineering Station
OS for the OS Servers and OS Clients
PRO_LIB for the Project Library

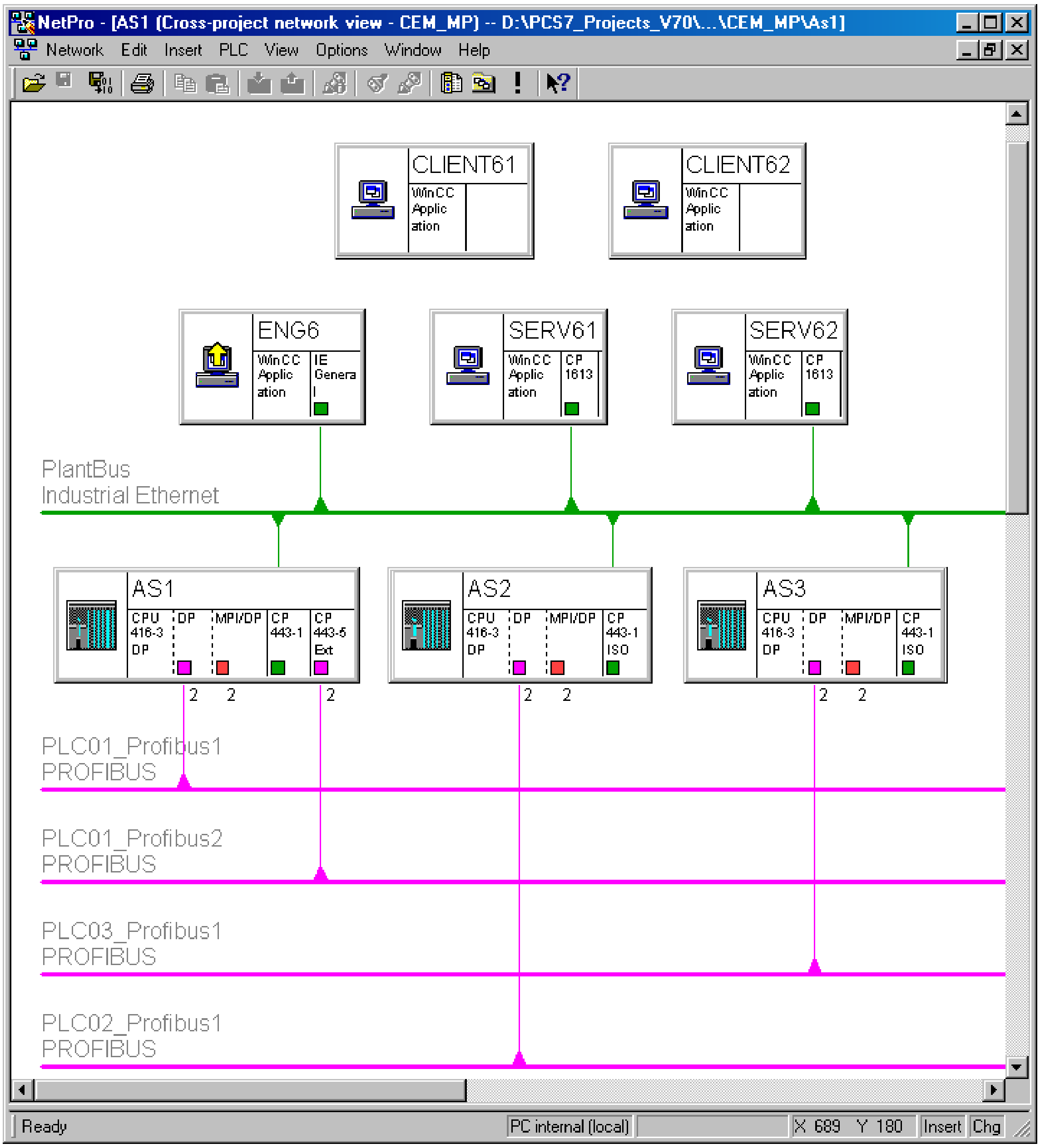
Caution: Avoid Underscore in the in the Name of the partial project for OS Server and Clients and also not in the name of the OS-Project for the Server because the package name is built of those two parts and the underscore is used as a separator in the package name, e. g. OS_OSSRV.
If you want to use the Cemat V6.0 Message Selection Function in the Alarm line, the underscore is absolutely forbidden.



The following screen shots show the Network Configuration of the Example Project.

Caution: The Terminal bus and the corresponding CPs cannot be displayed in the Network Configuration. Otherwise there will be errors during download to the Configuration Editor.

Network Configuration for the complete Multiproject (Cross-Project Network View)



Note: In our Example the Engineering Station has two Network Cards (1 internal CP and one 3COM) but no CP1613. In the Hardware Configuration of the PC only one Network Card for the connection to the Plant Bus can be configured!

General engineering rules

Before you start creating your own PCS 7 Project please get familiar with the most important engineering rules in order to avoid double work:

- The PC Names, OS Project Names and the Addresses (MAC Addresses, TCP/IP Addresses, PROFIBUS Addresses) used in the description have to be replaced by your PC Names etc.
- All variable names and function name must be used exactly as it is mentioned in the description (including Capital/small letters).
- In PCS 7 V8.2 the complete Engineering has to be done on the Engineering Station before the OS-Project is downloaded to the PCs. Don’t carry out any engineering step on the Server or OS Client itself.
- Never activate the OS-Project for Server on the Engineering Station if the Redundancy is already activated. On the Engineering Station the OS-Project can be activated in Simulation Mode with “Start OS Simulation”. In this case the OS-Project is duplicated and the Copy is activated.
- Don’t forget to take a backup of the complete Project (or MultiProject) once in a while, especially before you carry out extensive “Cut” and “Paste” functions and also if more than one person is working on the project.
- Don’t forget to save the modifications you made in the CEMAT Standard part. (Function blocks, Config Files, Standard Pictures). In case of a CEMAT Update these data will be overwritten.
- Create a common Library for the project (based on the CEMAT Library) and make sure that everybody working for this project is using the same blocks.
- The time synchronization is essential for the Alarm logging and Tag logging system. Make sure that there is only one (redundant) time Server in the system and that all the Stations (OS and AS) get the correct time. In PCS 7 V8.2 the AS has always GMT!!

The following engineering manual explains all steps for the installation and configuration of the PCS 7 Project, including AS, Engineering Station, Single Station, OS Server and OS Client:

For the Installation of PCS 7 V8.2
→ see "Installation of PCS 7 " and the Engineering Manual of PCS 7 V8.2

For the Installation of CEMAT
→ see "Installation of the CEMAT Software"

For the location of the CEMAT Software on your PC
→ see "File Structure"

For the additional settings on your PC which are required after the Installation of CEMAT
→ see "Adaptations in PCS 7 for CEMAT"

For the creation of a PCS 7 Project with AS and OS Stations (including Hardware Configuration and Network Configuration)
→ see "Create a new PCS 7 Project"

For the definition of the plant structure
→ see" Plant structure definition (Plant view and Process Object view)"

For all the required settings in a CEMAT AS
→ see" AS Configuration"

For the Configuration of a Single Station
→ see "OS Configuration (Single Station)"

For the Configuration of an OS Server
→ see "OS Configuration (Server)"

For the Configuration of a Standby-Server
→ see "OS Configuration (Standby-Server)"

For the Configuration of an OS Client
→ see "OS Configuration (OS Client)"

For the creation of the Server Data
→ see "Generating and loading of the Server Data"

For the Project Download to the OS Stations
→ see "Project Download"

Installation of PCS 7

For each PC Station (ES, Server, Standby-Server and OS Client) the appropriate startup procedure must be carried out. Please follow the instructions in PCS 7 V8.2 description and the readme file which is delivered with the PCS 7 CDs.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2

Make sure that the PCS 7 updates are installed before using CEMT V8.2!

Only a few additional remarks for the installation:

Language of the Operation System

Make sure that all PC Stations are installed with the same operation system language (normally English or German).

Regional Settings

Make sure that the regional settings of your PC are correct for the language you want to work with.

Computer Name

For the computer name follow the restrictions of PCS 7 V8.2!

In the PCS 7 Manual "PCS 7 PC – Configuration and Authorization you find the following note: **Use only letters and numbers for the computer names.**

Additional notes you find in the Online help of WinCC. Special characters are not allowed in the PC Name.

The following characters <u>are not allowed</u> in PCS 7 and Cemat:

. , ; : ! ? " ' ^ ´ ` ~ - + = / \ ¦ @ * # $ % & § ° ( ) [ ] { } < > _
blanks
only uppercase relevant

The first character must be a letter!

Screen Resolution

The recommended screen resolution for the CEMAT Faceplates and Pictures is 1920x1080.
Four more screen resolutions are possible: 1280x1024, 1600x1200, 1680x1050 and 1920x1200. In this case a few standard pictures have to be exchanged.
For Holcim Standard (004) and Lafarge Standard (027) the screen resolution 1600x1200 is not released.
Less than 1280x1024 is not possible; otherwise the Cemat Faceplate cannot be displayed.

Power Options

Switch off the Power saving system (for all Devices like Hard Disk, Monitor etc.) in BIOS and in the Control Panel

Domain controller

The installation of a Domain controller is **not** released for Cemat!

Installation of the software for PCS 7

Please, always use the actual installation instructions, which are delivered with the PCS 7 installation CD. As installation language please select the language in which the system is finally delivered. (Some system messages are only displayed in the installation language!) The installation is described in the document „PCS 7 PC Configuration and Authorization“.

Hardware definition with Commissioning Wizard

When you start your PC the first time after installation of PCS 7 the Commissioning Wizard will be started automatically in order to ask you for the Settings of the PC Station. Enter these settings according to the Hardware of your PC.

The configuration can be checked under:

Start → Simatic → SIMAITC NET → Settings → Configuration Console.

For detail information refer to the PCS 7 Manual “PC Configuration and Authorization”

Configuration Editor

In the Configuration Editor add the WinCC Application according to the Station Type and give the correct Station Name.

User rights for Cemat

The installation of Cemat requires Administrator rights.

During the runtime Cemat requires the rights for 'Siemens TIA Engineer', 'SIMATIC HMI' and 'SIMATIC NET'. The user must have write permissions for folder CEMAT_CS and all subfolders.

Installation of the CEMAT Software

The following procedure has to be carried out on all PC Stations (ES, Single Stations, Server and OS Client):

Start the delivered Setup Programs from the CD and follow the instructions on the screen.

- Select the installation language (English and German is possible).
- Accept the license agreement
- The Destination Folder for CEMAT is D:\CEMAT_CS.
  Don't modify the destination directory from the default. To ensure that all CEMAT functions working correctly CEMAT must be installed under D:\CEMAT_CS!
- Select the correct Setup-Type:

ENGINEERINGSTATION	for an Engineering Station
OS SERVER /CLIENT	for a Single Station, Server, Standby-Server or OS Client

- The list of Installation keys for the different project standards appears (the correct key must be entered later):

  000 = CEMAT Standard
  001 = CEMAT Minerals
  004 = Holcim
  006 = Dyckerhoff
  007 = Heidelberg Cement
  023 = Vigier
  024 = Bushehr
  025 = Caima
  026 = Alsen
  027 = Lafarge
  028 = Rossi

- Now enter the project key and start the installation.

File Structure

After running the Setup the CEMAT specific files are located in drive D, directory CEMAT_CS.

D:\CEMAT_CS

```
├──── \AddOn                      AddOns for Special Functions (Optional)
├──── \BIN                        CEMAT IndustrialX Con
├──── \BITMAPS                    CEMAT Bitmaps (zip-file)
├──── \CONFIG                     Configuration Files
├──── \DOCU                       Documentation in PDF format
├──── \HARDCOPY                   For Print Screen functions (print faceplates)
├──── \IMPORT                     Import files for WinCC Configuration Studio
├──── \MULTIMEDIA                 Documents for Info-Dialog:
│       ├──── \AS                        AS Cabinet drawings
│       ├──── \ELECTRICAL_DIAGRAM        Electrical schematic drawings
│       ├──── \IO                        I/O cabinet drawings
│       ├──── \LOOP_DIAGRAM              Software Interlocking diagrams
│       ├──── \MANUAL                    General function descriptions
│       ├──── \MCC                       MCC cabinet drawings
│       ├──── \OPERATING_PROCEDURES      Operating descriptions
│       ├──── \PERIPHERY                 I/O information from HW Config
│       ├──── \SOUND                     Sound replay
│       └──── \VIDEO                     Video description for trouble-shooting
├──── \SIMOCODE                   SIMOCODE files for 3UF70
├──── \SOUNDS                     Sound files for alarms
├──── \TOOLS                      Software Tools for CEMAT
└──── \WinCC                      WinCC files:
        ├──── \GraCS                     CEMAT Standard pictures and bitmaps
        ├──── \Library                   WinCC Project Functions for CEMAT (C-Scripts)
        ├──── \PRT                       WinCC Report layout
        ├──── \ScriptLib                 WinCC Visual Basic Scripts for CEMAT
        └──── \WScripts                  WinCC Property configuration for CEMAT
```

The CEMAT libraries ILS_CEM and PRO_CEM are installed in the PCS 7 system directory and will be found after the installation under

C:\Program Files (x86)\SIEMENS\STEP7\S7LIBS (Windows 7)

Adaptation in PCS 7 for CEMAT

1. In the configuration files under D:\CEMAT_CS\Config the system can be adapted to the project requirements.

   The main configuration file **C_CONFIG.CFG** contains the settings for the Multimedia Server and some general Cemat Settings which may need adaptation. Make sure that the file C_CONFIG.CFC has identical settings at each OS Station.

   Multimedia Server:
   If you don't want to save the Multimedia files on every Client, you have to enter name of the PC where the Multimedia files are located under "MServer=……..".

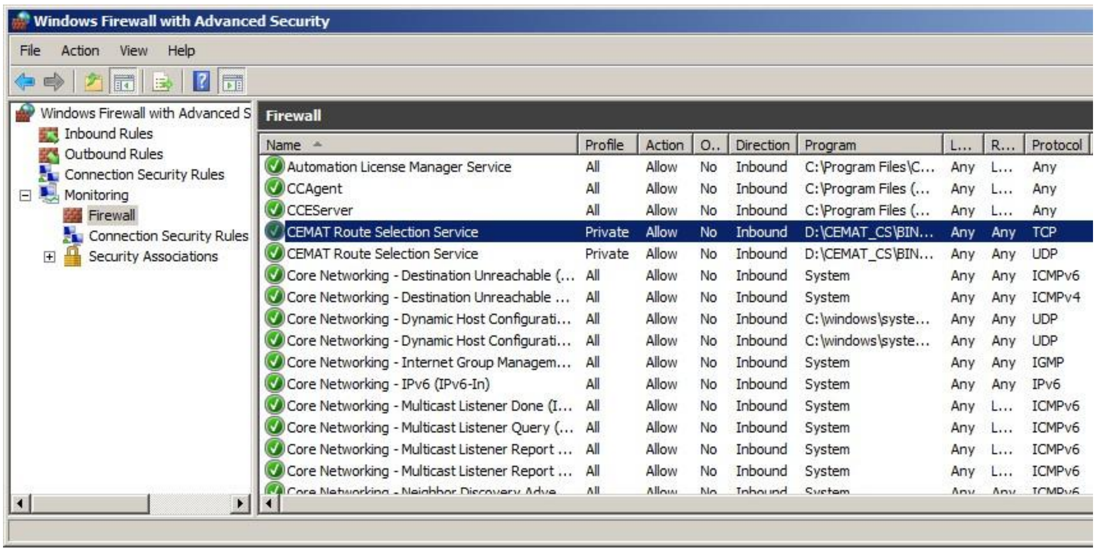
```
[MultimediaServer]
;for Example
;MServer=CLIENT62  -->  The name of the CEMAT Multimedia Server is
CLIENT62
;MServer=@RM_SERVER_NAME  -->  The actual server for this client
with the redundancy functionality
;List here the name of the multimedia server or the redundancy
functionality
MServer=
```

   General Cemat Settings:
   The rest of the settings are used to evaluate the Process Value archive, to select the printer option (if available), to choose the method how to open the object list, and to configure the Path for the Adobe Reader function (which is needed for opening the help files). For more information see 14_Tips_Tricks.pdf.

```
[CematSettings]
;Individual Settings for Cemat Objects (0=No, 1=Yes)
;
;C_MEASUR - see "Process Value Archive" in 14_Tips_Tricks.pdf
SkipTagArchiveName=0
;
;No Hard copy of the actual faceplate to the default printer - see
"Hardcopy to default printer" in 14_Tips_Tricks.pdf
HardCopyToDefaultPrinter=0
;
;Object List: Subobjects not expanded - see "Display mode of the
ObjectList" in 14_Tips_Tricks.pdf
ObjListExpanded=0
;
;USE_PDF_READER=1 ==> help files can read/show with an installed
pdf-reader,
;find class name with "CEMAT Window Class Finder" - see "Define
Window Class Name" in 08_OS_Engineering_009.pdf
USE_PDF_READER=1
READER_PATH=C:\Program Files (x86)\Adobe\Reader
10.0\Reader\AcroRd32.exe
READER_CLASS_NAME=AcrobatSDIWindow
```

2. If the Windows Firewall is activated, the following Settings have to be carried out on each Server Station under *Start → Control Panel → Windows Firewall*:

   Select "Allow a program or feature…" and chose "CEMAT Route Selection".
   After that you should see the file in window "Advanced Settings".



Create a new PCS 7 Project

Define a directory where all the PCS 7 projects will be stored in the future. We don't recommend using the default directories under D:\SIEMENS\STEP7\S7Proj. We recommend creating a separate directory on D drive.

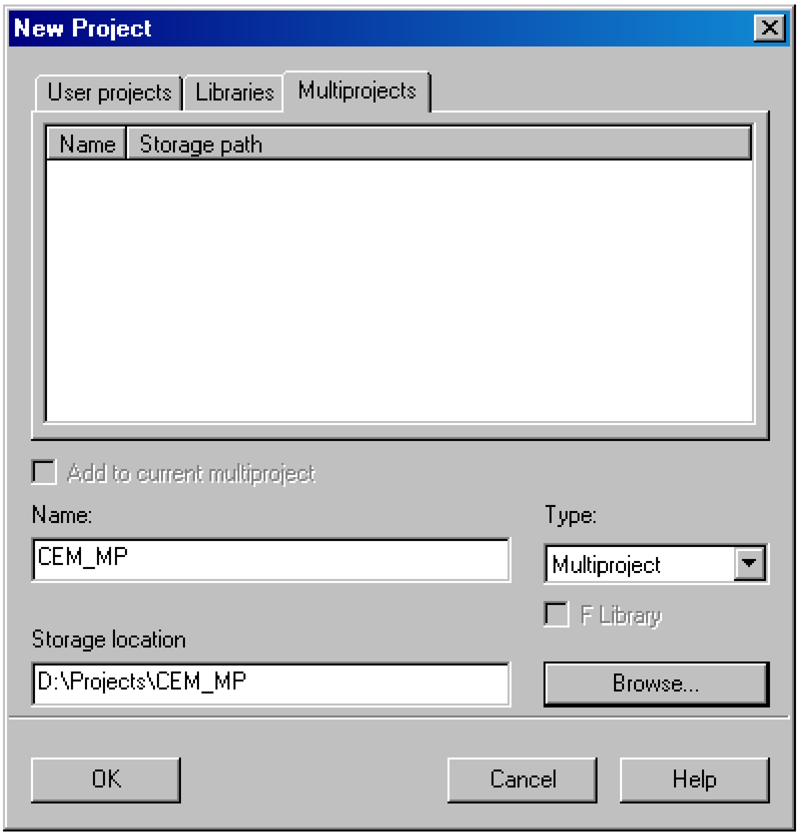
Use *Options -> Customize* to select the new directory under 'Storage location for projects'. Use directory D:\PROJECTS to save your PCS 7 projects. If you use Multiprojects, create a Subfolder as Storage location for the Multiproject, e. g. CEM_MP

You can use the Wizard to create a new Project or Multiproject or you can create the Project or Multiproject manually. The wizard will create a Project Structure with Plant Hierarchy which you can modify or delete.

In the following description the project is created manually:

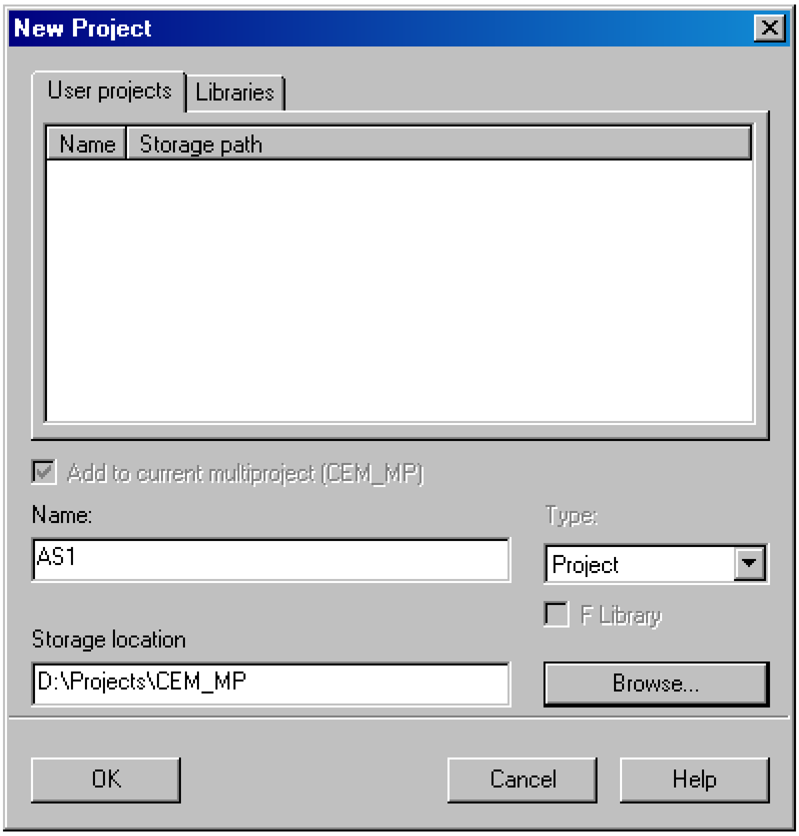
In the SIMATIC Manager create a new project using *File -> New*.

Select Type *Multiproject* and define the Project *name*. Check the *Storage location* (it should be the Multiproject directory).



Confirm with OK

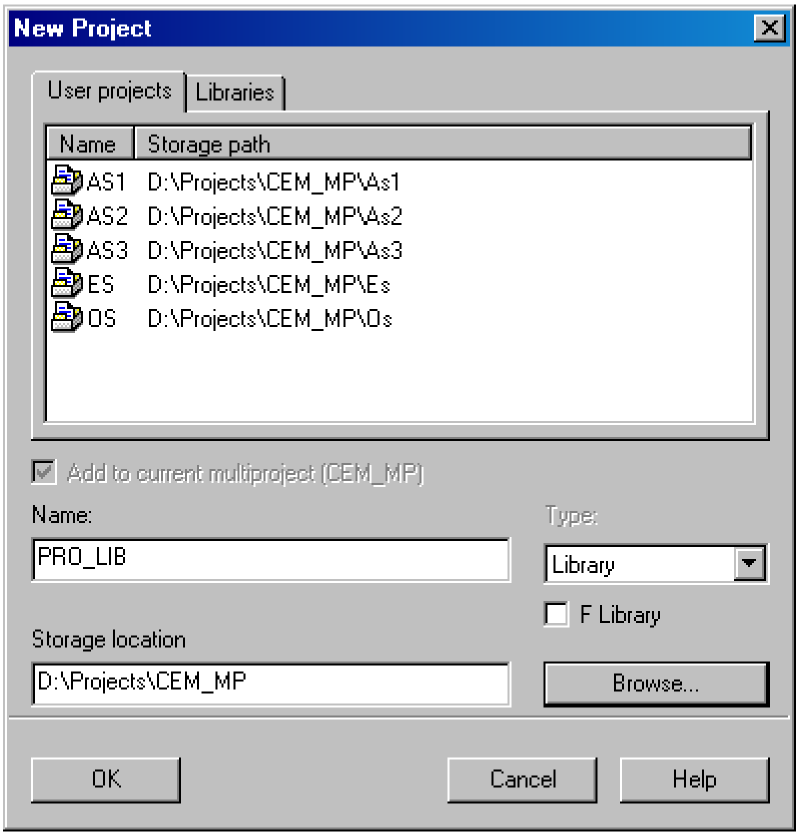
Select the Multiproject in the SIMATIC Manager and use *Multiproject -> Create in Multiproject* to create the first Project within the Multiproject.



Confirm with OK.

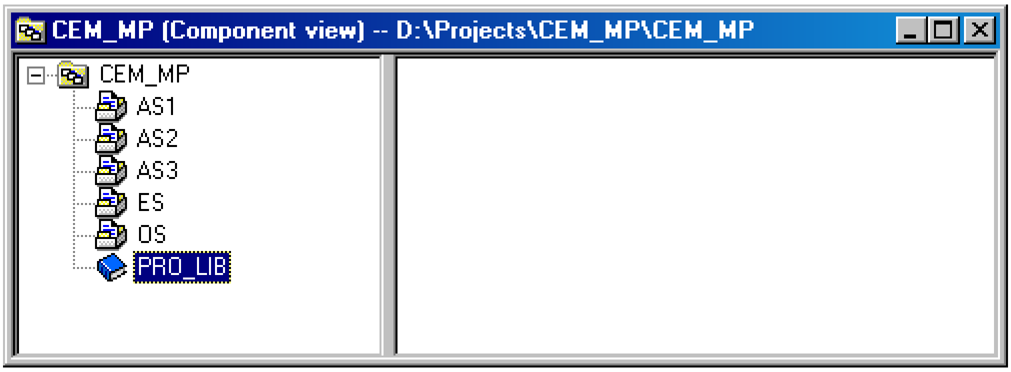
Repeat this step for all your Projects.

Create the Project Library using the same method. For the storage location the Folder of the Multiproject Folder is used.



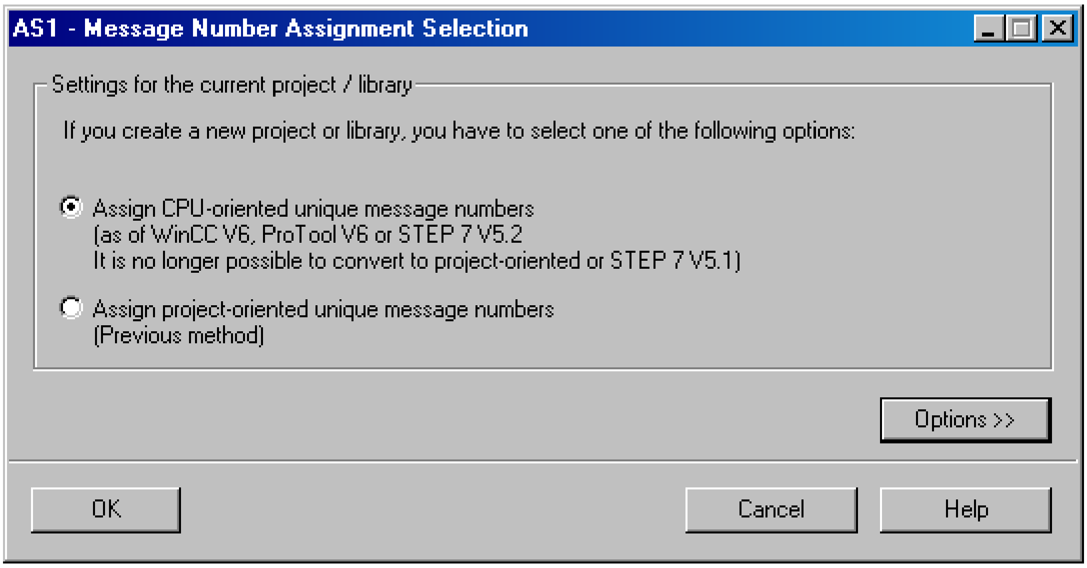
This library will be the Master Data Library for the Project. It should contain all Blocks and Typicals used in the Project (in the CFCs of each AS-Program).

To select the library as “Master” use *Multiproject -> Define as Master Data Library*.



For each partial Project select the display language. Select *Options → Language for Display Devices.*

The first time a window will pop up for Message Number Assignment Selection.



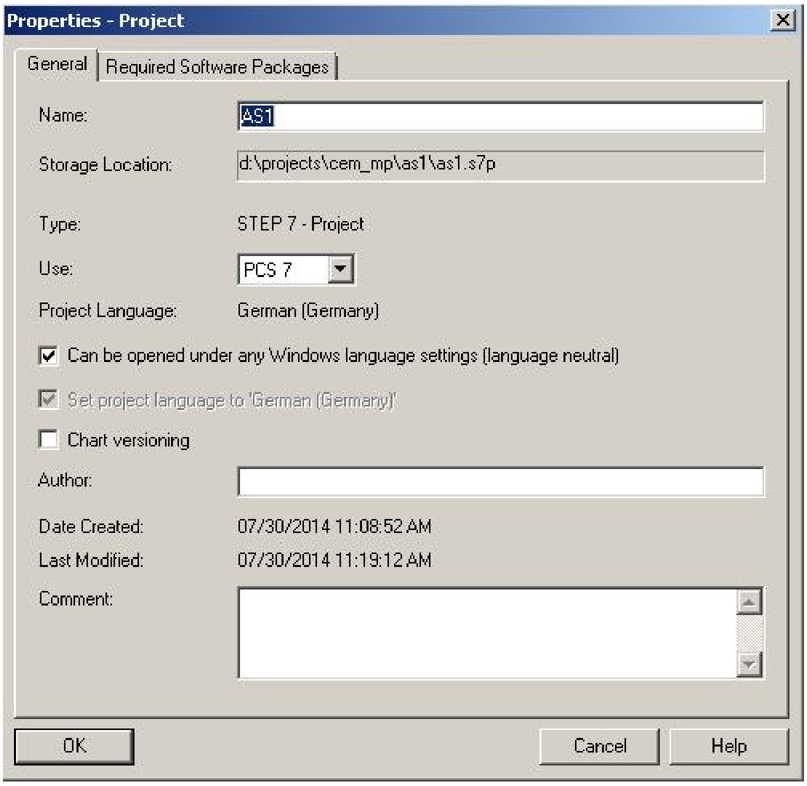
Assign CPU-oriented unique message numbers.

Choose the installed languages in the project and select the default language.

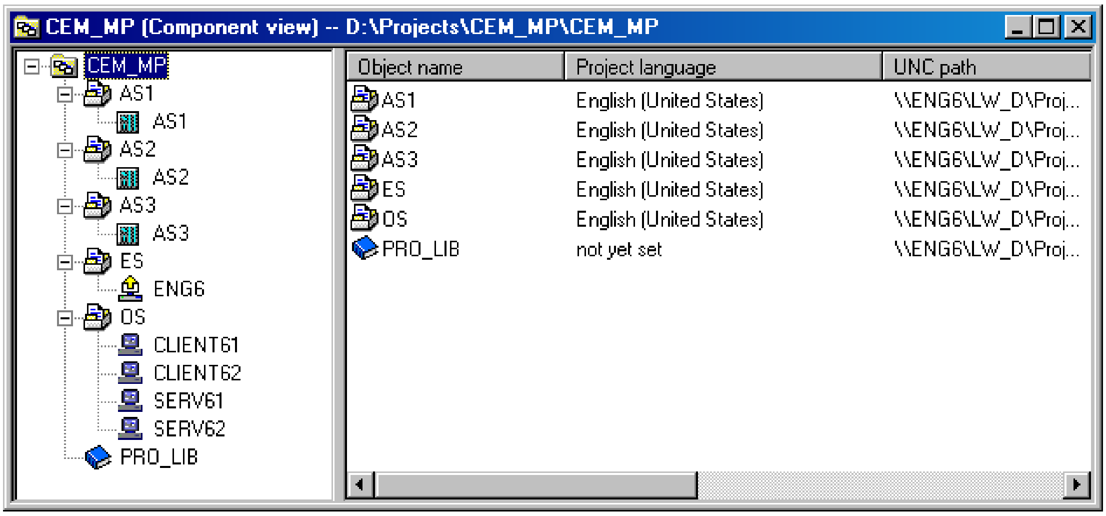
The display language must be similar in all partial projects.

⚠ **Caution:** For English and Spanish you have to select **English (United States)** and **Spanish (international Sort)** as these languages are used in the Cemat Standard blocks and Faceplates.

Especially if you work in a Project Language different than English or German, make sure that the project can be opened under any Windows language settings (language neutral).



Once the Projects are created and the display language is selected, you have to create the AS and the OS Stations according to the Hardware Configuration of your plant. New AS/OS can be created either in the Component View of the SIMATIC Manager using right mouse button to *Insert New Object* or in the Network Configuration.



Change name of the OS Stations to the real Station name (e.g. CLIENT62, SERV61). For the AS give a meaningful and unique name (e. g. AS1 or KILN1).

Hardware Configuration for a CEMAT AS

The following pages show the settings in the Hardware Configuration for a CEMAT AS.

Of course it is not possible to describe the wide range of possible PLC Hardware components in this document. In the following description we want to highlight the crucial settings in a Cemat Project.

The Screenshots below shows the relevant settings based on two examples:

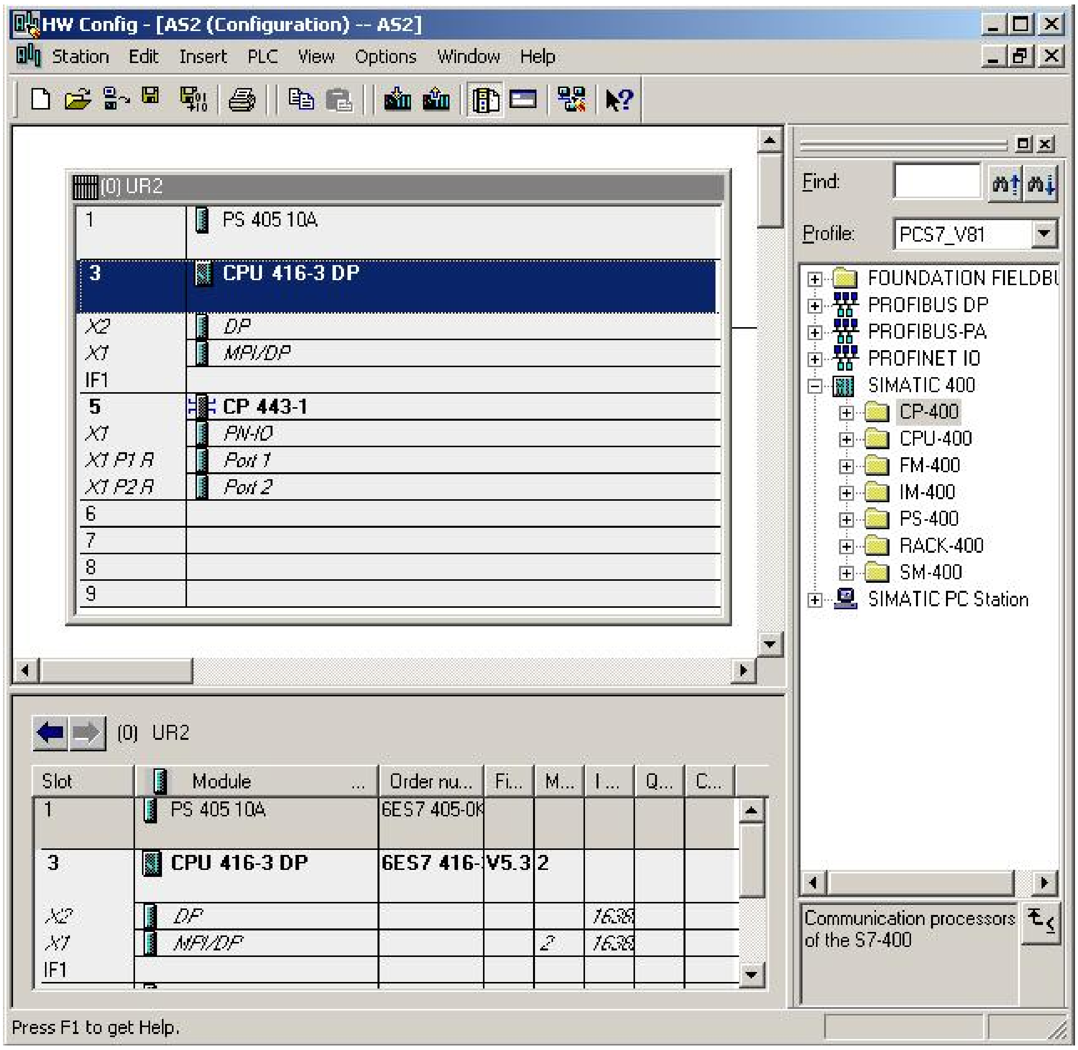
- CPU416-3 DP plus CP443-1
- CPU410-5H.

Other CPU and CP Types are possible as well, but in this case the configuration options and the property windows may differ. For details please refer to the PCS 7 Manuals and FAQs.

Define AS Hardware

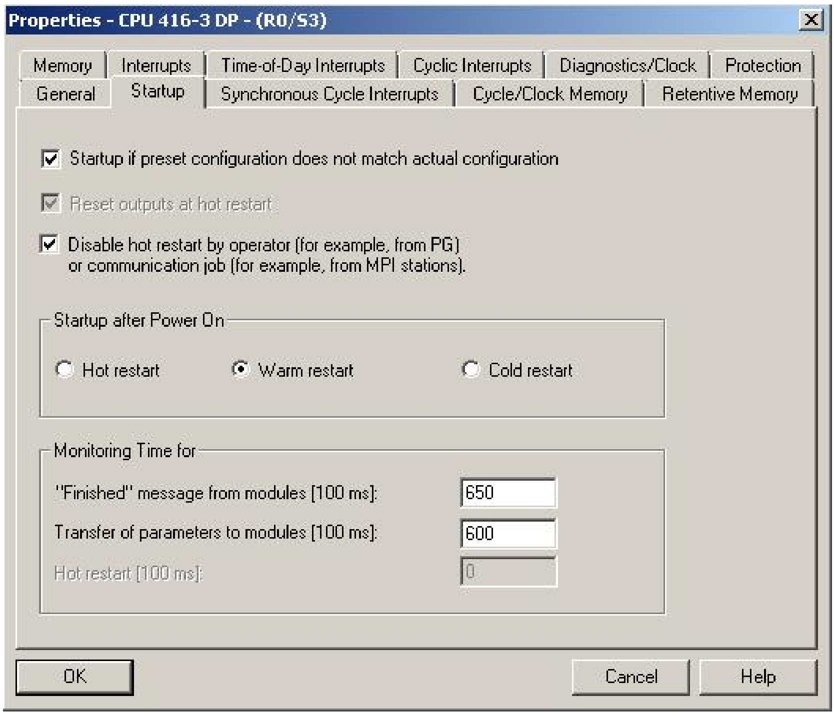
Select from *Hardware Catalog -> SIMATIC 400 -> RACK-400* the Rack you want to use for your AS and drop it to the sheet. Select a power supply, CPU and a CP according to your hardware configuration.

Example configuration for an AS with CP416-3 DP:



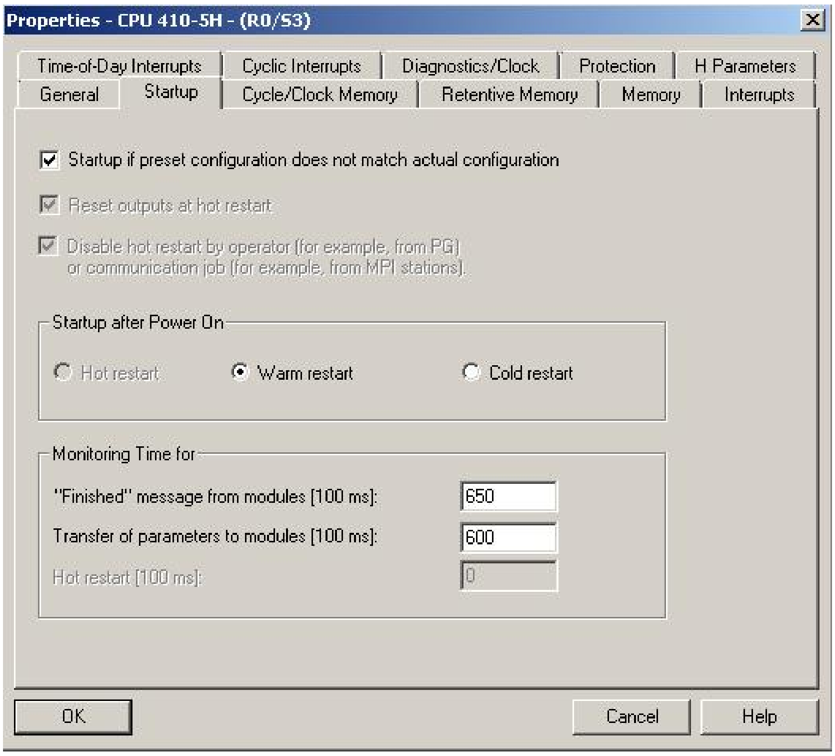
When dropping the CPU create a new PROFIBUS and connect it.

Settings in tab Start-up for CPU416-3 DP:



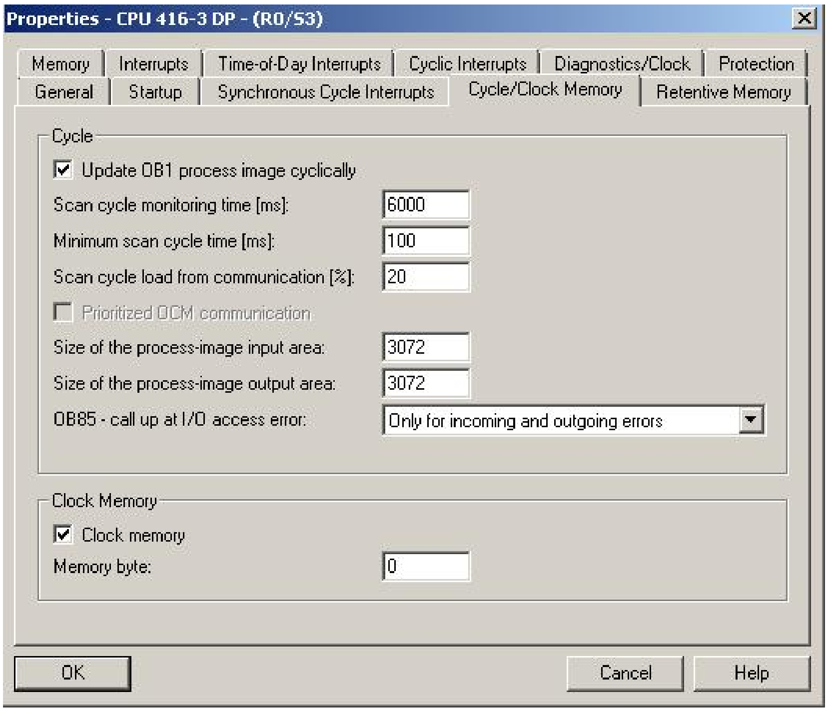
Under “Start-up after Power ON” **Warm Restart** must remain selected

Settings in tab Start-up for CPU410-5H:



Under “Start-up after Power ON” **Warm Restart** must remain selected

Settings in tab Cycle/Clock Memory for CPU416-3 DP:

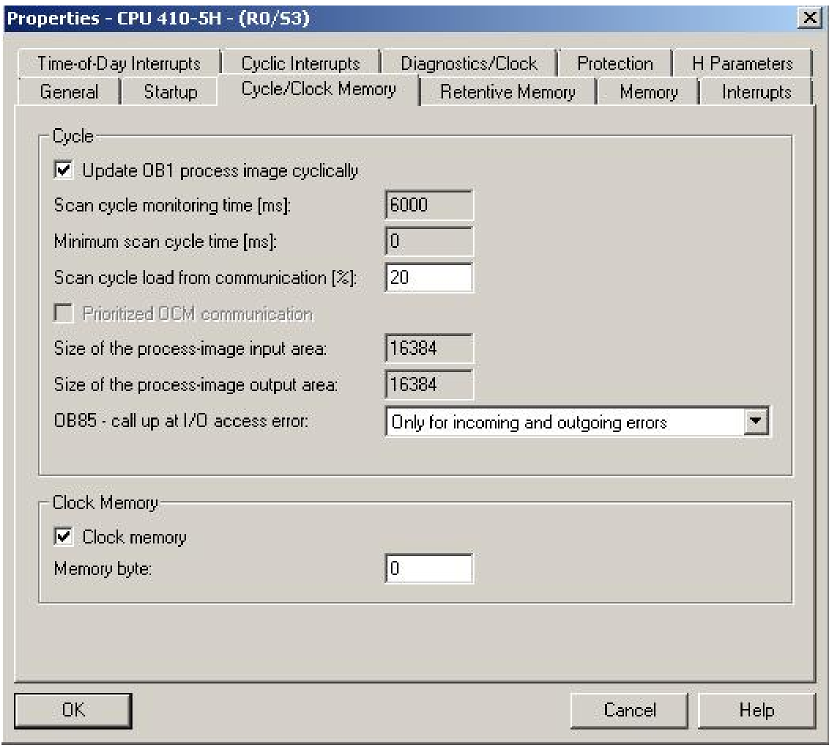


Enter the Scan Cycle Monitoring Time.

Enter the Minimum Scan Cycle Time of 100ms

Select Clock Memory. The memory Byte must be 0

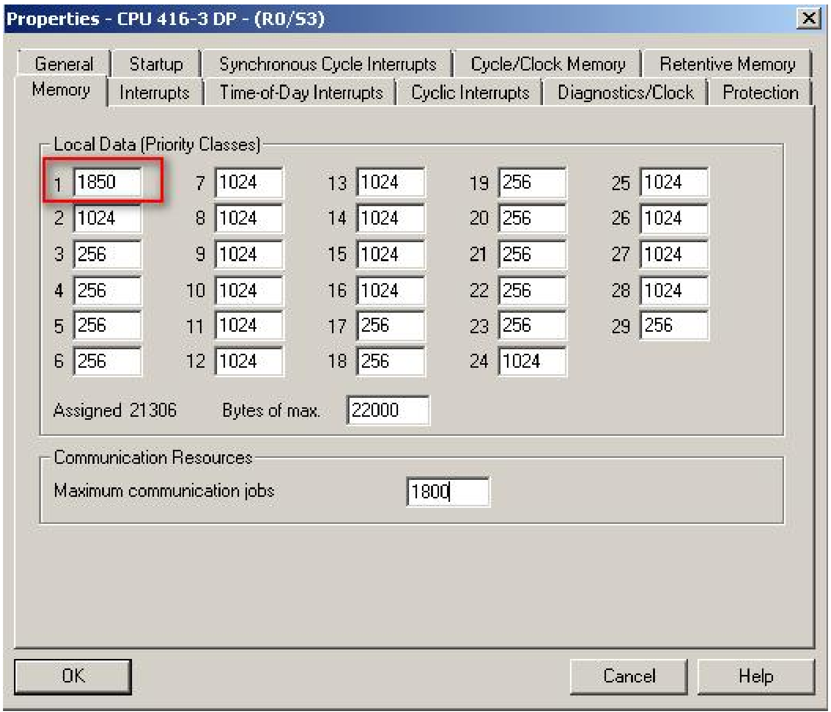
Settings in tab Cycle/Clock Memory for CPU410-5H:



Scan cycle monitoring time and Minimum scan cycle time cannot be configured.

Select Clock Memory. The memory Byte must be 0

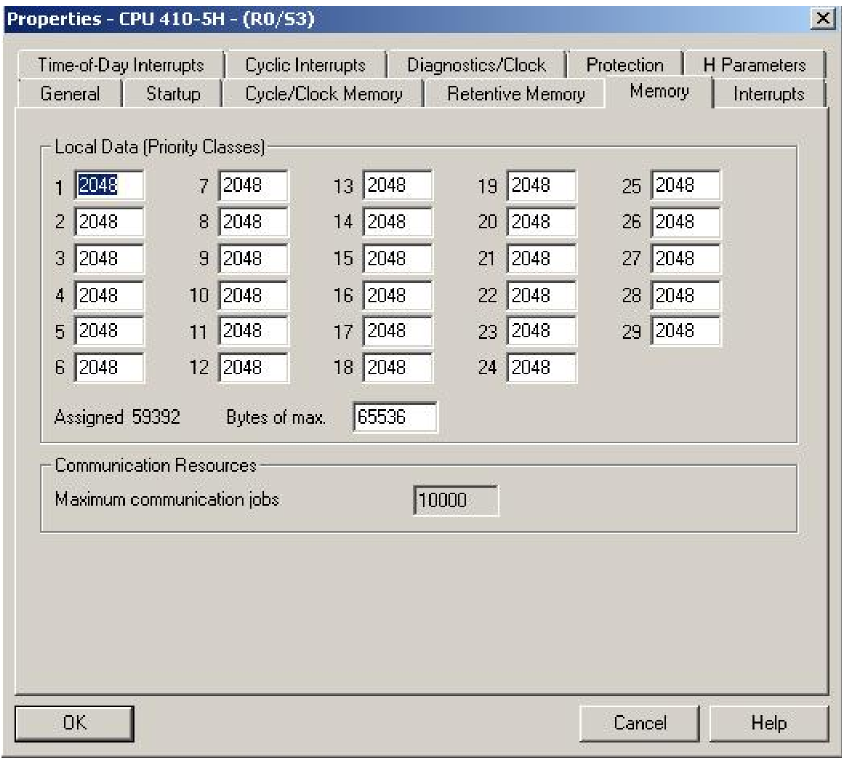
Settings in tab Memory for CPU416-3 DP:



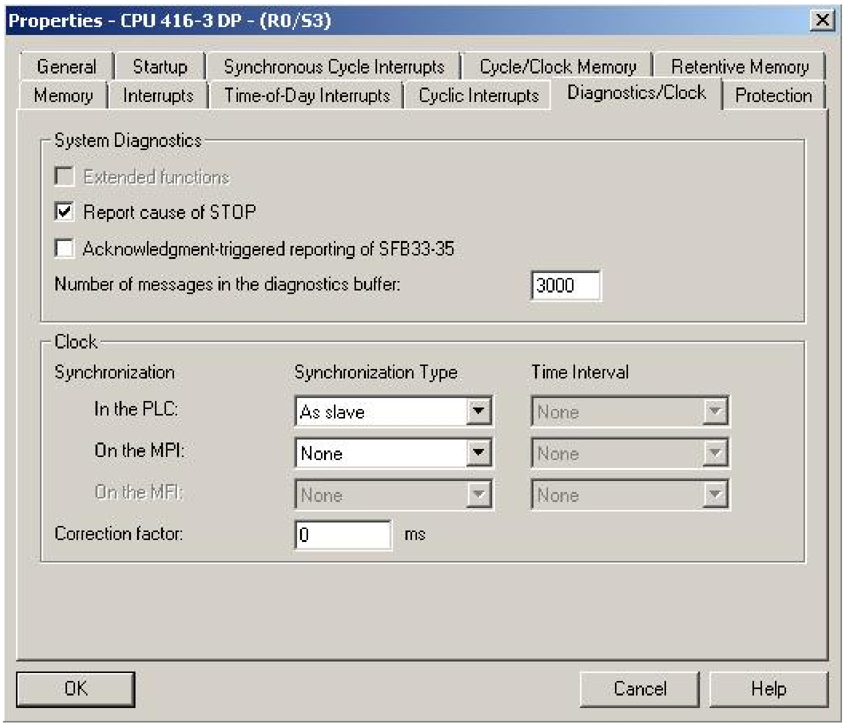
On this page the minimum values for Local Data per each priority class is configured. Bigger CPUs probably have more memory for local data.
In the screen shot the local data for Priority class 1 has already been increased (from 1024 to 1850). You have to check the required local data after completing the user program and modify the values if necessary.

The maximum number of Communication Jobs can be increased as well.

Settings in tab Memory for CPU410-5H:

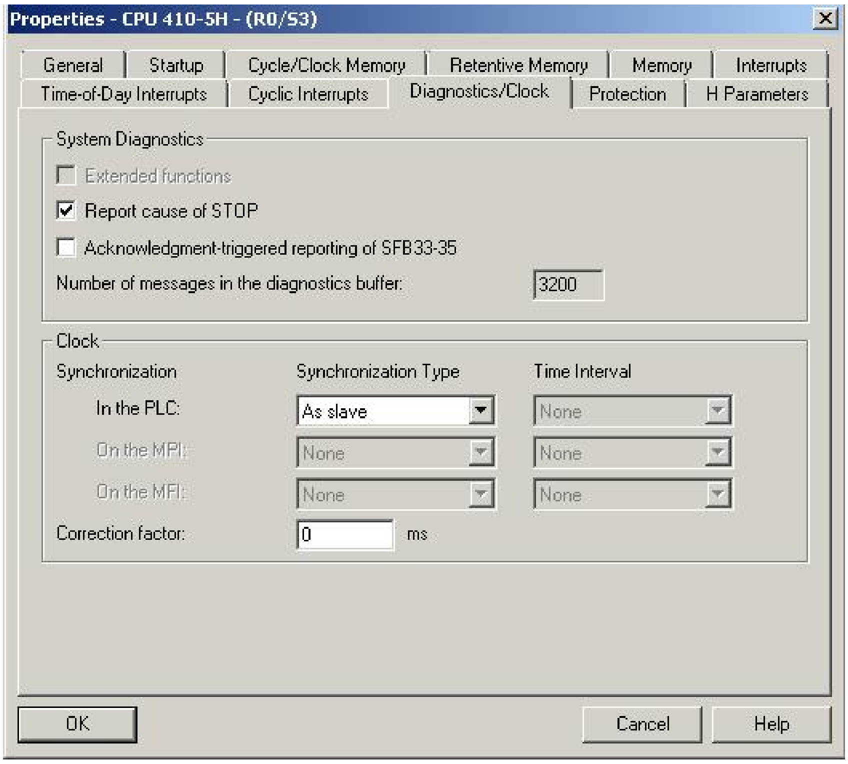


Settings in tab Diagnostics/Clock for CPU416-3 DP:



The AS synchronization type should be “As Slave”

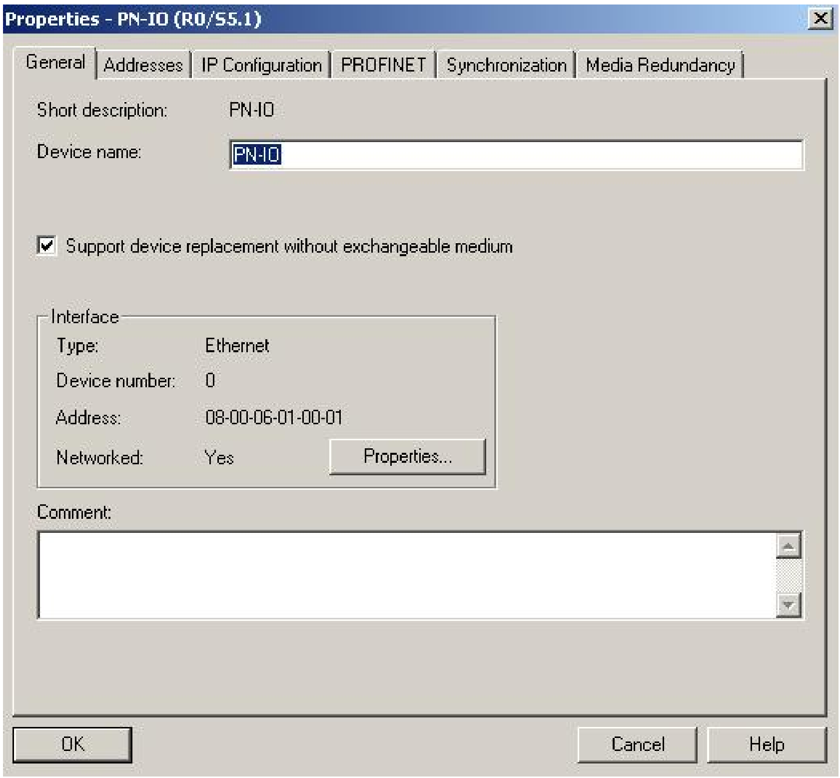
Settings in tab Diagnostics/Clock for CPU410-5H:



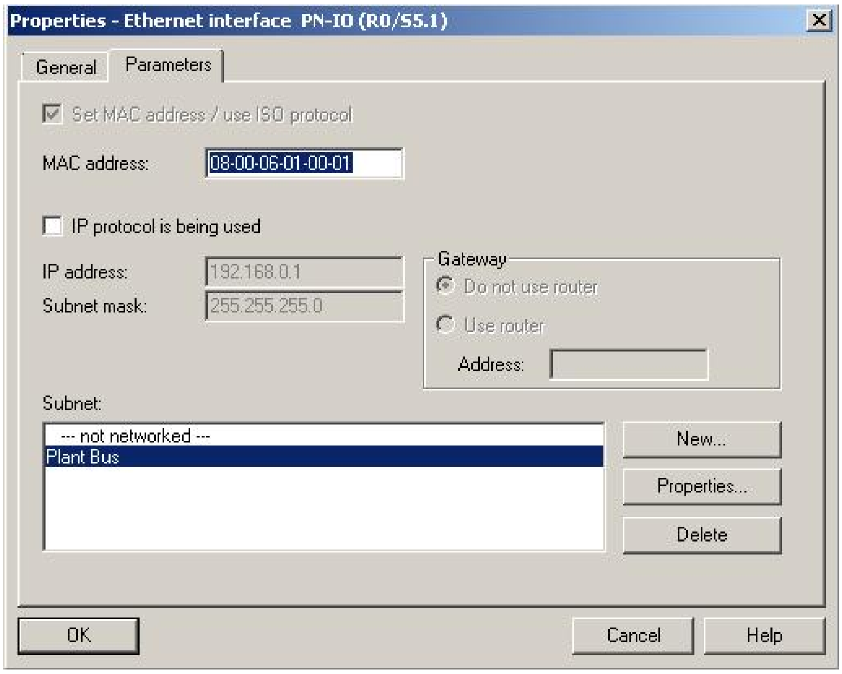
The AS synchronization type should be “As Slave”

Note: In PCS 7 V8.2 the AS always work with GMT.

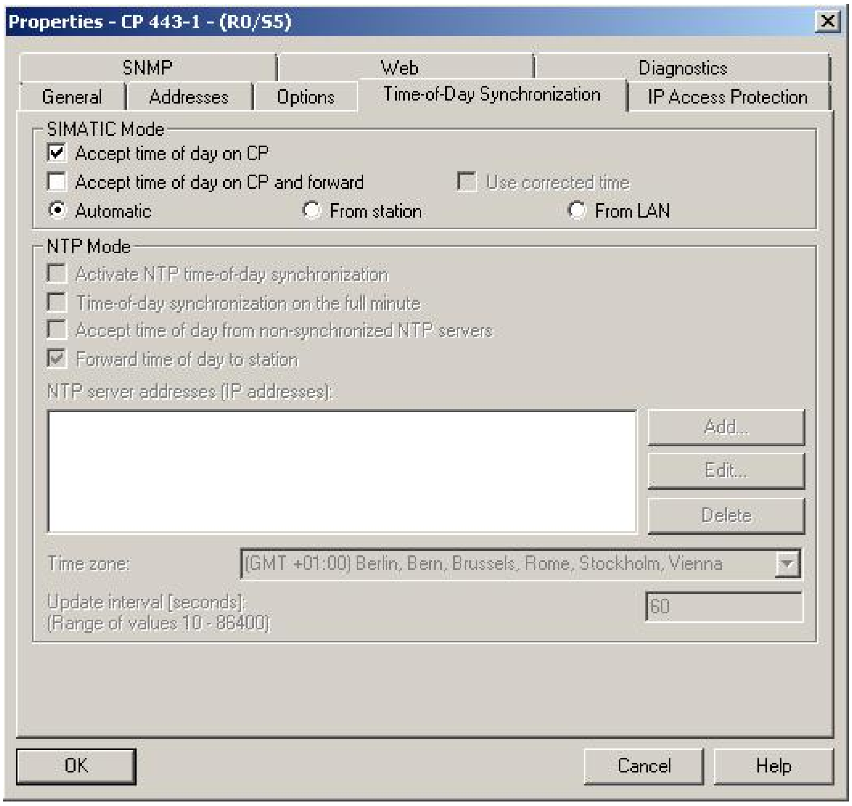
When using a CP443 connect the same to the Ethernet bus.



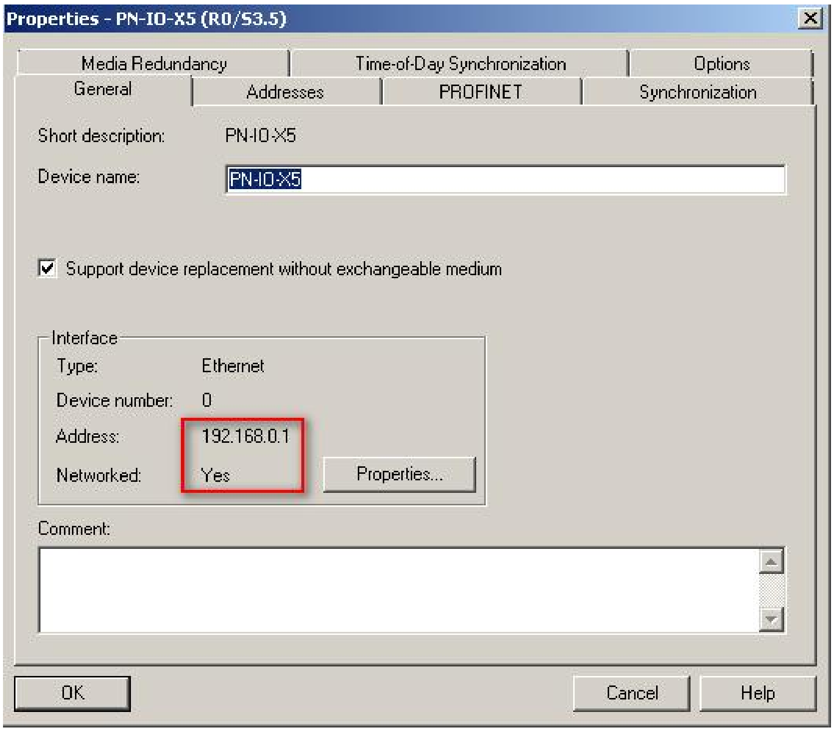
In the Properties dialog define the Address (MAC Address or IP address) and connect the CP with the Ethernet bus. If no Ethernet bus exists yet, you can create it here.



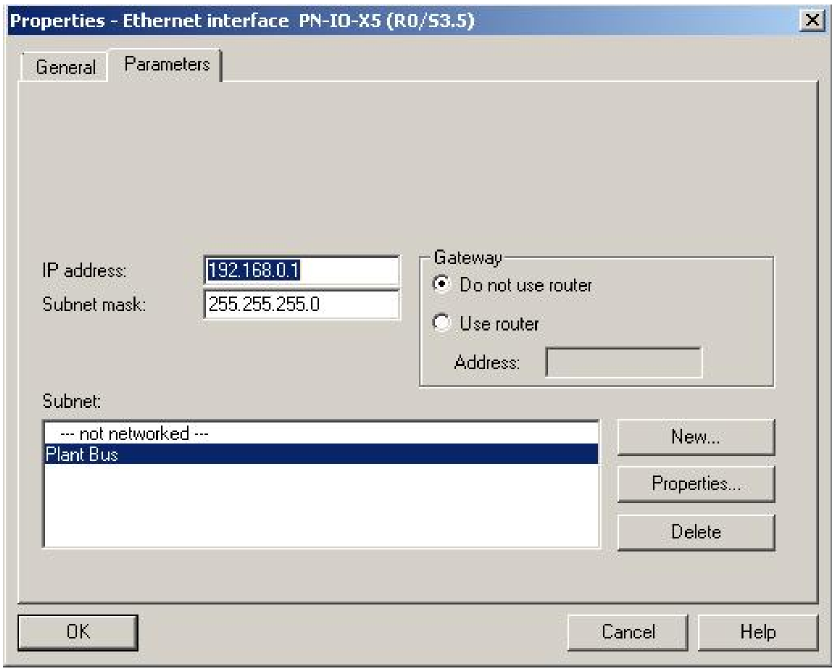
Under Time-of-Day Synchronization, SIMATIC Mode you have to activate the time-of-day synchronization for the CP:



When using a CPU-5H the PROFINET interface can be used for the connection to the Plant Bus and the CP443 may not be required.



In this case the connection is only possible via TCP/IP.



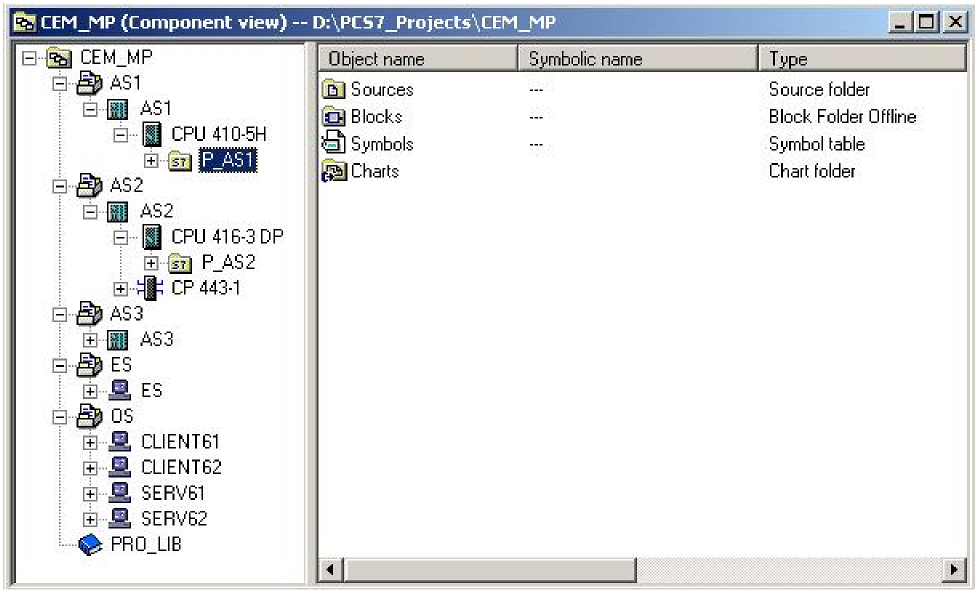
For the I/O Periphery again a huge variety exists and it is therefore not described in this document.

Please define the I/O Periphery according to your plant configuration. For details refer to the SIMATIC and PCS 7 Manuals.

After the Hardware configuration is completed, save and compile the hardware definitions and load it to the AS.

Rename S7-Program

Back in the SIMATIC Manager you see the elements of the Hardware Configuration (CPU, CP) under each AS Station. Below this you find the Program Container for the elements. For better identification we recommend to rename the S7 Program Container for the CPU (e. g. P_AS1).



Configuration of a SIMATIC PC Station

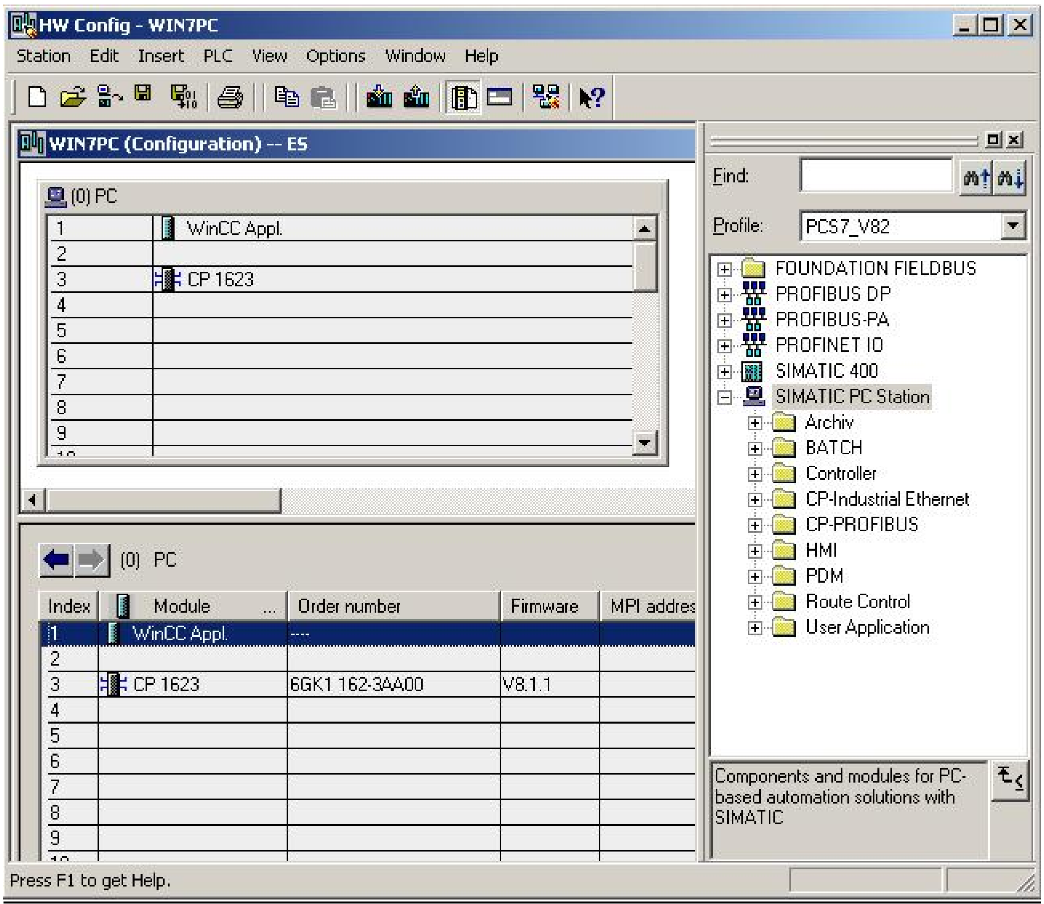
The SIMATIC PC Stations (Single Station, Server, Standby-Server and Clients) have to be configured according to the Hardware of the PC.

Again it is not possible to describe the wide range of possible Hardware components for the PC Station in this document. In the following description we want to show the general procedure for adding PC Stations in a Cemat Project.

Select the appropriate WinCC Application. Select the CP Industrial Ethernet for the Plant Bus communication according to the Hardware of your PC (Industrial CP or IE General) The Index must be the same as the Index in the Configuration Editor. In the Properties for each CP configure the right addresses (MAC Address; IP Address).

Note: It is important to use exactly the same Configuration as it is defined with the commissioning wizard.

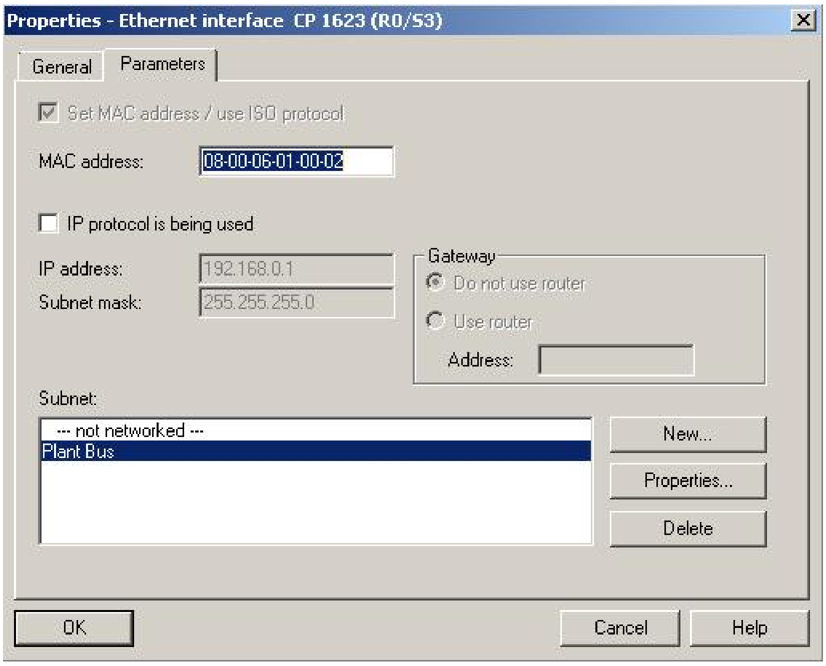
Example for the Configuration of the Engineering Station:



In some plants the Engineering Station has two CPs (one internal CP and one 3COM) but no Industrial CP (CP1613. CP1623 or CP1628). Instead of the Industrial CP the 3COM is used for the communication to the Plant Bus.

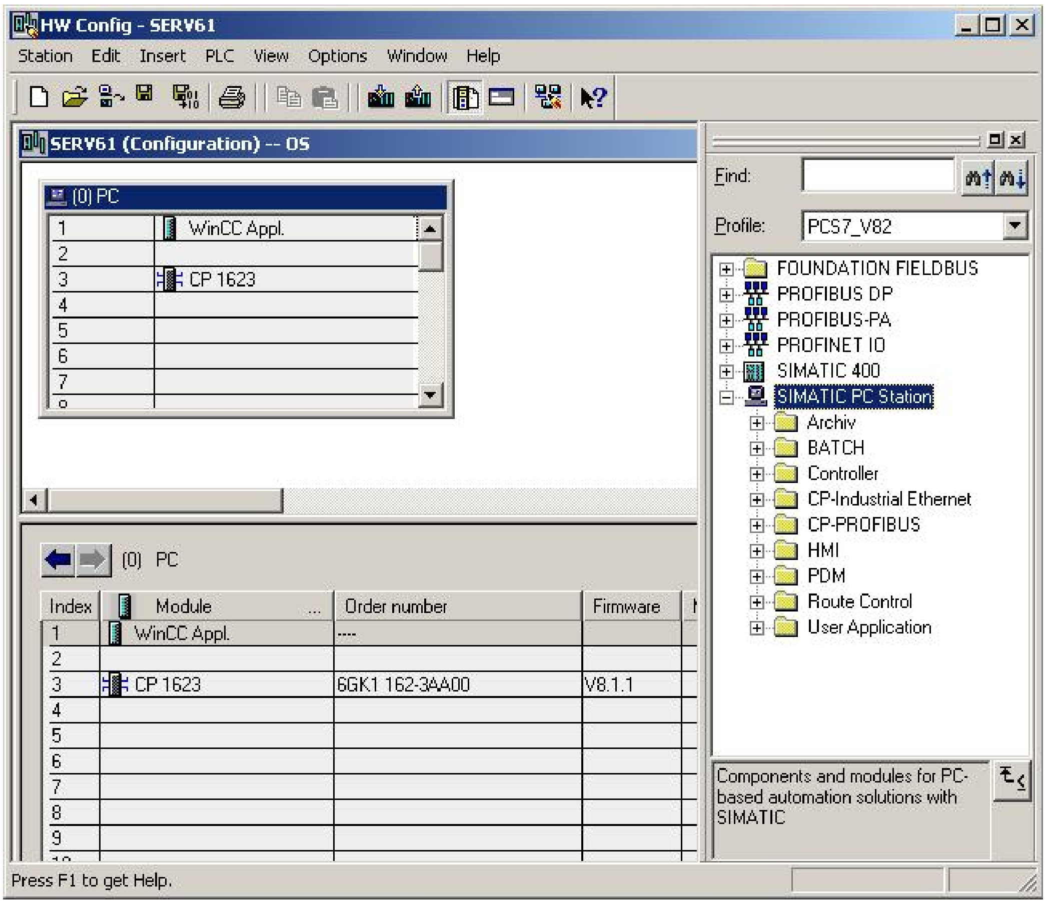
In this case insert a CP "IE General" instead of an Industrial CP for the communication to the Plant bus. The CP for the communication to the Terminal bus must not be inserted in the Hardware Configuration.

Select the *Object Properties* for the CP in order to enter the MAC address and connect it to the Plant Bus.



When saving the HW Configuration the OS-Project for the Engineering Station is created. Change the OS Project name (e.g. ES). Die OS Project name must be unique within the complete MultiProject.

Example for the Configuration of a Single-User Station or a Server:



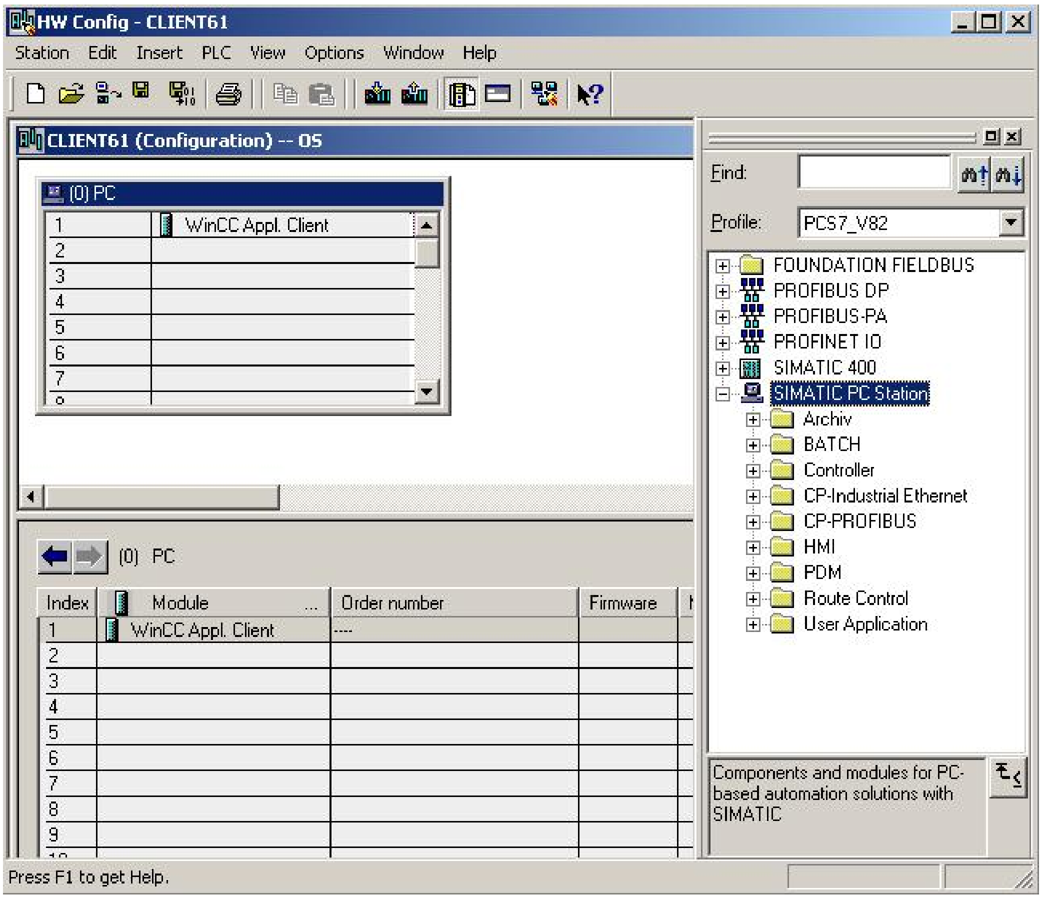
Select the *Object Properties* for the Industrial CP in order to enter the MAC address and to connect it to the Plant Bus.

When saving the HW Configuration the OS-Project for the station is created.

Change the OS Project name (e.g. OSSRV). Die OS Project name must be unique within the complete MultiProject.

For the Standby-Server use WinCC Application (stby) instead of WinCC Application. The rest is identical to the Server Configuration.

Example for the Configuration of an OS Client:



When saving the HW Configuration the OS-Project for the Client is created.

Change the OS Project name (e.g. OSC61). Die OS Project name must be unique within the complete MultiProject.

Configuration of the Network Connections

After all the Stations are defined you have to configure the Network Connections for the Communication between the OS Stations and the AS Stations and for the Communication between AS.

The settings for the Network Connections between the AS are not part of this description.

This description only refers to the communication between the AS und OS Stations. For each Connection between AS and OS Station an S7-Connection must be configured.

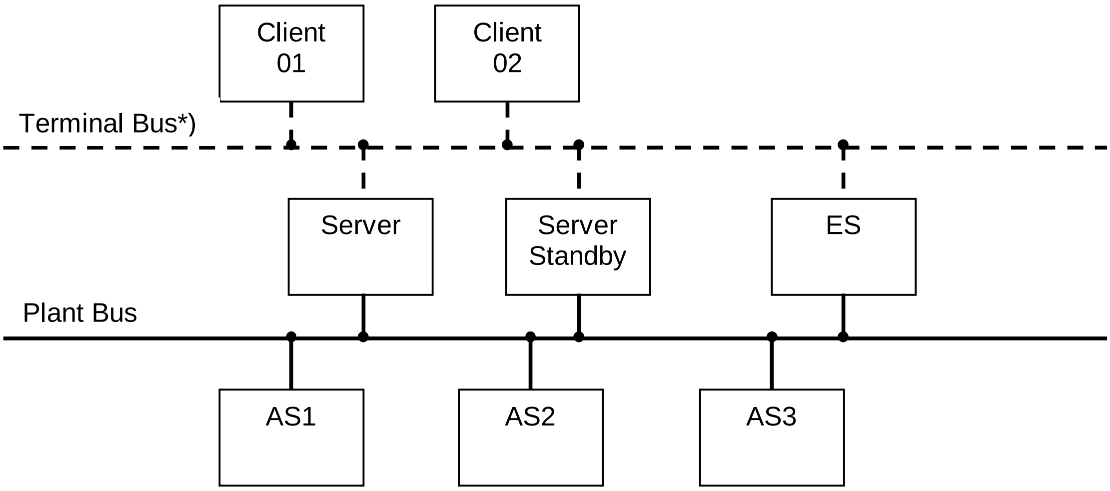
To define the connection, the CPs of the different Stations must be linked to an Ethernet Bus.

The CP of the AS is linked via “Plant Bus” or "Process Bus" to the Server Stations. For the Process Communication normally ISO Protocol is used.

The OS Clients are linked via “Terminal bus” to the Server Stations. In this Network usually TCP/IP Protocol is used.

In a complete Network this would look as follows:

Example with 3 AS, 1 redundant Server-Pair, 1 ES and 2 OS Clients:

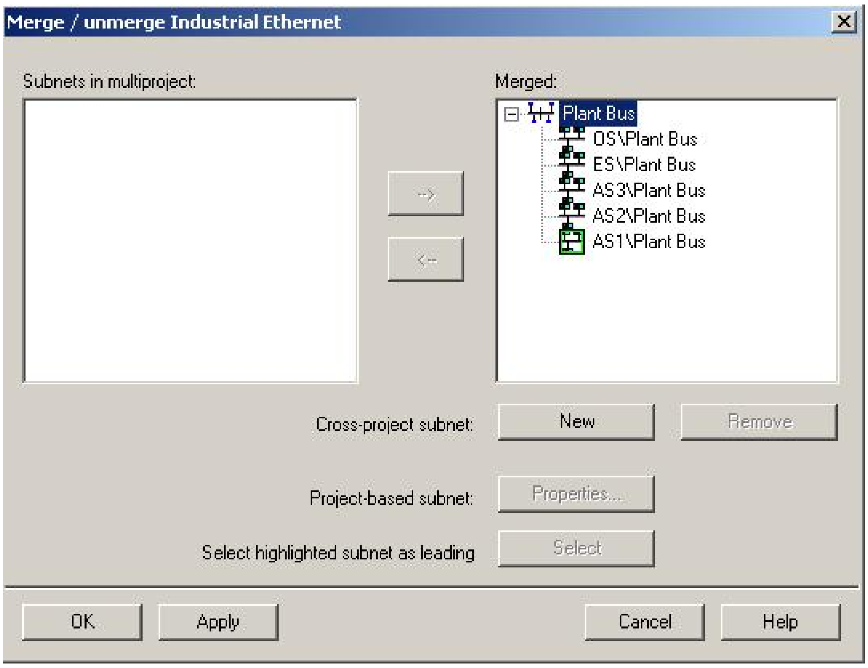


*) The CPs for the Terminal bus communication cannot be displayed in the Network Configuration of PCS 7 , otherwise the Hardware cannot be downloaded to the PC Stations. For the communication to the Terminal bus no configuration in Netpro is needed.

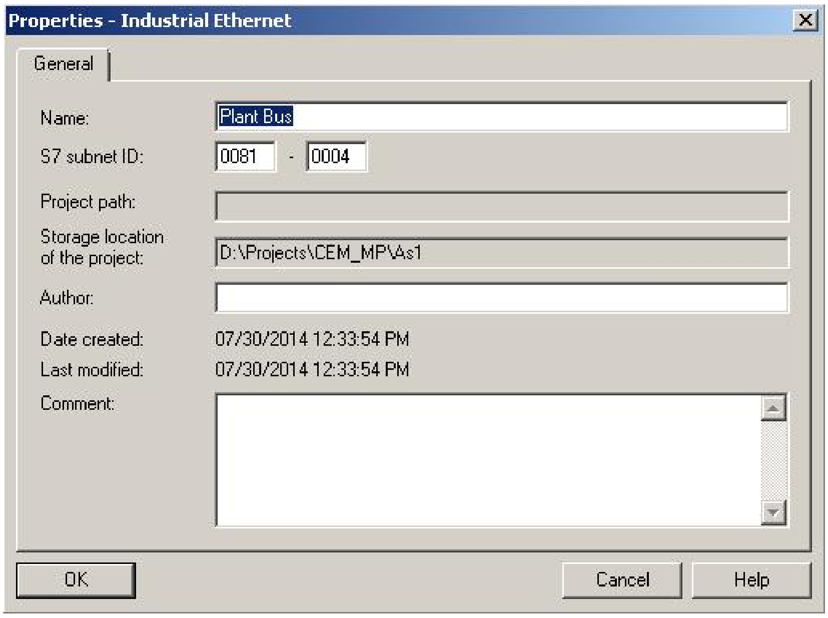
In a Multiproject the Stations are often located in different Projects. Each Project has its Network Configuration which contains Ethernet Connections to Plant Bus or Terminal bus.

To enable a Network Connections between different Network Configurations the Buses must be merged. In the SIMATIC Manager you will find the merge function if you select the Multiproject and use right mouse button option *Multiproject → Adjust Projects*.

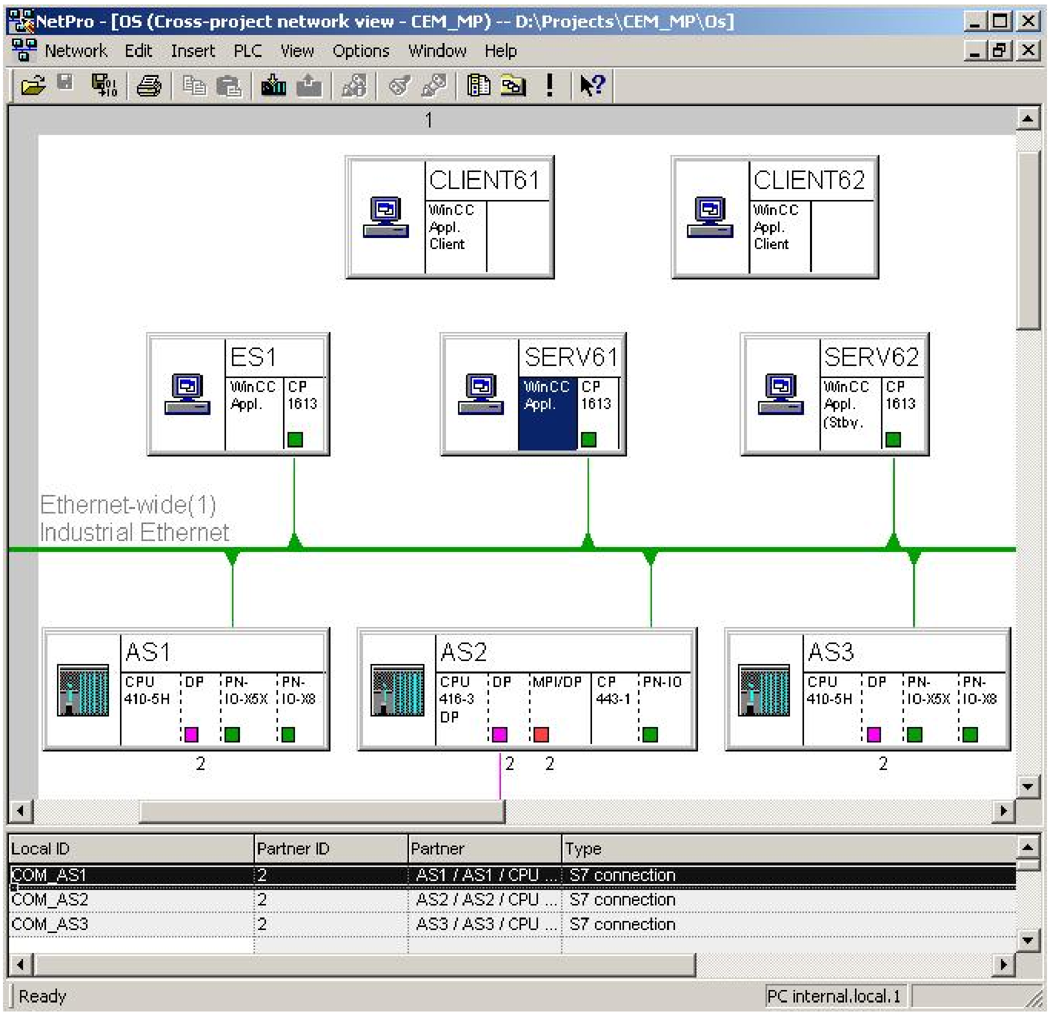
Select Ethernet and press Execute and the Window for Industrial Ethernet merge/unmerge will open and allow you to combine the Ethernet buses from different Networks.



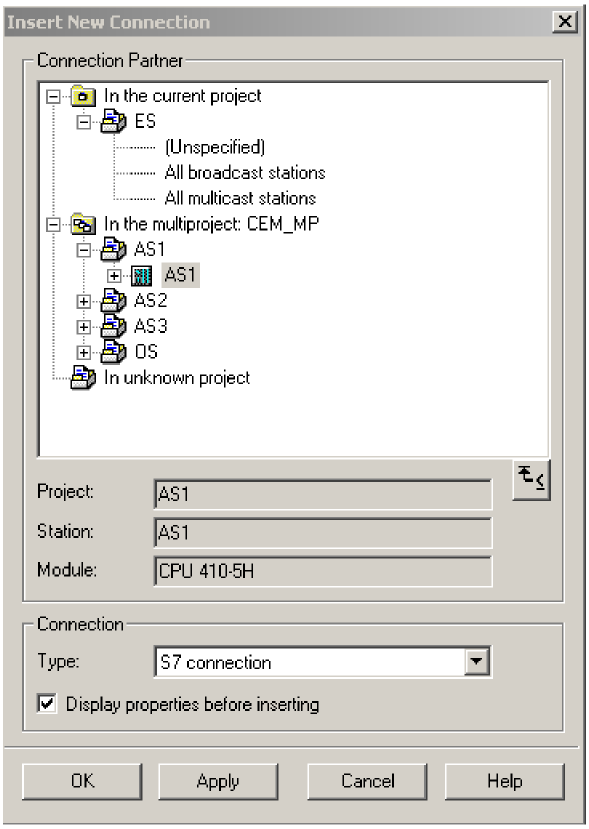
The highlighted subnet determines the global properties for the common Network. Under Properties you will find the S7 Subnet ID:



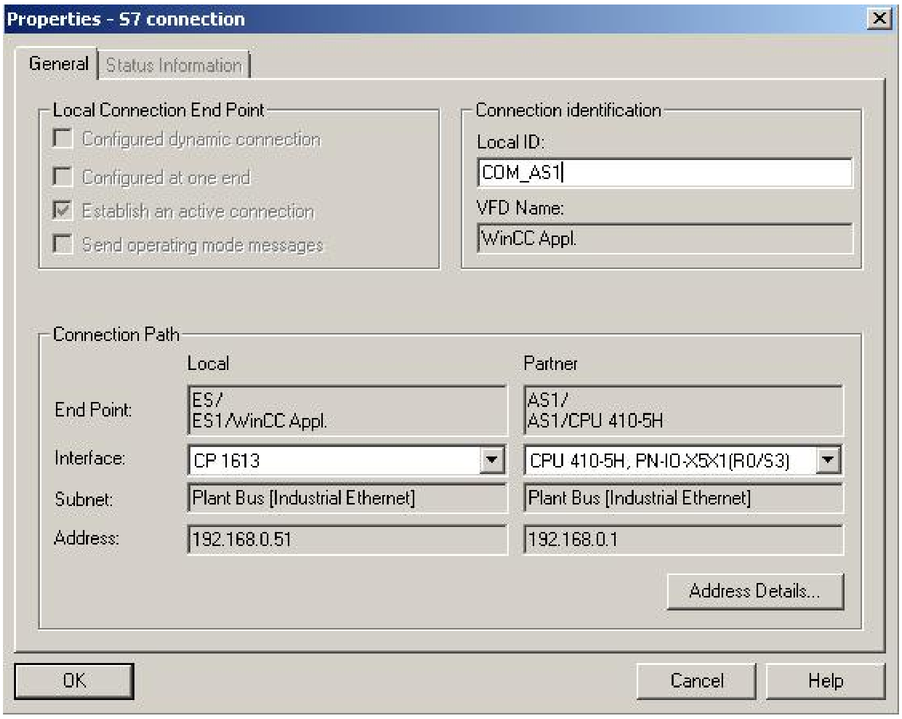
After the Network is merged the Network Connections between OS Stations and the AS Stations can be defined.



To add the S7 connection, select the WinCC Application of the OS-Project or the CPU of the AS and use right mouse button for selection of “Insert New Connection”.



Select the AS (or the WinCC Application of the OS) and press Apply. Save with ok.



We recommend changing the Connection identification for Local ID from S7 connection_1 into a more meaningful name, e. g. COM_AS1.

Note: The Connection identification for Local ID must be identical for the connection of all OS-Projects to this AS.
Example:
COM_AS1 for the Communication to AS1
COM_AS2 for the Communication to AS2
COM_AS3 for the Communication to AS3
etc.
The name will be used as Connection Name in the OS Compile for “Named Connections”.

Confirm the Settings with OK.

Enter the Network Connections between all the OS Stations (Server, Standby-Server and Single Station) and the AS.

Download for HW Configuration and Network Connections

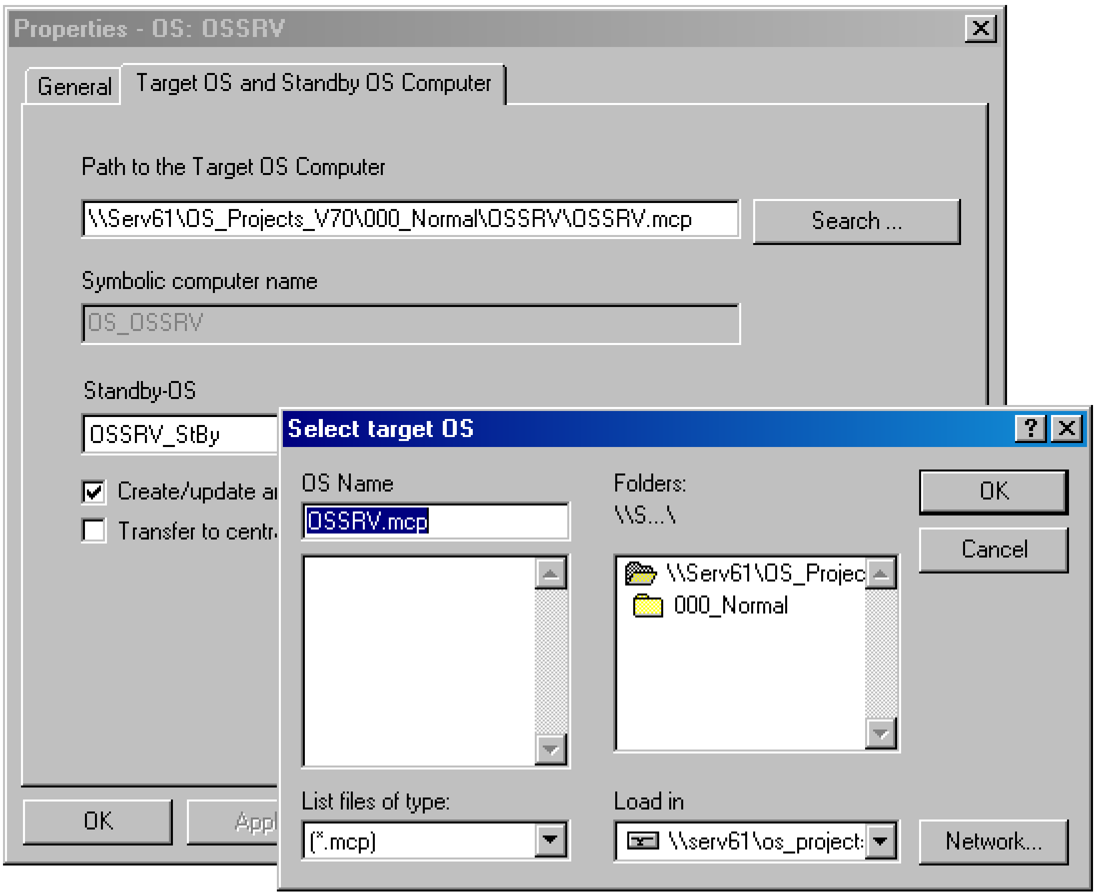
After the Station Configuration is finished and the Network Connections are complete the Configuration Data has to be transferred to the AS Stations and to the OS Stations.

Before this can be done, a few preparations are required:

In the SIMATIC Manager select the OS-Project for each Station and go to Object Properties

- For the Server
  select the Path to the Target OS Computer
  select the Standby-OS
- For the Standby-Server
  select the Path to the Target OS Computer
  the Primary Server is already selected
- For the Single Stations and OS Clients
  select the Path to the Target OS Computer

Example for selection of the target OS:



Download

1. Before you can download the configuration you have to select the Access Path to PC internal (local).
2. For the Download to the Stations always start with the Engineering Station. The Network Configuration for the Engineering Station has to be exact in order to enable the Download to the AS Stations and to the other PC Stations.

   Check the Configuration Editor to be sure that everything is configured correctly!
3. Download the Hardware Configuration to the AS.
4. Download the Network Configuration to the AS
5. Download the Hardware Configuration for each PC Station (Single Station, Server, Standby-Server) and double-check the Configuration Editor on each Target Station in order to make sure that the configuration is correct.
6. The OS Clients don’t have a Configuration Editor and the Connection via Terminal bus is not configured at all. For this reason the Hardware Configuration must not be downloaded to the OS Clients!

Plant Structure definition

In the component view of the SIMATIC Manager the Stations were inserted and the plant configuration was defined. Now you have to use the plant view or the process object view of the SIMATIC Manager to structure the plant according to technological criteria.

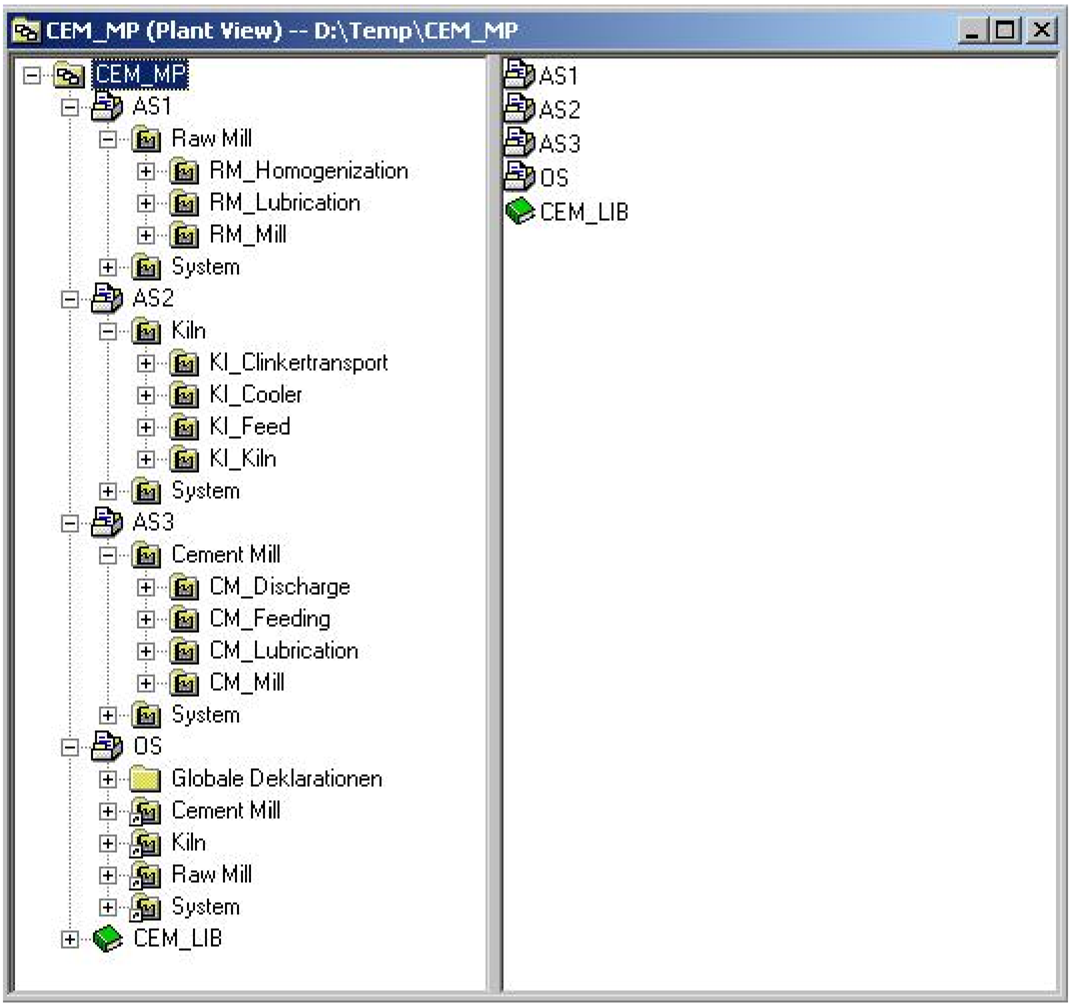
In the plant view the plant is structured technologically into plant sections (AREA), as in the example into Raw Mill, Kiln, Cement Mill. Coincidentally we have only one Area per AS but there could be several areas as well.

Typically the first hierarchy folder is defined as AREA (there are other options in PCS 7 but they are less useful).

Caution: The plant structure is later on used in the overview area of the OS system. Therefore it is extremely important to consider the picture hierarchy from the very beginning. The areas are shown in the selections buttons of the overview area and user rights can be given per area. Also the messages are filtered per area.

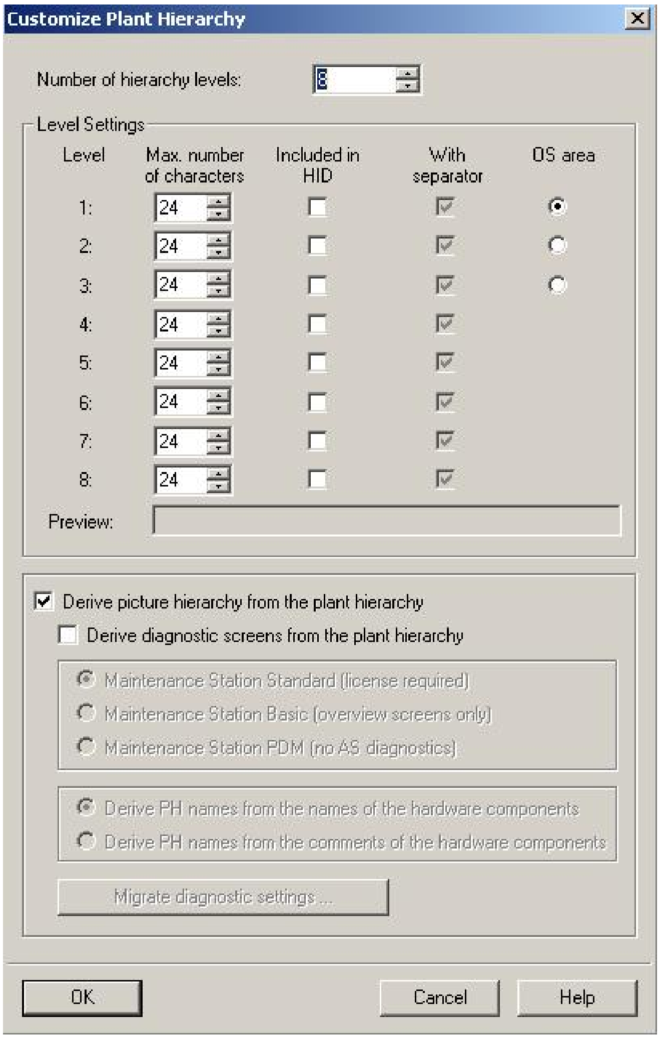
Below the areas there can be units, to which you may assign detail pictures. Please make sure that the CFCs area assigned to the correct units, otherwise the summarizing indication in the overview area will not work.

Example:



The names of the hierarchy folders can be limited for maximum number of characters. The names can be included completely or partly for the designation system.

The settings for max. number of hierarchy folders, max. number of characters, include in designation you will find if you select a hierarchy folder with the right mouse button and use *Options -> Plant Hierarchy -> Settings.*



For a Cement plant 3 hierarchy folders should be (more than) enough. For the diagnostic of the hardware up to 8 hierarchy levels are required. If you want to derive the diagnostic screens from the plant hierarchy, you have to select the function "Derive diagnostic screens from the plant hierarchy".

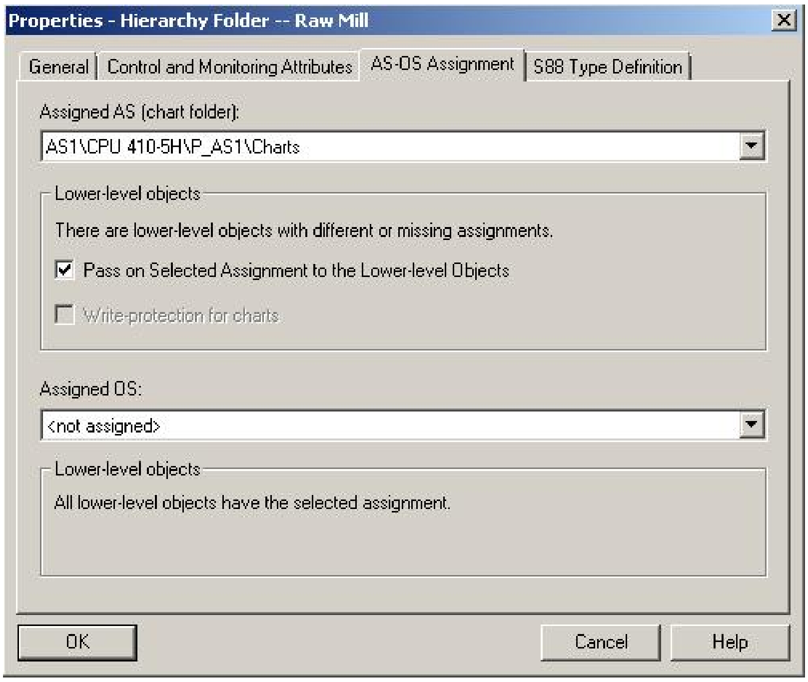
With the OS Area you specify which Hierarchy level is used the Selection Buttons in the Overview Area. Also the Operator right can later on be specified per Area.

The picture hierarchy must be derived from the plant hierarchy. Otherwise some functions cannot be guaranteed any more. During OS Compile the WinCC Picture tree is then generated automatically.

We don't recommend including the Hierarchy folder name in the Tagging system because the tagnames would be too long.

Note: If you insist to include the Hierarchy folder name in the tagging system you have to deselect this option for the System Chart. This is done in the Properties of the Hierarchy folder under "Control and Monitoring Attributes".

Before a CFC or SFC can be inserted in the Plant View or the Process Object View the Hierarchy folder has to be assigned to a chart folder in a AS. Before Pictures can be inserted in the Plant View the OS assignment is required.
Select the hierarchy folder and open the object properties. Use tab *AS and OS assignment* to select the chart folder for the assigned AS and the assigned OS.



All Elements you add to this hierarchy folder will be assigned to this AS/OS. If you pass the Selected Assignments to the Lower-level Objects, the subordinated hierarchy folders get the same AS/OS Assignment.

Important: Always use the Plant view or Process Object view for creating new elements (CFC, SFC, Pictures). If you create the elements in the component view, the plant assignation will be missing.
Exception: Pictures which shall not appear in the overview must be assigned directly to the OS-Project (in the component view).

Create a Project library

Before you start with the configuration of the individual AS, you have to create a project library, which must contain the latest version of all standard symbols, blocks and default charts which are used later on in the program.

The project library contains at all symbols, blocks and charts from the CEMAT library. It may also contain additional blocks, charts and models.

During the installation of CEMAT the CEMAT library (or libraries) was (were) installed in directory C:\….\SIEMENS\STEP7\S7LIBS:

- The CEMAT library ILS_CEM contains the S7-Program CEM_ALL with standard symbols, blocks and charts for CEMAT.
- For project standards with key unequal to ‘000’ the library PRO_CEM contains the project specific blocks. (For project key = ‘000’ the library PRO_CEM does not exist.)

In order to create the project library proceed as follows:

- At the moment the project library does not contain an S7 program or it contains an empty S7 program folder (in case the wizard was used to generate the project). In the second case you have to delete the S7 program folder.
- Then copy the complete S7 program folder CEM_ALL of the library ILS_CEM (including all symbols, blocks and charts) into your project library.
- The program folder of the Project library can be renamed, e. g. to P_LIB.
- In case of a project standard you have to open the library PRO_CEM and to copy all blocks into you project library as well (overwrite the existing blocks!).
- If you want to use further blocks in addition in your project you should also copy these block into the project library.

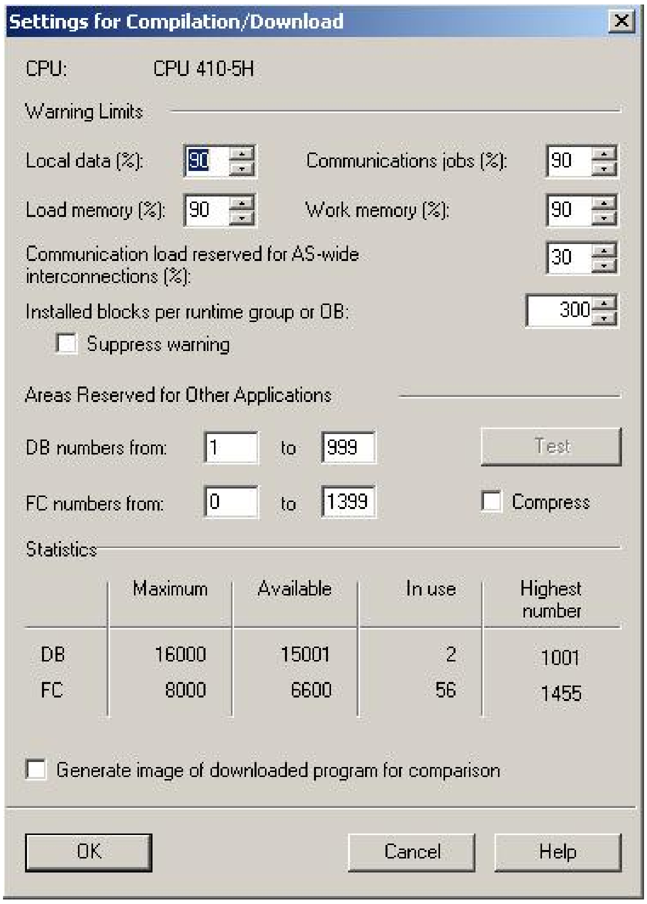
AS Configuration

Preparations for programming with the CFC

The following settings have to be carried out in the CFC Editor. This is only possible if a CFC exists. Therefore in the component view open the chart folder of your AS and create a new (Dummy-)CFC. With a double-click on the CFC the CFC editor will open.

Define illegal Ranges for CFC blocks:

Under *Options -> Customize -> Compile/Download…* you have to define the illegal ranges:



DB 1	to	999
FC 0	to	1399

The default value for Installed blocks per runtime group is 50. This leads to warnings during the compilation of the CFC. You can increase this number (e.g. to 300).

Confirm with ok. A window appears which tells you that the CFC is already using the area which is reserved for other applications. Confirm this as well in order to accept the changes.

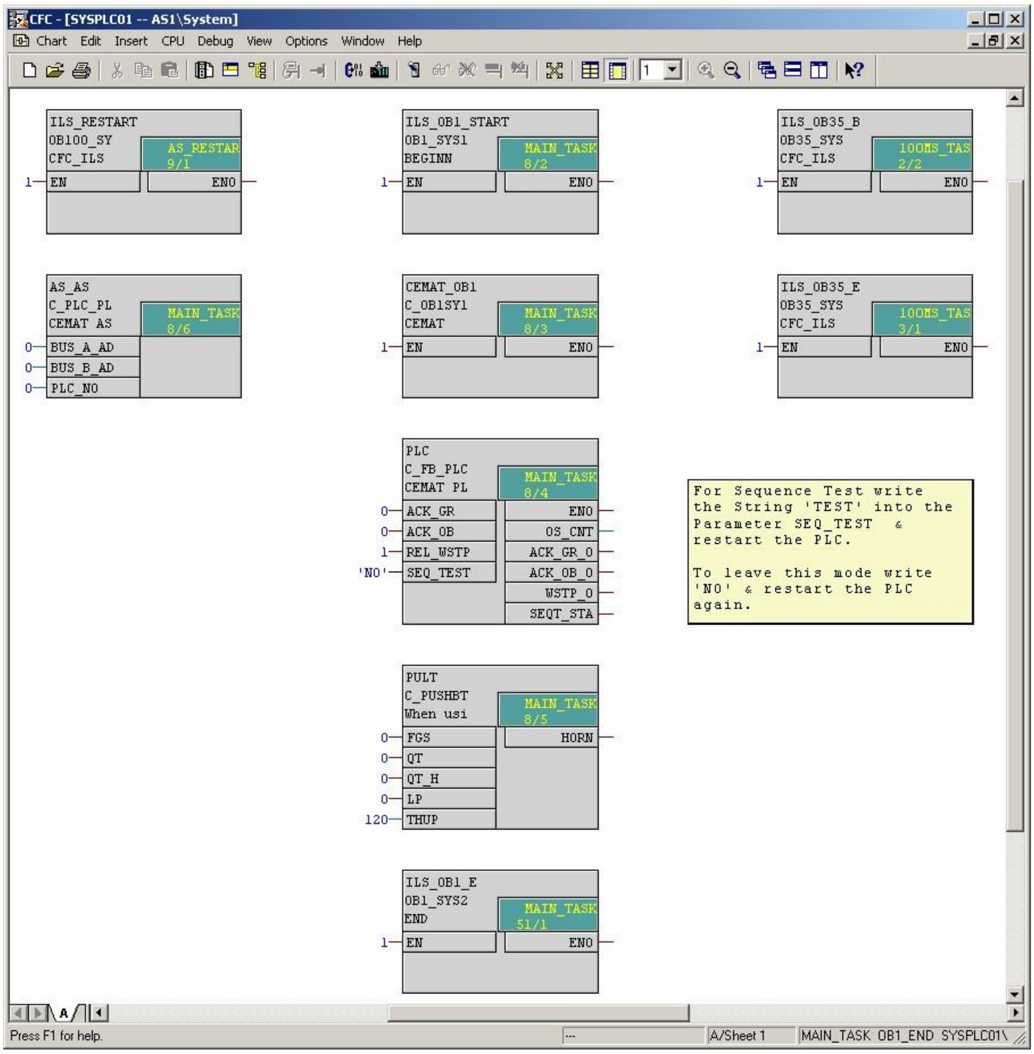
The (Dummy-)CFC which you have created in the component view was only needed in order to make the settings in the CFC-Editor. You can delete it now.

Copy Standard Symbols and blocks into the PCS 7 Project

After the project library was created which contains all symbols, blocks and charts for CEMAT, you can copy this to the AS.

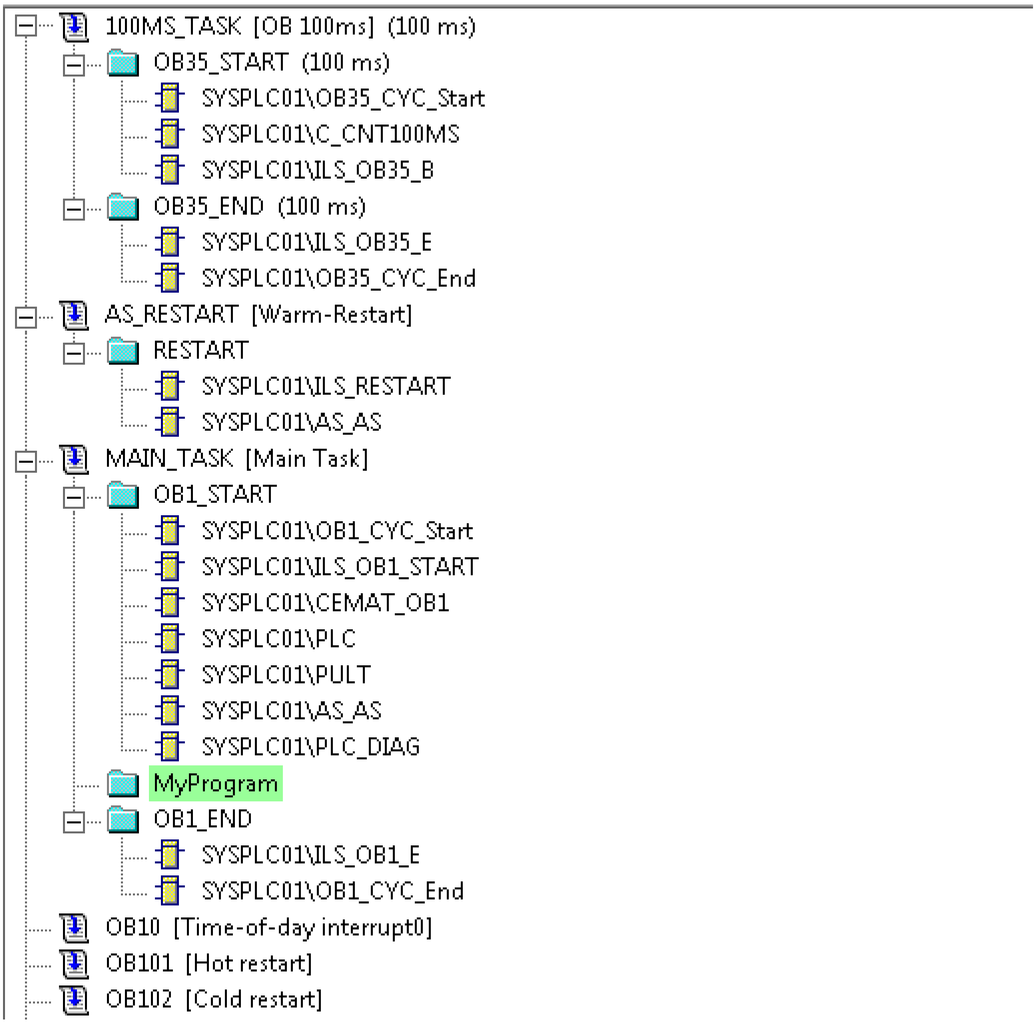
1. Copy first the symbols from the project library into the S7 Program Container of your AS. As the AS does not contain any symbols, you can copy the complete SYMBOLS from the program folder of the project library into the program folder of the AS.
2. Copy all blocks from the project library into the block Container of your AS.
3. Copy the system chart into the AS. The chart must be copied in the plant view!

   In the chart folder of your project library you will find the system chart SYSPLC00. This system chart is required once in each AS.



First define in the plant view a hierarchy folder for the system chart. Then copy the system chart from the library into this hierarchy folder. Change the name of the system chart according to the AS number (e. g. SYSPLC02).

4. Make sure that the blocks from system chart are installed at the correct position in the Runtime sequence, which should be as follows:



The Cemat blocks need to be installed in OB1, between Runtime group OB1_START and OB1_END.

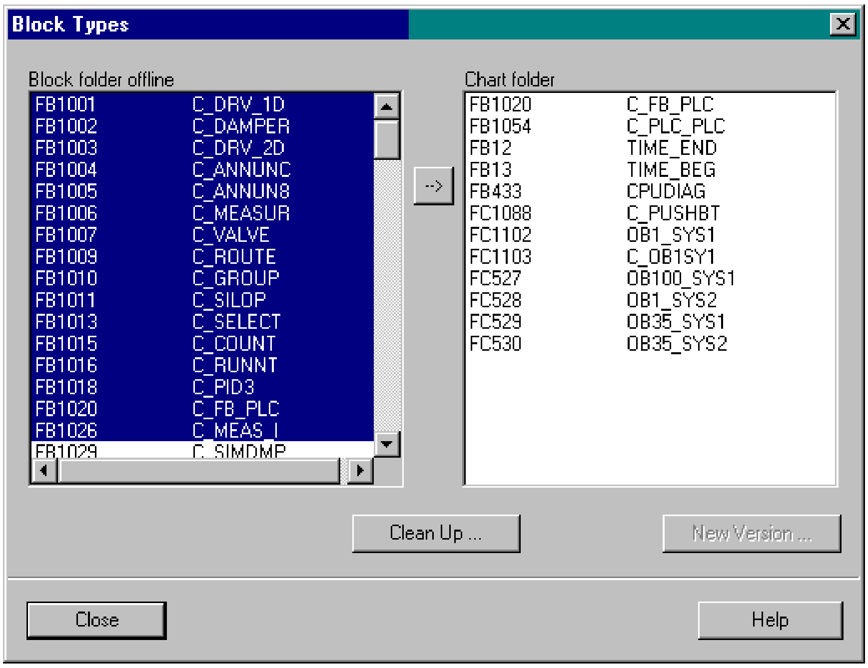
HOLCIM Standard: Parameter REL_WSTP at block C_FB_PLC must be set to 0-Signal. Otherwise the group start will be interrupted in case of a warning:

Caution: Block C_PLC_PLC must only be configured if the CEMAT AS-AS-Coupling is used (for communications to older CEMAT systems, e.g. CEMAT V4). **Only in this case** change Parameter PLC_NO into the PLC-NO of the own AS. The CP-Address of the CP443 for Bus A and Bus B (e. g. 16372 and 16376) must then be entered into the parameters BUS_A_AD and BUS_B_AD.
The function block is called once in the restart (PLC_RESTART) and once in OB1 (MAIN_TASK). For detailed description see chapter AS-AS-Coupling.

5. The following step is optional and may be used as a preparation for the CFC engineering.

After updating the Chart Folder with the Cemat blocks from the offline block folder, the Cemat blocks are available under Catalog CEMAT.

In the CFC-Editor go to *Options -> Block types*. Select the blocks from the offline block folder and copy it into the chart folder:



List of blocks, which have to be copied into the chart folder:

Absolute	Symbol	Task
FB1001	C_DRV_1D	OB1
FB1002	C_DAMPER	OB1
FB1003	C_DRV_2D	OB1
FB1004	C_ANNUNC	OB1
FB1005	C_ANNUN8	OB1
FB1006	C_MEASUR	OB1
FB1007	C_VALVE	OB1
FB1008	C_PROFB **)	OB1
FB1009	C_ROUTE	OB1
FB1010	C_GROUP	OB1
FB1011	C_SILOP	OB1
FB1013	C_SELECT	OB1
FB1015	C_COUNT	OB35
FB1016	C_RUNNT	OB1
FB1018	C_PID3	OB35*)
FB1021	C_MAX_CYC	
FB1026	C_MEAS_I	OB1

Absolute	Symbol	Task
FB1034	C_SIMOS	OB1
FB1036	C_STO_MA	OB1
FB1037	C_STORAG	OB1
FB1038	C_ANA_SEL	OB1
FB1039	C_POLY3	OB1
FB1046	C_ODA	OB1
FB1057	C_INTERL	OB1
FB1058	C_INTER5	OB1
FB1059	C_RELMOD	OB1
FB1060	C_Intlk08	OB1
FB1061	C_Intlk04	OB1
FB1095	C_TIS_B **)	OB1
FB1096	C_TIS_N **)	OB1
FB1097	C_TIS_S **)	OB1
FB61	**CTRL_PID**	**OB35*)**
FC1017	C_MUX	OB1
FC1018	C_ADAPT	OB1

*) This block can be called from any Time interrupt OB.

**) These blocks only exist in Project Version 004 Holcim

Compile and Download CFC

Once the configuration settings have been made in the system chart the AS Program is loadable and executable.

Warning: To start the AS with the Programmer only **Warm Restart** is allowed!

After these steps you can start with the AS Engineering. In chapter 6 of this manual "AS Engineering" you find some additional advises.

You may also continue with the configuration of the OS System. To enable the OS compile, the compile of the CFC must be carried out before. For the first time a complete compile is required.

OS Compile

Compile the OS (for all Servers and Single User Systems). With the OS Compile, in the Tag Management of the OS the driver SIMATIC S7 PROTOCOL SUITE is added. Under "Named connections" you will find the variables of the group instance list and of the System chart.

OS Configuration (Single-User System)

The WinCC Project (OS-Project) for the Single-User System was already defined with the SIMATIC Manager or Network Configuration. The following settings have to be carried out in the OS-Project, which means you have to open WinCC. This can be done directly from the SIMATIC Manager through selection of the OS with right mouse button and *Open Object* of from the Start Menu under *Start -> Simatic -> WinCC -> WinCC Explorer*.

Important: The following steps must be carried out from the beginning in the engineering language, which is used for the project. If you change the language afterwards the language dependent setting must be repeated.
The language settings in SIMATIC Manager are not consistent with the settings for WinCC!!

Project Properties

Adaptations for the Project itself can be carried out under Properties.

Folder “General” contains the Project Type.
Generating a new OS-Project of type “WinCC Application“, automatically a “Multi-User Project” is created. As in a Single User System no Server License is available you have to select the Project type Single-User Project.

Note: Don’t delete the startup list!

In Folder “Update Cycles” the time values for the Update cycles are defined. The CEMAT Symbols generally use “User Cycle 1”. The default value of this is 2000ms. You have to adapt this value according to your requirement, e.g. 1000ms.

OS Project Editor

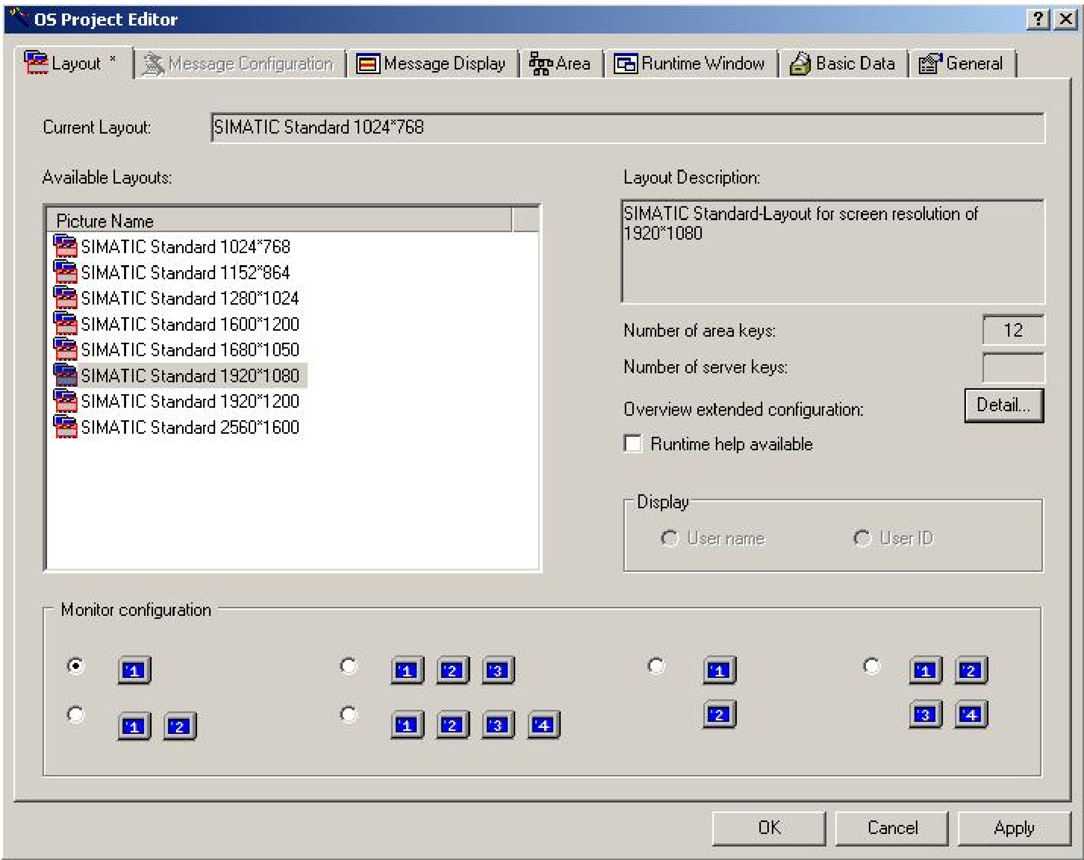
Settings for the user interface used by the plant operator for monitoring and controlling the plant during process operation In the OS Project Editor.

A detailed description to this item you find in the PCS 7 Configuration Manual Operator Station. The following pages describe only the settings relevant for CEMAT.

In Tab Layout choose the screen resolution according to your Monitor. In Cemat V8.2 the default resolution is 1920x1080, but the Cemat Standard pictures are also available for 1280x1024, 1600x1200 (not for Project Version 004), 1680x1050 and 1920x1200.

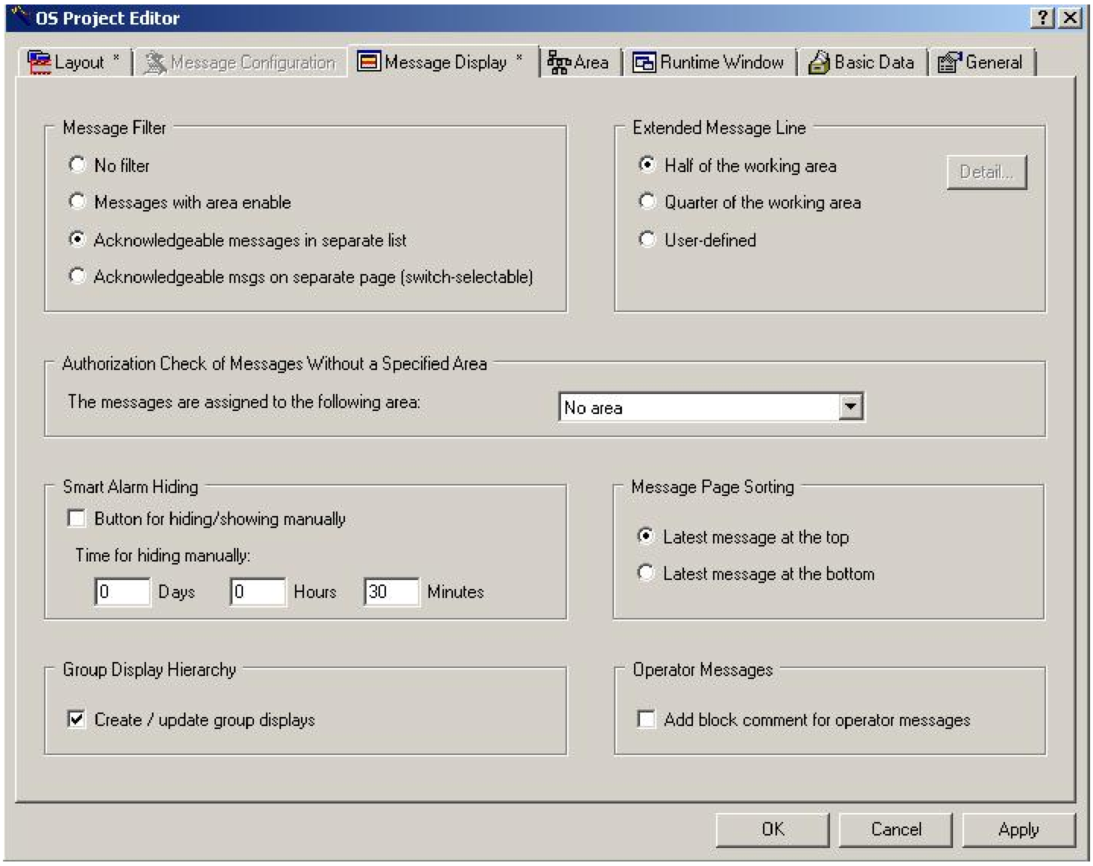
Also select the Number of horizontal and vertical area keys under Detail (Picture Tree) and the Monitor Configuration of your PC.

Example for the Layout of a Single Station:



Note: If you select a different screen resolution (1280x1024, 1600x1200, 1680x1050 or 1920x1200), the standard pictures @AlarmOneLine.pdl, @TopAlarmNew.pdl, @C_AlarmListing_Cemat.pdl, @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl and @Overview1.pdl have to be exchanged.

Message display for a Single Station:



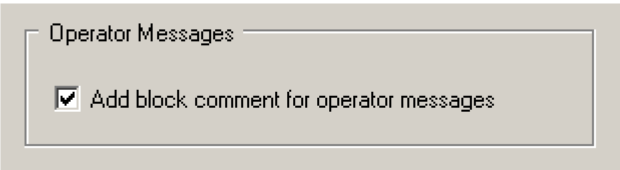
The Message filter has to be set to “Acknowledgeable messages in separate list”.

The group display should be created and updated automatically.

Note: The sorting of the message pages is only used for the Standard WinCC Alarm list, and not relevant for the Cemat Alarm list.
In the Cemat Alarm list the default sorting is "Latest message at the bottom". This setting can be changed under D:\Cemat_CS_Config in file C_Messages.cfg (see chapter Tips and Tricks).

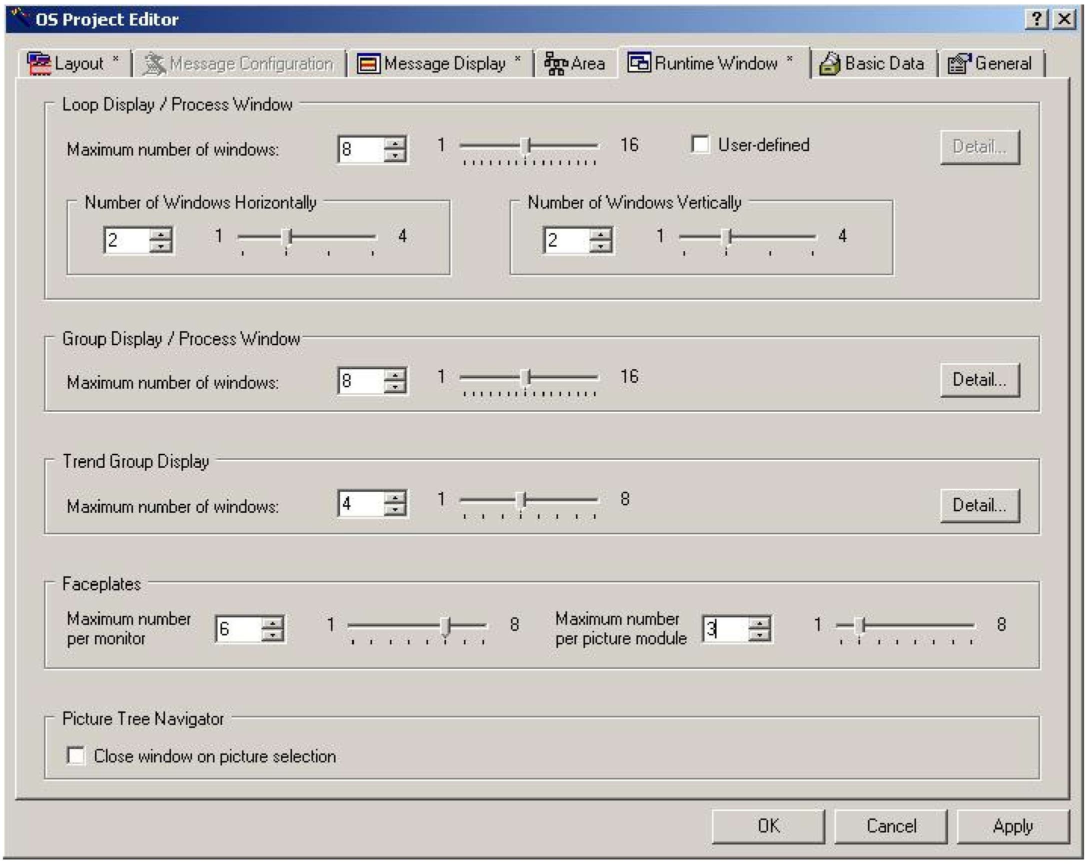
Optional Setting:

Operator Messages created by HMI come without block comment (default setting). If you like to see the Block comment with this kind of messages you have to select the following Option:



Note: Different to the Operator messages created in the AS, in the Operator messages from HMI the block comment is part of the Event Text.

Runtime-Window for a Single Station



Define the maximum number of windows that can be opened when faceplates and curves are called as well as the maximum number of faceplates that can be opened on a monitor.

Set the maximum number of picture modules per monitor to 6 and the maximum number per picture module to 3.

The rest of the settings in the OS Project Editor can remain as per default or can be adapted later.

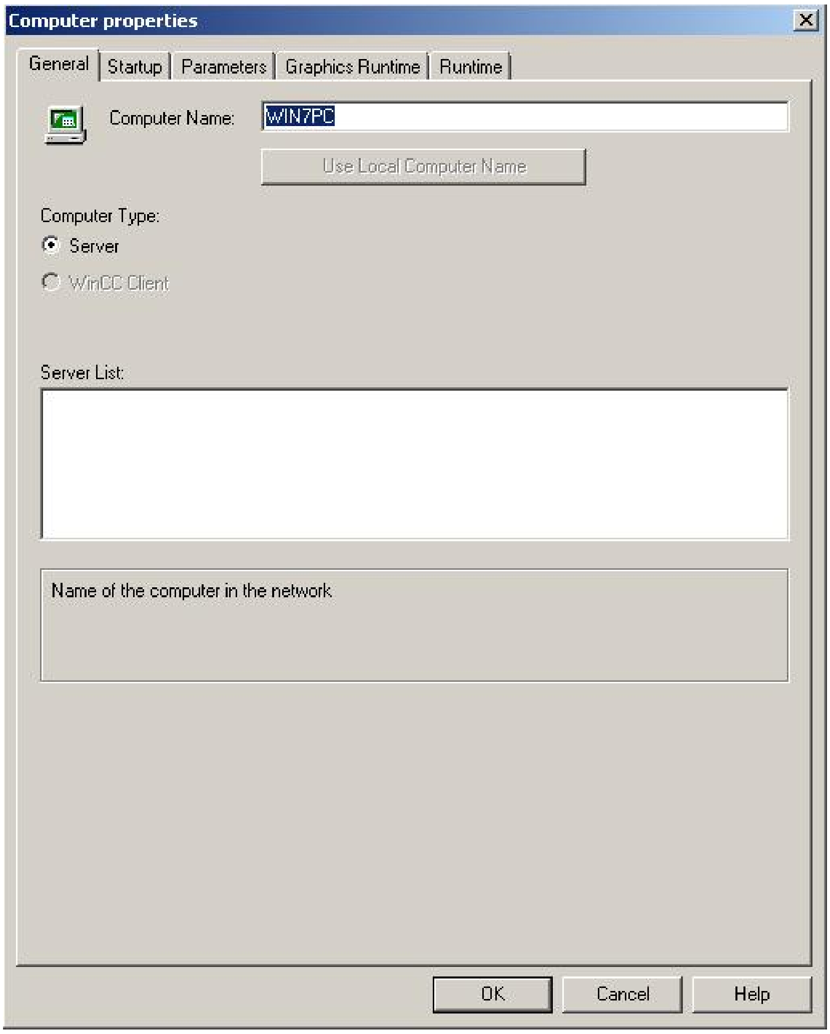
With “OK” all settings are carried out.

Caution: If for any reason you have to run the OS-Project Editor again, make sure that all files mentioned under CEMAT Specific Preparations are deselected in Folder “Basic Data”. Otherwise the CEMAT Settings are overwritten with the delivery state of PCS 7 .
→ See chapter CEMAT Specific Preparations!

Computer properties

The computer properties must be checked and adapted to the requirements of the plant. Most of the settings can probably not be entered at the beginning of the engineering and must be adapted later (sometimes after commissioning).

Folder General shows the Computer Name and the Computer Type

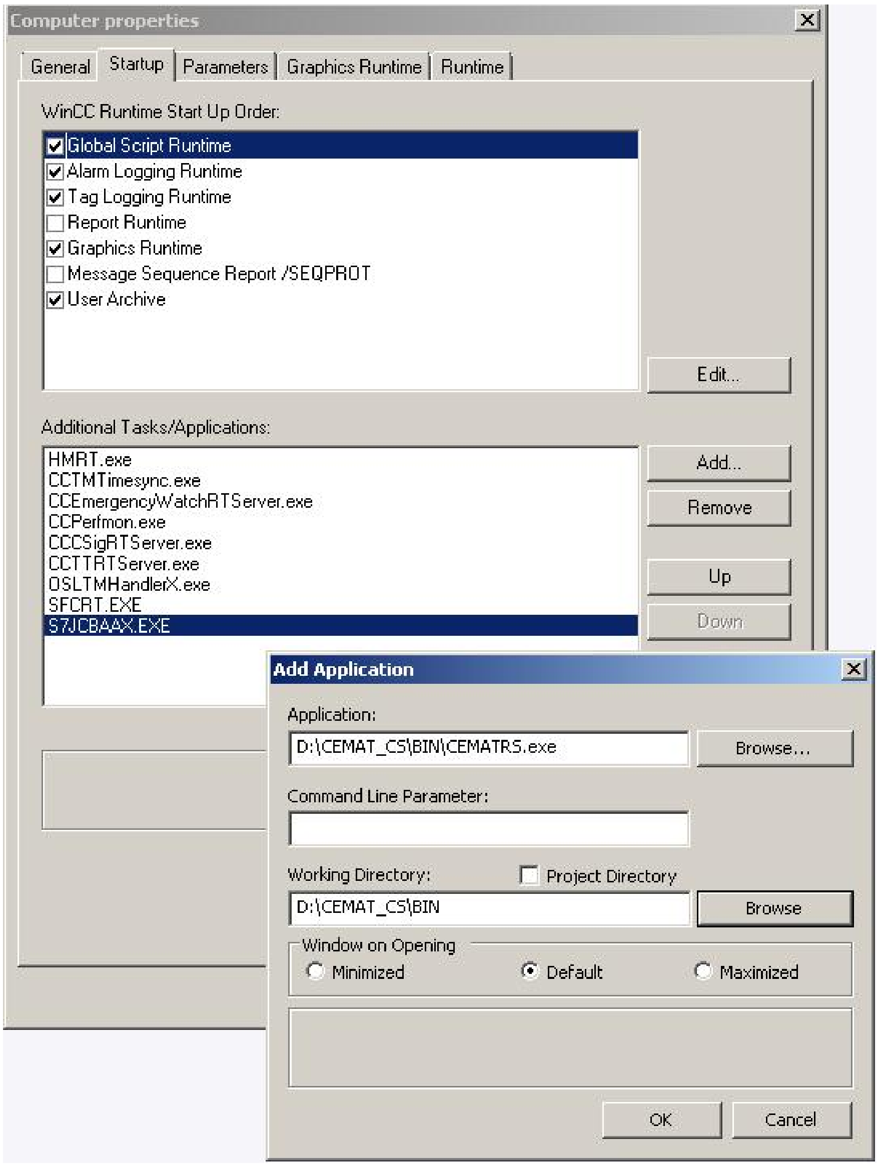


Caution: The Computer Name for all OS Projects is the name of the Engineering Station. During the download of the OS-Project to the individual PCs the name is automatically replaced by the Name of the PC Station.

Folder Startup shows the applications which are started after Activation of the Runtime System

Under Additional Tasks/Applications you have to add the CEMATRS.exe to the Startup list. The application is located in Directory D:\CEMAT_CS\BIN.

CEMATRS.exe should be called at the end.



Insert both, the Application and the working Directory and save with ok. Move the application to the last position.

Tag Management

Cemat Tags

Cemat needs a number of internal variables which must be added in the WinCC Configuration Studio. The variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

Open the WinCC Configuration Studio for the Tag Management

Use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

After a successful Import the following Groups and Variables have been added to the Tag Management:

Group “Cemat”

Name	**Data Type**	**Length**
C_ALM	Unsigned 16-bit value	2
C_Empty	Text tag 16bit character set	255
C_AlarmNavigation *)	Text tag 16bit character set	255

*) C_AlarmNavigation is only needed for Holcim Standard (004)

Group “Horn”:

Name	**Data Type**	**Length**
C_horn_alarm	Binary Tag	1
C_horn_PLC	Binary Tag	1
C_horn_process	Binary Tag	1
C_horn_system	Binary Tag	1
C_horn_warning	Binary Tag	1

Redundancy Tags

CEMAT needs the Redundancy tags in any case, even if this is a Single User Station. The easiest way to create the Redundancy tags is to activate the Redundancy temporarily (and to deactivate afterwards).

After activating the redundancy the following internal variables have been added in group „Redundancy“:

@RM_MASTER
@RM_MASTER_NAME
@RM_SERVER_NAME
@RM_OFFLINE_UA_NAME

Note: In case of a Single User Station you have to deactivate the redundancy!

Alarm logging

All definitions for the alarm logging are carried out in the WinCC Configuration Studio.

Cemat projects require adaptations regarding color definitions and acknowledgement/flashing behavior. These adaptations are already prepared in Excel files Import_MessageBlocks.xlsx and Import_MessageBlocks.xlsx and can be imported from D:\CEMAT_CS\IMPORT.

Open the WinCC Configuration Studio and import the predefined configuration:

Message blocks

For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

The following values get imported:

Message blocks	Number of characters	Flash
Date	0	0
Time	0	0
Status	2	0
Comment	2	0
Info	2	0
Tagname	32	0
Area	24	0
Event	50	0
Tag Comment	80	0
FCL	80	0

Message configuration

For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

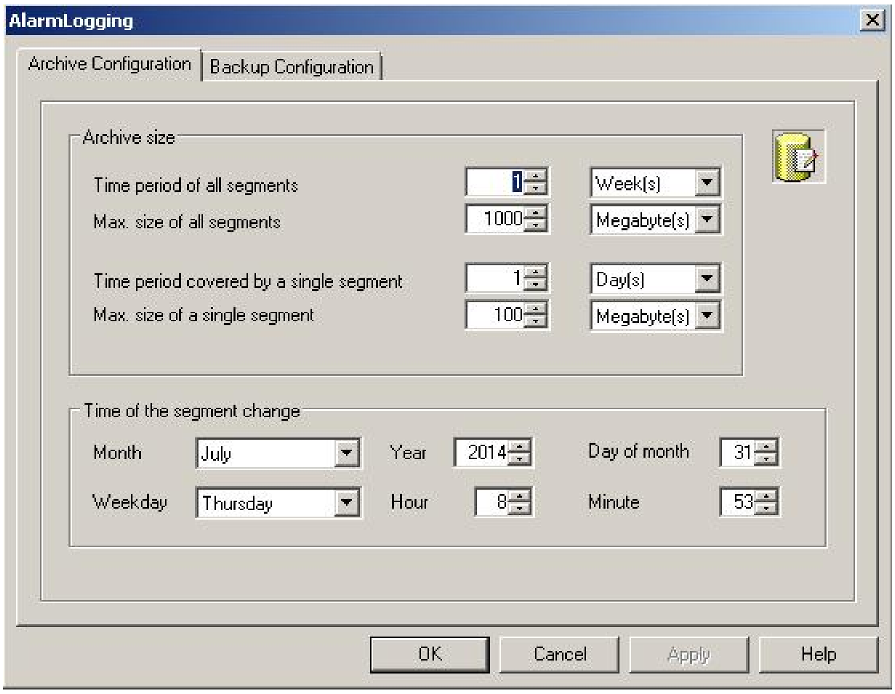
The following values get imported:

		Color definition		
Message class	Message types	come in	went out	acknowledged
Alarm	Alarm High	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Alarm	Alarm Low	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Warning	Warning High	F: 0; 0; 0 B: 255; 255; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 255; 128
Warning	Warning Low	F: 0; 0; 0 B: 255; 255; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 255; 128
PLC process control messages	Failure	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Process message	Process message.	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Operator Message	Operator Message.	F: 0; 0; 0 B: 0; 255; 255		

F = Font color
B = Background color

Archive Configuration

Under Messages → Archive Configuration → Properties the Archive Configuration must be adapted according to the requirements of your project:



Tag logging

For archiving of the Process Values a Process Value Archive must be configured in the WinCC Configuration Studio.

Dependent on the number of Process Values to be archived, the Acquisition and Archiving cycle and the requirements regarding archiving period, the Process Value Archive may be located locally on the OS Station or on a separate PC Station dedicated for Archiving, the PCS 7 Process Historian.

For detailed information refer to PCS 7 Manuals for Process Historian and Information Server.

For small and medium-sized applications the Process Value Archive can be located on OS Station. For the calculation you have to consider the hard disk memory, the number of Segments and the Segment size.

The Archive Tags will be created during the Engineering, preferable in the CFC via a selection of the I/O for “Archiving” or “Long-term archiving”.

“Archiving”
The I/O will be archived on the OS or on an archive server

“Long-term archiving”
The values archived in the OS of on the archive server will be stored for long-term archiving on CD, DVD, MOD, tape, etc.

During OS compilation the Archive Tags are created in the Process Value Archive named “SystemArchive”. As acquisition cycle the minimum acquisition cycle is used. For values with different acquisition cycle it must be adapted later in the Tag Logging.

Note: PCS 7 can only create one common Process Value Archive. Multiple Process Value Archives (e. g. splitted by AREA) are not supported in the automatic generation and in general not necessary.
Archives and Archive Tags could be created manually in the Tag Logging, but it is not recommended, because it means much more Engineering effort.

Required settings for CEMAT

In Cemat it is possible to open the Archive curve of a measured value directly via block icon or the faceplate of the C_MEASUR block. In order to achieve this, an archive tag with correct naming must exist for output MV of the C_MEASUR block.

Example:

Measured value name: 123_BC1/J01
Archive tag name: 123_BC1/J01.MV

Caution: If the name of the archive tag differs from the above mentioned rules, the CEMAT Faceplate does not find the Archive data and the archive curve of the measure cannot be shown.
Make sure that the Tag name of the measure does not contain any illegal characters for archive variables!

More information regarding Process Value Archives you will find in chapters AS Engineering and OS Engineering.
In case of Archive Splitting or manual creation of Process Value Archives please refer to chapter 14_Tips_Tricks – Process Value Archive.

User Archives

WinCC User Archives are used for the following CEMAT Functions:

Cemat Curve groups

User Archive **C_CURVE** is used to save the Cemat curve groups. The data is created/ updated in the Runtime system.

Note: In order to allow saving of a new curve group, at least one (dummy) line must exist and therefore be created during the setup of the project.

Cemat Object list

User archive **C_DriveList** is used to save the object list (list of objects associated to a route or group and list of objects associated to a drive). It gets filled when you open the Object list in the Runtime system and press the save button.

During the setup of a project this User Archive may remain empty.

Cemat Object Information

User archive **C_INFO** contains the object information for Display in the Information Dialog and in the Maintenance Dialog of the Cemat Objects.

For each Cemat Object one data set must be provided, containing all the information to be displayed. The data sets can be created with Engineering tool C_INFO.xlsx (see Engineering Manual, chapter 09_Engineering_Tools.doc) after the completion of the software engineering. In the Runtime System the information can be modified and/or completed.

Note: In order to allow adding of new object information, at least one (dummy) line must exist and therefore be created during the setup of the project.

Cemat Polygon Function

User Archive **C_POLY3** contains the parameters for the Polygon function and is only required if the Function Polygon is used. The information is created/ updated in the Runtime system.

Note: In order to allow saving of a new Polygon parameters, at least one (dummy) line must exist and therefore be created during the setup of the project.

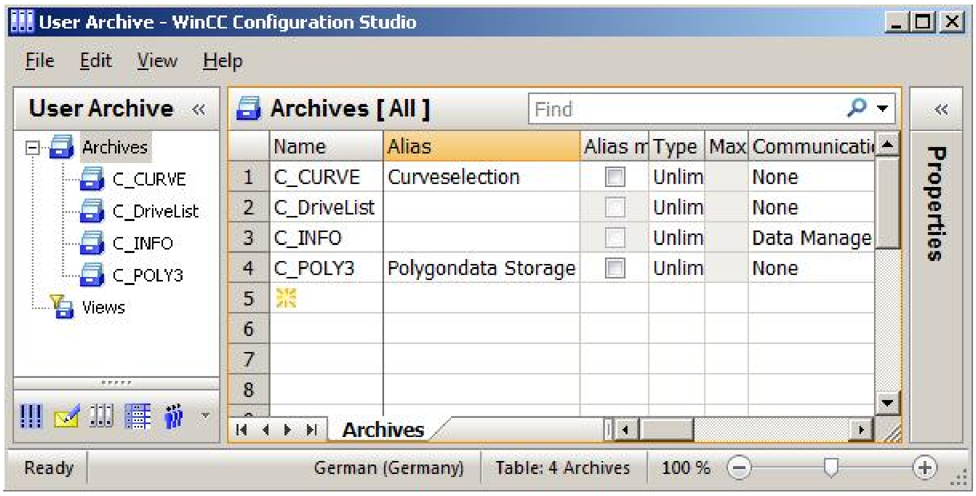
How to create the Cemat user archives

The Cemat User Archives are configured in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for the User Archive

In order to import the structure use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_UserArchives.xlsx and press Button *Import.*

After the import the following Archives exist:

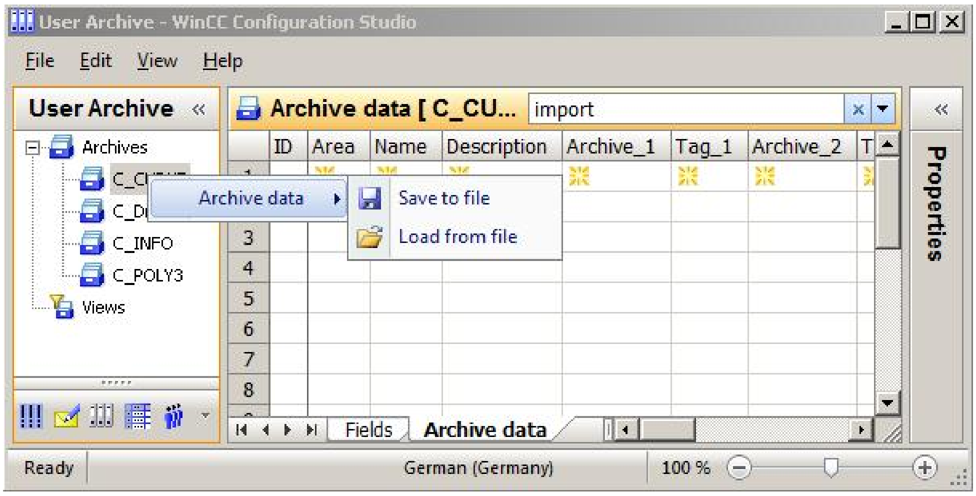


How to create the user archive data

For the user archives C_CURVE, C_INFO and C_POLY3 at least one line of data must be created during the setup of the OS-Project.

Therefore directory D:\CEMAT_CS\IMPORT contains csv-files with dummy lines for C_CURVE, C_INFO and C_POLY3 which can be imported. To import the data go to tab Archive data and proceed as follows:

With right mouse click on C_CURVE → *Archive data → Load from file*, select from D:\CEMAT_CS\IMPORT file C_CURVE.csv , and press Button *Import.*



With right mouse click on C_INFO → *Archive data → Load from file*, select from D:\CEMAT_CS\IMPORT file C_INFO.csv , and press Button *Import.*

With right mouse click on C_POLY3 → *Archive data → Load from file*, select from D:\CEMAT_CS\IMPORT file C_POLY3.csv , and press Button *Import.*

The user Archive C_DriveList does not need to contain data.

Add Authorization levels

Each Operation in the HMI is linked to an Authorization level which the person must have to carry out the function. Beside the Authorization levels of PCS 7 the operations in Cemat require additional Authorizations (18 – 29) for specific functions.

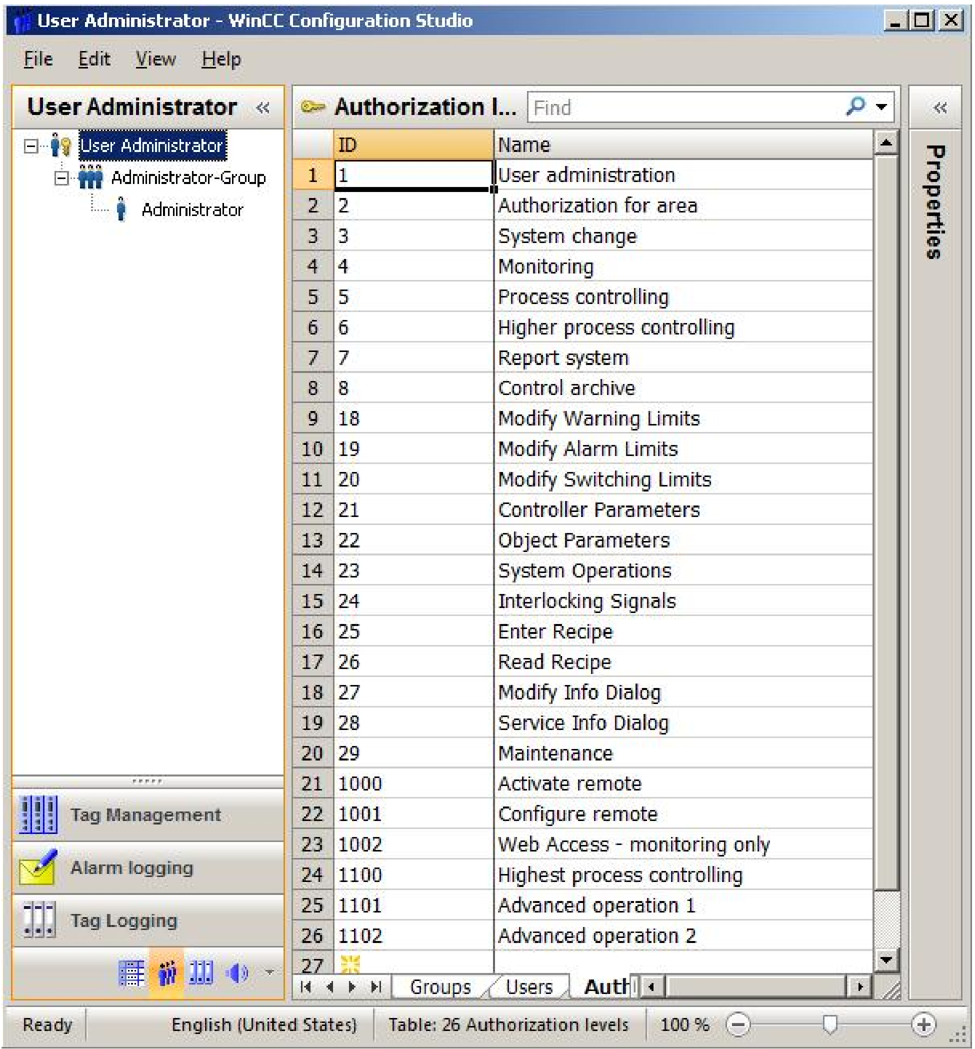
Each User group/User may get the permission for certain authorization levels in certain areas.

The Authorization levels, User groups and Users are defined in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for User Administrator

In order to import the additional Authorization levels 18 - 29 use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_AuthorizationLevels.xlsx and press Button *Import.*

After the import you will have the following Authorization levels:



If operations shall be permitted to restricted personal only, additional user rights can be defined.

Authorization levels “25 - Enter Recipe” and “26 - Read Recipe” are currently not used.

Operations and Authorization levels

The following table shows the user rights for the particular the CEMAT Functions. For some operations it is possible to give instance specific rights via block icon attributes “Processcontrolling_backup” and “HigherProcesscontrolling_backup” (see chapter OS-Engineering):

Object type	Operation	Right/ default right	Instance specific settings possible via Attribute	Project Code
C_DRV_1D	Start/Stop	05: Process controlling	Processcontrolling_backup	all
	Process parameter	22: Object Parameters		all
	Reset Operation	23: System Operations		all
	Bypass speed monitor	24: Interlocking Signals	HigherProcesscontrolling_backup	all
	Maintenance functions	29: Maintenance		all
	auto/man. Interl./man. Non Interl.	05: Process controlling	Processcontrolling_backup	006
	non interl. Single mode Rel.	23: System Operations		007
	non interl. Single mode on/off	24: Interlocking Signals	HigherProcesscontrolling_backup	004/007
C_DRV_2D	R1/R2/Stop	05: Process controlling	Processcontrolling_backup	all
	Process parameter	22: Object Parameters		all
	Reset Operation	23: System Operations		all
	Bypass speed monitor	24: Interlocking Signals	HigherProcesscontrolling_backup	all
	Maintenance functions	29: Maintenance		all
	auto/man. Interl./man. Non Interl.	05: Process controlling	Processcontrolling_backup	006
	non interl. Single mode Rel.	23: System Operations		007
	non interl. Single mode on/off	24: Interlocking Signals	HigherProcesscontrolling_backup	004/007
C_DAMPER	R1/R2/Stop	05: Process controlling	Processcontrolling_backup	all
	Process parameter	22: Object Parameters		all
	Reset Operation	23: System Operations		all
	Button “up”	05: Process controlling	Processcontrolling_backup	all
	Button “down”	05: Process controlling	Processcontrolling_backup	all
	Set point	05: Process controlling	Processcontrolling_backup	all
	Release inching mode	05: Process controlling	Processcontrolling_backup	all
	Maintenance functions	29: Maintenance		all
	auto/man. Interl./man. Non Interl.	05: Process controlling	Processcontrolling_backup	006
	non interl. Single mode Rel.	23: System Operations		007
	non interl. Single mode on/off	24: Interlocking Signals	HigherProcesscontrolling_backup	004/007

Object type	Operation	Right/ default right	Instance specific settings possible via Attribute	Project Code
C_VALVE	R1/R2/Stop	05: Process controlling	Processcontrolling_backup	all
	Process parameter	22: Object Parameters		all
	Reset Operation	23: System Operations		all
	Maintenance functions	29: Maintenance		all
	auto/man. Interl./man. Non Interl.	05: Process controlling	Processcontrolling_backup	006
	non interl. Single mode Rel.	23: System Operations		007
	non interl. Single mode on/off	24: Interlocking Signals	HigherProcesscontrolling_backup	004/007
	Override limit position	24: Interlocking Signals	HigherProcesscontrolling_backup	004
C_MEASUR	Warning Limits	18: Modify Warning Limits	HigherProcesscontrolling_backup	all
	Alarm Limits	19: Modify Alarm Limits		all
	Switching Limits	20: Modify Switching Limits	Processcontrolling_backup	all
	Simulation	24: Interlocking Signals		all
	Bypass Meas. Channel	24: Interlocking Signals		all
	Process Parameters	22: Object Parameters		all
C_GROUP	Start/Stop/Interrupt/ Local/Auto/ Single/QuickStop	05: Process controlling	Processcontrolling_backup	all
	Process Parameters	22: Object Parameters		all
	Auto	05: Process controlling	Processcontrolling_backup	006
	Standby mode	05: Process controlling	Processcontrolling_backup	004
	not empty	24: Interlocking Signals	HigherProcesscontrolling_backup	004
	Bypass Interlock	23: System Operations		004
C_ROUTE	Select/Deselect	05: Process controlling	Processcontrolling_backup	all
	Auto	05: Process controlling	Processcontrolling_backup	006
C_SELECT	Select/Deselect	05: Process controlling	Processcontrolling_backup	all
C_ANNUNC	Process Parameters	22: Object Parameters		all
	Simulation	24: Interlocking Signals	HigherProcesscontrolling_backup	all
C_PROFB	Process Parameters	22: Object Parameters		004
	Simulation	24: Interlocking Signals	HigherProcesscontrolling_backup	004
C_ANNUN8	Process Parameters	22: Object Parameters		all
	Simulation	24: Interlocking Signals	HigherProcesscontrolling_backup	all

Object type	Operation	Right/ default right	Instance specific settings possible via Attribute	Project Code
C_RUNNT	Reset	23: System Operations		all
C_COUNT	Reset	23: System Operations		all
C_SILOP	Start	05: Process controlling	Processcontrolling_backup	all
	Process Parameters	22: Object Parameters		all
CTRL_PID	Set point, Output	05: Process controlling	Processcontrolling_backup	all
	Process Parameters	21: Controller Parameters	HigherProcesscontrolling_backup	all

Info-Dialog	All registers without Service	27: Info Dialog Input		all
	Service Register	28: Info Dialog Service		all
	Save Button	05: Process controlling		all

Selection of area specific Authorization levels

After the definition of the additional authorization levels for CEMAT you must define the user groups and the users and in order to specify the authorization level for each user.

In the example we have defined the user groups “Administrator”, “Supervisor”, “Master Operator”, “Operator” and “Guest”. Within these groups different users may exist.

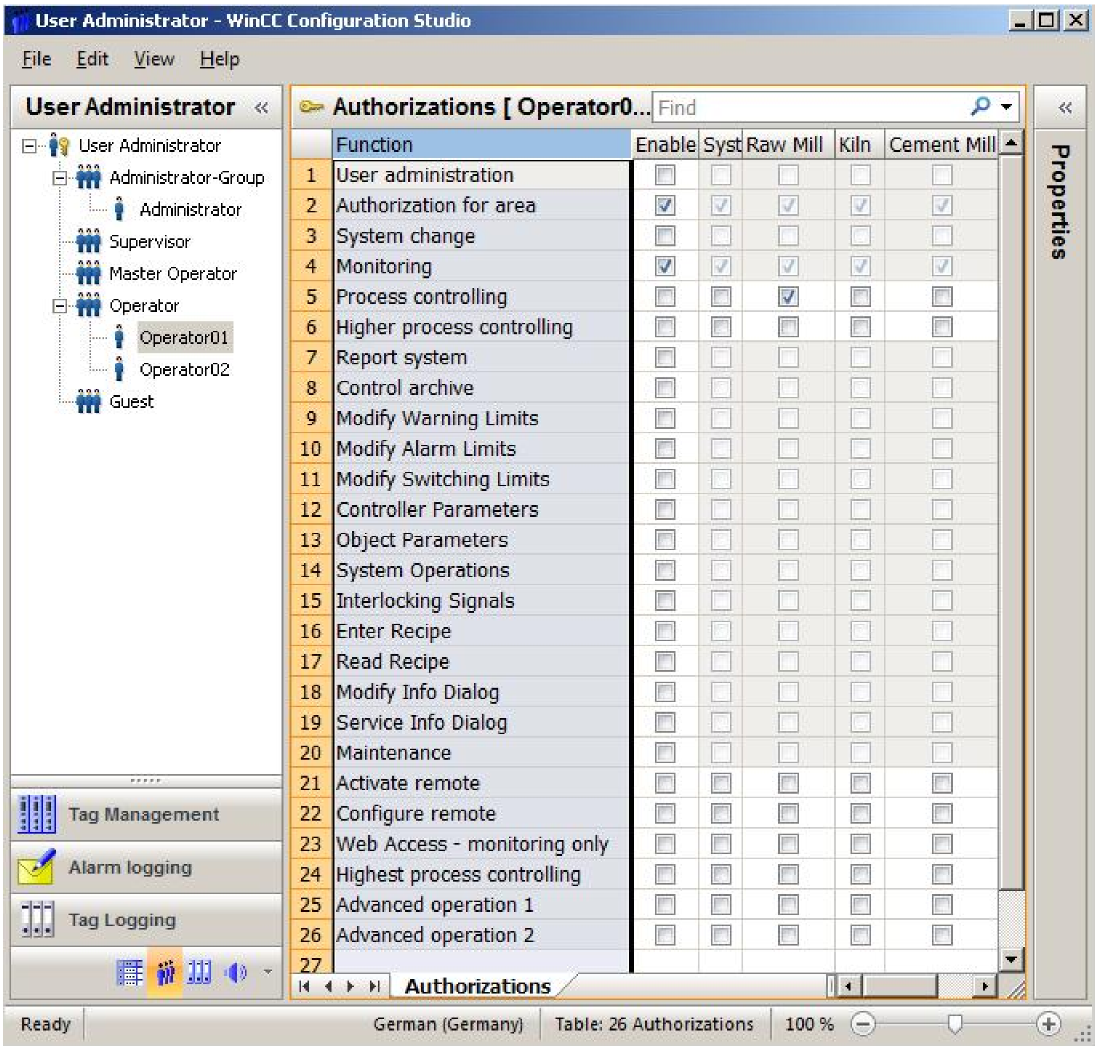
You can e.g. enable a complete area or only certain operations within an area.

With the level 2 “Authorization for area” it is defined which area is released for the user.

With the level 4 "Monitoring" is defined whether the area can be monitored.

With the level 5 „Process controlling“, it is defined whether a user is allowed to operate the area. He will then also get the messages from this area and the horn is activated.

Example:



The user Operator01 is allowed to:

open, watch → System, Raw Mill, Kiln and Cement Mill
operate → Raw Mill

Messages are only shown for areas which are enabled for Process controlling. Operator01 will get only the messages from the Raw Mill area and he can acknowledge only these messages.

Horn configuration

Via Horn configuration the alarm sounds can be defined per message class (and if needed dependent on Priority, Source, Area or Event).

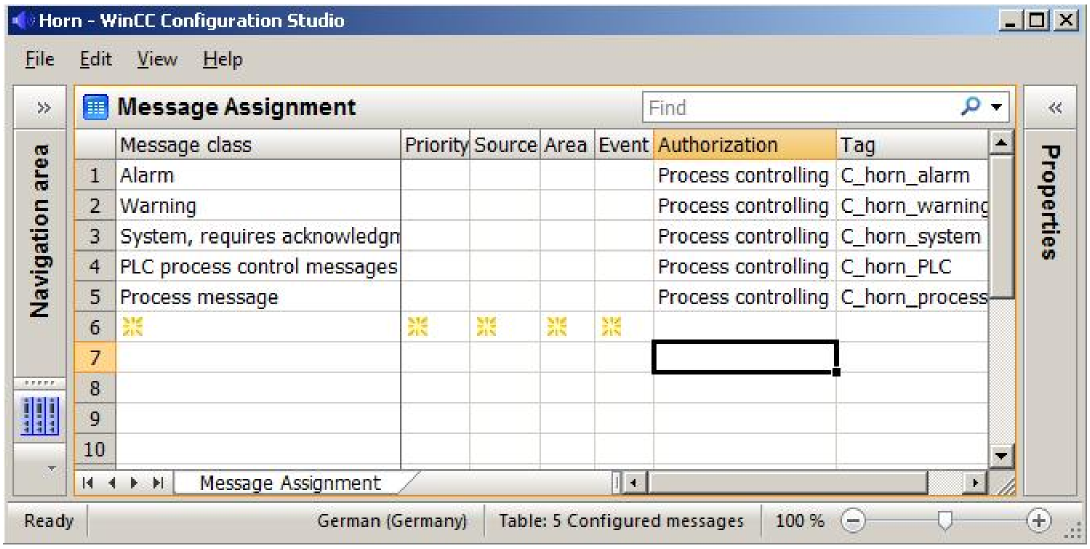
As the alarms are linked to authorization level “Process controlling” (users with this authorization will see the alarms), the alarm sound must be linked to the same authorization.

The Horn configuration is defined in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for Horn configuration

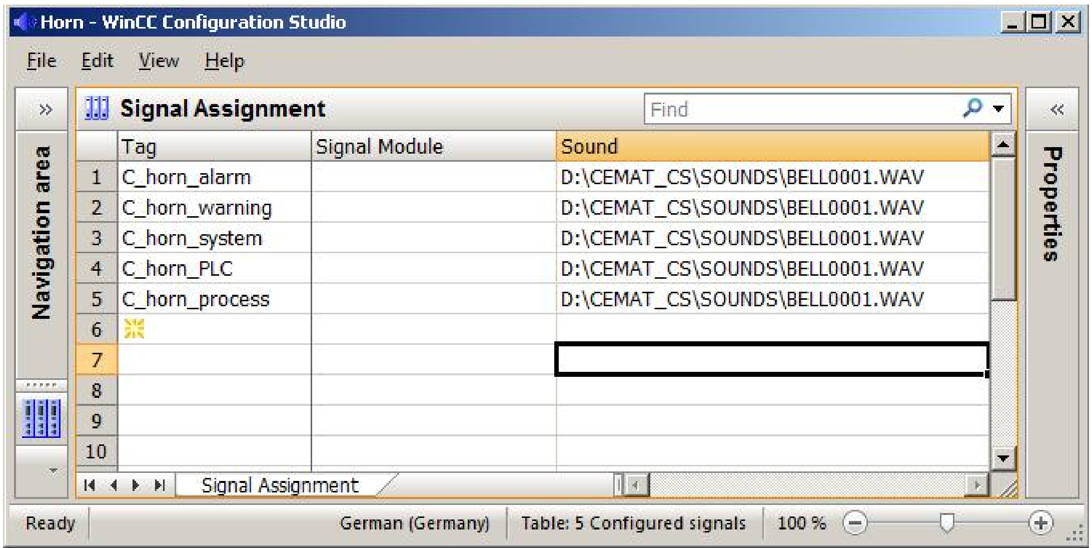
In order to import the message assignment, select tab message assignment and use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_HornMessgeAssignment.xlsx and press Button *Import.*

After the import you will see the following message assignment:



In order to import the Signal Assignment for the different sounds, select tab signal assignment and use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_HornSignalAssignment.xlsx and press Button *Import.*

After the import you will see the following signal assignment:



Update the WinCC Project with Cemat Functions

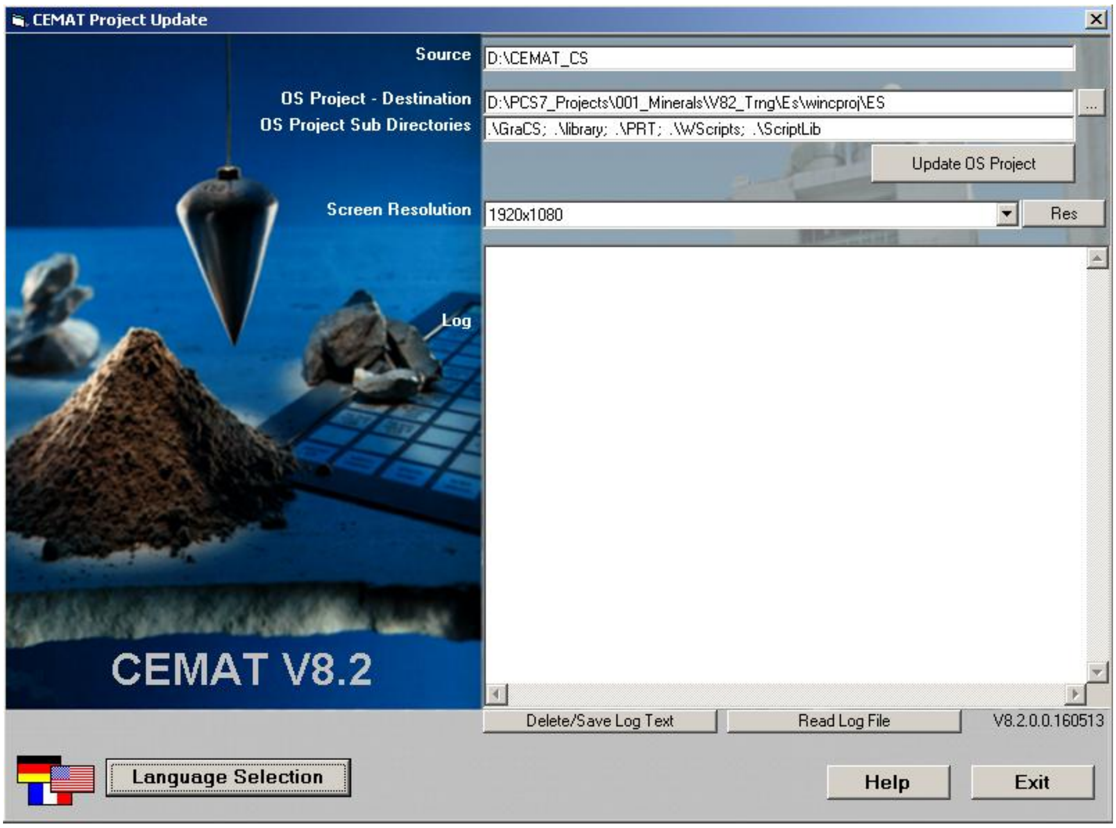
In the following step the WinCC Project gets updated with the Cemat functions. During the Cemat installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Libray, PRT and Wscipts. These files must now be copied to the equivalent folders of the OS Project.

Additionally, some PDLs are available in different Screen resolutions (1280x1024, 1600x1200, 1680x1050, 1920x1080 or 1920x1200) and must be exchanged.

GraCS-Files for which different Screen Resolutions exist:

<table>
<tr><th></th><th>1280x1024</th><th>1600x1200</th><th>1680x1050</th><th>1920x1080</th><th>1920x1200</th></tr>
<tr><td>@AlarmEmergency.pdl</td><td>x</td><td>x</td><td colspan="3">x</td></tr>
<tr><td>@AlarmEmergencyOP.pdl</td><td>x</td><td>x</td><td colspan="3">x</td></tr>
<tr><td>@AlarmOneLine.pdl</td><td>x</td><td>x</td><td>x</td><td colspan="2">x</td></tr>
<tr><td>@C_AlarmListing.pdl</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
<tr><td>@Overview1.pdl</td><td>x</td><td>x</td><td>x</td><td colspan="2">x</td></tr>
<tr><td>@TopAlarmNew.pdl</td><td>x</td><td>x</td><td colspan="3">x</td></tr>
</table>

In order to update the WinCC Project, please go to D:\CEMAT_CS and start 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project)

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

Select the desired **Screen Resolution** for your Project (1280x1024, 1600x1200, 1680x1050, 1920x1080 or 1920x1200). Via an option, during the OS Project Update the above listed Standard pictures are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

One by one the directories GraCS, Library, PRT and WScripts are copied into your OS Project and after this the PDLs are exchanged by the proper Screen Resolution. You have to acknowledge each step with OK and everything is entered into a LOG file.

During the update procedure the following GraCS files are overwritten by Cemat files.

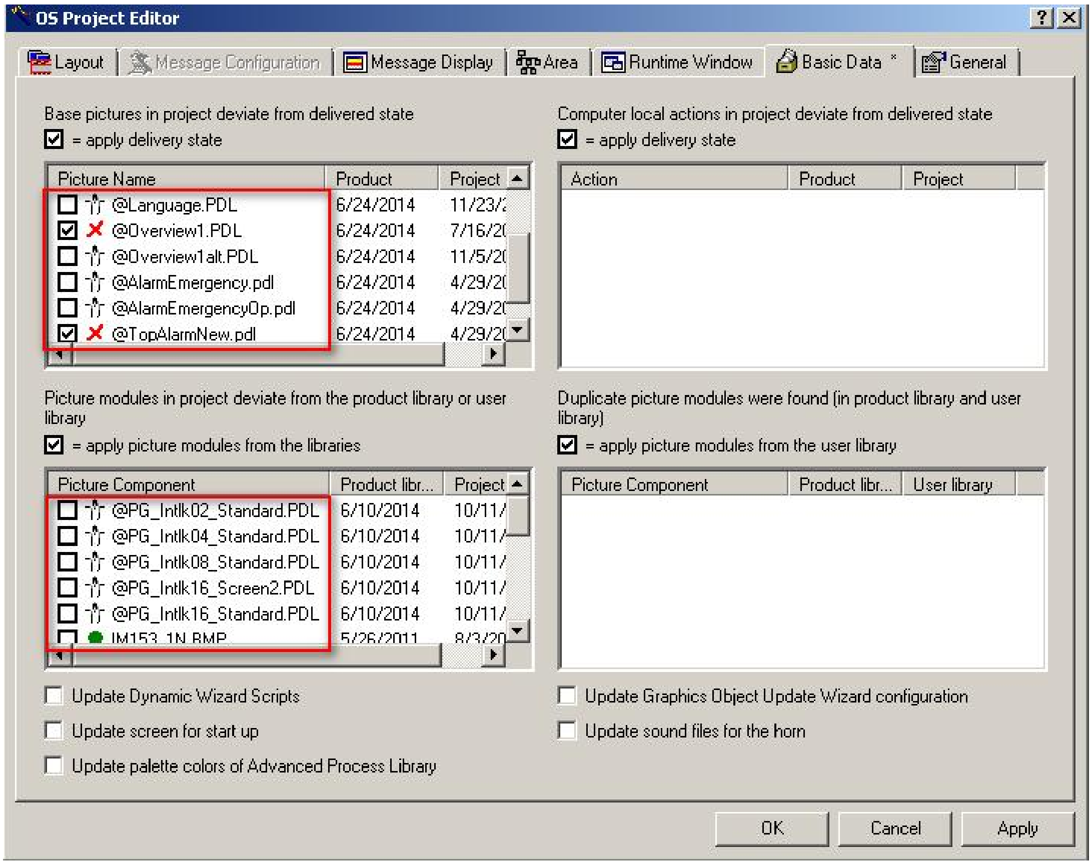
@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

After the update of the WinCC Project, proceed as follows:

- Open the WinCC Project.
- Open 'Global Script' selecting C-Editor and regenerate Header. (Menu Options -> Regenerate Header)

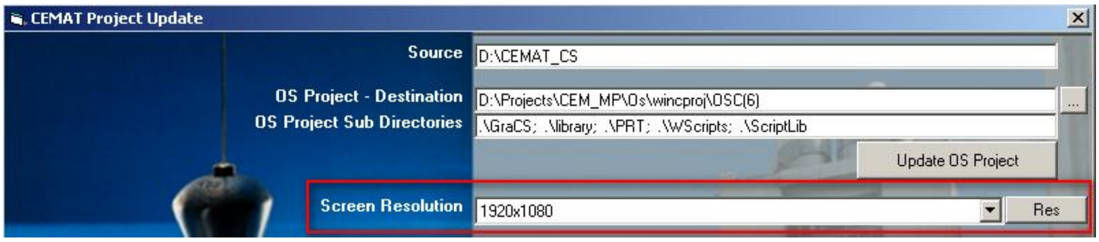
Tab “Basic Data”

Make sure that the system files which are modified by Cemat are <u>deselected</u> in the OS Project Editor under Basic Data. Otherwise, the next time when you run the OS Project Editor (e. g. in order to change the Area definition) the Cemat PDLs are overwritten by the PCS 7 system files.



Caution: @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must be replaced by the files from D:\CEMAT_CS\WinCC\GraCS again.

After running the OS Project Editor, use program 'CematProjectUpdate.exe' and press Button “Res” in order to update only the Screen resolution dependent Cemat pictures.



Via this function the following pictures are replaced by the Cemat Version again:

@AlarmEmergency.pdl,
@AlarmEmergencyOP.pdl,
@AlarmOneLine.pdl,
@C_AlarmListing.pdl,
@Overview1.pdl and
@TopAlarmNew.pdl

Generation of the Template Pictures

With the CEMAT project update various template pictures is copied into the GraCS folder of the OS Project. The template pictures contain default symbols for automatic generation of block icons from SIMATIC Manager.

The symbols of the new template pictures C_@PCS7Typicals_CemV8_000.pdl and C_@PCS 7 Typicals_CemV8_004.pdl contain the attributes for the new functions.

In each template picture you find examples for different representations of all CEMAT block types.

- The picture **C_@PCS7Typicals_CemV8_000.pdl** contains the block icons for Normal Standard and for the Project standards 006, 007 and 026 (not for 004 and 027).
  (For all non-mentioned Project standards the symbols for Normal standard can be used.)
  The block icons <u>can be used for all screen resolutions</u>.
- The picture **C_@PCS7Typicals_CemV8_004.pdl** contains the block icons for Holcim (Project Standard 004).
  The picture **C_@PCS7Typicals_CemV8_027.pdl** contains the block icons for Lafarge (Project Standard 027).
  In case of Project Standard 004 or 027, the block icons of template pictures **C_@PCS7Typicals_CemV8_000.pdl** <u>cannot be used.</u>
- For Holcim (Project Standard 004) an additional Template picture **C_Holcim_Symbols.pdf** is available. It contains picture elements (extended Status Display) for Motors, Dampers and Valves, which are displayed additionally to the block icon.

For the automatic generation of block icons PCS 7 V8.2 uses the picture **@PCS7Typicals.pdl** or the pictures with this name plus an extension (e. g. @PCS7Typical_1.pdl). The program searches in ascending sequence.
Before you can use the automatic generation of block icons you must create a template picture with the name @PCS7Typicals_extension.pdl which contains the symbols you want to use in the project.

You can use the CEMAT template pictures and rename it or you can copy parts of it into your template picture.

Caution: Only use the block icons which fit to your Project version. Using block icons from different project versions does not guarantee full functionality.

Further information regarding Template pictures you find in chapter OS Engineering.

OS Configuration (Server)

The WinCC Project (OS-Project) for the Server was already defined with the SIMATIC Manager or Network Configuration. The following settings have to be carried out in the OS-Project, which means you have to open WinCC. This can be done directly from the SIMATIC Manager through selection of the OS with right mouse button and *Open Object* of from the Start Menu under *Start -> Simatic -> WinCC -> WinCC Explorer*

Important: The following steps must be carried out from the beginning in the engineering language, which is used for the project. If you change the language afterwards the language dependent settings must be repeated.
The language settings in SIMATIC Manager are not consistent with the settings for WinCC!!

Project Properties

Adaptations for the Project itself can be carried out under Properties.

Folder “General” contains the Project Type.
A Server must be a “Multi-User Project”. This is the default setting and must not be changed.

In Folder “Update Cycles” the time values for the Update cycles are defined. The CEMAT Symbols generally use “User Cycle 1”. The default value of this is 2000ms. You have to adapt this value according to your requirement, e.g. 1000ms.

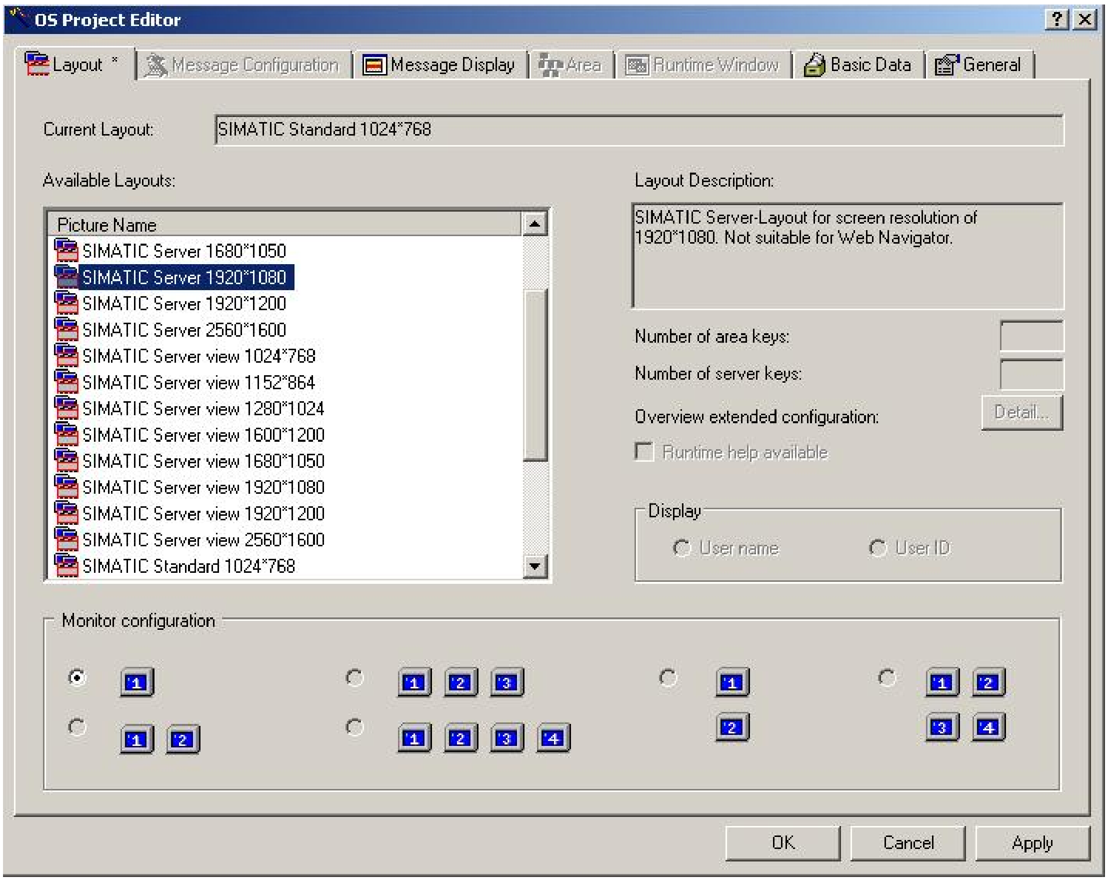
OS Project Editor

Settings for the user interface used by the plant operator for monitoring and controlling the plant during process operation In the OS Project Editor.

A detailed description to this item you find in the PCS 7 Configuration Manual Operator Station. The following pages describe only the settings relevant for CEMAT.

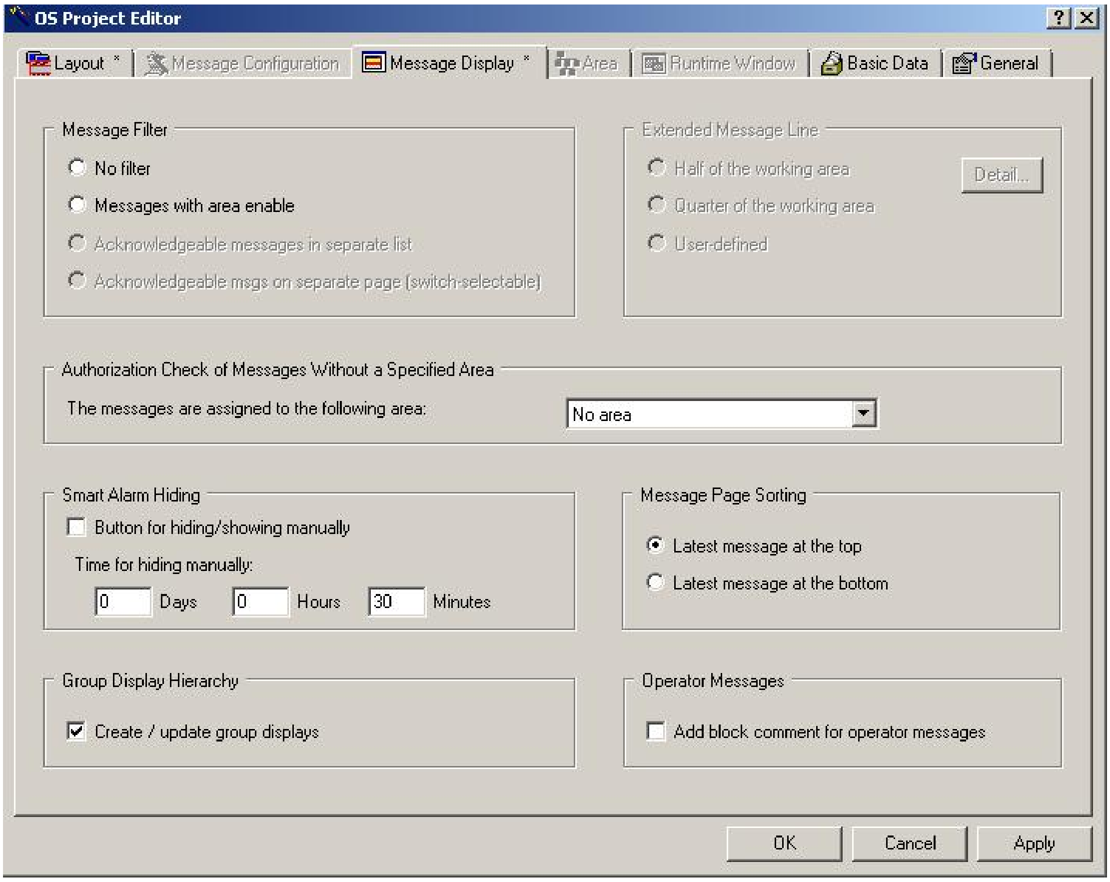
In Tab Layout choose the screen resolution according to your Monitor. In Cemat V8.2 the default resolution is 1920x1080, but the Cemat Standard pictures are also available for 1280x1024, 1600x1200 (not for Project Version 004), 1680x1050 and 1920x1200.

Example for the Layout of a Server:



The recommended screen resolution for CEMAT is 1920x1080. Sometimes smaller monitors are used for the servers which may not allow this screen resolution. Select the appropriate screen resolution for your monitor.

Message display for a Server:

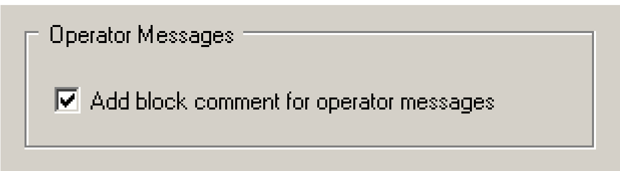


The group display should be created and updated automatically.

Note: The sorting of the message pages is only used for the Standard WinCC Alarm list, and not relevant for the Cemat Alarm list.
In the Cemat Alarm list the default sorting is "Latest message at the bottom". This setting can be changed under D:\Cemat_CS_Config in file C_Messages.cfg (see chapter Tips and Tricks).

Optional Setting:

Operator Messages created by HMI come without block comment (default setting). If you like to see the Block comment with this kind of messages you have to select the following Option:



Note: Different to the Operator messages created in the AS, in the Operator messages from HMI the block comment is part of the Event Text.

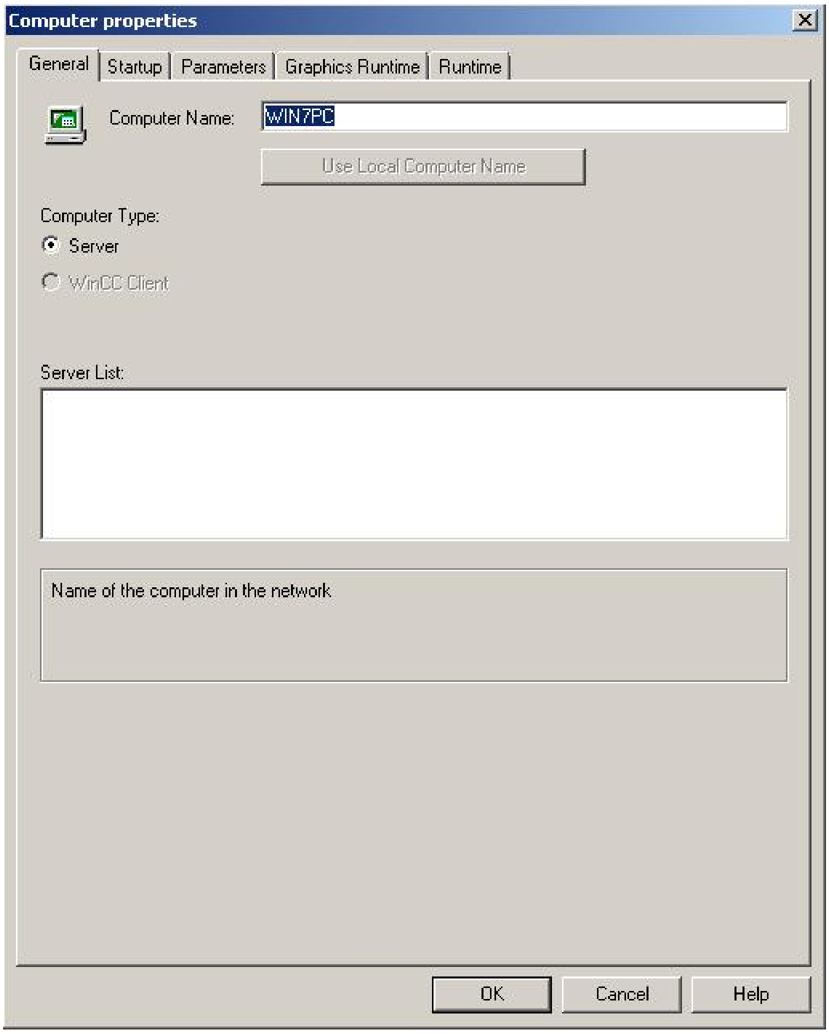
The rest of the settings in the OS Project Editor can remain as per default or can be adapted later.

With “OK” all settings are carried out.

Computer properties

The computer properties must be checked and adapted to the requirements of the plant. Most of the settings can probably not be entered at the beginning of the engineering and must be adapted later (sometimes after commissioning).

Folder General shows the Computer Name and the Computer Type

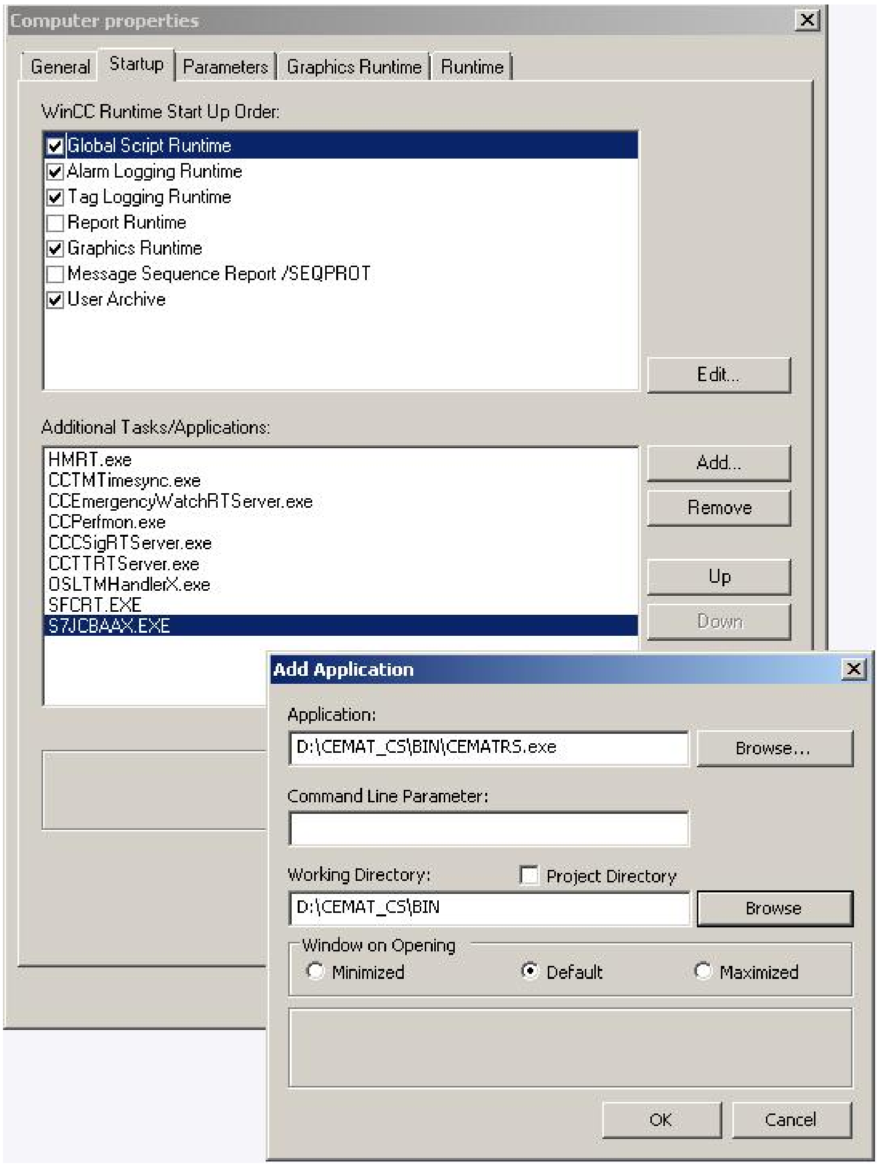


Caution: The Computer Name for all OS Projects is the name of the Engineering Station. During the download of the OS-Project to the individual PCs the name is automatically replaced by the Name of the PC Station.

Folder Startup shows the applications which are started after Activation of the Runtime System

Under Additional Tasks/Applications you have to add the CEMATRS.exe to the Startup list. The application is located in Directory D:\CEMAT_CS\BIN.

CEMATRS.exe should be called at the end.



Insert both, the Application <u>and</u> the working Directory and save with ok. Move the application to the last position.

Tag Management

Cemat Tags

Cemat needs a number of internal variables which must be added in the WinCC Configuration Studio. The variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

Open the WinCC Configuration Studio for the Tag Management

Use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

After a successful Import the following Groups and Variables have been added to the Tag Management:

Group “Cemat”

Name	**Data Type**	**Length**
C_ALM	Unsigned 16-bit value	2
C_Empty	Text tag 16bit character set	255
C_AlarmNavigation *)	Text tag 16bit character set	255

*) C_AlarmNavigation is only needed for Holcim Standard (004)

Group “Horn”:

Name	**Data Type**	**Length**
C_horn_alarm	Binary Tag	1
C_horn_PLC	Binary Tag	1
C_horn_process	Binary Tag	1
C_horn_system	Binary Tag	1
C_horn_warning	Binary Tag	1

Horn Reset Tags:

For the Horn Reset Tag it depends on whether you want to acknowledge the Horn on each Client independently or you want to acknowledge the Horn for different Clients at the same time.

The first solution may be useful if each WinCC Client is located in a different Room. If the Clients are located close to each other and if the same plant section can be operated and controlled by different Clients at the same time, the WinCC clients should act in unison.

More about Horn configuration you can read in the online-help of WinCC.

In case of WinCC clients with signaling devices acting in unison you have to create one or more reset tags on the Server. If the Horn is acknowledged on one client the variable on the server will be reset. This leads to an Acknowledgement for all Clients which use the same reset variable.

One or more reset variables are needed in group ”Horn“

Name	**Data Type**	**Length**	**update**
HornResetMC	Binary Tag	1	Project-wide

This variable must be selected in the Horn Configuration of the Client.

Redundancy Tags

CEMAT always needs the Redundancy tags, even if you may not have a redundant Server at the moment. The following Redundancy tags get automatically created when activating the Redundancy and will be found as internal variables in group „Redundancy“:

@RM_MASTER
@RM_MASTER_NAME
@RM_SERVER_NAME
@RM_OFFLINE_UA_NAME

⚠ **Note:** If no redundant Server is available (yet), the Redundancy must afterwards be deactivated again!

Alarm logging

All definitions for the alarm logging are carried out in the WinCC Configuration Studio.

Cemat projects require adaptations regarding color definitions and acknowledgement/flashing behavior. These adaptations are already prepared in Excel files Import_MessageBlocks.xlsx and Import_MessageBlocks.xlsx and can be imported from D:\CEMAT_CS\IMPORT.

Open the WinCC Configuration Studio and import the predefined configuration:

Message blocks

For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

The following values get imported:

Message blocks	Number of characters	Flash
Date	0	0
Time	0	0
Status	2	0
Comment	2	0
Info	2	0
Tagname	32	0
Area	24	0
Event	50	0
Tag Comment	80	0
FCL	80	0

Message configuration

For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

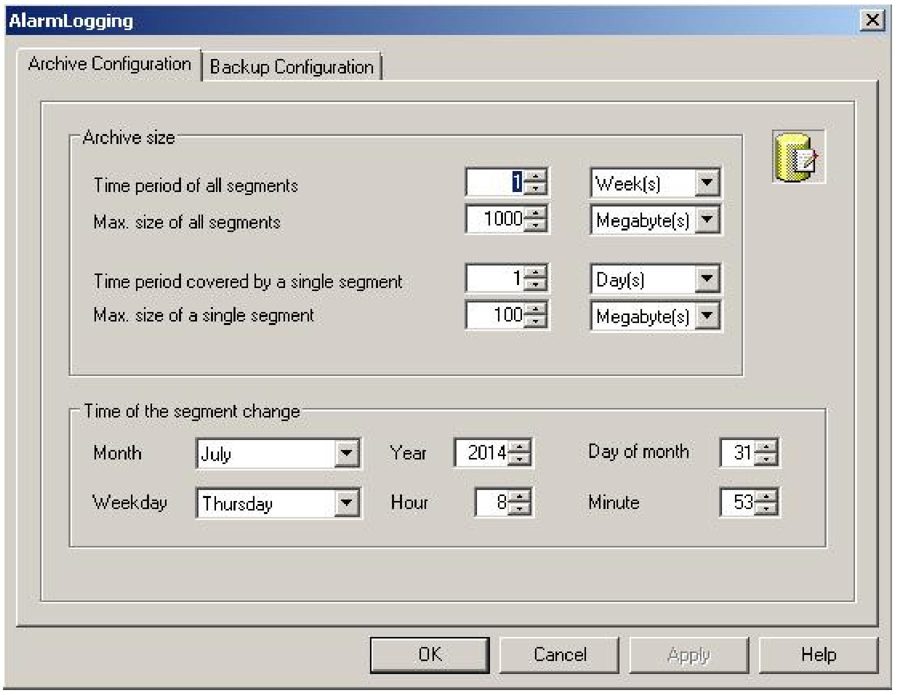
The following values get imported:

		Color definition		
Message class	Message types	come in	went out	acknowledged
Alarm	Alarm High	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Alarm	Alarm Low	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Warning	Warning High	F: 0; 0; 0 B: 255; 255; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 255; 128
Warning	Warning Low	F: 0; 0; 0 B: 255; 255; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 255; 128
PLC process control messages	Failure	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Process message	Process message.	F: 255; 255; 255 B: 255; 0; 0	F: 0; 0; 0 B: 0; 255; 0	F: 0; 0; 0 B: 255; 128; 128
Operator Message	Operator Message.	F: 0; 0; 0 B: 0; 255; 255		

F = Font color
B = Background color

Archive Configuration

Under Messages → Archive Configuration → Properties the Archive Configuration must be adapted according to the requirements of your project:



Tag logging

For archiving of the Process Values a Process Value Archive must be configured in the WinCC Configuration Studio.

Dependent on the number of Process Values to be archived, the Acquisition and Archiving cycle and the requirements regarding archiving period, the Process Value Archive may be located locally on the OS Station or on a separate PC Station dedicated for Archiving, the PCS 7 Process Historian.

For detailed information refer to PCS 7 Manuals for Process Historian and Information Server.

For small and medium-sized applications the Process Value Archive can be located on OS Station. For the calculation you have to consider the hard disk memory, the number of Segments and the Segment size.

The Archive Tags will be created during the Engineering, preferable in the CFC via a selection of the I/O for “Archiving” or “Long-term archiving”.

“Archiving”
The I/O will be archived on the OS or on an archive server

“Long-term archiving”
The values archived in the OS of on the archive server will be stored for long-term archiving on CD, DVD, MOD, tape, etc.

During OS compilation the Archive Tags are created in the Process Value Archive named “SystemArchive”. As acquisition cycle the minimum acquisition cycle is used. For values with different acquisition cycle it must be adapted later in the Tag Logging.

Note: PCS 7 can only create one common Process Value Archive. Multiple Process Value Archives (e. g. splitted by AREA) are not supported in the automatic generation and in general not necessary.
Archives and Archive Tags could be created manually in the Tag Logging, but it is not recommended, because it means much more Engineering effort.

Required settings for CEMAT

In Cemat it is possible to open the Archive curve of a measured value directly via block icon or the faceplate of the C_MEASUR block. In order to achieve this, an archive tag with correct naming must exist for output MV of the C_MEASUR block.

Example:

Measured value name: 123_BC1/J01
Archive tag name: 123_BC1/J01.MV

Caution: If the name of the archive tag differs from the above mentioned rules, the CEMAT Faceplate does not find the Archive data and the archive curve of the measure cannot be shown.
Make sure that the Tag name of the measure does not contain any illegal characters for archive variables!

More information regarding Process Value Archives you will find in chapters AS Engineering and OS Engineering.
In case of Archive Splitting or manual creation of Process Value Archives please refer to chapter 14_Tips_Tricks – Process Value Archive.

User Archives

WinCC User Archives are used for the following CEMAT Functions:

Cemat Curve groups

User Archive **C_CURVE** is used to save the Cemat curve groups. The data is created/ updated in the Runtime system.

Note: In order to allow saving of a new curve group, at least one (dummy) line must exist and therefore be created during the setup of the project.

Cemat Object list

User archive **C_DriveList** is used to save the object list (list of objects associated to a route or group and list of objects associated to a drive). It gets filled when you open the Object list in the Runtime system and press the save button.

During the setup of a project this User Archive may remain empty.

Cemat Object Information

User archive **C_INFO** contains the object information for Display in the Information Dialog and in the Maintenance Dialog of the Cemat Objects.

For each Cemat Object one data set must be provided, containing all the information to be displayed. The data sets can be created with Engineering tool C_INFO.xlsx (see Engineering Manual, chapter 09_Engineering_Tools.doc) after the completion of the software engineering. In the Runtime System the information can be modified and/or completed.

Note: In order to allow adding of new object information, at least one (dummy) line must exist and therefore be created during the setup of the project.

Cemat Polygon Function

User Archive **C_POLY3** contains the parameters for the Polygon function and is only required if the Function Polygon is used. The information is created/ updated in the Runtime system.

Note: In order to allow saving of a new Polygon parameters, at least one (dummy) line must exist and therefore be created during the setup of the project.

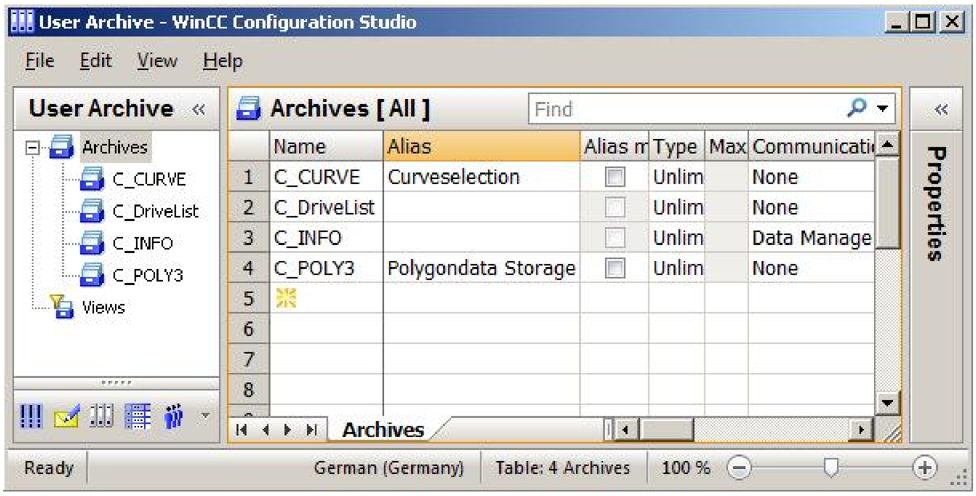
How to create the Cemat user archives

The Cemat User Archives are configured in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for the User Archive

In order to import the structure use function *Edit* → *Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_UserArchives.xlsx and press Button *Import.*

After the import the following Archives exist:

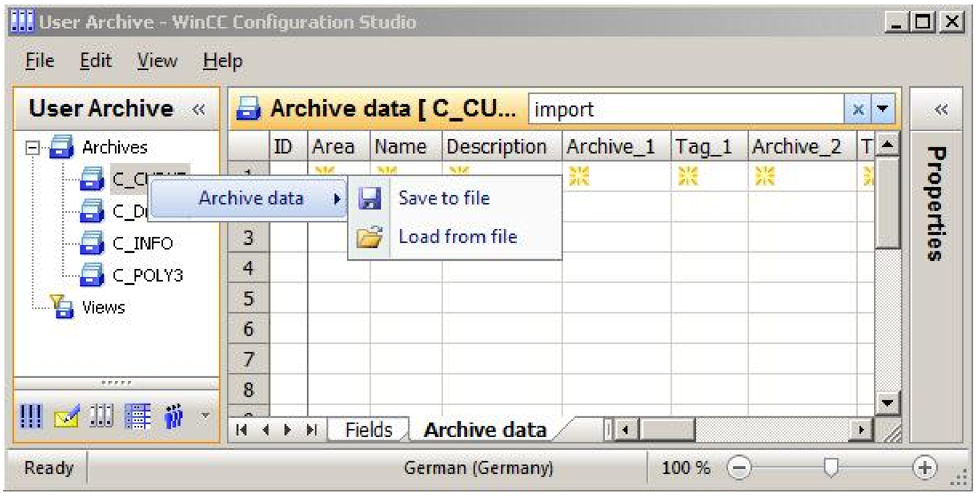


How to create the user archive data

For the user archives C_CURVE, C_INFO and C_POLY3 at least one line of data must be created during the setup of the OS-Project.

Therefore directory D:\CEMAT_CS\IMPORT contains csv-files with dummy lines for C_CURVE, C_INFO and C_POLY3 which can be imported. To import the data go to tab Archive data and proceed as follows:

With right mouse click on C_CURVE → *Archive data* → *Load from file*, select from D:\CEMAT_CS\IMPORT file C_CURVE.csv , and press Button *Import.*



With right mouse click on C_INFO → *Archive data* → *Load from file*, select from D:\CEMAT_CS\IMPORT file C_INFO.csv , and press Button *Import.*

With right mouse click on C_POLY3 → *Archive data* → *Load from file*, select from D:\CEMAT_CS\IMPORT file C_POLY3.csv , and press Button *Import.*

The user Archive C_DriveList does not need to contain data.

Add Authorization levels

Each Operation in the HMI is linked to an Authorization level which the person must have to carry out the function. Beside the Authorization levels of PCS 7 the operations in Cemat require additional Authorizations (18 – 29) for specific functions.

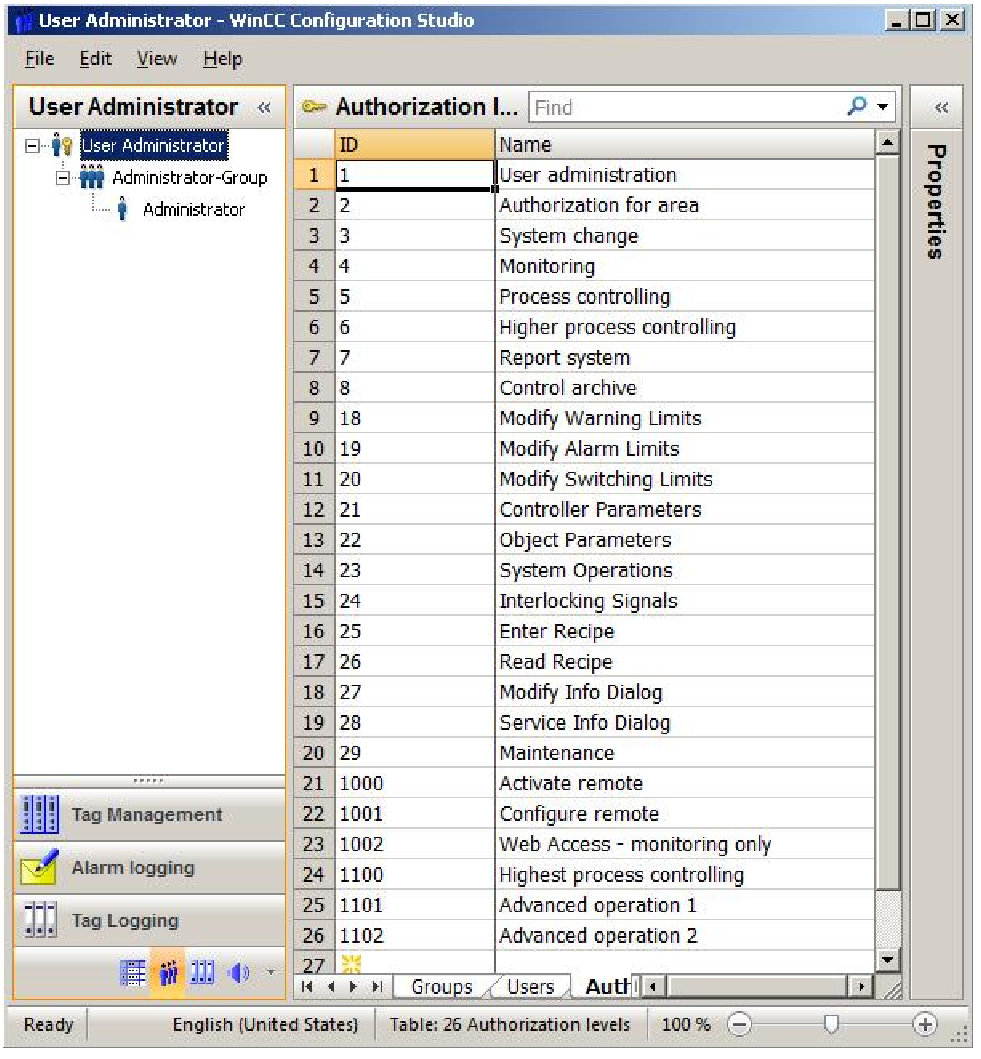
Each User group/User may get the permission for certain authorization levels in certain areas.

The Authorization levels, User groups and Users are defined in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for User Administrator

In order to import the additional Authorization levels 18 - 29 use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_AuthorizationLevels.xlsx and press Button *Import.*

After the import you will have the following Authorization levels:



	ID	Name
1	1	User administration
2	2	Authorization for area
3	3	System change
4	4	Monitoring
5	5	Process controlling
6	6	Higher process controlling
7	7	Report system
8	8	Control archive
9	18	Modify Warning Limits
10	19	Modify Alarm Limits
11	20	Modify Switching Limits
12	21	Controller Parameters
13	22	Object Parameters
14	23	System Operations
15	24	Interlocking Signals
16	25	Enter Recipe
17	26	Read Recipe
18	27	Modify Info Dialog
19	28	Service Info Dialog
20	29	Maintenance
21	1000	Activate remote
22	1001	Configure remote
23	1002	Web Access - monitoring only
24	1100	Highest process controlling
25	1101	Advanced operation 1
26	1102	Advanced operation 2
27		

If operations shall be permitted to restricted personal only, additional user rights can be defined.

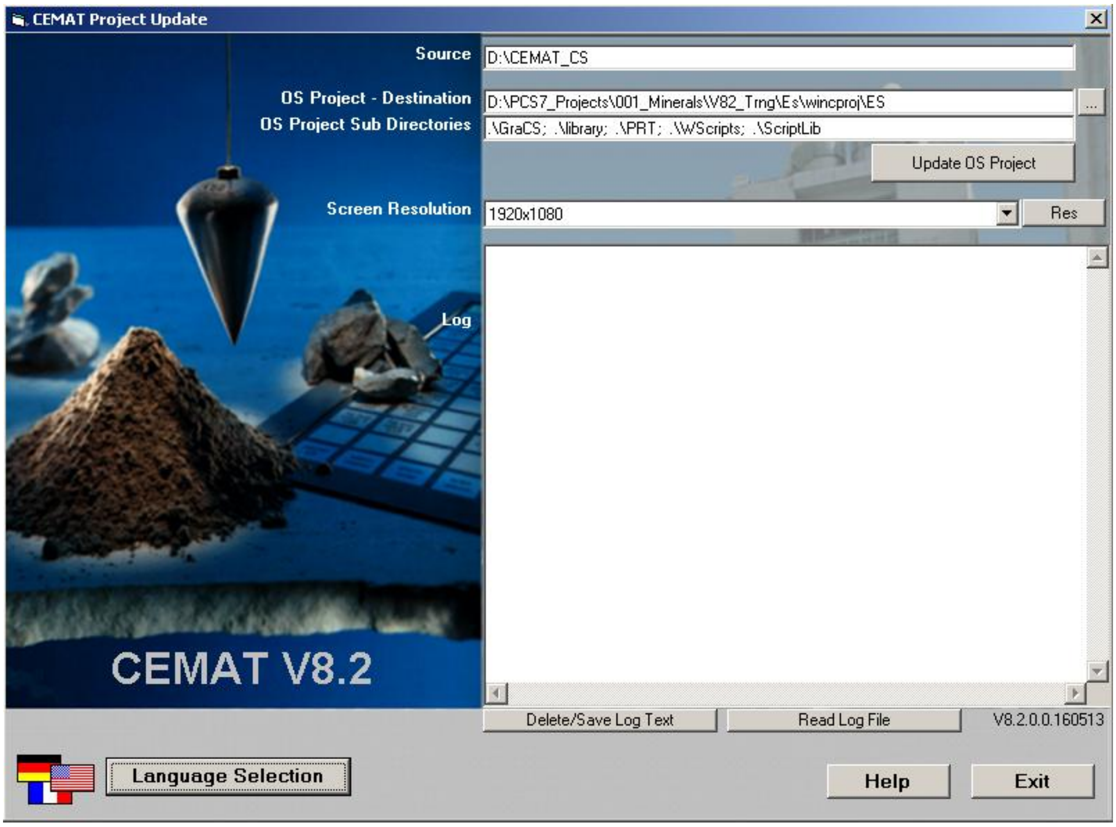
Under 'OS Configuration (Single-User System)' you find a list of the Cemat Operations and the required Authorization level.

Authorization levels "25 - Enter Recipe" and "26 - Read Recipe" are currently not used.

Update the WinCC Project with Cemat Functions

In the following step the WinCC Project gets updated with the Cemat functions. During the Cemat installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Libray, PRT and WScipts. These files must now be copied to the equivalent folders of the OS Project.

In order to update the WinCC Project, please go to D:\CEMAT_CS and start 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project)

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1600x1200, 1680x1050, 1920x1080 or 1920x1200) is only important for OS Projects with Operation Interface. Via an option, during the OS Project Update the Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl and @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

One by one the directories GraCS, Library, PRT and WScripts are copied into your OS Project and after this the PDLs are exchanged by the proper Screen Resolution. You have to acknowledge each step with OK and everything is entered into a LOG file.

During the update procedure the following GraCS files are overwritten by Cemat files.

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

After the Update the WinCC Project, proceed as follows:

- Open the WinCC Project
- Run the OS Project Editor again in order to overwrite the Standard Pictures with the original pictures from PCS 7 .
- Open 'Global Script' selecting C-Editor and regenerate Header. (Menu Options -> Regenerate Header)

Tab “Basic Data”

Normally the Server has no User interface. In this case the Replacement of @Overview1.pdl by the Cemat Version is not correct. Therefore, run the OS Project Editor again in order to replace the files under “Basic data” by the PCS 7 Version.

If your Server has a User interface, make sure that the Cemat PDLs are used. Cemat Files must be deselected in tab “Basic Data” and you have to run the CEMAT Project Update every time after running the OS Project Editor (see description for a Single Station).

Generation of the Template Pictures

With the CEMAT project update various template pictures is copied into the GraCS folder of the OS Project. The template pictures contain default symbols for automatic generation of block icons from SIMATIC Manager.

The symbols of the new template pictures C_@PCS7Typicals_CemV8_000.pdl and C_@PCS7Typicals_CemV8_004.pdl are slightly bigger than the symbols of older Cemat Versions and contain the attributes for the new functions. In general they can be used for all screen resolutions.

The old template picture for screen resolution 1280x1024 (without new functions) and the corresponding bitmaps you find on the Cemat installation DVD in directory Cemat_AddOn\CEMAT_1280\CEMAT_CS\WinCC\GraCS. You can still use these but in this case you have to copy it manually into the GraCS directory of your OS project.

In each template picture you find examples for different representations of all CEMAT block types.

- The picture **C_@PCS7Typicals_CemV8_000.pdl** contains the block icons for Normal Standard and for the Project standards 006, 007 and 026 (not for 004 and 027).
  (For all non-mentioned Project standards the symbols for Normal standard can be used.)
  The block icons can be used for all screen resolutions.
- The picture **C_@PCS7Typicals_CemV8_004.pdl** contains the block icons for Holcim (Project Standard 004).
  The picture **C_@PCS7Typicals_CemV8_027.pdl** contains the block icons for Lafarge (Project Standard 027).
  In case of Project Standard 004 or 027, the block icons of template pictures **C_@PCS7Typicals_CemV8_000.pdl** cannot be used.
- For Holcim (Project Standard 004) an additional Template picture **C_Holcim_Symbols.pdf** is available. It contains picture elements (extended Status Display) for Motors, Dampers and Valves, which are displayed additionally to the block icon.

For the automatic generation of block icons PCS 7 V8.2 uses the picture **@PCS7Typicals.pdl** or the pictures with this name plus an extension (e. g. @PCS7Typical_1.pdl). The program searches in ascending sequence.
Before you can use the automatic generation of block icons you must create a template picture with the name @PCS7Typicals_extension.pdl which contains the symbols you want to use in the project.

You can use the CEMAT template pictures and rename it or you can copy parts of it into your template picture.

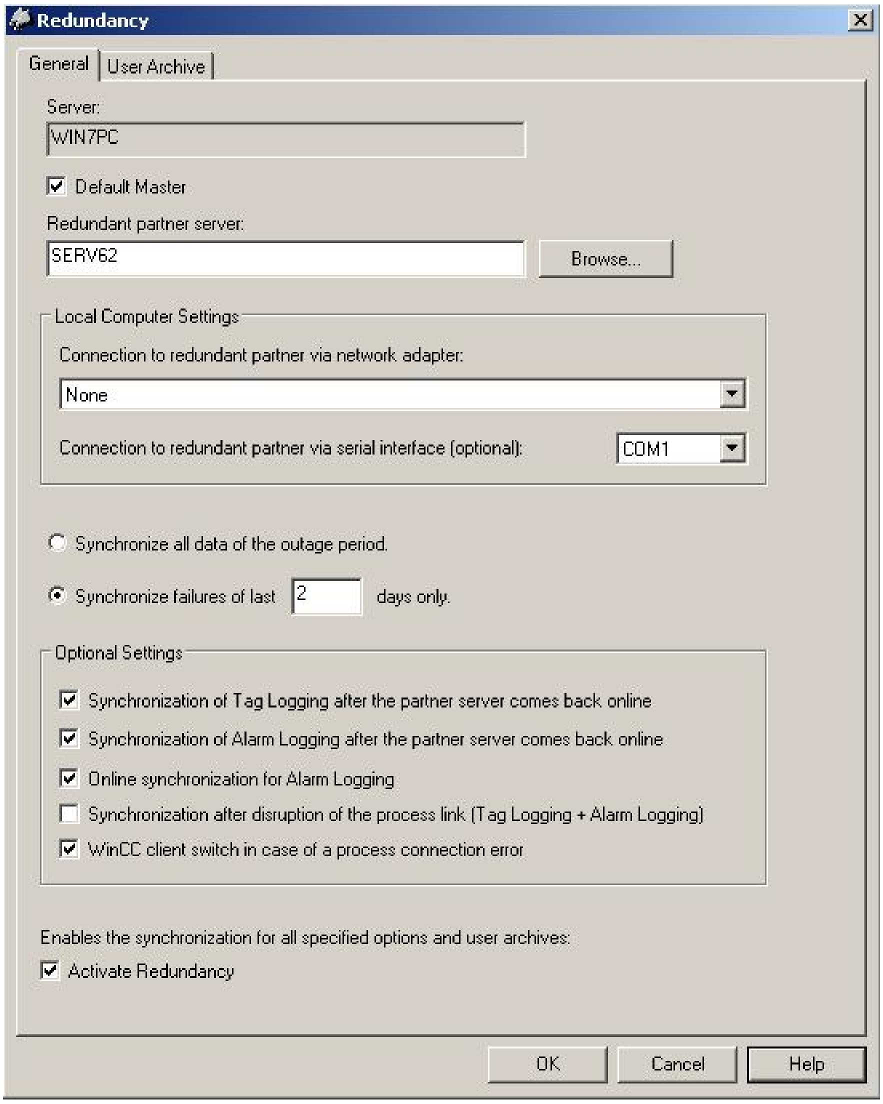
Caution: Only use the block icons which fit to your Project version. Using block icons from different project versions does not guarantee full functionality.

Further information regarding Template pictures you find in chapter OS Engineering.

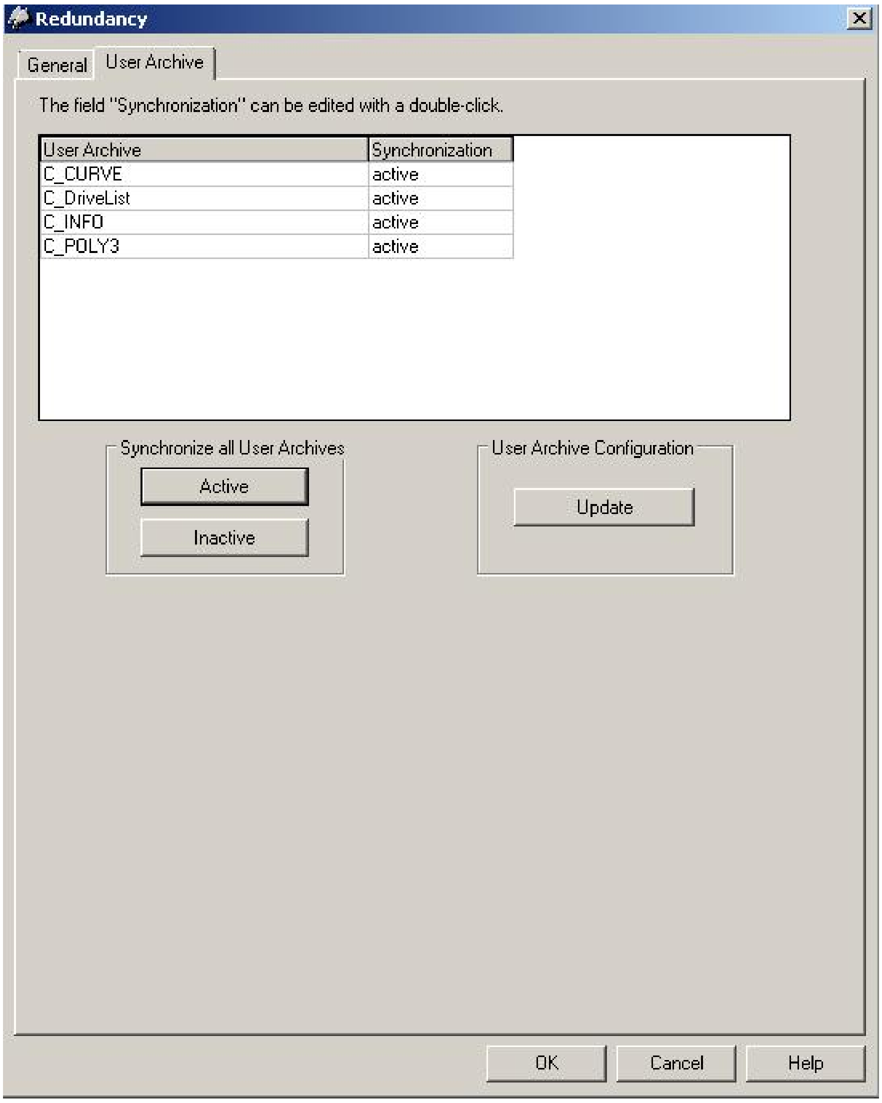
Redundancy Settings in the Server-Project

After the Server-Project and the Standby-Server-Project are configured, in the Server-Project the corresponding Standby-Server-Project must be selected and the redundancy settings have to be performed.

1. Select the Server-Project in the SIMATIC Manager and select in the Properties under Target OS and Standby-OS the corresponding Standby-OS.
2. Open the Server-Project and carry out the redundancy settings:
   - Activate Redundancy
   - Select the Redundant Partner Server.
   - Select the options according to your requirement.



3. Activate the redundancy also for the user archives C_INFO, C_DriveList, C_POLY3 and C_CURVE.



OS Configuration (Standby-Server)

Standby-Server-Project

The Standby-Server-Project doesn't contain any data. There is only a reference to the Master- Server-Project. No additional Settings required.

OS Configuration (Client)

The following settings have to be performed in the OS-Project of each Client.

Important: The following steps must be performed from the beginning in the engineering language, which is used for the project. If you change the language afterwards the language dependent setting have to be performed again.
The language settings in SIMATIC Manager are not consistent with the settings for WinCC!!

Project Properties

Adaptations for the Project itself can be carried out under Properties.

Folder “General” contains the Project Type.
An OS Client is of type “Client Project”. This is the default setting and must not be changed.

In Folder “Update Cycles” the time values for the Update cycles are defined. The CEMAT Symbols generally use “User Cycle 1”. The default value of this is 2000ms. You have to adapt this value according to your requirement, e.g. 1000ms.

OS Project Editor

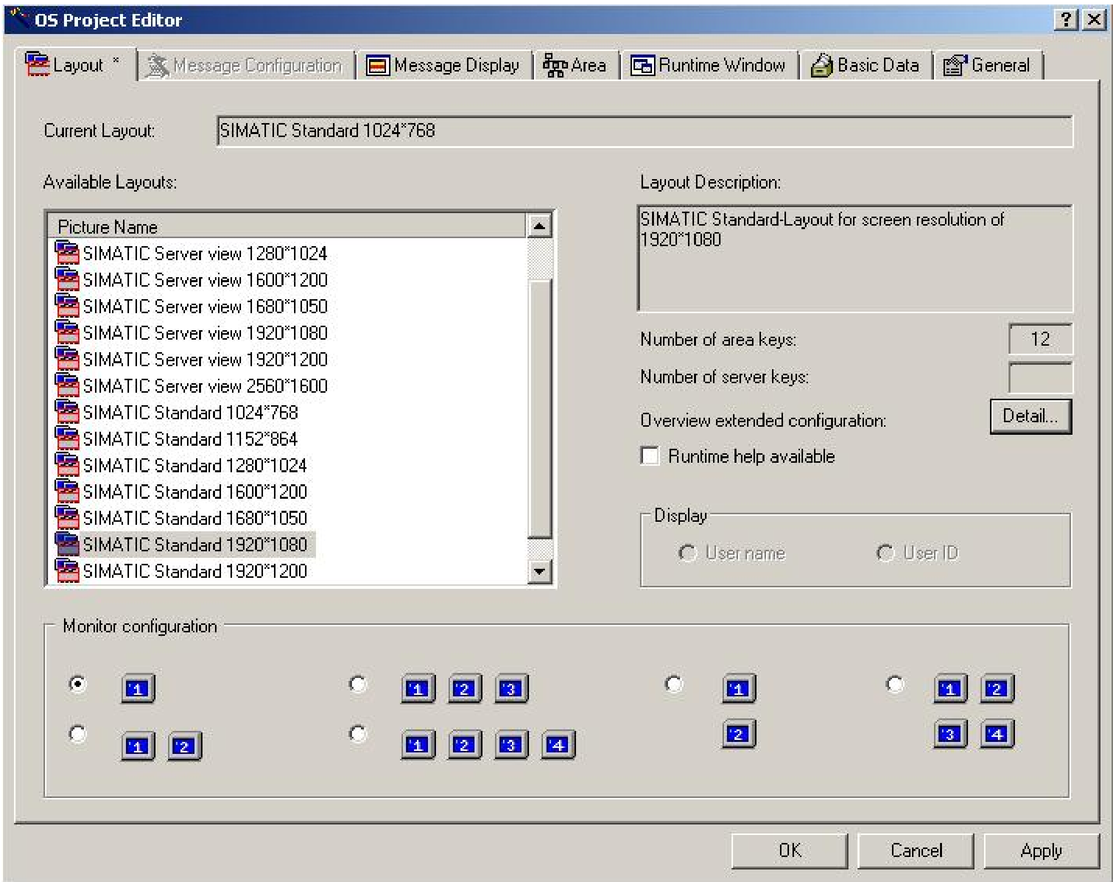
Settings for the user interface used by the plant operator for monitoring and controlling the plant during process operation In the OS Project Editor.

A detailed description to this item you find in the PCS 7 Configuration Manual Operator Station. The following pages describe only the settings relevant for CEMAT.

In Tab Layout choose the screen resolution according to your Monitor. In Cemat V8.2 the default resolution is 1920x1080, but the Cemat Standard pictures are also available for 1280x1024, 1600x1200 (not for Project Version 004), 1680x1050 and 1920x1200.

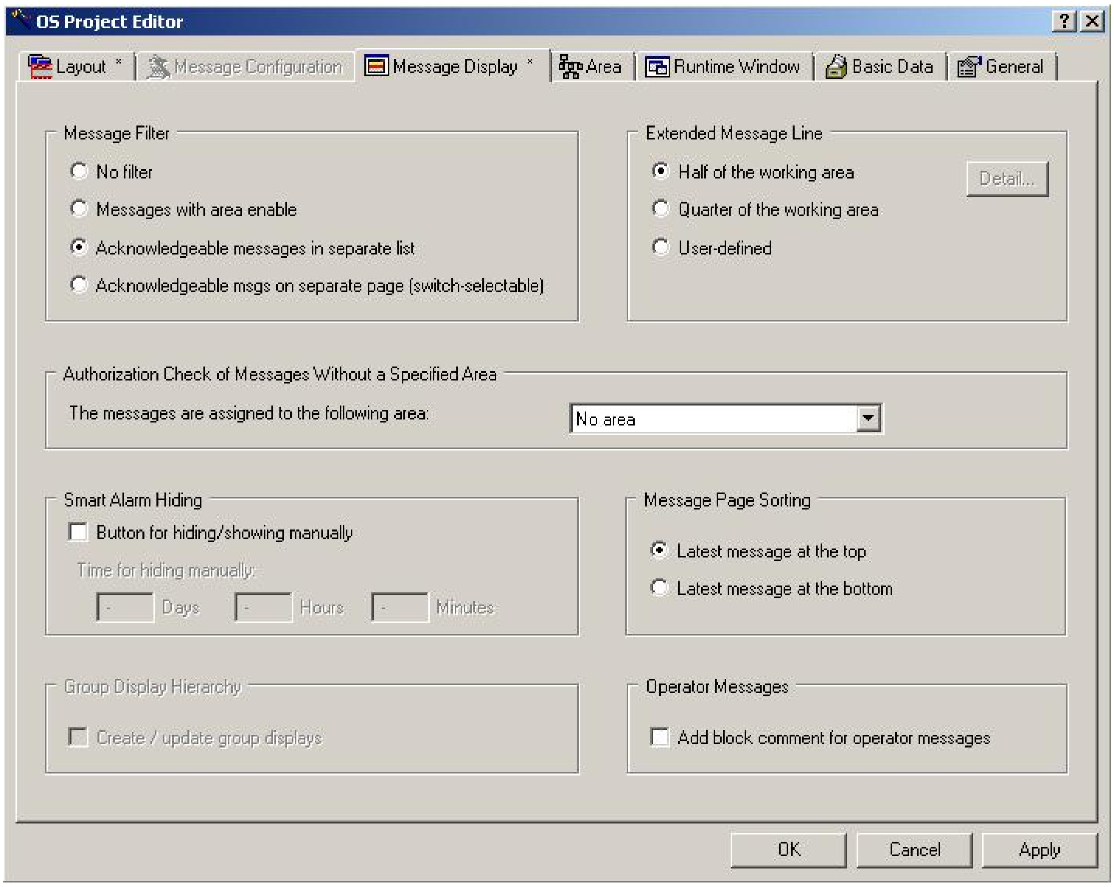
Also select the Number of horizontal and vertical area keys and the number of Servers under Detail (Picture Tree) and the Monitor Configuration of your PC.

Example for the Layout of an OS Client:



Note: If you select a different screen resolution (1280x1024, 1600x1200, 1680x1050 or 1920x1200), the standard pictures @AlarmOneLine.pdl, @TopAlarmNew.pdl, @C_AlarmListing_Cemat.pdl, @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl and @Overview1.pdl have to be exchanged.

Message display for a Client Station:



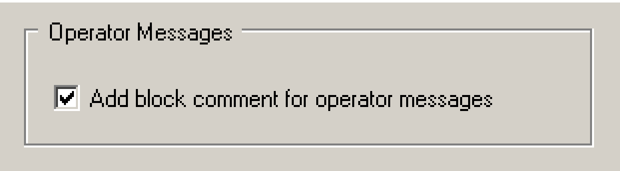
The Message filter has to be set to “Acknowledgeable messages in separate list”.

The group display should be created and updated automatically.

Note: The sorting of the message pages is only used for the Standard WinCC Alarm list, and not relevant for the Cemat Alarm list.
In the Cemat Alarm list the default sorting is "Latest message at the bottom". This setting can be changed under D:\Cemat_CS_Config in file C_Messages.cfg (see chapter Tips and Tricks).

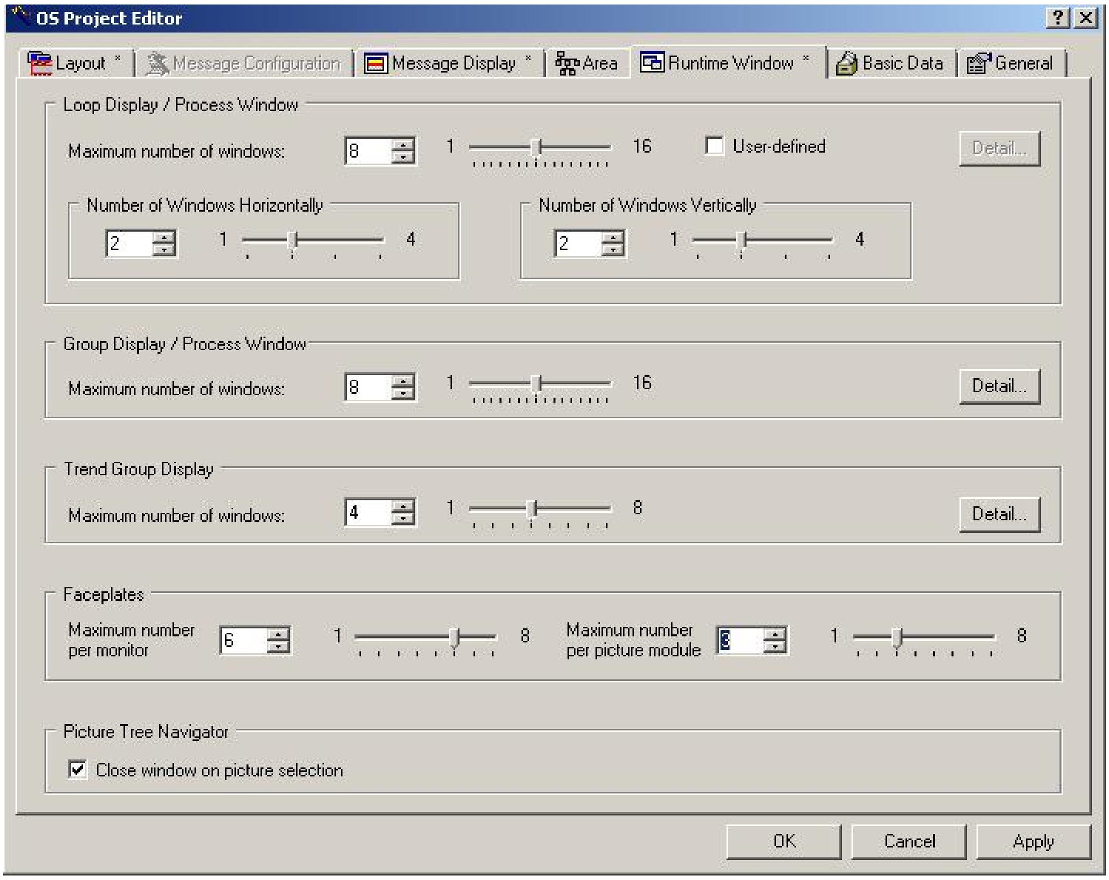
Optional Setting:

Operator Messages created by HMI come without block comment (default setting). If you like to see the Block comment with this kind of messages you have to select the following Option:



Note: Different to the Operator messages created in the AS, in the Operator messages from HMI the block comment is part of the Event Text.

Runtime Window for an OS Client:



Define the maximum number of windows that can be opened when faceplates and curves are called as well as the maximum number of faceplates that can be opened on a monitor.

Set the maximum number of picture modules per monitor to 6 and the maximum number per picture module to 3.

The rest of the settings in the OS Project Editor can remain as per default or can be adapted later.

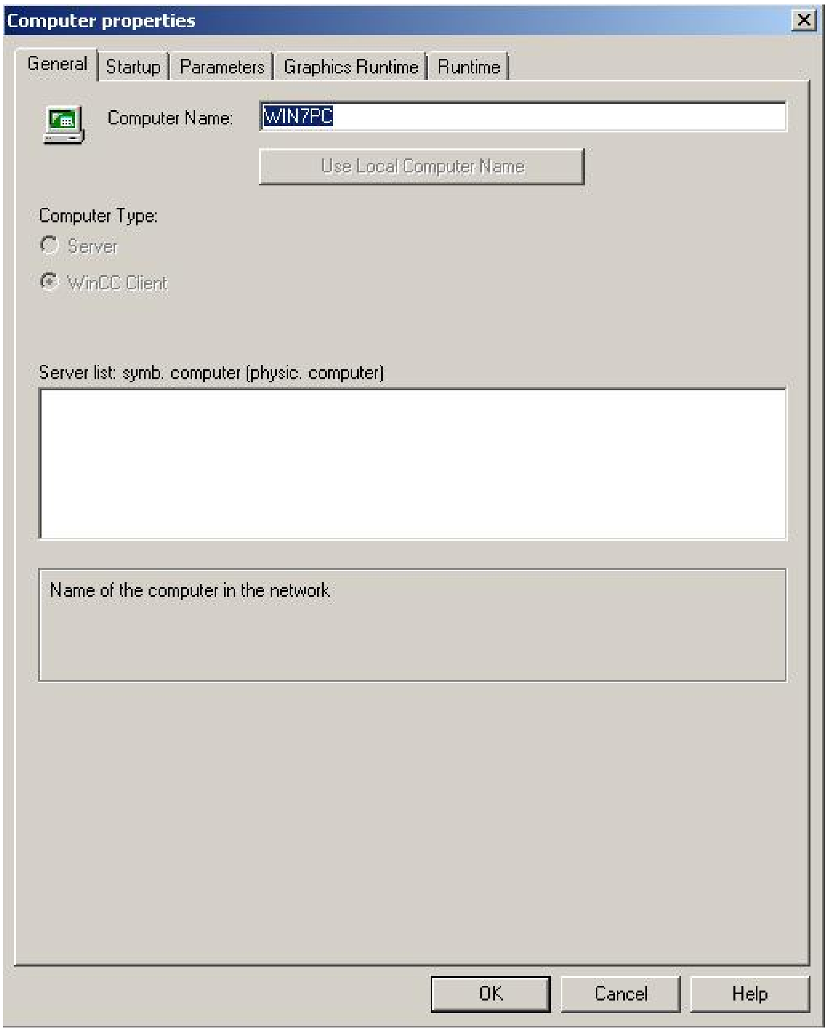
With “OK” all settings are carried out.

Caution: If for any reason you have to run the OS-Project Editor again, make sure that all files mentioned under CEMAT Specific Preparations are deselected in Folder “Basic Data”. Otherwise the CEMAT Settings are overwritten with the delivery state of PCS 7 .
→ See chapter CEMAT Specific Preparations!

Computer properties

The computer properties must be checked and adapted to the requirements of the plant. Most of the settings can probably not be entered at the beginning of the engineering and must be adapted later (sometimes after commissioning).

Folder General shows the Computer Name and the Computer Type



Caution: The Computer Name for all OS Projects is the name of the Engineering Station. During the download of the OS-Project to the individual PCs the name is automatically replaced by the Name of the PC Station.

The additional application CEMATRS.exe must not be added for a Cemat Client (not required).

Tag Management

Cemat Tags

Cemat needs a number of internal variables which must be added in the WinCC Configuration Studio. The variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

Open the WinCC Configuration Studio for the Tag Management

Use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

After a successful Import the following Groups and Variables have been added to the Tag Management:

Group “Cemat”

Name	Data Type	Length
C_ALM	Unsigned 16-bit value	2
C_Empty	Text tag 16bit character set	255
C_AlarmNavigation *)	Text tag 16bit character set	255

*) C_AlarmNavigation is only needed for Holcim Standard (004)

Group “Horn”:

Name	Data Type	Length
C_horn_alarm	Binary Tag	1
C_horn_PLC	Binary Tag	1
C_horn_process	Binary Tag	1
C_horn_system	Binary Tag	1
C_horn_warning	Binary Tag	1

Add User Authorization levels

Each Operation in the HMI is linked to an Authorization level which the person must have to carry out the function. Beside the Authorization levels of PCS 7 the operations in Cemat require additional Authorizations (18 – 29) for specific functions.

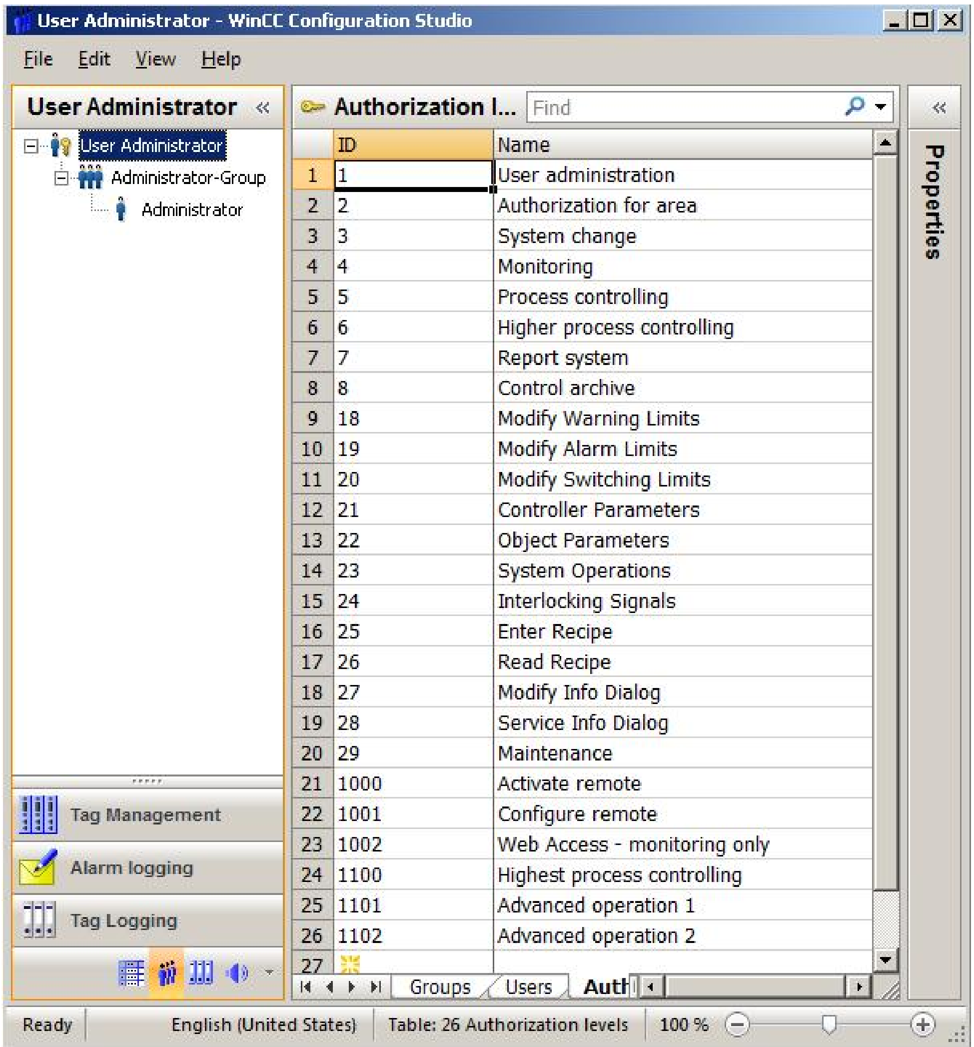
Each User group/User may get the permission for certain authorization levels in certain areas.

The Authorization levels, User groups and Users are defined in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for User Administrator

In order to import the additional Authorization levels 18 - 29 use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_AuthorizationLevels.xlsx and press Button *Import.*

After the import you will have the following Authorization levels:



If operations shall be permitted to restricted personal only, additional user rights can be defined.

Authorization levels “25 - Enter Recipe” and “26 - Read Recipe” are currently not used.

Update the WinCC Project with Cemat Functions

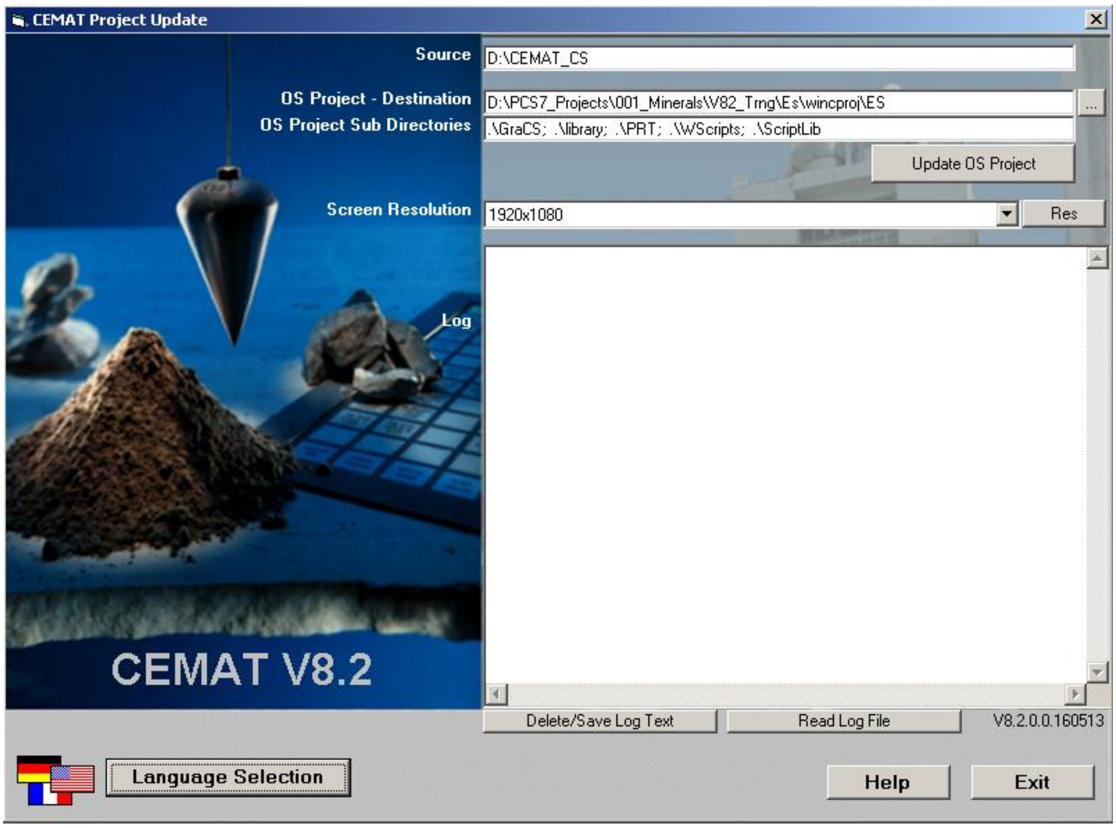
In the following step the WinCC Project gets updated with the Cemat functions. During the Cemat installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Libray, PRT and WScipts. These files must now be copied to the equivalent folders of the OS Project.

Additionally, some PDLs are available in different Screen resolutions (1280x1024, 1600x1200, 1680x1050, 1920x1080 or 1920x1200) and must be exchanged.

GraCS-Files for which different Screen Resolutions exist:

	1280x1024	1600x1200	1680x1050	**1920x1080**	1920x1200
@AlarmEmergency.pdl	x	x		**x**	
@AlarmEmergencyOP.pdl	x	x		**x**	
@AlarmOneLine.pdl	x	x	x	**x**	
@C_AlarmListing.pdl	x	x	x	**x**	x
@Overview1.pdl	x	x	x	**x**	
@TopAlarmNew.pdl	x	x		**x**	

In order to update the WinCC Project, please go to D:\CEMAT_CS and start 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project)

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

Select the desired **Screen Resolution** for your Project (1280x1024, 1600x1200, 1680x1050, 1920x1080 or 1920x1200). Via an option, during the OS Project Update the above listed Standard pictures are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

One by one the directories GraCS, Library, PRT and WScripts are copied into your OS Project and after this the PDLs are exchanged by the proper Screen Resolution. You have to acknowledge each step with OK and everything is entered into a LOG file.

During the update procedure the following GraCS files are overwritten by Cemat files.

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

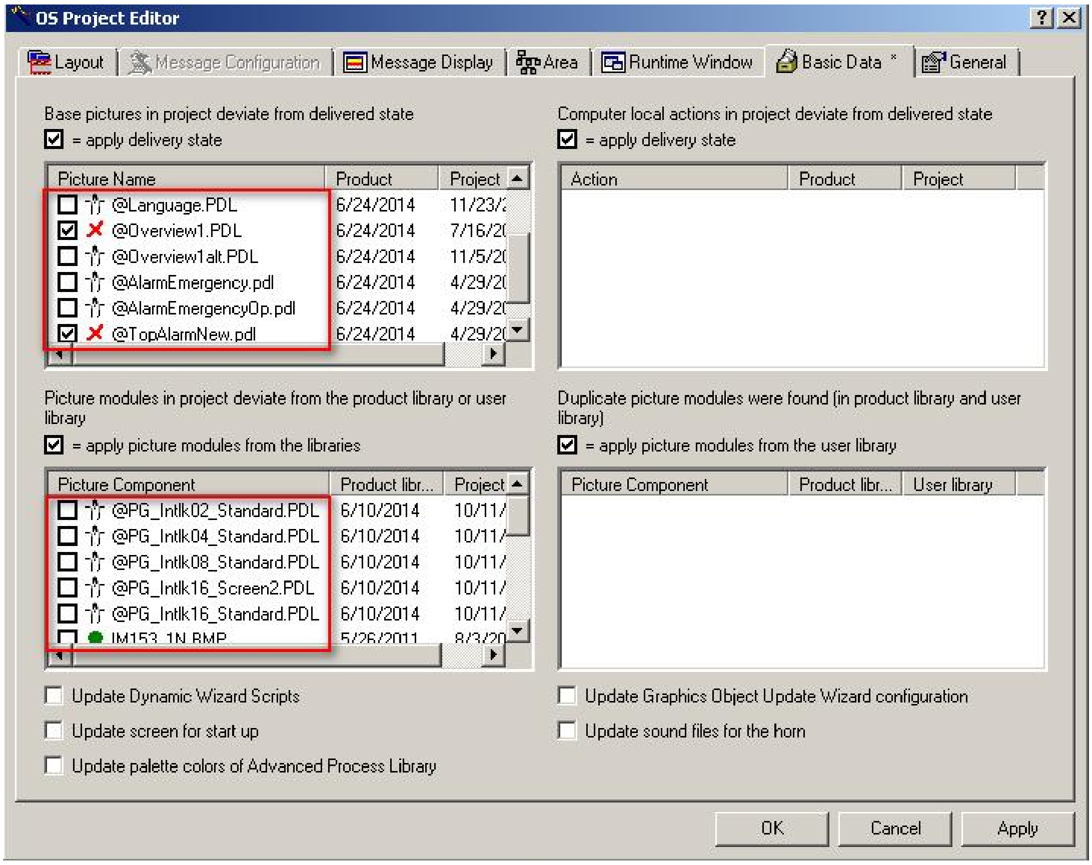
After the update of the WinCC Project, proceed as follows:

- Open the WinCC Project.

- Open 'Global Script' selecting C-Editor and regenerate Header. (Menu Options -> Regenerate Header)

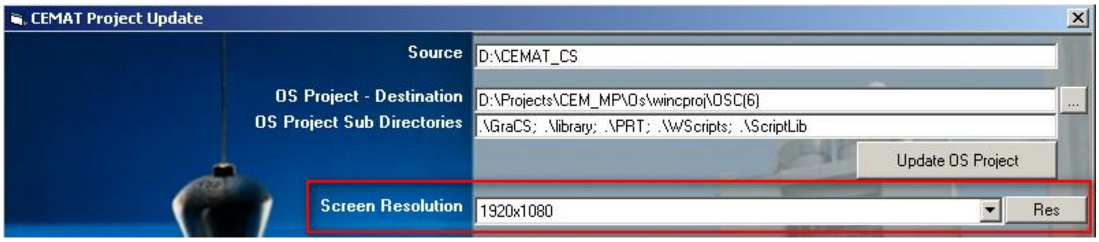
Tab “Basic Data”

Make sure that the system files which are modified by Cemat are deselected in the OS Project Editor under Basic Data. Otherwise, the next time when you run the OS Project Editor (e. g. in order to change the Area definition) the Cemat PDLs are overwritten by the PCS 7 system files.



Caution: @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must be replaced by the files from D:\CEMAT_CS\WinCC\GraCS again.

After running the OS Project Editor, use program 'CematProjectUpdate.exe' and press Button “Res” in order to update only the Screen resolution dependent Cemat pictures.



Via this function the following pictures are replaced by the Cemat Version again:

@AlarmEmergency.pdl,
@AlarmEmergencyOP.pdl,
@AlarmOneLine.pdl,
@C_AlarmListing.pdl,
@Overview1.pdl and
@TopAlarmNew.pdl

Generating and loading of the Server Data

In order to make the Server data available for the client, the Server data (Package) must be created on the Server Project and the OS Server must be assigned to the client.

There are three options to create the Server data:

- During OS Compile to the Server select the option “Create server data”.
- Via selecting the OS-Project of the Server with the right Mouse Button and choosing the Option “Generate Server Data…”.
- The same function you will find in the SIMATIC Manager under Options → OS → Generate Server Data.

In order to assign the OS Server to the Client:

- Select the OS-Project of the OS Client in the SIMATIC Manager with right Mouse button and choose “Assign OS Server…”
- The same function you will find in the SIMATIC Manager under Options → OS → Assign to Server…
- In order to load the Server data on the OS Client, open the OS Client Project, select Server data → Loading…, browse to the package file in the OS Server Project and select the Package file (.pck).

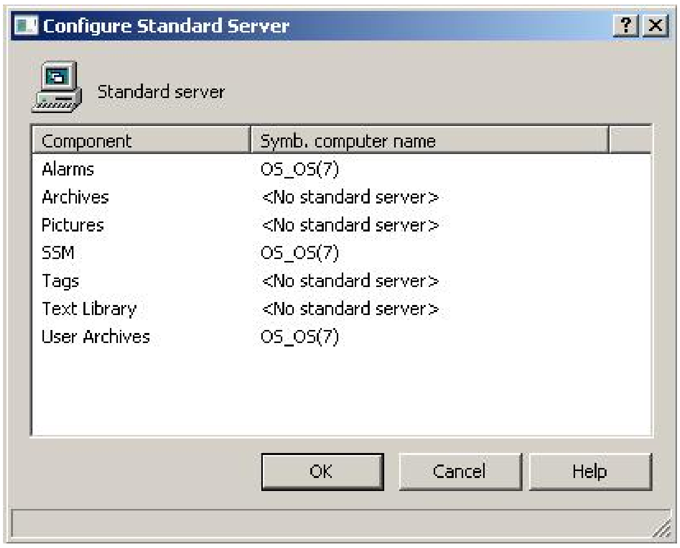
Once the Server data is loaded the package gets automatically updated during compilation of the OS-Data.

Now open the OS Client Project. The configuration of the standard server and the selection of the preferred server are carried out under WinCC.

Loading of the Server Data in a Client Project

To select the standard server, use context menu of Server data and select menu option Standard server...

- Select the standard server (exclusively!) for the components Alarms, SSM and User Archives.



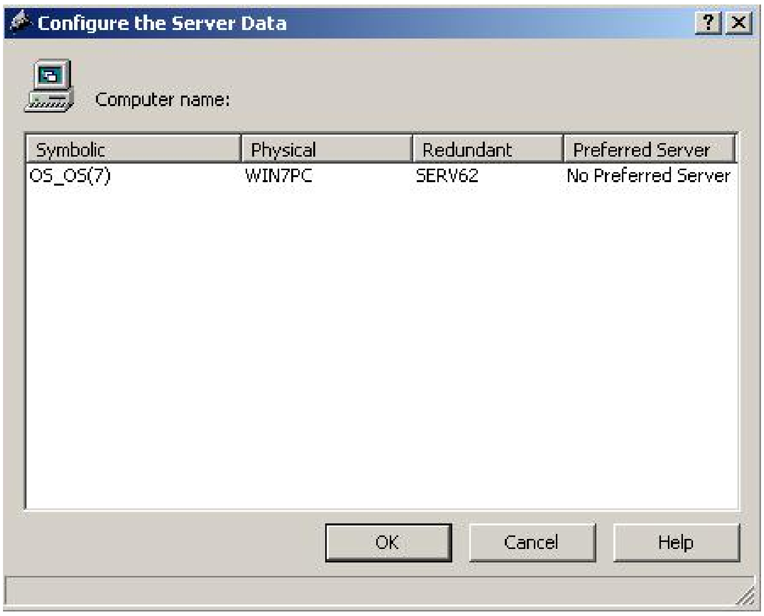
SSM = Split Screen Manager (e. g. the curve selection will be stored on the here selected Server).

Selection of a Preferred Server

To distribute the load for the servers uniformly it is useful to connect a part of the Clients to one Server and the other part to the Standby-Server.

Therefore a Preferred Server must be configured under Serverdata -> Configure ...

Example:



If the preferred Server is stopped the Client automatically switches to the Partner-Server. Once the preferred Server runs again, the Client automatically switches back to the preferred Server.

Selection of area specific Authorization levels

After the package is loaded the user groups and the users can be defined and the authorization levels can be defined area specific.

You can e.g. enable a complete area or only certain operations within an area.

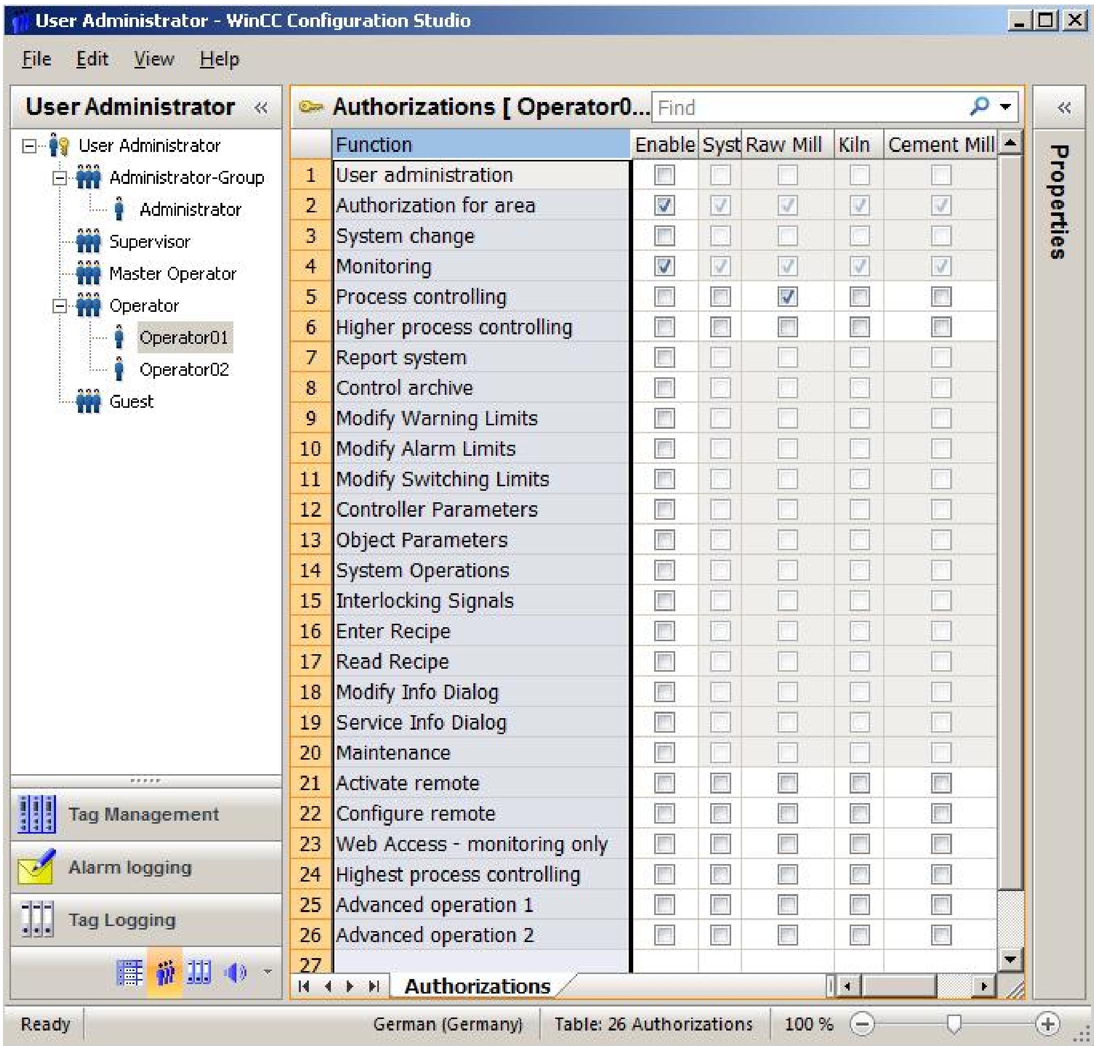
In the example we have defined the user groups “Administrator”, “Supervisor”, “Master Operator”, “Operator” and “Guest”. Within these groups different users may exist.

With the level 2 “Authorization for area” it is defined which area is released for the user.

With the level 4 "Monitoring" is defined whether the area can be monitored.

With the level 5 „Process controlling“, it is defined whether a user is allowed to operate the area. He will then also get the messages from this area and the horn is activated.

Example:



The user Operator01 is allowed to:

open, watch → System, Raw Mill, Kiln and Cement Mill
operate → Raw Mill

Messages are only shown for areas which are enabled for Process controlling. Operator01 will get only the messages from the Raw Mill area and he can acknowledge only these messages.

Horn configuration

Via Horn configuration the alarm sounds can be defined per message class (and if needed dependent on Priority, Source, Area or Event).

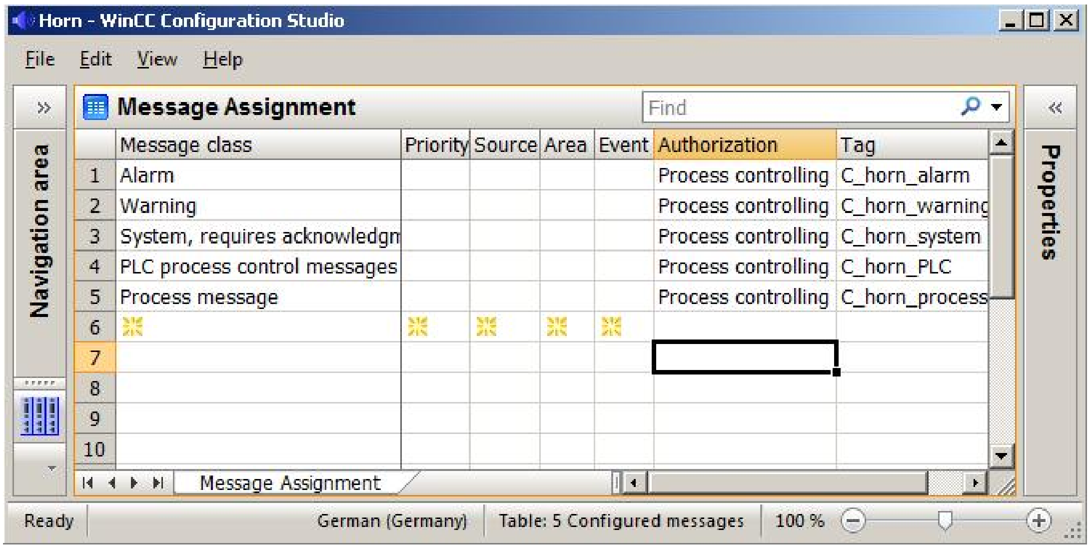
As the alarms are linked to authorization level “Process controlling” (users with this authorization will see the alarms), the alarm sound must be linked to the same authorization.

The Horn configuration is defined in the WinCC Configuration Studio and can be imported from a predefined file.

Open the WinCC Configuration Studio for Horn Configuration

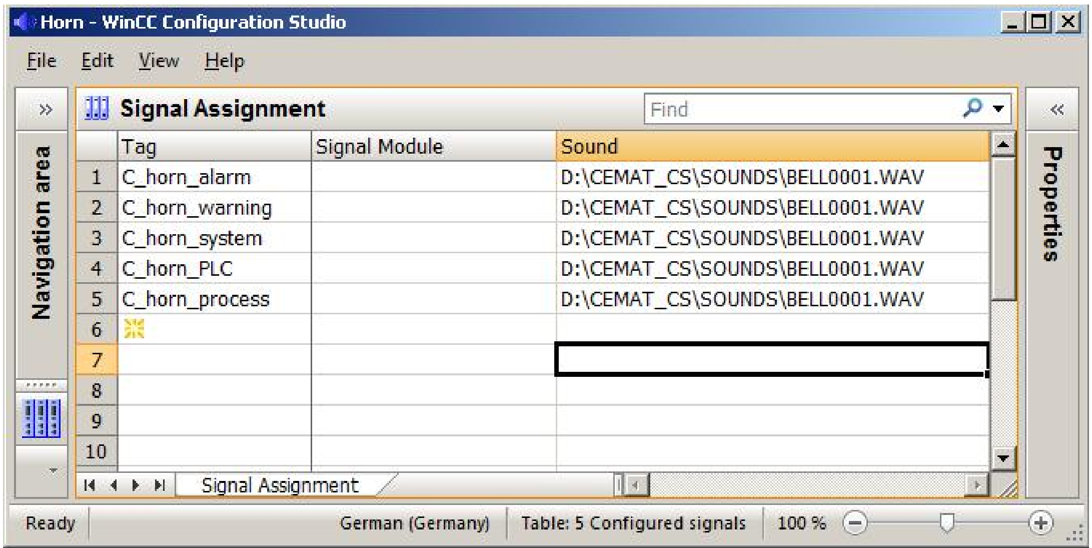
In order to import the message assignment, select tab message assignment and use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_HornMessgeAssignment.xlsx and press Button *Import.*

After the import you will see the following message assignment:



In order to import the Signal Assignment for the different sounds, select tab signal assignment and use function *Edit → Import*, browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_HornSignalAssignment.xlsx and press Button *Import.*

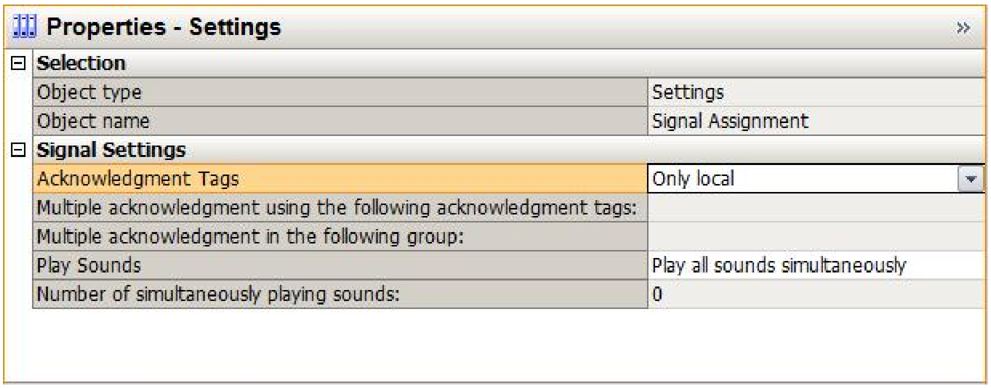
After the import you will see the following signal assignment:



Under “Properties” you can configure the Horn acknowledgement and how you want to play the sounds.

Acknowledgement:

- For stand-alone WinCC Clients the Horn Acknowledgement is “only local”.



- For WinCC Clients with signal devices acting in unison a Reset Variable must exist on in the Tag Management of the Server. This tag you have already created during the Server Configuration.

  In this case select the Horn acknowledgement option „Multiple acknowledgement using tags” and after that select the reset variable from the internal variables of the Server.

- For further details refer to PCS 7 descriptions

Time Synchronization

Time synchronization can be carried out in multiple ways, depending on your hardware and network configuration.

The easiest way of synchronization is to define the OS Servers as time Master. The AS and the Clients are defined as slaves. Please keep in mind that the AS works with Greenwich Mean Time and all OS Stations must be configured for the same time zone.

But you may also use an external time master (e. g. SICLOCK).

We cannot describe all possible options in this manual and kindly suggest consulting the PCS 7 Manuals and FAQs and if necessary the PCS 7 hotline support. The most important documents you find with the following links:

For the time synchronization:

http://support.automation.siemens.com/WW/view/en/57265478

For the NTP configuration of the PCs:

http://support.automation.siemens.com/WW/view/en/61931975

At this point we only want to emphasize the importance of the time synchronization in the system. If the time is not synchronized within your control system the alarm system and the archiving will not work properly. Even the timer functions in the Cemat blocks program are affected by wrong time stamps.

How to create a PDL Cache

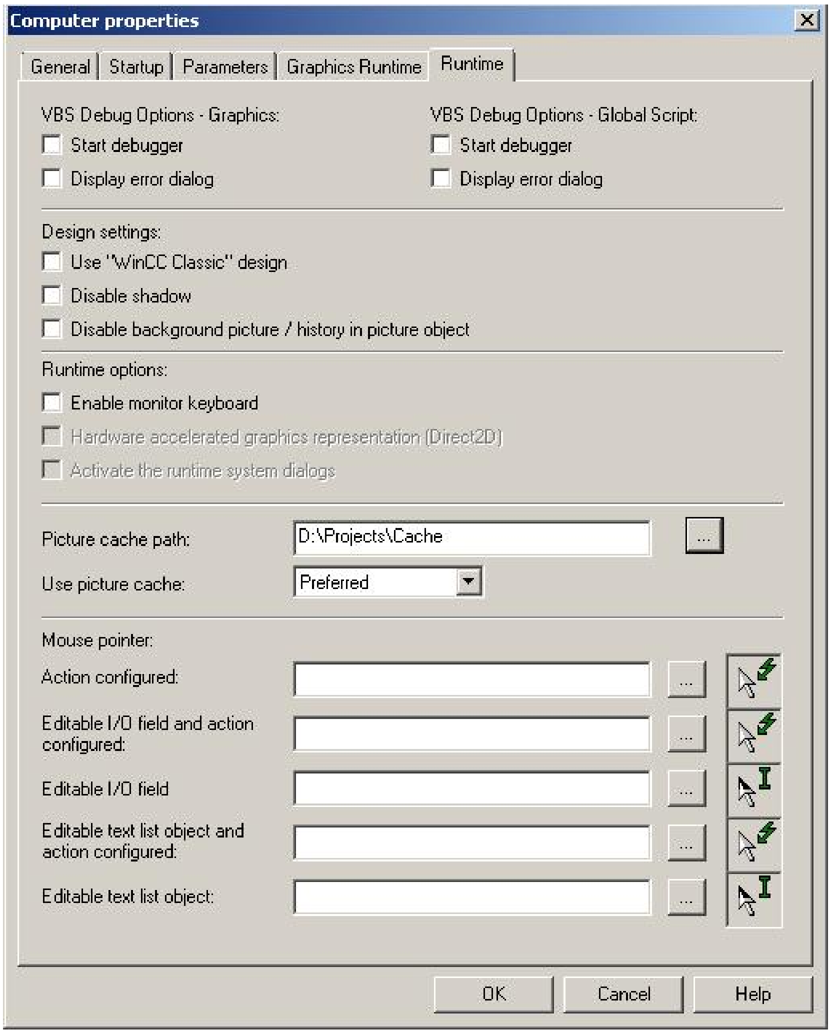
To display the runtime pictures, WinCC normally accesses the corresponding WinCC Server and Retrieves the current pictures form it. Using the Picture Cache, it is possible to store the WinCC pictures locally for display in runtime. When a Picture Cache is used, the WinCC Client does not need to reload the pictures continually. Thus, shorter picture change times can be achieved.

The required pictures must be manually saved on the computer which should use the Picture Cache. To do this, create a folder on the computer in the standard directory C:\…..\Siemens\WinCC\Bin\PDLCache. The name of the folder must be the symbolic computer name of the package stored on the server, e. g. OS_OSSRV

The settings for the PDL Cache are carried out under Computer Properties, Register Runtime. There you can select the following options for reading the Cache.

not	The Picture Cache will not be used
Preferred	Modified pictures will be read from Server, unchanged pictures will be read from the Picture Cache.
Always	The pictures will always be read from the Picture Cache.

Under “Path” you have the possibility to choose a different Location where the pictures are stored. The Path specification must only be entered up to the directory in which the PDLCache folder is located. If the standard directory is used, the path doesn’t need to be specified.



Assignments

Content

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\04_Assignments_009.doc

Assignments FB, FC, DB

FB

Range	Assignment
000 – 499	PCS 7
950 – 1199	CEMAT
1200 – 1799	User
1800 – 2047	AP Library PCS 7

FC

Range	Assignment
001 – 400	PCS 7
1000 – 1199	CEMAT
1200 – 1399	User
1400 – 2047 *	CFC-Tasks

DB

Range	Assignment
001 – 399	PCS 7
400 – 599	User
600 – 999	CEMAT
1000 – 4095	CFC-Instances

UDT

Range	Assignment
001 – 1499	CEMAT
1500 – 4095	User

Memory, Timer, Counter

Memory	
0000.0	
	PCS 7
0000.7	
0001.0	
	CEMAT
0199.7	
0200.0	
	User
2047.7	

The biggest memory address depends on the selected CPU Type.

CEMAT Standards don‘t use Timers and Counters

Please consider that the quantity differs from CPU type to CPU type. The above mentioned numbers refer to CPU 416-2.

Engineering Examples

Content

General

This description should provide support in the solution of control tasks using the CEMAT software.

Please adhere to these recommendations that have proved themselves in practice.

Many block parameters start for historical reasons with a certain codification (name was never changed because of Migration of existing plants). The codification is as follows:

Object type	**Block name**	**Code**
Unidirectional drive	C_DRV_1D	E
Damper	C_DAMPER	K
Valve	C_VALVE	V
Annunciation module	C_ANNUNC	M
Measured value	C_MEASUR	UM
Route	C_ROUTE	W
Group	C_GROUP	G
Selection module	C_SELECT	AW
Silo pilot	C_SILOP	SP
Counter	C_COUNT	CNT
Running time	C_RUNNT	RT

Basic Rules

It is **essential to** conform to the following basic rules.

Limit Switches

The break contacts from these switches should be connected directly in the contactor control circuit. Connect the make contact elements to the AS inputs.

Binary Special Signals

Signals such as belt drift switch, pull-rope switch, silo full must always be connected to an annunciation module. The module flag of the annunciation module (OutSig or MAU) should then be used for the further use in the program.

Start/Operation Interlock of the Group

In order to display also the interlocking conditions in the status call-up of the group, the interlock signals (also internal flags) should be connected to annunciation modules.

Damper Directions

The following definition applies for the end position of the damper:
direction 1 = closed
direction 2 = open

Connection rules and recommendations

Selection and Route Module

We provide two module types for selections:

- selection module
- route module

The selection module is particularly suitable for smaller branches, standby circuits.

Advantage:

- reduced configuration effort
- reduced program run-time

The classical route module is particularly suitable for (long) transport routes that use different drives.

Advantage:

- Through assignment of the objects to a route 'partial groups' are defined.
- Only the status of the objects belonging to a selected route will be forwarded to the group.
- Status call and object list call can be carried out for the individual route, which permits the diagnostic function per destination

The question whether a selection module, a route module or various groups should be used must be decided individually.

Annunciation Release

In all Cemat blocks through plausibility logic it is assured that in case of several simultaneous faults only the "perpetrator" creates an alarm message.
For example, in case of a protection interlock of a motor, no alarm is created by the motor block, it is created by the associated annunciation module.

In order not to create an alarm flush in case of loss of control power, each Cemat block has an interface "Annunciation Release" xMFR, which has to be connected with the control power signal.

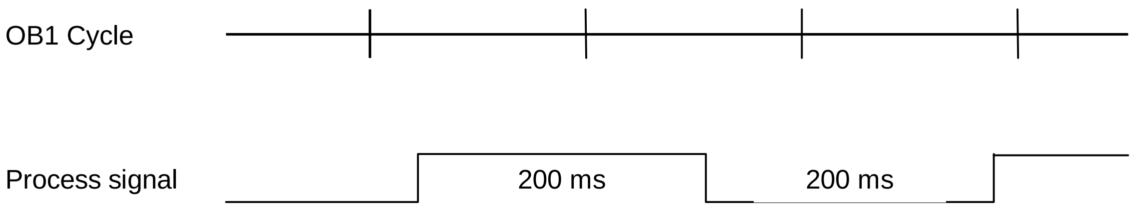
In case of a periphery failure (such as bus failure, card failure, MCC failure) this signal must get "0". At this moment the Cemat block does not create further messages (no incoming, no outgoing messages).

For the annunciation of the control power failure itself an additional annunciation block has to be programmed.

⚠ The annunciation release is one of the most important interfaces in order to achieve a smooth operation. The project engineer must assure that only the perpetrators of a fault create alarm messages.

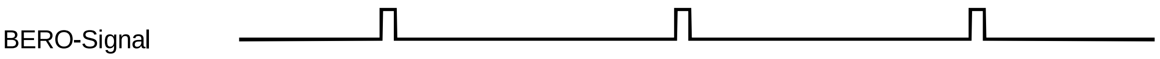
Impulse Processing

Both the pulse length and pause length must be longer than the OB1-cycle for signals that are to be recorded from the cyclical program (OB1). As guidance value, it should be longer than 200ms.



If the signals are recorded in the 100ms-OB cycle (time-controlled alarm processing), the pulse times and the pause times of the signal must both be longer than 150 ms.

There are signals that are created with a flag attached to the drive shaft and a BERO switch. Such signals, for example, have a short pulse and a long pause.



Such signals must not be wired **directly** to an input module.

The signals must be converted with C1-technic hardware circuits or with series-connected mini programmable controllers (binary converter) so that the programs can record them reliably.

Such signals can occur in silo pilots or speed monitors.

Group functions

Group start/stop

Starting and stopping of a group can be achieved via Operator Faceplate, through the program or via a conventional Control desk (Push Buttons).

Group Start and Stop via operator faceplate

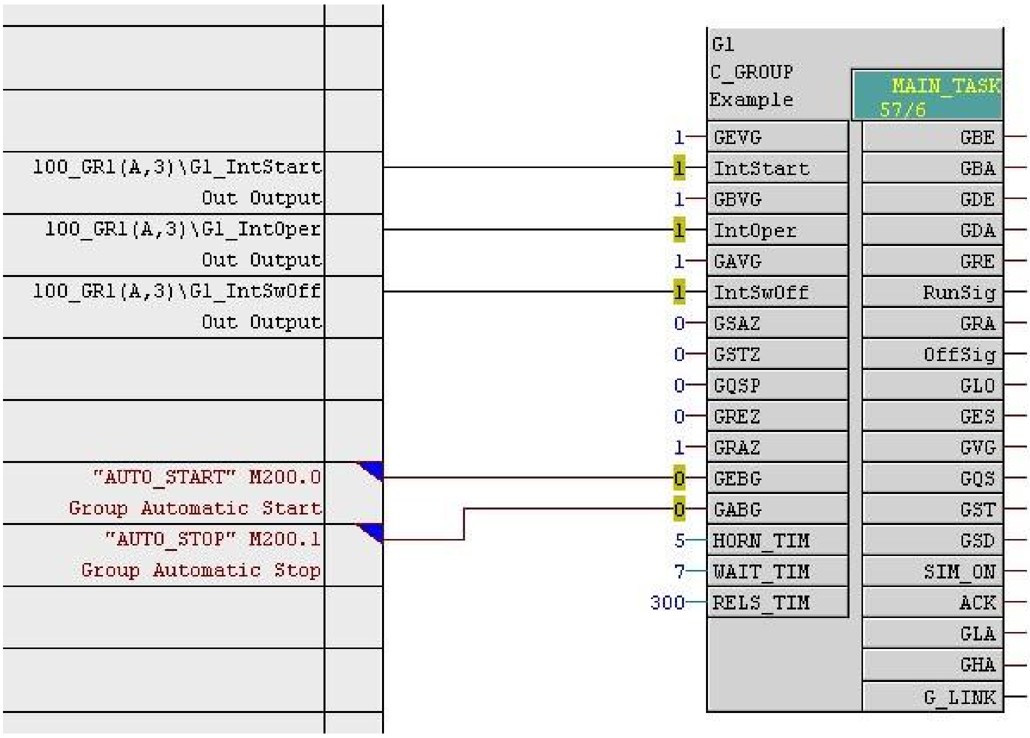
For the start and stop via operator faceplate no programming in the CFC is needed.

Group start via operator faceplate is only permitted if the start interlock and the operation interlock have 1-Signal.

Group Stop is only permitted if the switch-off interlock has 1-Signal.

Group start and stop through the program

In order to start and stop the group via program the interface GEBG must be connected with the start command and the interface GABG with the stop command.



Group start via interface GEBG is only permitted if the start interlock and the operation interlock have 1-Signal. A rising edge on GEBG is required.

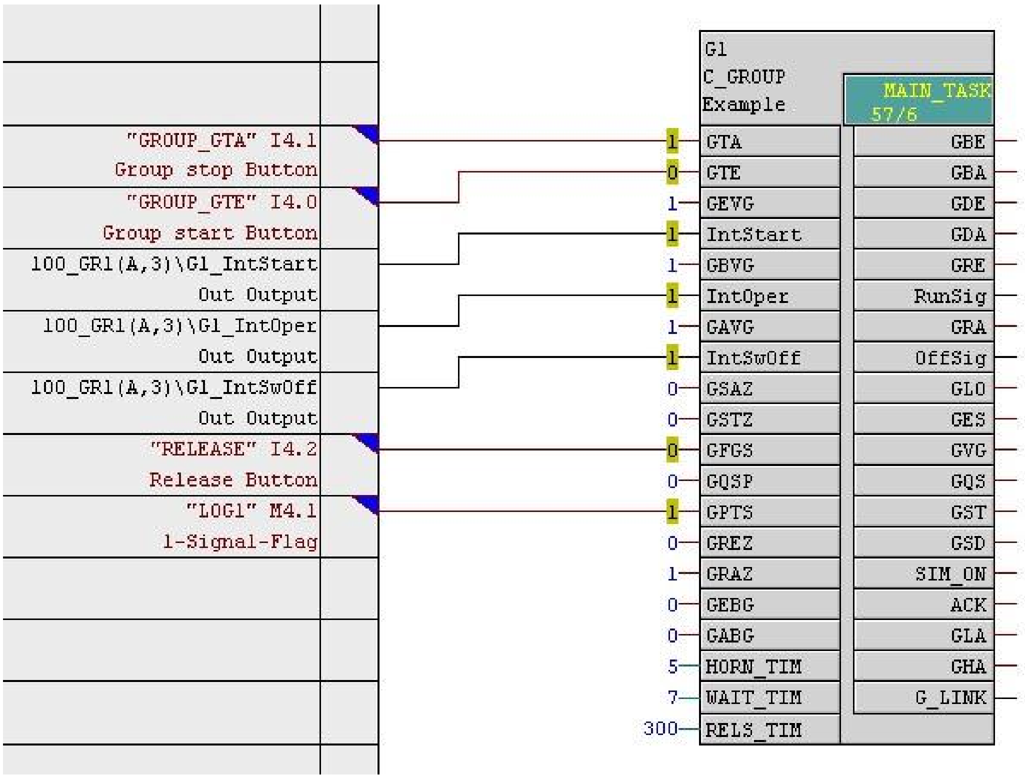
For the group stop via interface GABG the switch-off-interlock is not considered!

In project standard 004 (Holcim), the group start via interface GEBG is only permitted if the group is switched into Standby mode. The Standby mode must be enabled by the operator. The group stop via interface GABG is always permitted.

Group start and stop via conventional control desk (push buttons)

For the start and stop via conventional push-buttons the interface GTA must be connected with the Stop button and interface GTE with the start button. GTA is normally closed, which means it must be “1” if the button is not pressed. GTE is normally open.

In order to ensure a two-hand operation, the interface GFGS must additionally be connected with a Release button. The Release button must be pressed together with the GTE or GTA button.

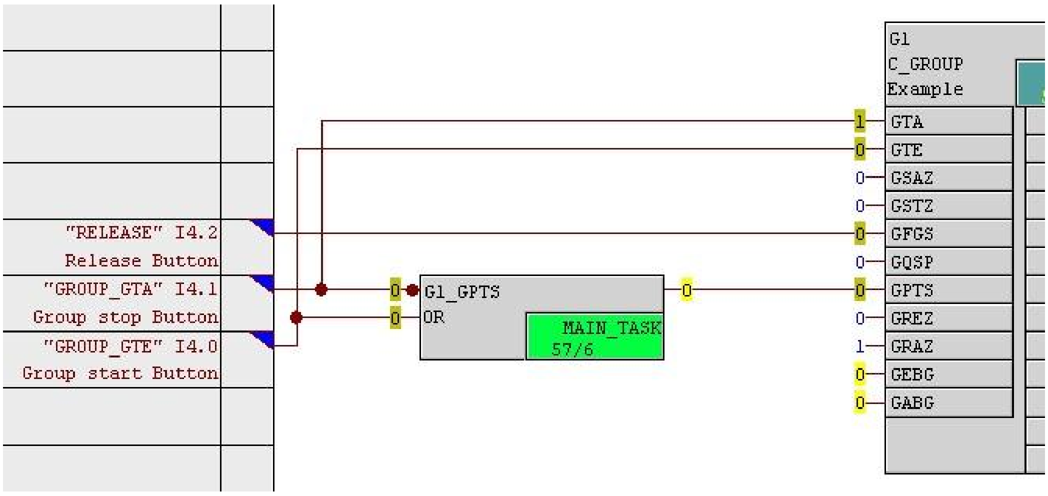


The group start via GTE is only permitted if the start interlock and operation interlock have 1-Signal. A rising edge on GTE is required.

Group Stop via GTA is only permitted if the switch-off interlock has 1-Signal.

In order to enable the interfaces GTA and GTA the pushbutton release GPTS must be connected with a 1-Signal. (For the operation via operator faceplate GPTS must have 0-Signal).

If you want to allow a parallel operation, via operator faceplate and via push buttons, this can be achieved with the following logic.



Start-up-warning

Before the group start is transmitted to the drives, a start-up-warning must be given. The group block has two outputs, one for a lamp and one for an acoustical signal (horn).

The lamp output GLA remains set during the complete start-up sequence and it will be switched off when the group start is completed.

The horn output GHA will be set for the duration of parameter HORN_TIM. It can be used for an acoustical alarm in the field.

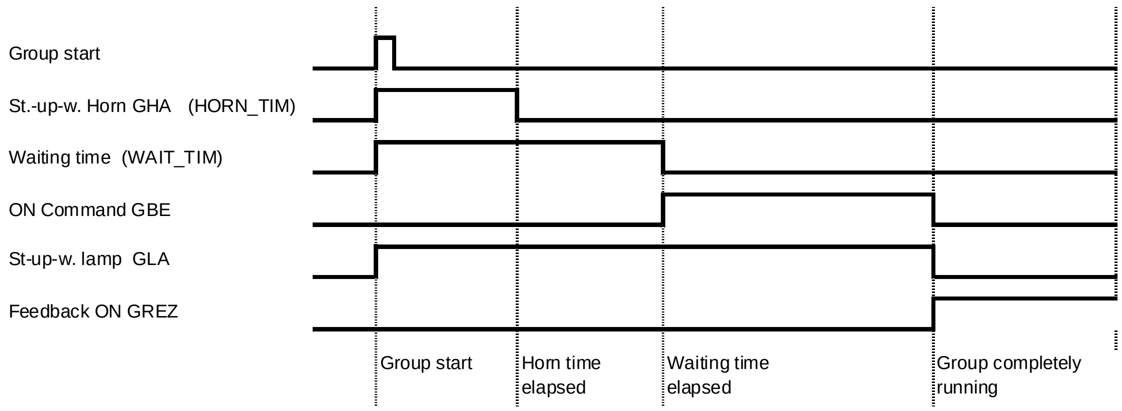
The drives get started only after the waiting time WAIT_TIM has elapsed.

Horn time and Waiting time start simultaneously, which means that the waiting time should be a little bit bigger than the horn time. This insures that after the horn goes off the start is still delayed for some seconds.

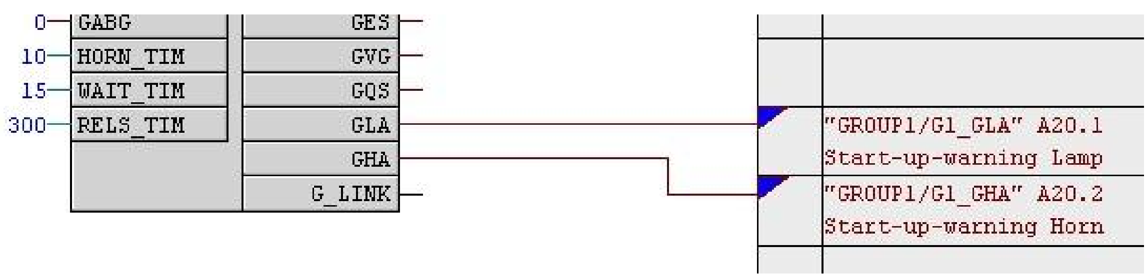
The start command is limited by the release time RELS_TIM. After the release time elapsed no start command is given to the drives.

In case of a problem in any of the group objects during the start procedure, the dynamic fault of the object interrupts the group start command, but in this case the release time still proceeds. If the problem can be solved within the release time the group start can be triggered without creating a new start-up warning.

Time diagram:



Parameters and Block outputs for the start-up-warning:



For the start-up-warning in single-start-mode the HORN outputs of the drives must additionally be considered.

Group start interlock

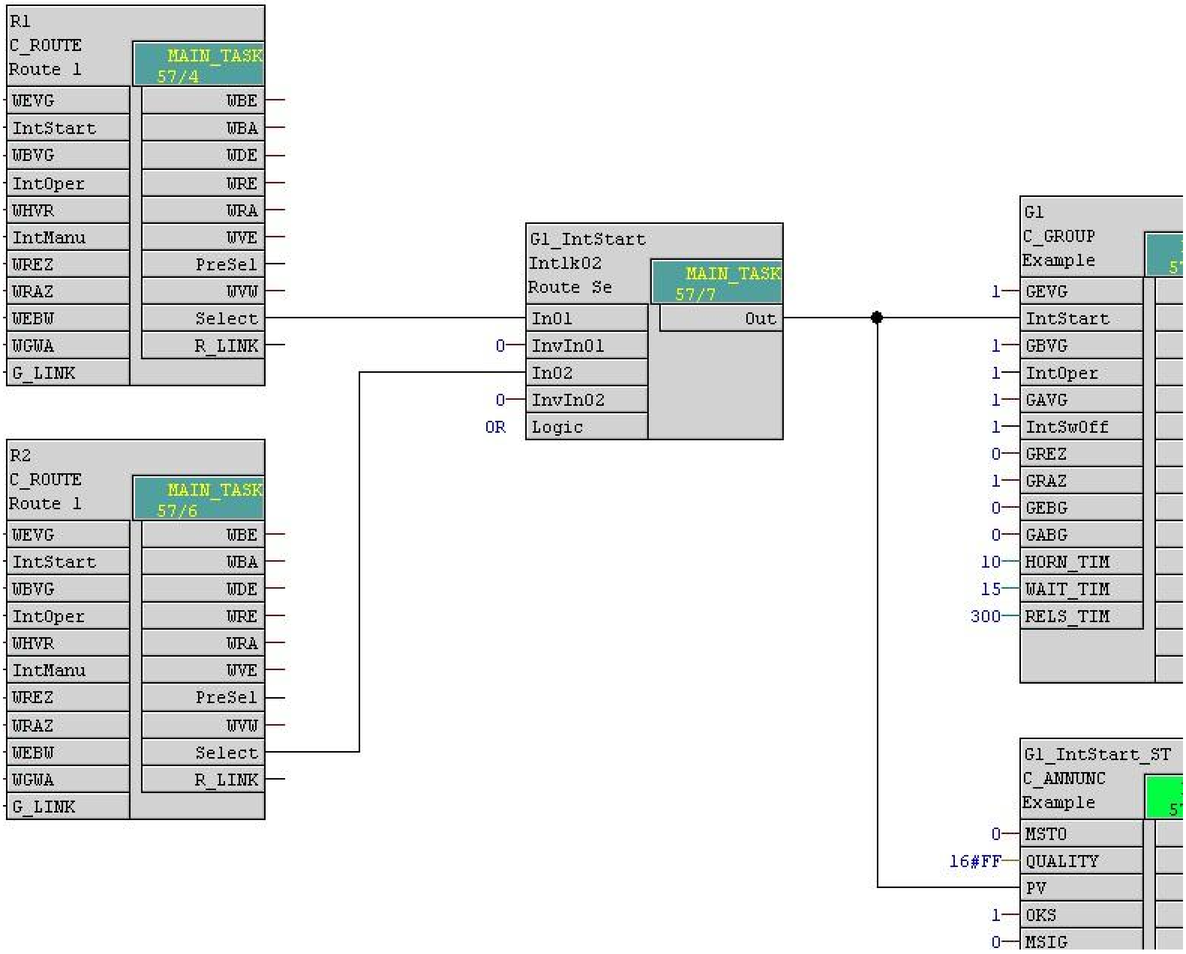
Through group start interlock you can avoid a start-up-warning for a group which is not ready to start (e. g. if a route must be selected before group start).

Two interfaces exist, which are effective at the same time, which means that they must both have 1-Signal in order to allow the group start.

The interface GEVG has format BOOL and can be used for connecting binary signals or one of the binary interlock blocks C_INTERL or C_INTER5.

The interface IntStart (from Cemat V7.1) has STRUCTURE format and can be connected with a structure output or with a structure interlock block, e. g. Intlk02. If this interface is used, the connected block can directly be opened via the faceplate of the group.

Example: One of two routes must be selected in order to allow the group start.



In order to provide this information in the status call of the group, annunciation blocks must be programmed for the start interlocks. See “Annunciation bocks for status call function”.

Group operation interlock

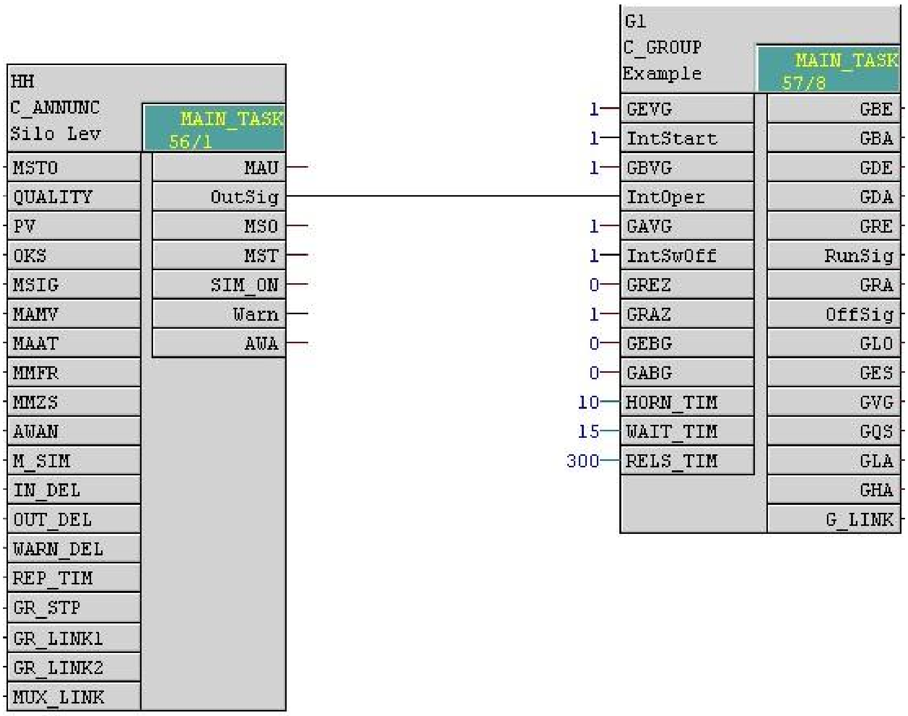
Process signals which must stop a group have to be connected to the operation interlock (e. g. Silo full or discharging group has tripped). Stopping of the group is similar to a normal group stop (no Quick Stop).
This kind of group stops must be acknowledged by the operator.

Two interfaces exist, which are effective at the same time, which means if one of these interfaces has 0-Signal the group will be stopped.

The interface GBVG has format BOOL and can be used for connecting binary signals or one of the binary interlock blocks C_INTERL or C_INTER5.

The interface IntOper (from Cemat V7.1) has STRUCTURE format and can be connected with a structure output or with a structure interlock block, e. g. Intlk02. If this interface is used, the connected block can directly be opened via the faceplate of the group.

Example: The feeding group must be stopped if the silo is full.



In order to provide this information in the status call of the group, annunciation blocks must be programmed for the operation interlocks. See “Annunciation bocks for status call function”.

Group switch-off interlock

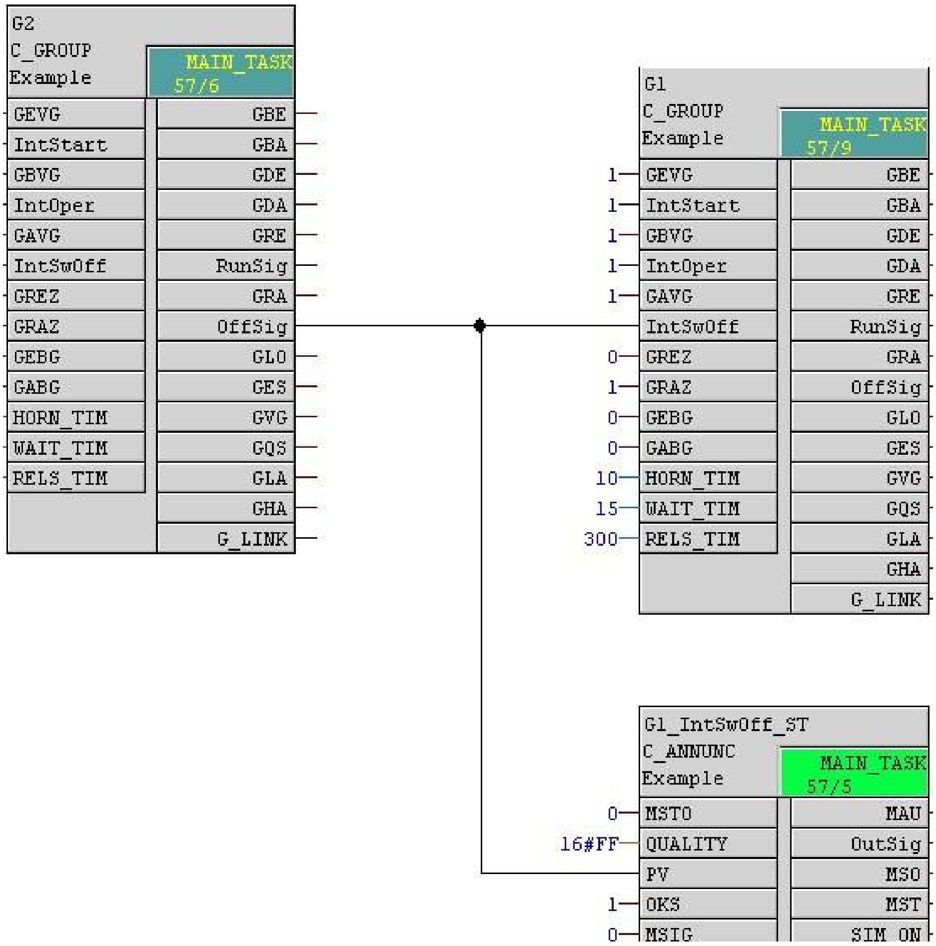
Via switch-off interlock, stopping of a group can be inhibited; e. g. if another group must be switched off before.

Two interfaces exist, which are effective at the same time, which means that they must both have 1-Signal in order to permit the group stop.

The interface GAVG has format BOOL and can be used for connecting binary signals or one of the binary interlock blocks C_INTERL or C_INTER5.

The interface IntSwOff (from Cemat V7.1) has STRUCTURE format and can be connected with a structure output or with a structure interlock block, e. g. Intlk02. If this interface is used, the connected block can directly be opened via the faceplate of the group.

Example: Before a group can be switched off, the feeding group must be stopped.



In order to provide this information in the status call of the group, annunciation blocks must be programmed for the switch-off interlocks. See “Annunciation bocks for status call function”.

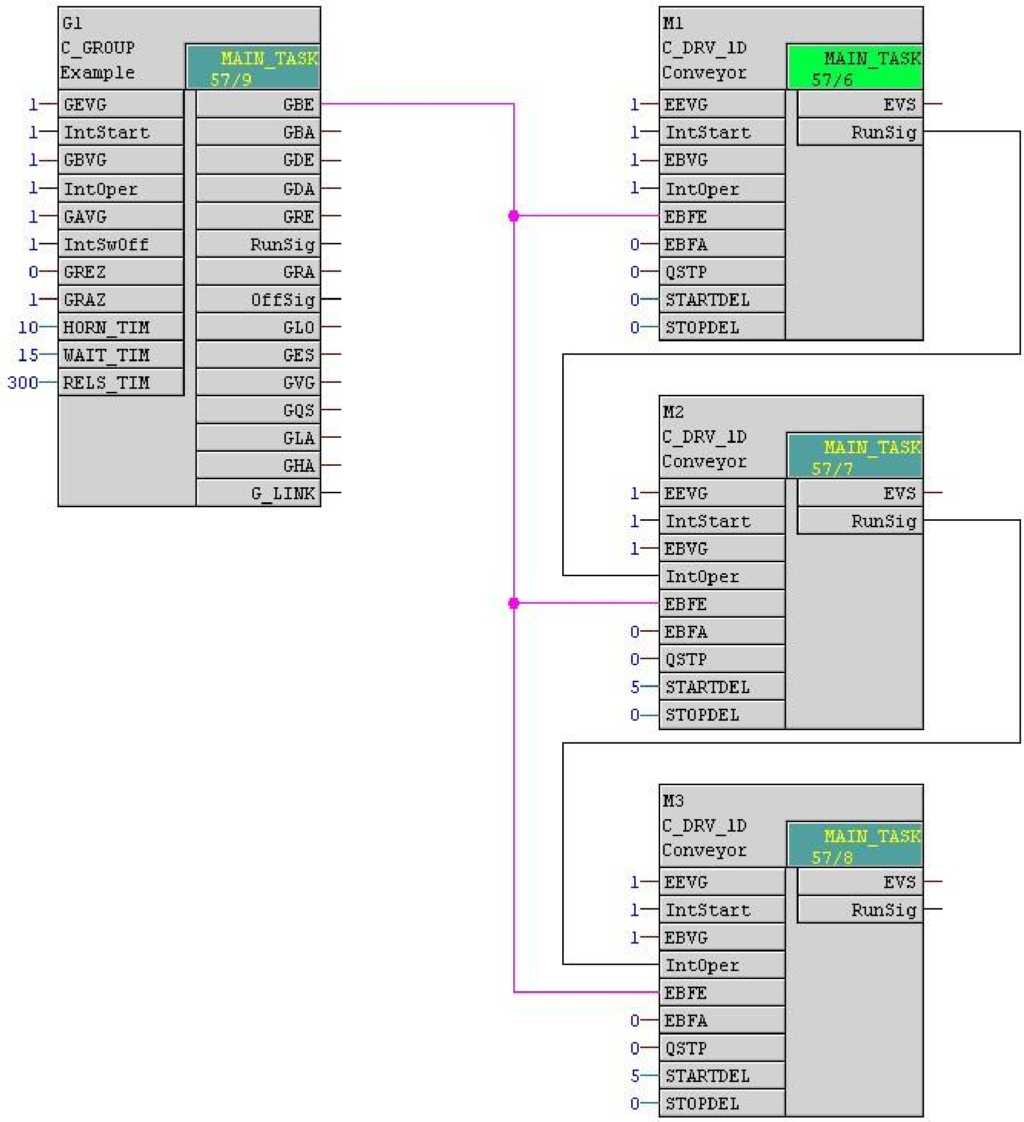
Start and Stop of the drives

For the **Start** of the drives, use output GBE of the group. The group output GBE is set after the start-up-warning (Horn time and waiting time) has elapsed and if there is no fault it remains “1” until the group is completely running.

If it comes to a fault within the start-up sequence, the group output GBE is reset.

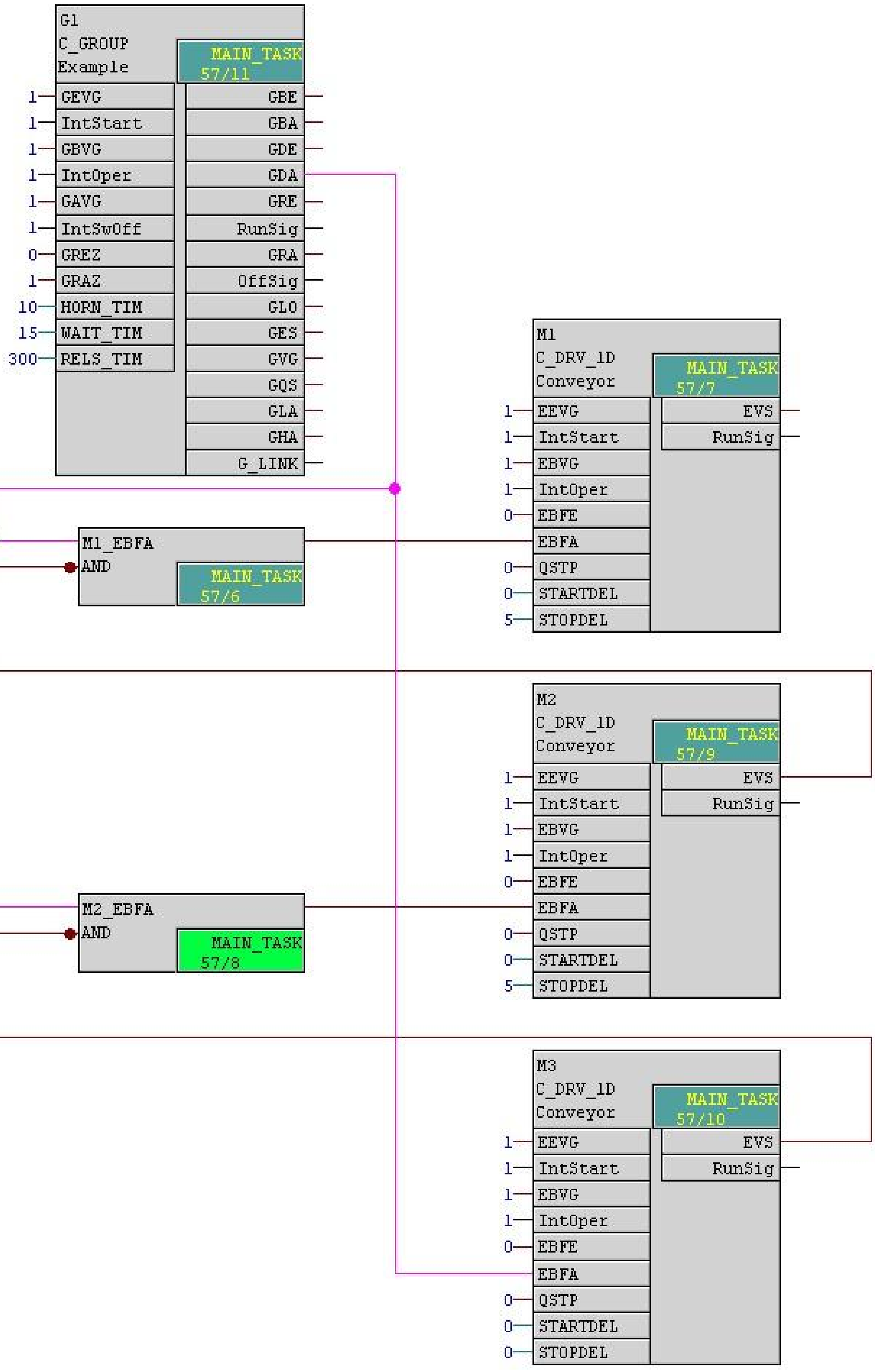
After the release time has elapsed the group output GBE is also reset.

The group output GBE must be simultaneously connected to all drives. The start sequence is programmed through process interlocking between the drive. The start of the drives can additionally be delayed via parameter STARTDEL (see drive blocks).



Beside output GBE the group has an additional output GDE (continuous on command). Signal GDE is not suitable for starting the drives, because it is not reset in case of a fault. **The drive fault can not be acknowledged as long as there is a continuous 1-Signal on the motor start command!**

For the **Stop** of the drives, use output GDA of the group. The group output GDA is set with the group stop and remains “1” until all drives of the group have been stopped.



There is no switch-off interlock available for the drives. For a sequential stop, the stop condition (feeding drive has been stopped) must be connected together with the group output GDA to the drive. The drive stop can additionally be delayed via parameter STOPDEL (see drive blocks).

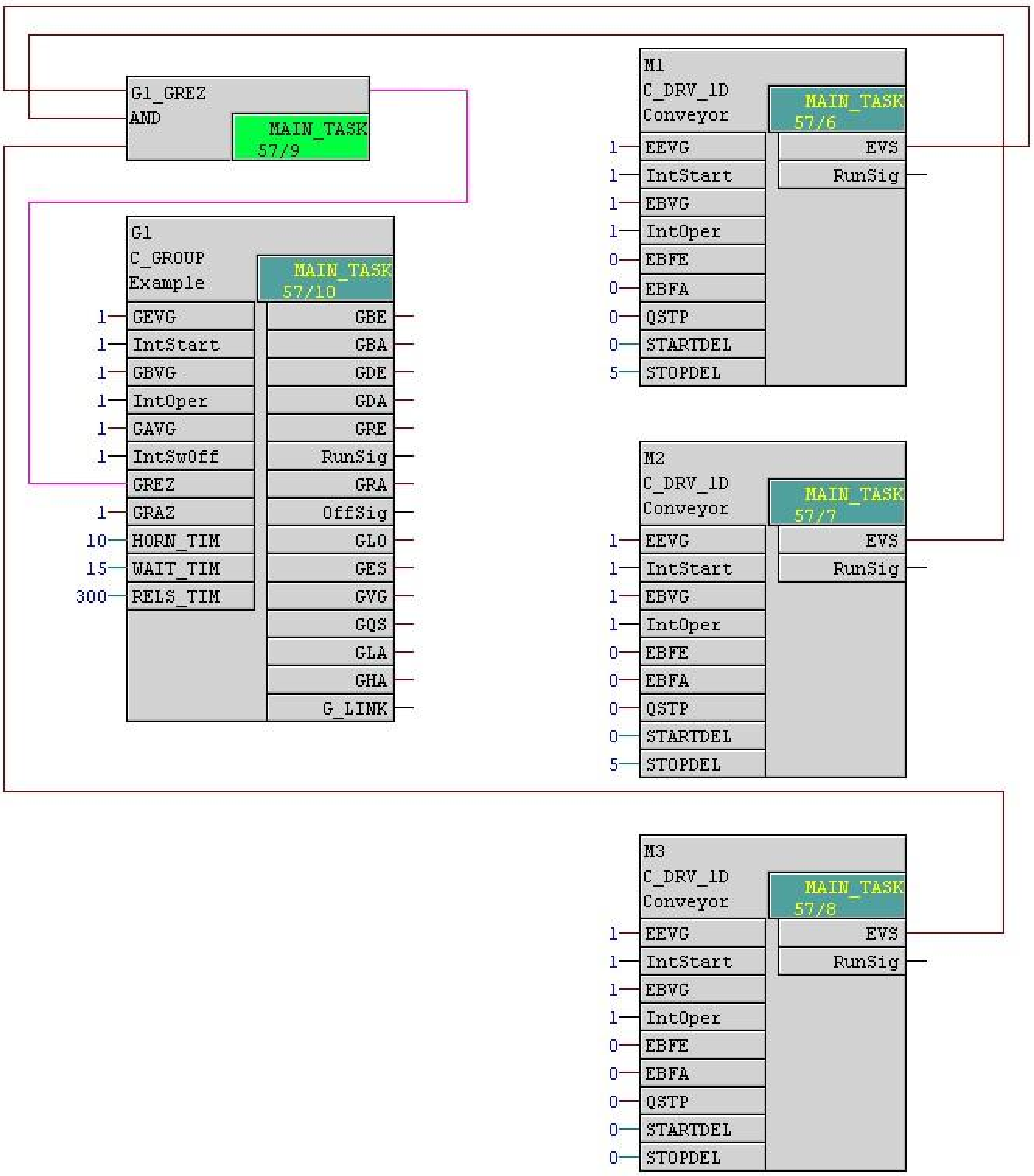
Beside output GDA the group has an additional output GBA (Off command). Output GBA is only a pulse, which means it is not suitable for stopping the drives (if a stop delay is programmed, the output GBA doesn’t stay long enough).

Feedback ON (all drives running)

Via the group feedback ON (interface GREZ) the group is informed that the start-up has been completed and all drives are running.

Depending on the application this can be an AND-Function with the running signal of all drives, or the last drive of a start sequence.

With 1-Signal at input GREZ the start-up warning lamp GLA goes off and the on command GBE is reset.



In case of a group with routes, the route outputs WRE (Route Feedback ON) of the relevant routes are connected to input GREZ of the group.

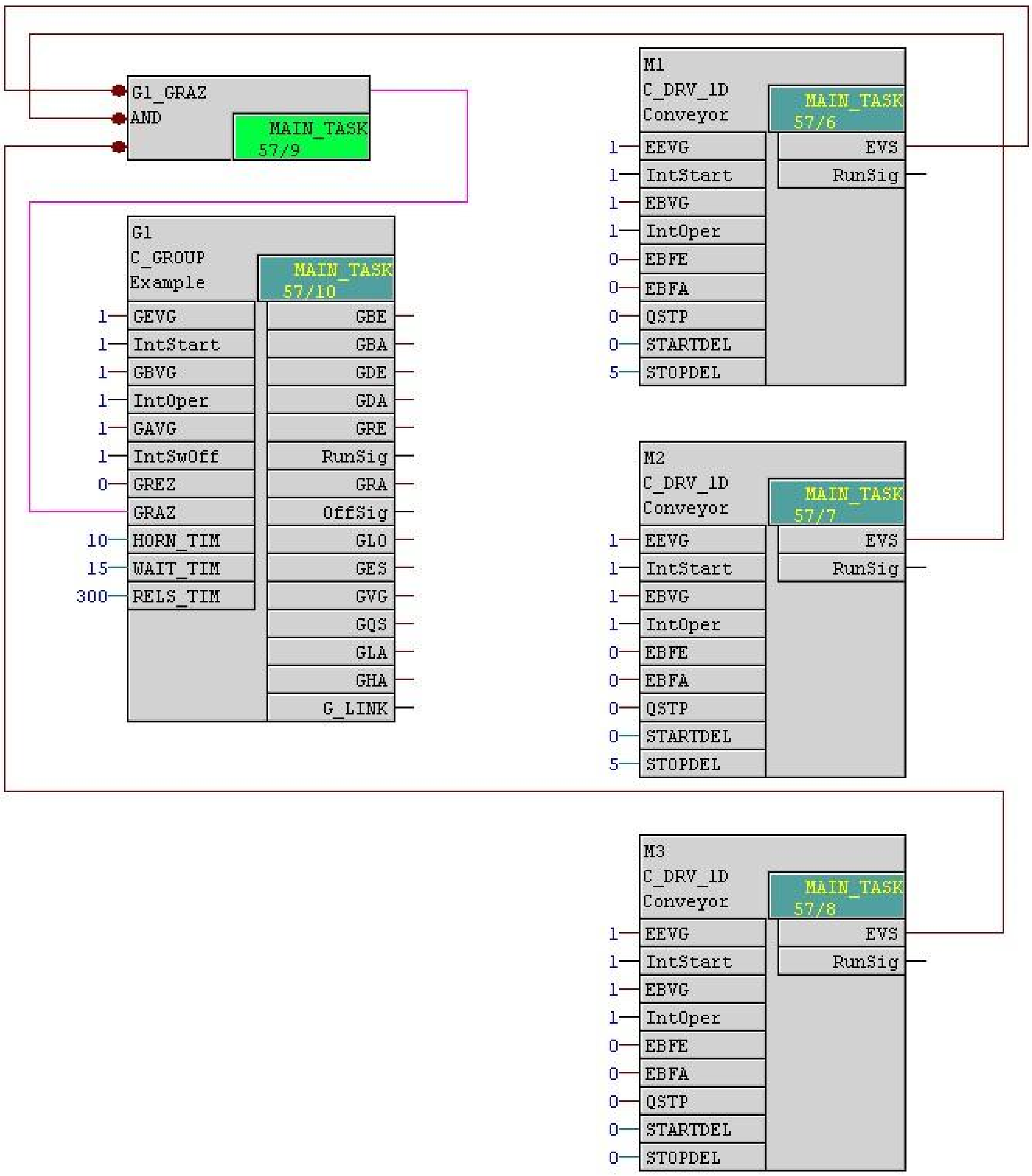
As long as GREZ has 1-Signal the start button of the group is disabled and restart is not possible. This especially has to be considered in case of an uninterrupted route change-over, if one route is still running while a second route shall be started.

Feedback OFF (all drives stopped)

Via group feedback OFF (interface GRAZ) the group is informed that the stop sequence has been completed and all drives are stopped.

Depending on the application this can be an AND-Function with the inverted running signal of all drives, or from the last drive of a stop sequence.

With 1-Signal at input GRAZ the off command GDA is reset as well as the output GVG of the group.



In case of a group with routes, the route outputs WRA (Route Feedback OFF) of the relevant routes are connected to input GRAZ of the group.

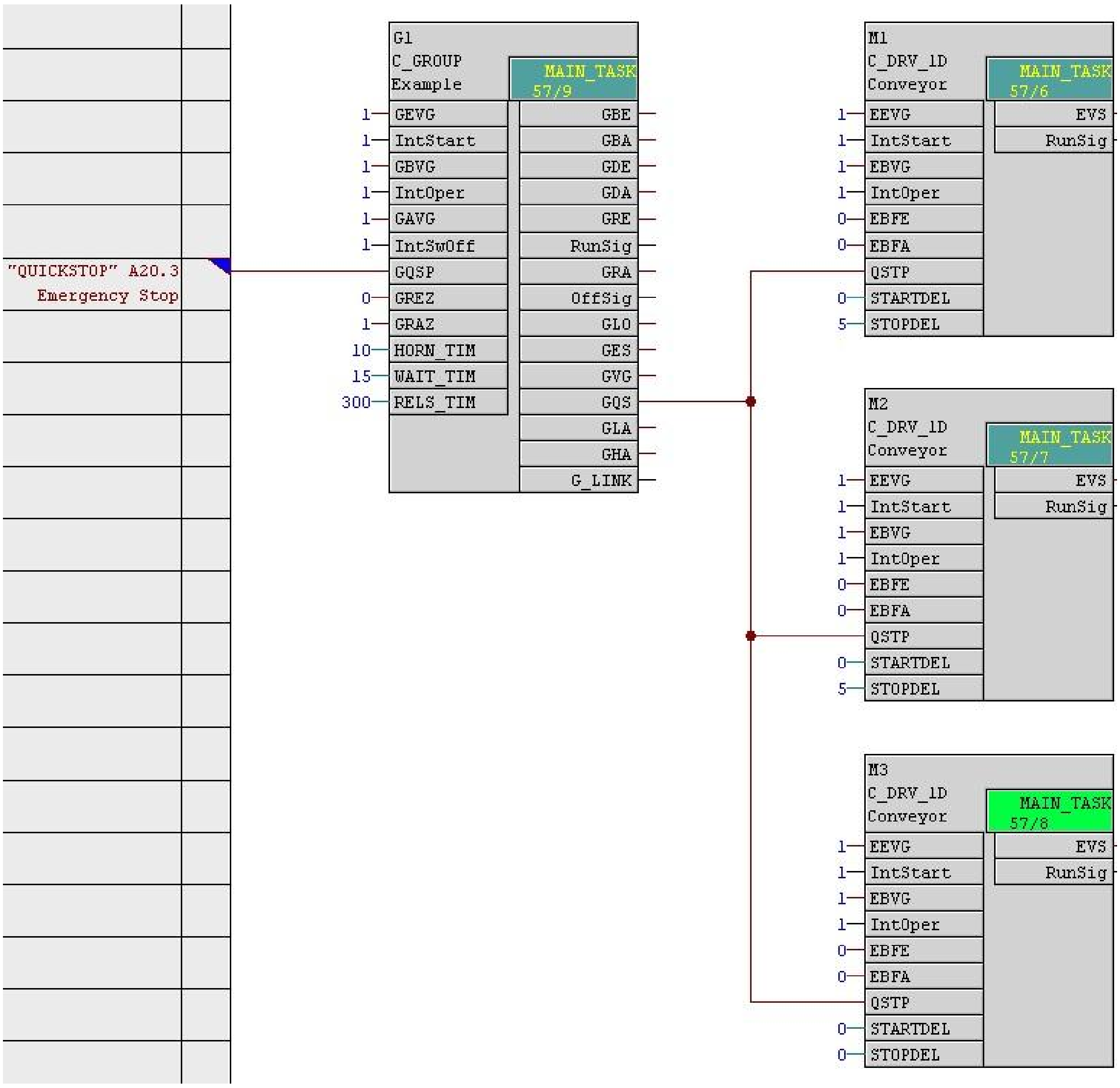
Route Feedback OFF (Signal WRA) has 0-Signal if the route is not selected or if the route is completely stopped. For this reason, WRA must be inverted for the connection to the group feedback OFF.

Quick stop

The group Quick Stop function is used for an immediate stop of the drives, which means the stop delay STOPDEL is not considered.

The Quick Stop can either be carried out via the operator faceplate through the program via interface GQSP.

In both cases the group sets an output GQS, which has to be connected to input QSTP of the drives.



In order to display the Quick Stop Button in the operator faceplate, in the property „Styles“ of the block icon the attribute „StyleTag“ must be adapted accordingly.

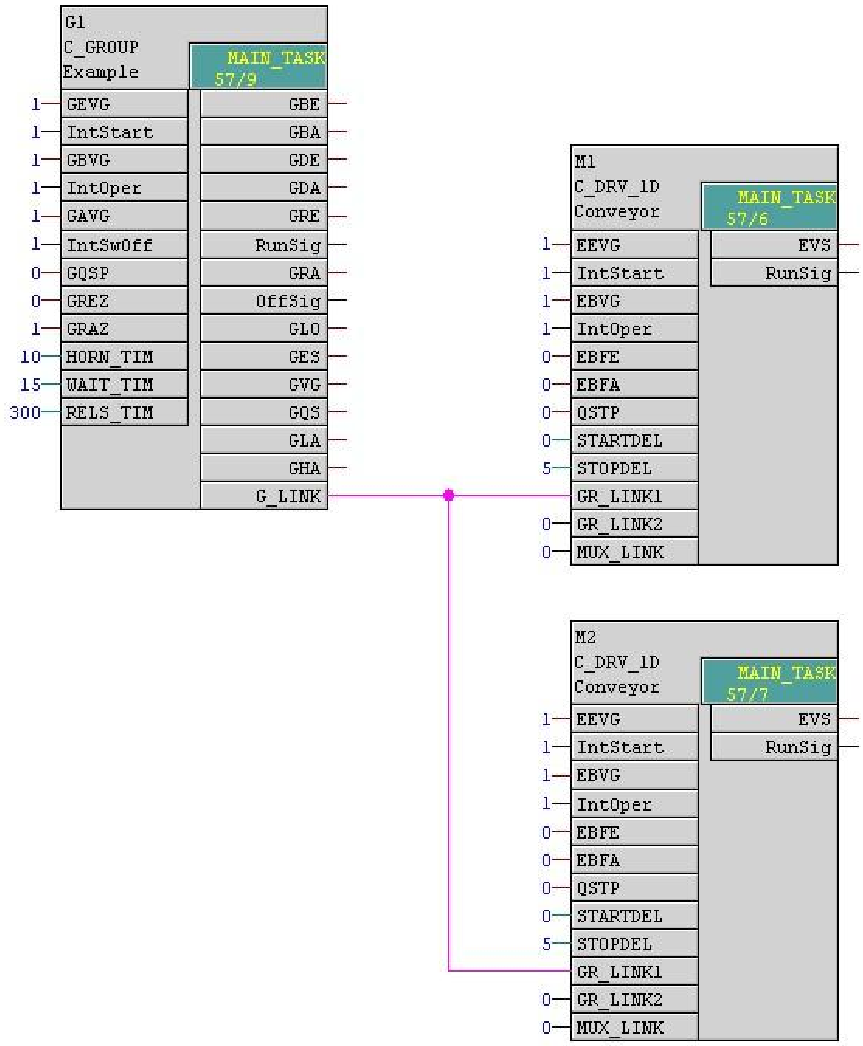
With StyleTag 2, 6, 7 and 8 or. 102, 106, 107 and 108; the button “Quick Stop" in the group faceplate will be visible. → See description 08_OS_Engineering_009.doc.

Group Links

All objects (Drives, Annunciation blocks, Measured values) belonging to a group must be assigned to this group via group link. This connection is essential, in order to inform the group about the status of the objects.

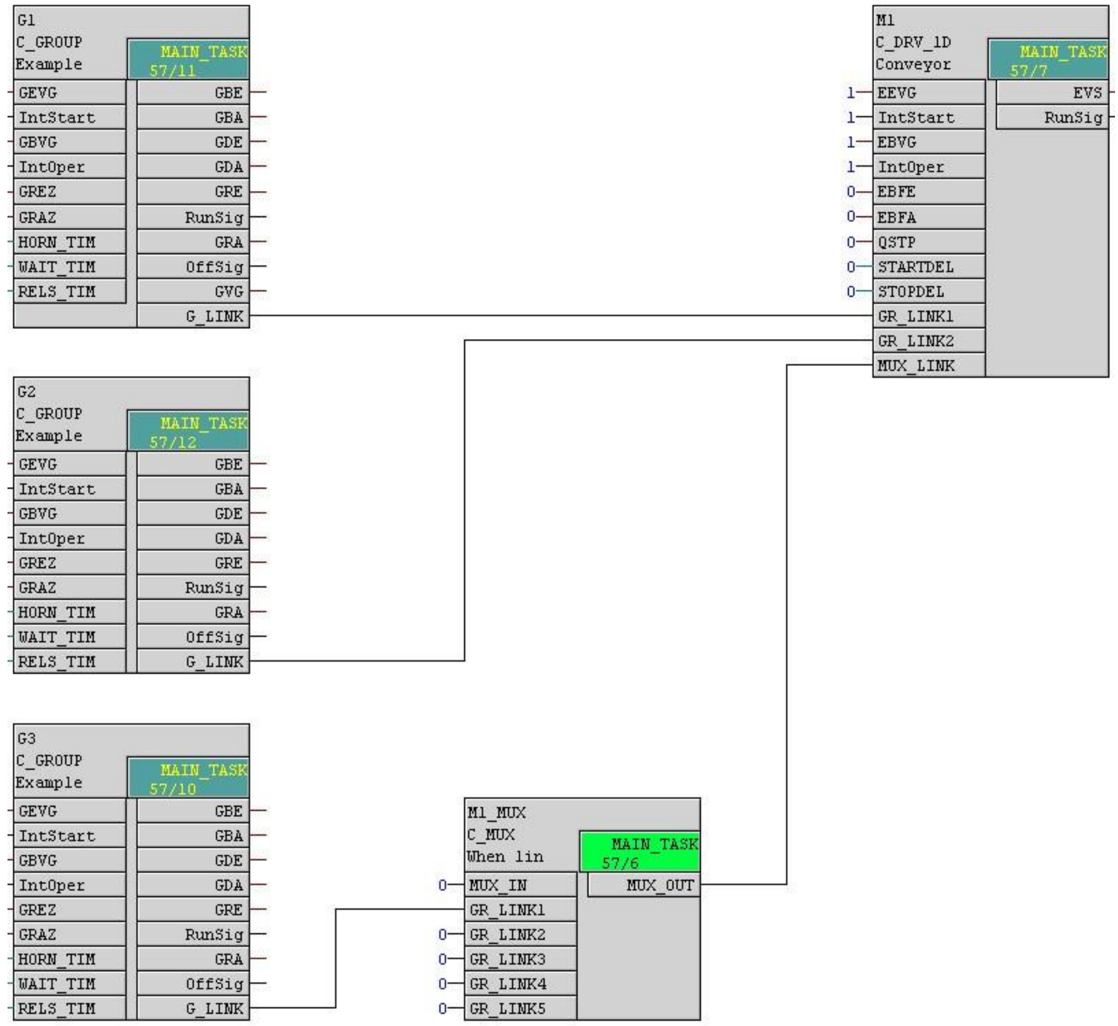
- Via the group link the group knows if one of the assigned objects has a problem. This is important for the summarizing indication (summarizing fault and summarizing warning in the block icon of the group) and for the status call.
- Dynamic faults interrupt the group start (GBE is reset).
- From Cemat V7.1 the group has a function to highlight all assigned objects in the process picture.

Connect output G_LINK of the group to input GR_LINK1 (or GR_LINK2) off all related objects (Drives, Annunciation blocks and Measured value blocks).



As an object may belong to more than one group, each object has two link inputs GR_LINK1 and GR_LINK2 and an additional input MUX_LINK for the connection to a multiplex block C_MUX. With this method the object can be assigned to an unlimited number of groups.

The main group should be connected to input GR_LINK1 of the object, because the button “Related Objects“ opens the faceplate of this group.



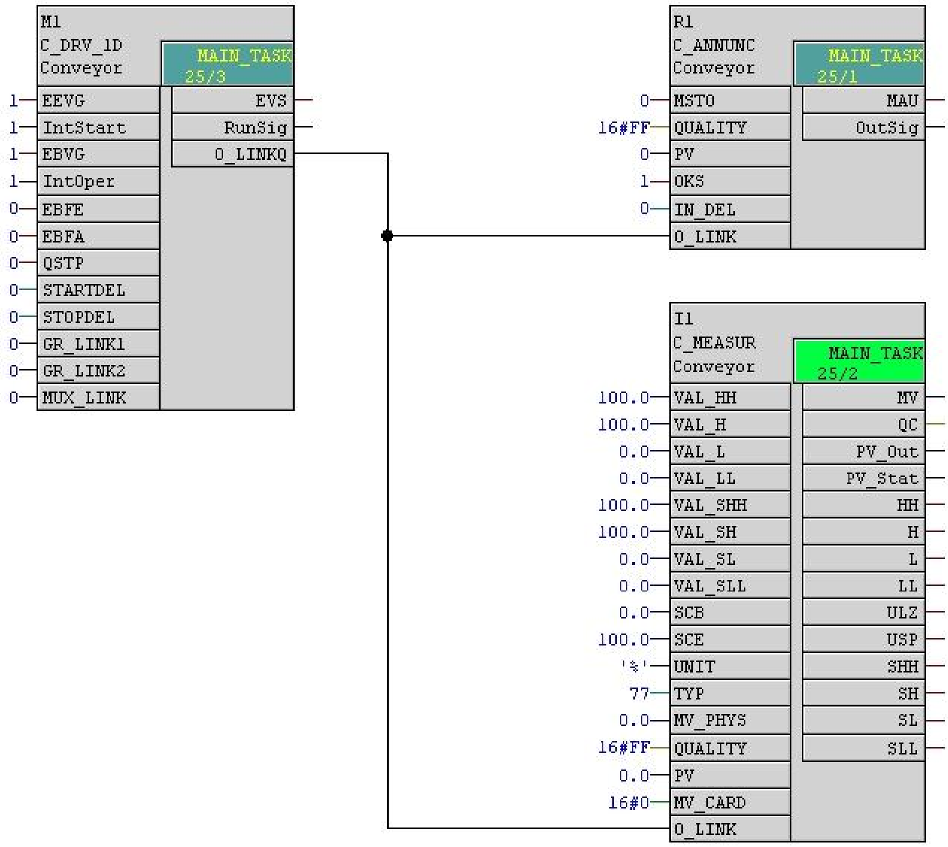
If multiplex blocks C_MUX are used the runtime sequence is crucial: The C_MUX must be called before the drive/annunciation/measured value.
In case of cascaded multiplex block the runtime sequence must be followed as well!
Refer to engineering tool "CEMAT_CheckTool.xlsm", which can be used for checking the availability of group links and the proper runtime sequence.

If the group has route modules, the output G_LINK of the group must be connected to input G_LINK of the route and output R_LINK of the route has to be connected to GR_LINK1/ GR_LINK2 of drives, annunciations and measured values.

Object Links

Since Cemat V8 SP1 it is possible to link annunciation blocks and measurements which belong to a motor via interface O_LINK directly to the drive block. In this case the group links GR_LINK1, GR_LINK2 and MUX_LINK must not be used!

- The drive block itself has an object browser function and is able to show a list of linked objects.
- Additionally the drive block transmits all information from the linked annunciation blocks and measured value blocks to the group, which interrupts the group start in case of a dynamic fault (or warning) and which is used for summarizing indication (summarizing fault and summarizing warning in the block icon of the group), object list and status call.



Caution: It is not allowed to use the Object link and the Group/Route link at the same time. If *O_LINK* is used, *GR_LINK1* and *GR_LINK2* or *C_MUX* must not be connected!

Acknowledgement mode

All dynamic faults, warnings and interlocks must be acknowledged.

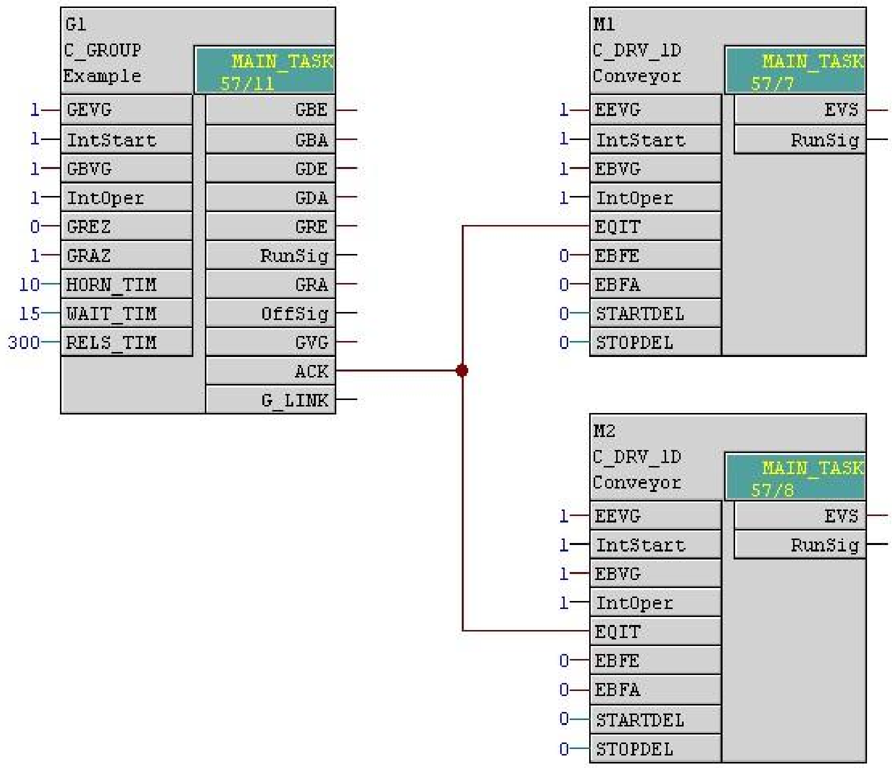
AS-wide Acknowledgement

Normally the fault acknowledgement goes together with the acknowledgement of the alarm message in the alarm line and faults are acknowledged for the complete AS. In case of an alarm message from AS1 in the alarm line, the acknowledgement will reset the faults of this AS (only the dynamic fault of the blocks, the messages must be acknowledged individually.)

This is the default setting in the system chart SYSPLCxx at block C_FB_PLC, if parameter ACK_GR and ACK_OB have value "0". In our opinion this is the most convenient method, because beside the acknowledgement of the alarm message no further action is required.

Group-wise acknowledgement

Beside this there is a possibility of acknowledgement per group. In this case, in the system chart SYSPLCxx at block C_FB_PLC the parameter ACK_GR must be changed to 1-Signal. Additionally the output ACK of the group must be connected to the acknowledgement inputs EQIT, KQT1, VQIT, MQIT and UQIT of all related objects.



Output ACK of the group is only set in this mode!

Object-wise acknowledgement

If in the setting in system chart SYSPLCxx at block C_FB_PLC the parameter ACK_OB is set to 1-Signal, the acknowledgement is be carried out per object, which means that via the acknowledgement of the message in the alarm line only the object which created the alarm is acknowledged.

This method has a disadvantage: For protection interlocks only the annunciation block creates the message but also the drive block which is stopped by protection interlock has a dynamic fault. In this case the drive fault must be acknowledged additionally via the drive faceplate.

Drive functions

Change of operation mode

Starting and stopping of drives can be carried out in three different operation modes. The most important operation mode is the **Automatic mode**. In the Automatic mode the drives are started and stopped via a group. All process interlocks are effective.

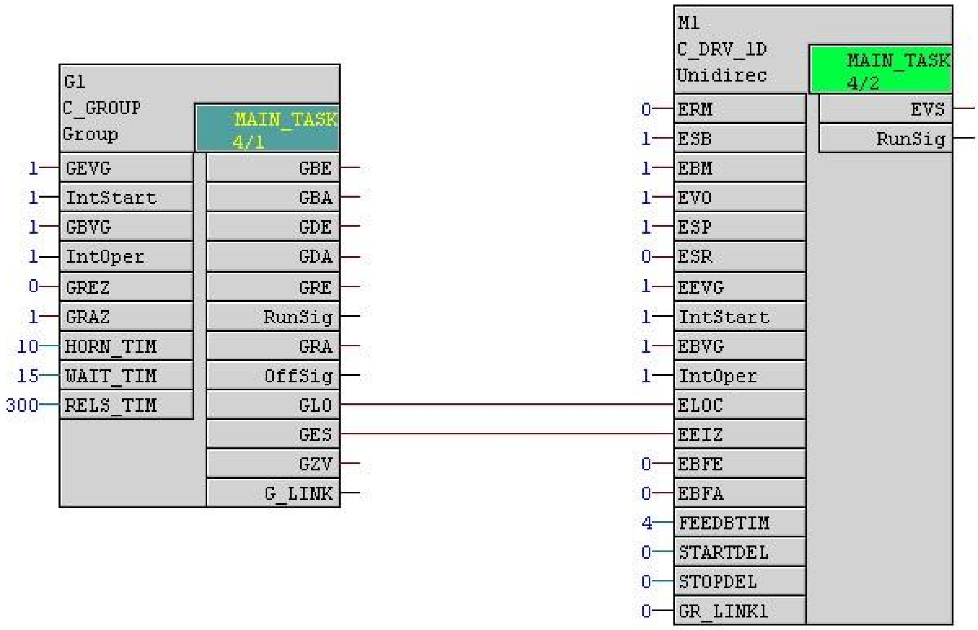
In **Local mode** the drives can be started and stopped via locally installed push-buttons. In this operating mode the process interlocks are not effective.

In **Single-start mode** the drives can be started and stopped via operator faceplate from HMI. Regarding process interlocks the behaviour is different in the individual project versions:

- In project version 004 the single-start mode is meant for maintenance only, and the process interlocks are therefore not effective.
- Im Project veraions 006 and 007 there is a possiblility of a non-interlocked single-start mode. But this mode must be explicitely released.

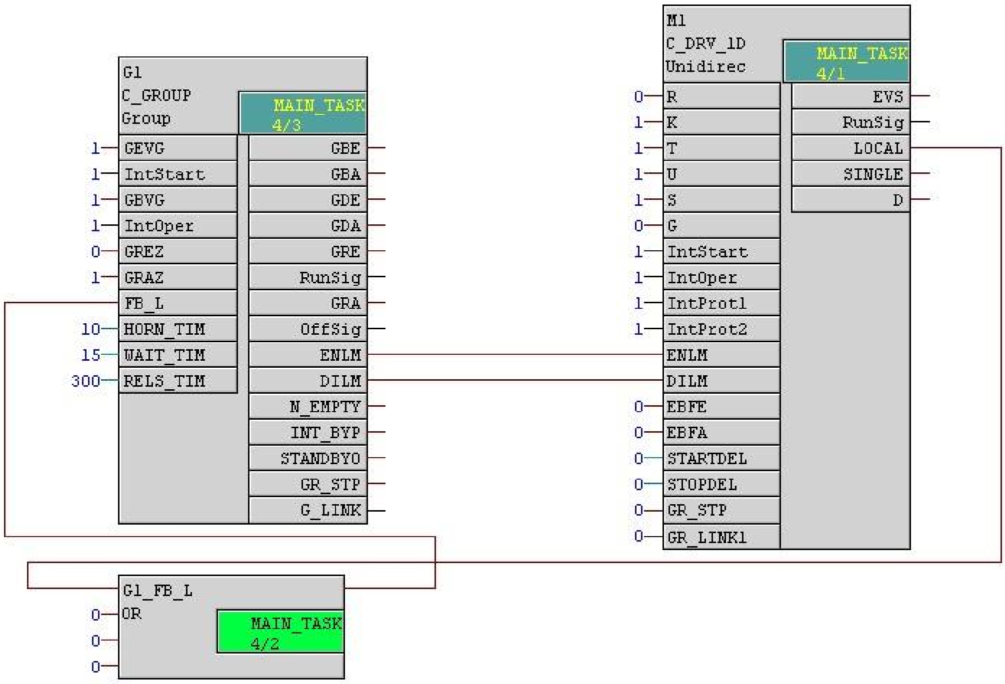
In some of the project versions (000, 007, 023, 024,025, 028) the operation mode is enabled by the group block, the faceplate of the group has buttons for operation mode change and the group block has outputs for enabling the corresponding operation mode of the drives.

Therefore the group output GES must be connected to drive input xEIZ and group output GLO must be connected to drive input xLOC.

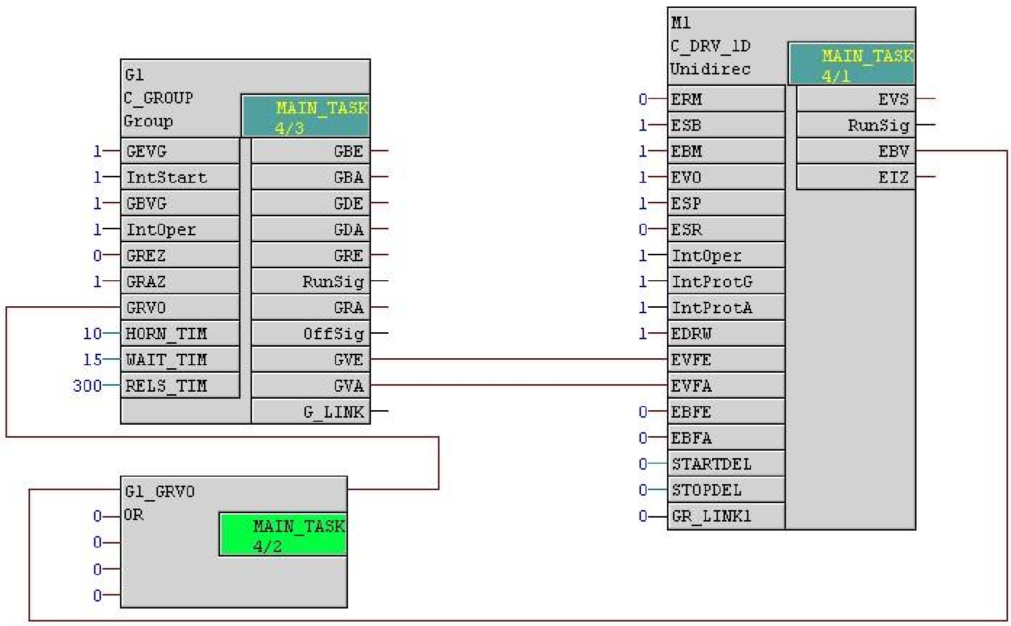


In all other project versions (004, 006, 026, 027) the operation mode can be changed individually via the drive faceplate. Some project versions additionally allow group-wise mode change from local to automatic (004, 002, 026) or from automatik to local (004, 026).

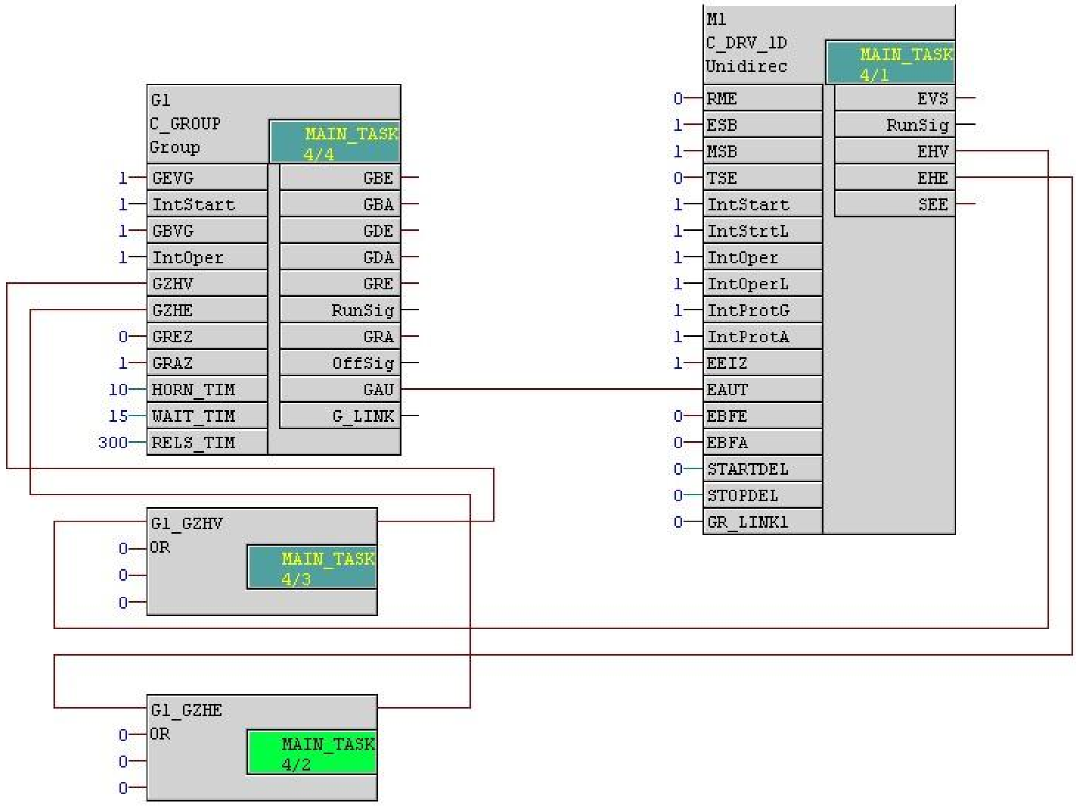
In **Project version 004** group output ENLM switches the drives into local mode and output DILM switches the drives back to automatic. Both outputs are only pulses and must be connected to the input of the drives with the same name.
The feedback to the group (LOCAL to FB_L) is used for the indication in the block icon and tells the operator that at least one drive is in local mode.



In **Project version 026** group output GVE must be connected to drive input xVFE and group output GVA must be connected to drive input xVFA. The outputs of the group are pulses and only used for switching over.
The feedback to the group (EBV to GRVO) is used for the indication in the block icon and tells the operator that at least one drive is in local mode.



In **Project version 006** the group has an output GAU (Pulse) which must be connected to input xAUT of the drives, in order to switch all drives of a group back into automatic mode. In case of a group with routes, the switch-back can be carried out per route. In this case output GAU of the group must be connected to input WAUT of the route and output WAU of the route must be connected to input xAUT of the drives.
The feedback to the group (EHV to GZHV) is used for the indication in the block icon and tells the operator that at least one drive is in ‘Manual interlocked’ mode.
The feedback to the group (EHE to GZHE) is used for the indication in the block icon and tells the operator that at least one drive is in ‘Manual non-interlocked’ mode.

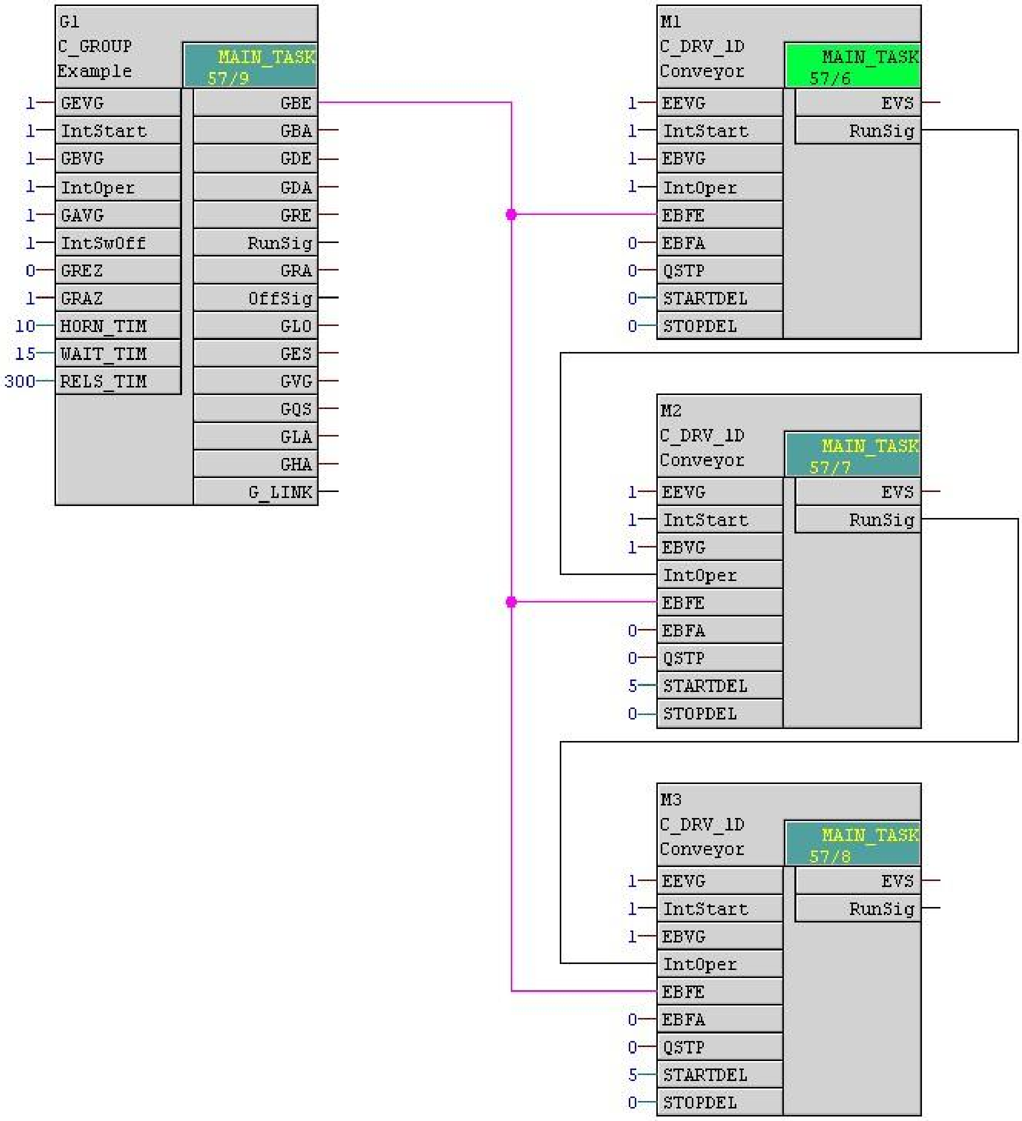


Start and Stop in Automatic mode

For the **Start** of the drives in automatic mode the output GBE of the group must be connected to input EBFE, KEB1, KEB2 or VBFE of the drives.

Output GBE remains "1" until the group is completely running. In case of a fault during the start-up of the group, output GBE is reset.

Output GBE is connected simultaneously to all drives. The start sequence is determined through interlocking among the drive. The start of the drive can be delayed additionally via parameter STARTDEL.

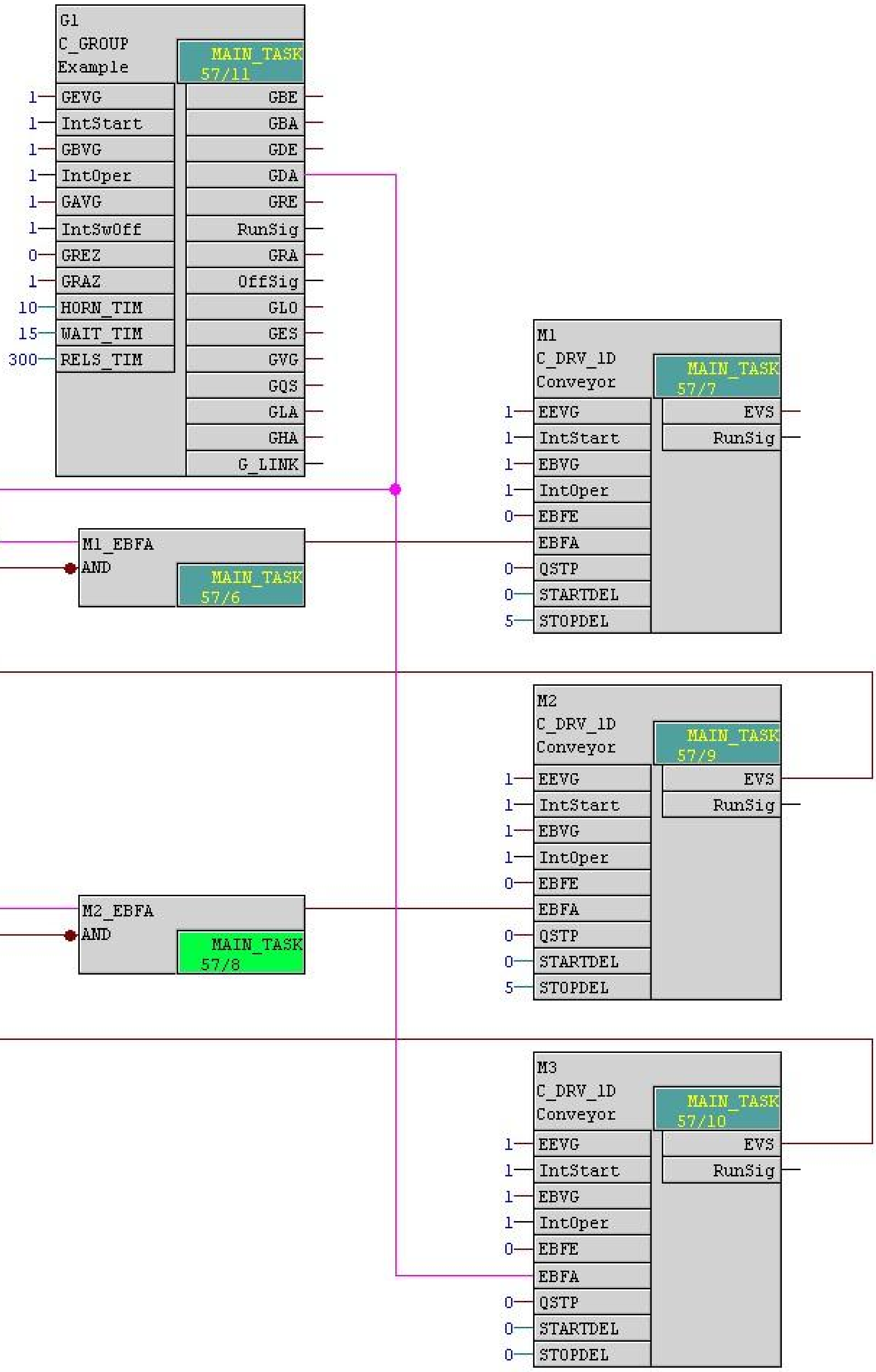


Output GDE of the group is not suitable as drive start command, because this signal is not reset in case of a fault. **The drive fault cannot be acknowledged as long a there is a continuous 1-Signal at the on command!!**

For the **Stop** of the drives in automatic mode the output GDA of the group must be connected to input EBFA or VBFA of the drives.

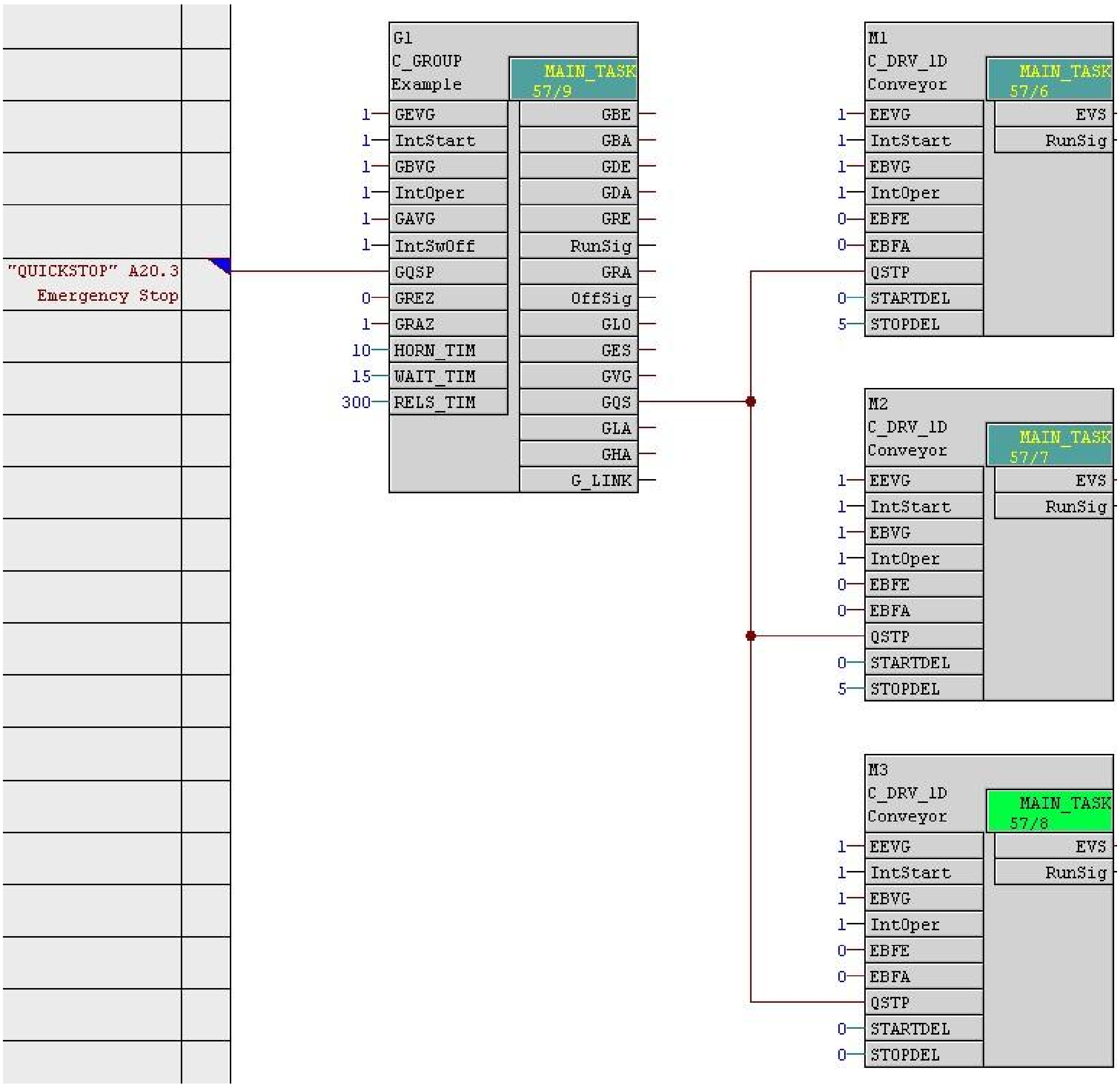
Output GDA is set with the group Stop and remains "1" until the group is completely stopped.

As the drives to not have a stop interlocking, for a sequential stop the stop condition (feeding drive is stopped) must be connected together with the group output GDA to the drive. The drive stop can additionally be delayed via parameter STOPDEL.



Group Output GBA (Command Off) is only a pulse and normally not suitable for stopping the drives (if a stop delay is used output GBA does not stay long enough).

For the **Quick stop** of the drives the group output GQS must be connected to input QSTP of the drives. In case of a Quick Stop the drive is stopped immediately (without considering the Stop delay).



Start and Stop in Single-start mode

For start and Stop in single-start mode no additional connections are required. Only the permission must be given.

In some of the project versions the change into single-start mode can be inhibited by switching the faceplate buttons of the drive to invisible. This can be achieved in project version 004 and 026 via interface EN_SINGL and in Project version 006 via interface EEIZ.

At this point we would like to mention again the non-interlocked single-start mode of project versions 004, 006 and 007.

- In Project version 004 the single-start mode is always non-interlocked.
- In Project version 006 the drive faceplate has a button fort he so-called non-interlocked mode. In this case only the fix start interlocks and operation interlocks are effective. Some of the start interlocks and operation interlocks will be bypassed.
- In Project version 007 there is also a possibility for a non-interlocked single-start mode, but it needs a special permission and must additionally be enabled.

Start and Stop in Local mode

In local mode the start and the stop of the drives is carried out via a local switch. This can either be a switch or start and stop buttons, located beside the device.

Local switches differ a lot and are described separately. In most cases, the signals of the local switch go into the PLC; which means, also in local mode the drive is started and stopped by the SIMATIC.

In old installations, it also happens that the signals from the local switch controls the drive directly, which allows the local mode even if the SIMATIC is not running. In this case a signal is needed which tells the SIMATIC that the drive is in local mode, otherwise the local start leads to a feedback fault.

Variants of Local switches:

1. Switch with positions *Remote – Local Stop – Local – Local Start*
   The position 'Local Start' is non-locking, which means after starting the switch goes back into position 'Local'. Local start is a make contact (normally open) and local stop is break contact (normally closed).
2. Switch with position *Local Stop – Remote – Local Start*
   The position 'Local Start' is non-locking, which means after starting the switch goes back into position 'Automatic'. Local start is a make contact and local stop is break contact.
3. Buttons for Start and Stop, where the start button is a make contact and the stop button is a break contact.
   An additional input permits the local mode (for the mode change from automatic to local).
4. Buttons for Start and Stop, where the start button is a make contact and the stop button is a break contact.
   The permission of the local mode is exclusively given through HMI.
5. Buttons for Start and Stop, where the start button is a make contact and the stop button is a break contact.
   An additional repair switch (with locks) is effective in all operation modes and inhibits the drive start during maintenance action (Signal 'Local Isolated'). The permission of the local mode is exclusively given through HMI.
6. Start button (make contact). An additional local switch is effective in all operation modes and inhibits the start of the drive.
   The local mode is always enabled or must be enabled exclusively through HMI.
7. Start button (make contact). The local mode is an inching mode, which means the motor remains started as long as the button is pressed.
   The permission of the local mode is exclusively given through HMI.

Regarding local mode the project versions are different, which means depending on the local switch which is used one project version is more or less suitable.

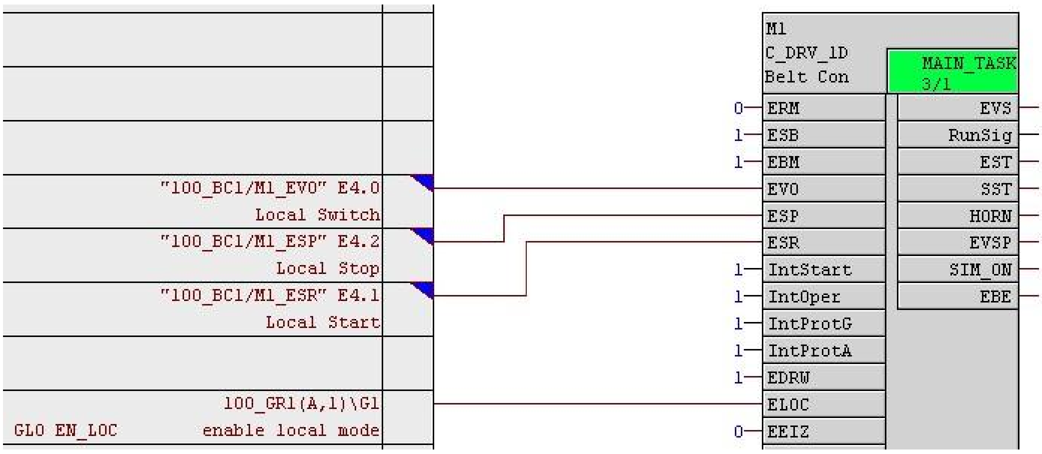
Probably other variants of local switches exist. In case you have any doubts and can not decide which project version fits best, please contact the Cemat Competence Center.

On the following pages you find examples for the different project versions:

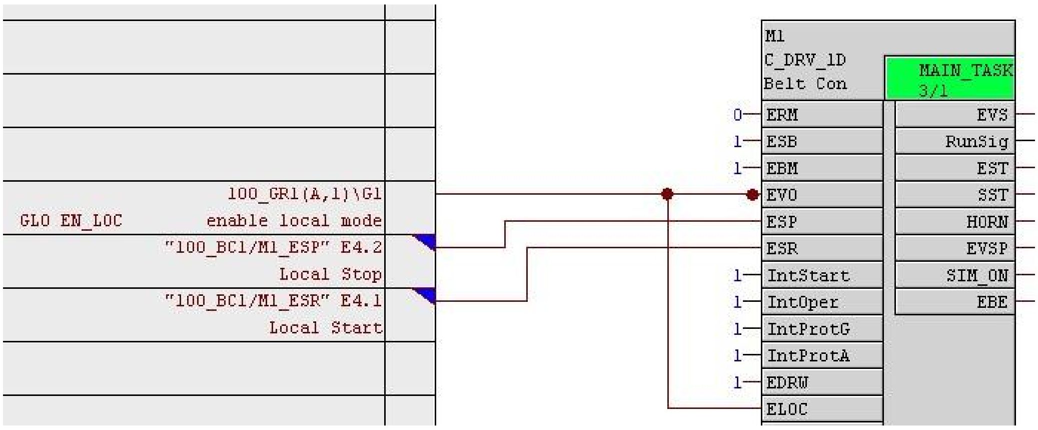
Project versions 000 and 028:

ESR	Local Start (normally open)
ESP	Local Stop (normally closed)
EVO	Local Selection Switch (1 = Remote; 0 = Local)
ELOC	Enable local mode via HMI

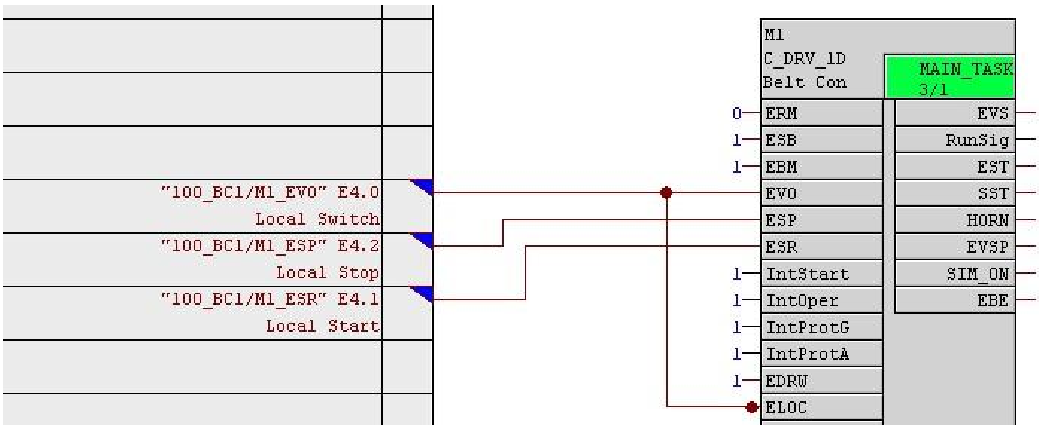
The Project versions 000 and 028 can be used for variant 1, variant 3 and variant 4.
The Local mode is enabled per group. Additionally the switch EVO must be in position Local.



If no additional switch for the selection between Remote/Local is available, the input EVO must be simulated using inverted group enable GLO.



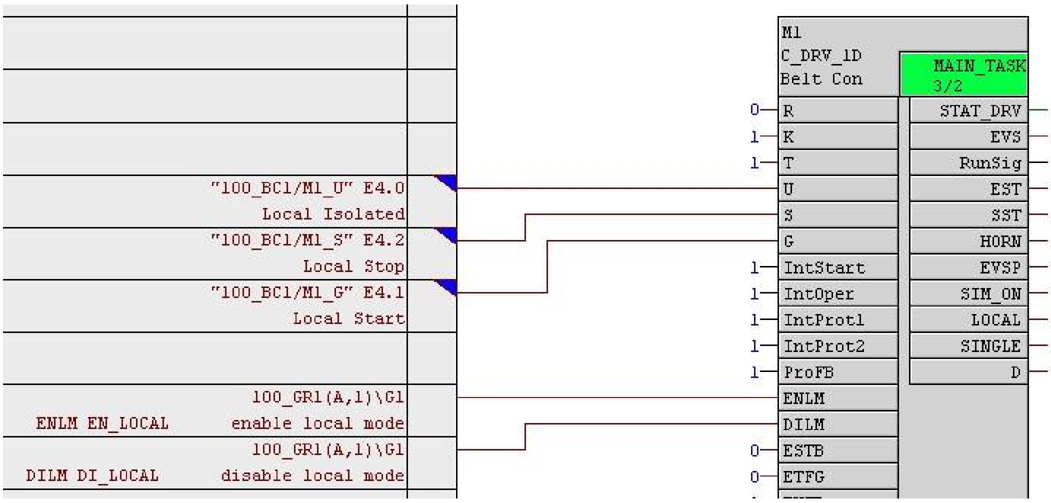
If the enable via HMI shall not be used, the input ELOC must be simulated by using inverted EVO signal.



Project version 004:

G	Local Start (normally open)
S	Local Stop (normally closed)
U	Local repair switch (Local Isolated) must always be “1“.
ENLM	Pulse for change into local mode
DILM	Pulse for change into automatic mode

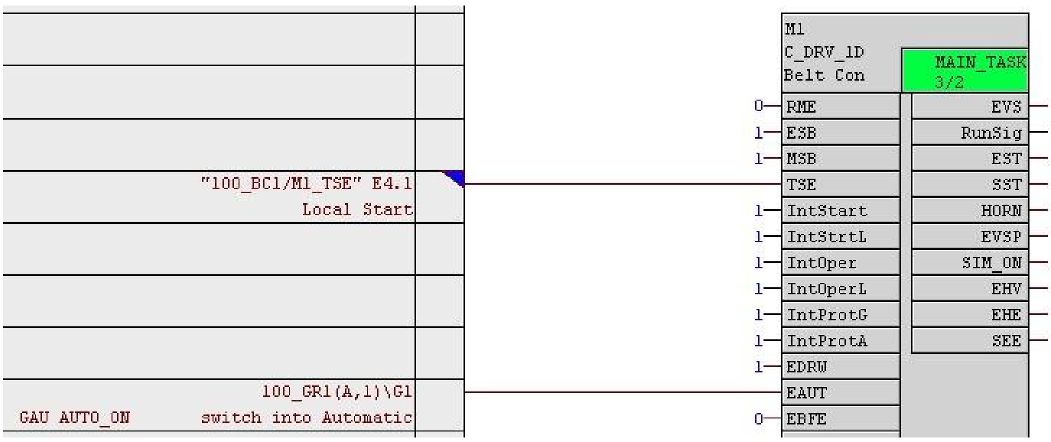
The Project version 004 is suitable for variant 5.
The local mode is enabled via the faceplate of the drive or the faceplate of the group and can only be given when the drive is not running.
Change into automatic mode can be carried out via the faceplate of the drive or via the faceplate of the group as well as automatically during group start.



Project version 006:

TSE	Local Start (normally open)
EAUT	Switch back into automatic mode

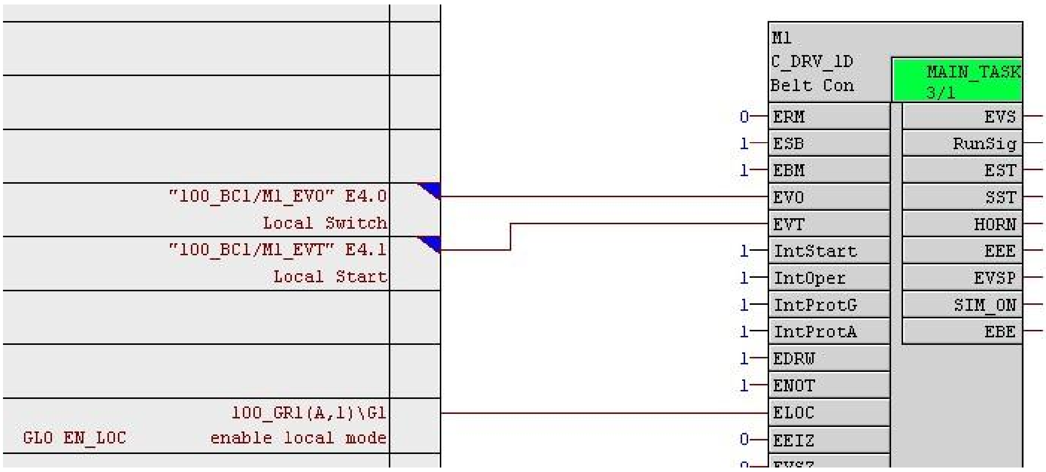
The project version 006 is suitable for Variant 7.
The local mode is enabled via the faceplate of the drive. Switching back into automatic mode can be carried out via the faceplate of the drive and also via the faceplate of the group (or the route).



Project version 007:

EVT	Local Start (normally open)
EVO	Local switch (must always be “1“)
ELOC	Enable local mode via HMI

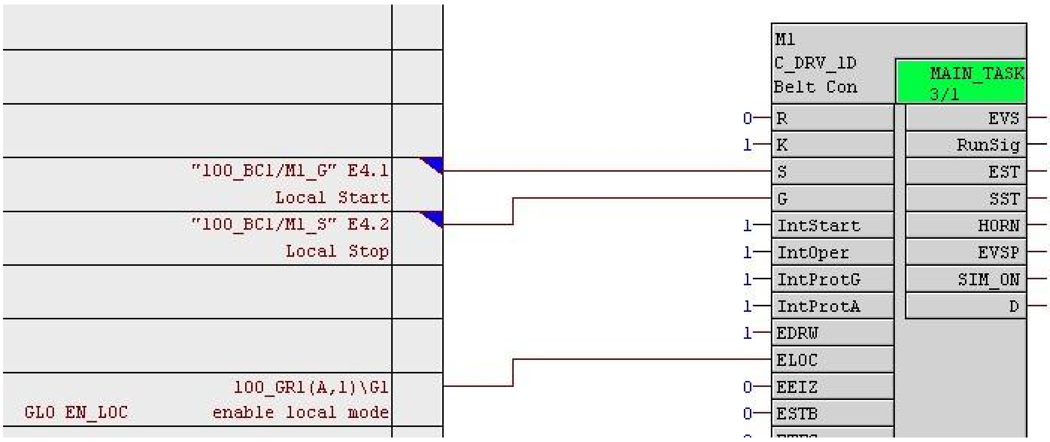
The Project version 007 is suitable for variant 6.
The local mode is either always enabled (ELOC is preset with “1“) or can be enabled by the group.



Project version 023:

G	Local Start (normally open)
S	Local Stop (normally closed)
ELOC	Enable local mode via HMI

The Project version 023 is suitable for variant 4.
The local mode is enabled group-wise (through the group faceplate).



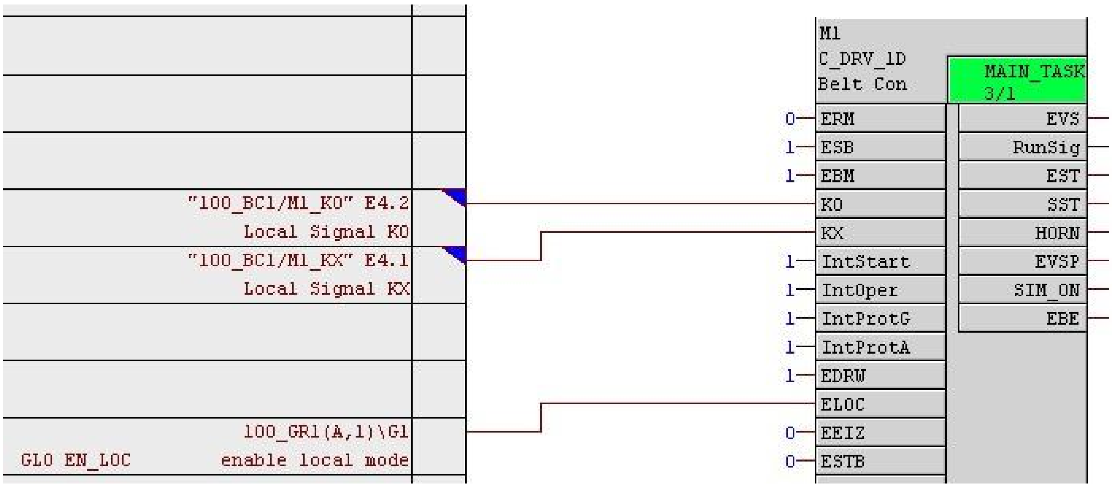
Project version 024:

KX Local input KX
K0 Local input K0
ELOC Enable local mode via HMI

The project version 024 is suitable for variant 1, but it requires a special matrix. In this matrix 4 different situations are simulated with only two signals.

Switch position	K0	KX
Automatic	1	0
Local Stop	0	0
Local	0	1
Local Start	1	1

The local mode is enabled group-wise (through the group faceplate).



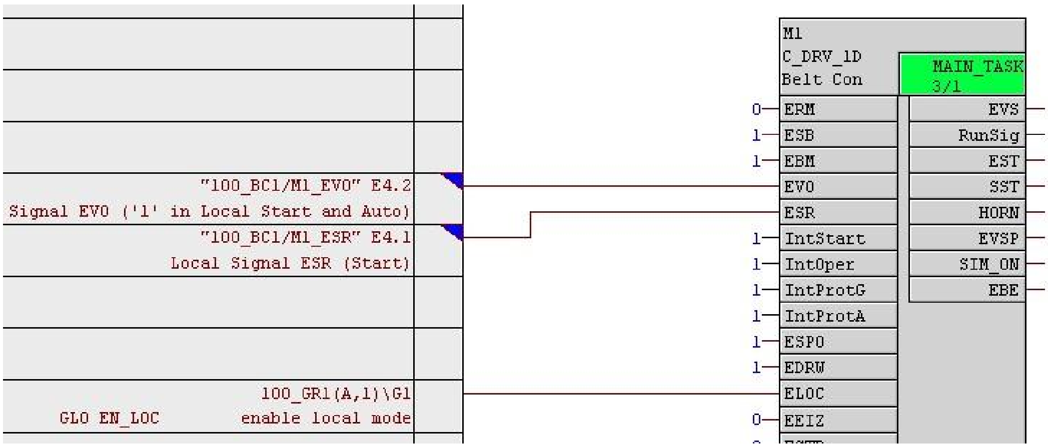
Project version 025:

EVO Local input EVO
ESR Local input ESR
ELOC Enable local mode via HMI

The project version 025 is suitable for variant 2, but it requires a special matrix. In this matrix 3 different situations are simulated with only two signals.

Switch position	EVO	ESR
Automatic	1	0
Local Stop	0	0
Local Start	1	1

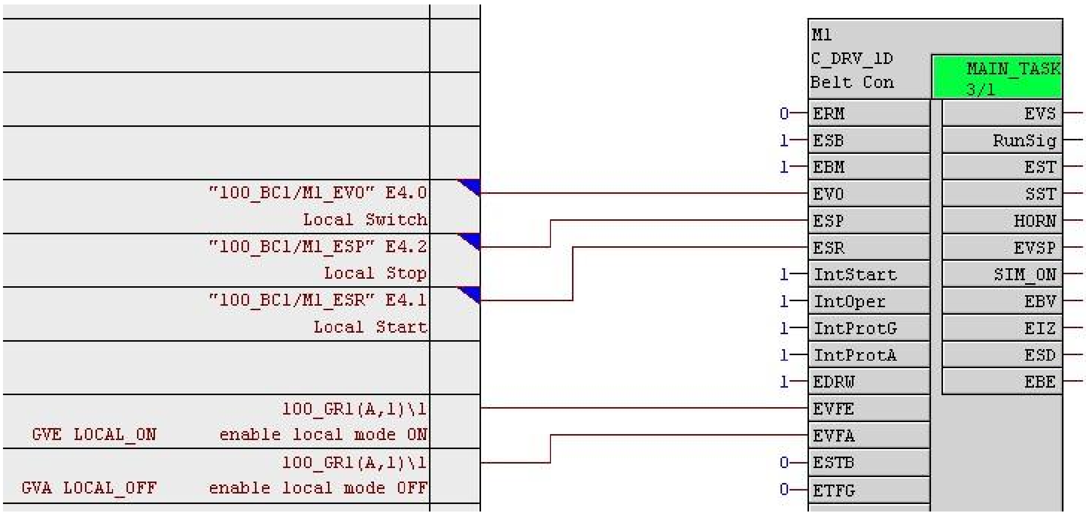
The local mode is enabled group-wise (through the group faceplate).



Project version 026:

ESR	Local Start (normally open)
ESP	Local Stop (normally closed)
EVO	Local repair switch (Local Isolated) must always be “1“.
EVFE	Pulse for change into local mode
EVFA	Pulse for change into automatic mode

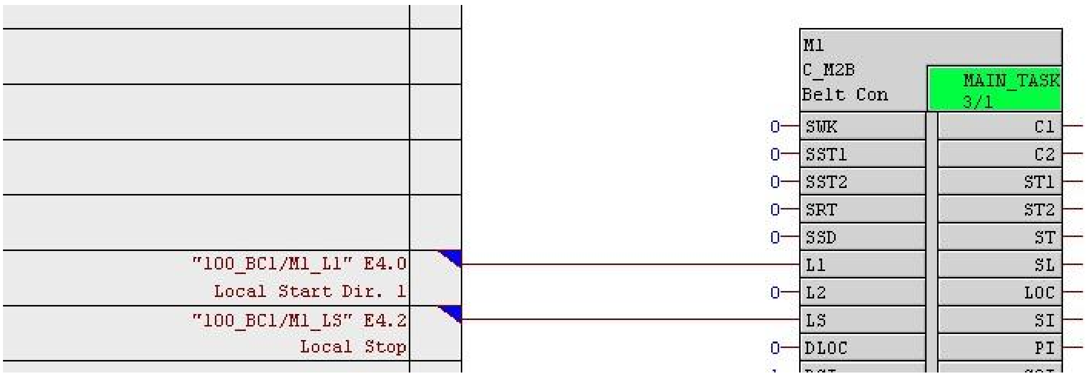
The Project version 026 is suitable for Variant 5
The local mode is enabled via the faceplate of the drive or the faceplate of the group and can only be given when the drive is not running.
Change into automatic mode can be carried out via the faceplate of the drive or via the faceplate of the group as well as automatically during group start.



Project version 027:

L1	Local Start (normally open)
LS	Local Stop (normally closed)
DLOC	Enable local mode

The Project version 027 is suitable for Variant 4.
Local mode is enabled via the faceplate of the drive or via interface DLOC.



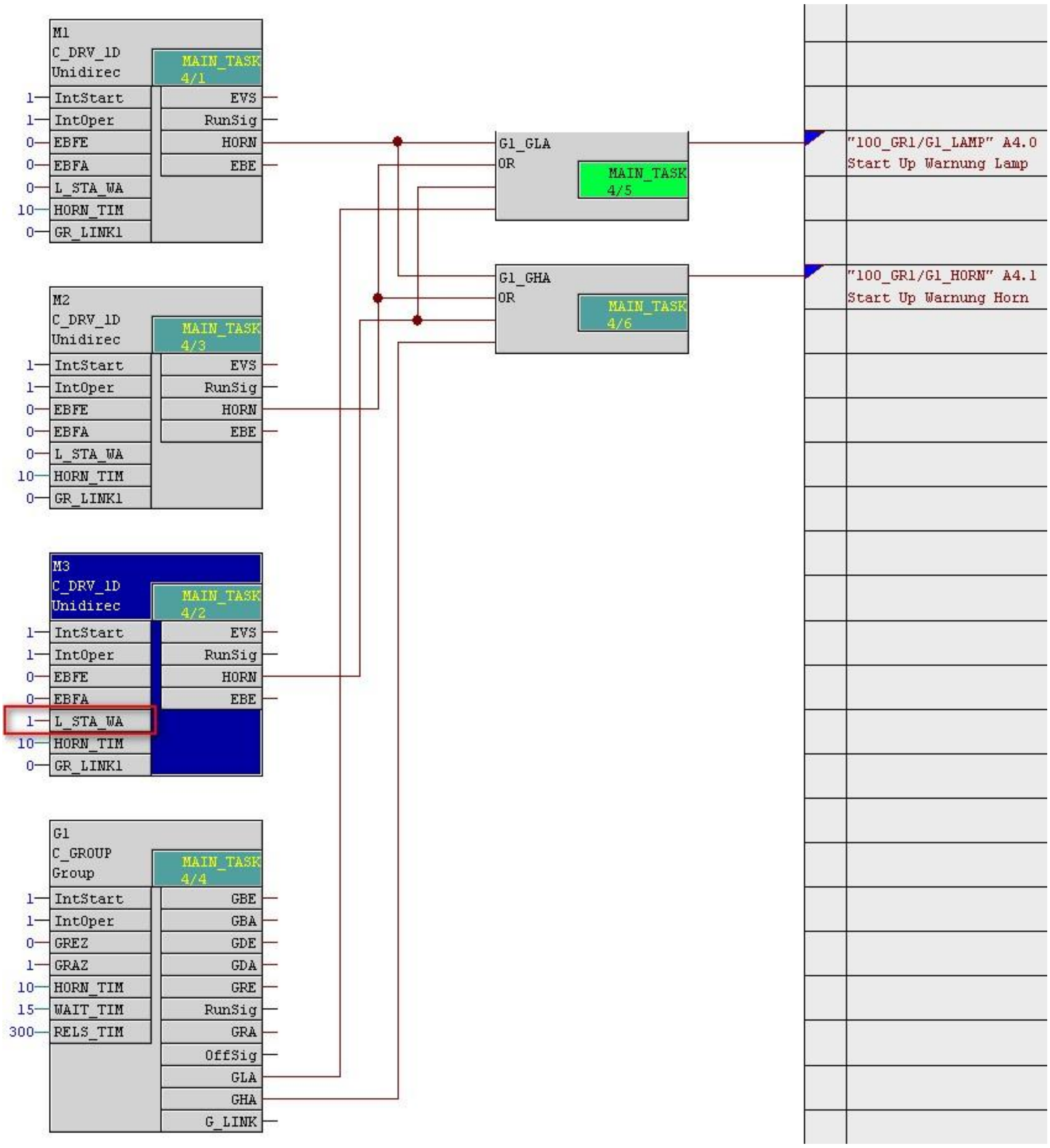
Caution: If DLOC is connected the mode change via drive faceplate is no longer possible!

Start-up warning in Single-start mode (and Local mode)

In the automatic mode the start-up warning is carried out through the corresponding group. However, if the drive is started in single-start mode then at least in remote it must be insured that before drive start an acoustic alarm is created.

This is done via output HORN of the drive blocks. The duration of the start-up warning can be set via process parameter HORN_TIM. The contactor command is only given after this time has elapsed.

Connect output HORN of the drives together with the start-up warning signal of the group to the corresponding hardware outputs.



If the start-up warning shall be given also in local mode, you have to change parameter L_STA_WA of the drive to 1-Signal. See drive M3.

Caution: In local mode the start button must remain pressed until the start-up warning is completed.

Sporadic mode

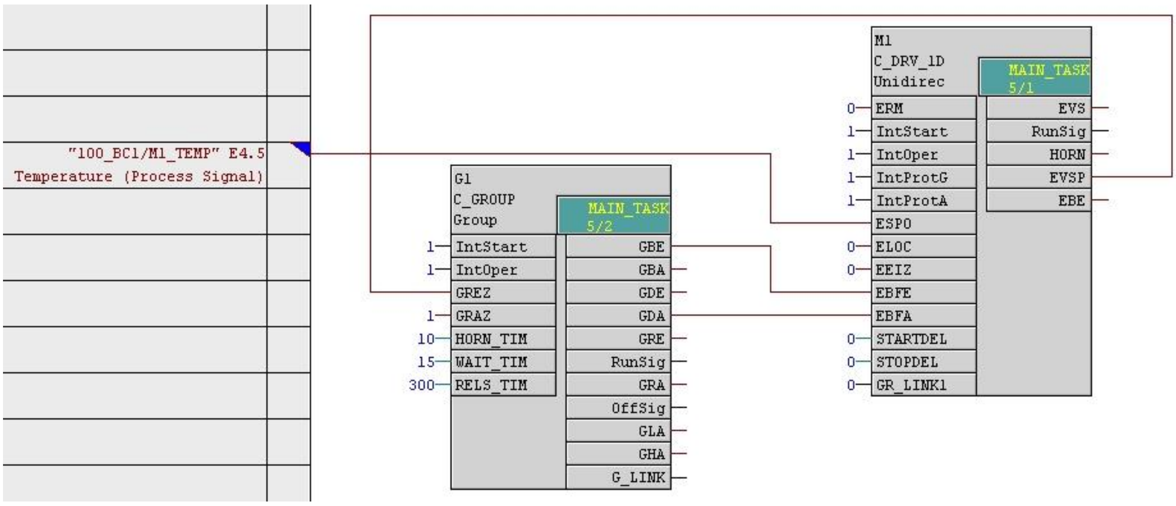
Concerns Unidirectional drive, Bi-directional drive and Valve

Sometimes it is required to start and stop drives dependent on a process signal or time controlled. This concerns devices for heating, cooling, lubrication or valves for silo aeration.

In this case the drives have to be activated via group start signal GBE or via route start signal WBE and can then be switched on and off via the status at interface ESPO.
0-Signal at interface ESPO stops the motor without reset of the command memory EKS. The motor remains activated and starts automatically with 1-Signal at this interface.

To switch off the drive completely it needs a 1-Signal at EBFA or 0-Signal at the operation interlock (EBVG or IntOper). If the drive has been stopped by fault the group must be started again.

For the feedback to the group signal EVSP must be used. EVSP has 1-Signal, as soon as the drive is activated.



This interface is only effective in automatic mode. In Single-start mode or in Local mode ESPO is not considered. In case of a mode change the behaviour is as follows:

- If a drive which is running in automatic mode is switched to single-start mode, it will then run continuously.

- If a drive which is running in single-start mode is switched into automatic mode, the signal status of ESPO is considered. If ESPO = 0-Signal the drive is stopped (EKS is reset). If ESPO = 1-Signal the drive will proceed in sporadic mode.

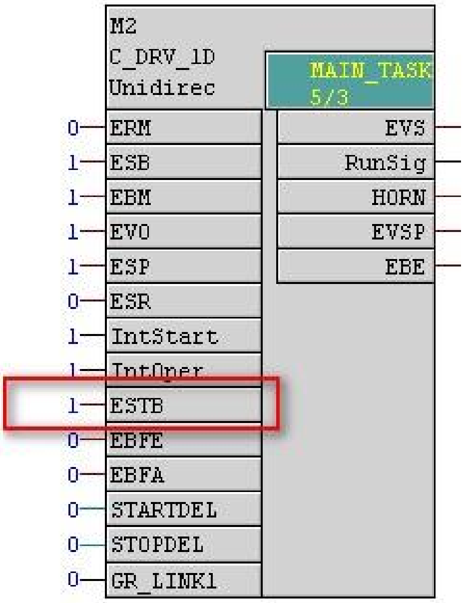
Caution: Modified function for project version 007
- If a drive which is running in automatic mode is switched into single-start mode it will continue to run sporadically until the drive gets stopped.
- If the drive was started in single-start mode or in local mode, ESPO is not considered, the drive is running continuously.
- If a drive which was started in single-start mode is switched into automatic mode, at the moment of the mode change the signal status of ESPO is considered: If ESPO = 0-Signal the motor will be stopped (EKS is reset). If ESPO is 1-Signal the drive continues to run in sporadic mode.

Drive in Standby (Alarm message also for non-running device)

If a drive is not started or if it is in local mode, in case of a fault no dynamic fault and no alarm message is created. This is important in order to avoid alarms during maintenance, or if the plant section is not in operation.

However, in some occasions it might be required to get the alarms, even if the drive is not running at this moment, e. g. if the drive has a 'Standby-Function'.

Through 1-Signal at interface ESTB (VSTB, KSTB) also for a non-running device alarm messages are created, which must be acknowledged by the operator.



Protection interlocks

Protection interlocks are interlocking conditions which must stop the drive in case of a fault, in order to protect machines or persons.

In case of a missing protection interlock the drive indicates 'faulty' (red indication), similar to normal drive faults (static fault if the drive was not running, dynamic fault if the drive was stopped through protection interlock.)

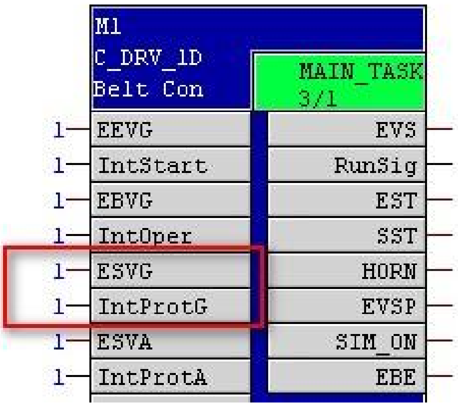
Unlike for normal drive signals as e. g. electrical availability or overload, the drive block does not create alarm messages for protection interlocks. An annunciation block has to be programmed for this purpose. See chapter 'Drive fault annunciations'.

Always active protection interlocks

In all drive blocks you find 2 interfaces which are effective in all operation modes. If one of these interfaces has 0-Signal the drive will be stopped.

The interface ESVG (VSVG, KSV1, KSV2) has format BOOL and can be used for connecting binary signals (e. g. output MAU of the annunciation block or an output of the binary interlock blocks C_INTERL or C_INTER5).

The interface IntProtG (from Cemat V7.1) has STRUCTURE format and can be connected with the structure output OutSig of the annunciation block or with a structure interlock block, e. g. Intlk02. If the structure interface is used, the connected block can directly be opened via the faceplate of the drive.

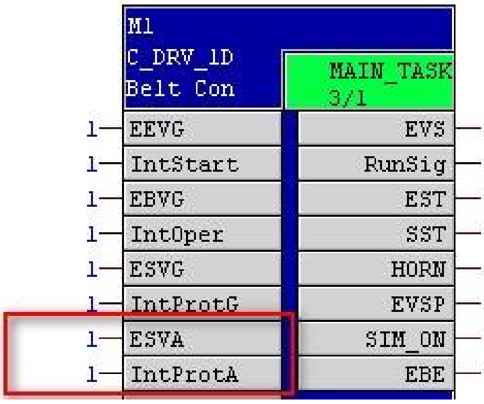


Protections only active in Automatic mode and Single-start mode

For Unidirectional drives and Bidirectional drives you additionally find two interfaces which are effective only in automatic mode and Single-start mode. These interfaces can be used for signals like Drift switches, where the local start must be permitted.

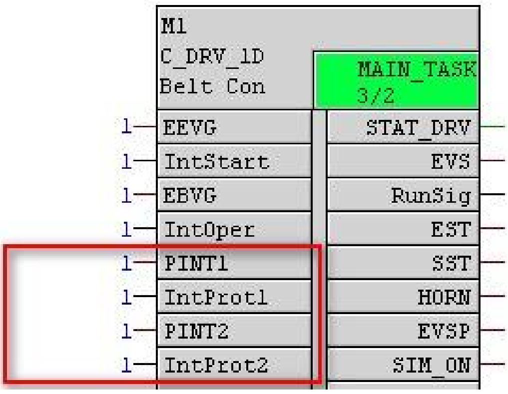
The interface ESVA has format BOOL and can be used for connecting binary signals (e. g. output MAU of the annunciation block or an output of the binary interlock blocks C_INTERL or C_INTER5).

The interface IntProtA (from Cemat V7.1) has STRUCTURE format and can be connected with the structure output OutSig of the annunciation block or with a structure interlock block, e. g. Intlk02. If the structure interface is used, the connected block can directly be opened via the faceplate of the drive.



In **Project version 004** the binary interfaces are called PINT1, PINT, PINTX and PINTY, the structure interfaces are called IntProt1, IntProt, IntProtX and IntProtY. These interfaces are effective in all operation modes.

The Unidirectional drive and the Bi-directional drive have an additional binary interface PINT2 and an additional structure interface IntProt2, which can be by-passed in local mode with the local start button. In this case the motor will run only as long as the start button is pressed.



Speed monitor supervision

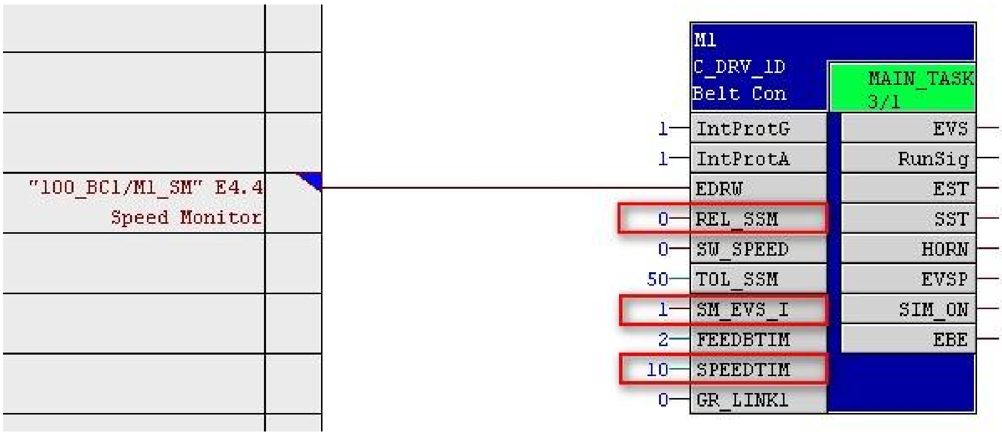
Speed Monitor signals are detected by the drive block and the drive block also creates the alarm message. This is valid for all project versions except 004. An example for project version 004 is described behind.

Input 'Speed Monitor' can be a continuous signal or a pulse. The drive block can handle both types. In both cases the process parameter SPEEDTIM must be set to the time after which the process feedback supervision has to be activated.

Via parameter SM_EVS_I you can decide whether the signal 'motor running' is created after the supervision time SPEEDTIM or if signal 'motor running' should immediately be given with the process feedback.

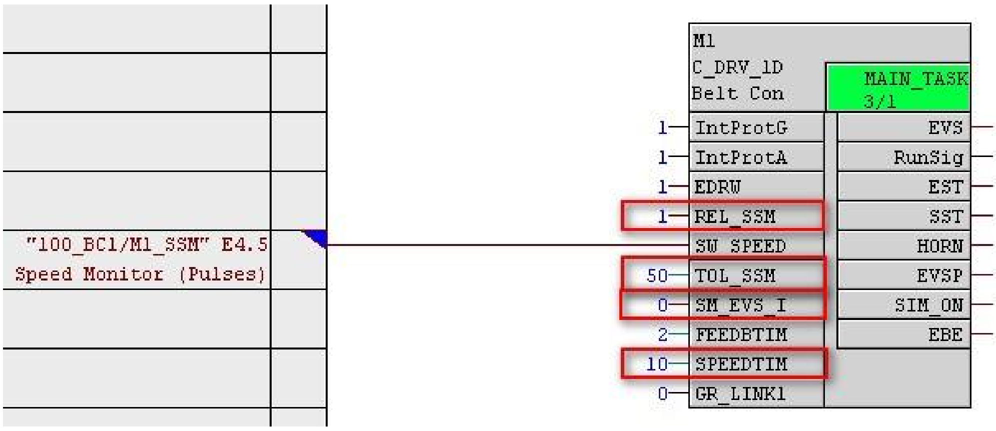
Some examples with explanations:

1. The process feedback is a continuous signal. The signal must be connected to interface EDRW and parameter REL_SSM must have 0-Signal.
   The feedback supervision time SPEEDTIM is set to 10 seconds and parameter SM_EVS_I is configured in a way that outputs EVS and RunSig are immediately set when the process feedback is true.



2. The process feedback signal is a pulse. The signal must be connected to interface SW_SPEED and parameter REL_SSM must have 1-Signal. The feedback supervision time SPEEDTIM is set to 10 seconds and parameter SM_EVS_I is configured in a way that outputs EVS and RunSing are set after the feedback supervision time.

   The tolerance value for the detection of pulses TOL_SSM is preset to 50 cycles, which means if after 50 OB1 cycles no signal change at input SW_SPEED is detected, the motor is switched off with 'Speed monitor fault'.



In **Project version 006** the drive block has an additional interface EVQD for the acknowledgement of the speed monitor fault. The speed monitor must be acknowledged from local. Via the normal fault acknowledgement the speed monitor fault cannot be reset.

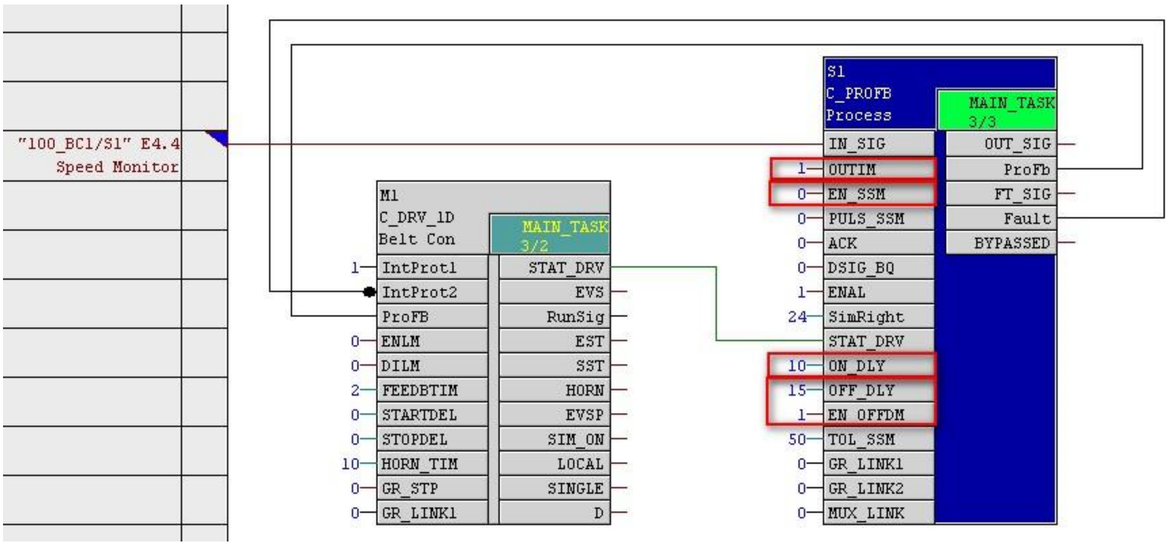
In **Project version 004,** block C_PROFB is used for process feedback supervision and alarm message. Also this block has two options, the supervision of a continuous signal or the supervision of pulses

Via connection of drive output STAT_DRV to input STAT_DRV of block C_PROFB the process feedback block receives the status of the drive.
The output 'ProFb' of block C_PROFB transmits the process feedback to the motor and output 'Fault' of block C_PROFB must be used as protection interlock in order to switch off the drive in case of a fault.

Dependent on the function, different options exist:

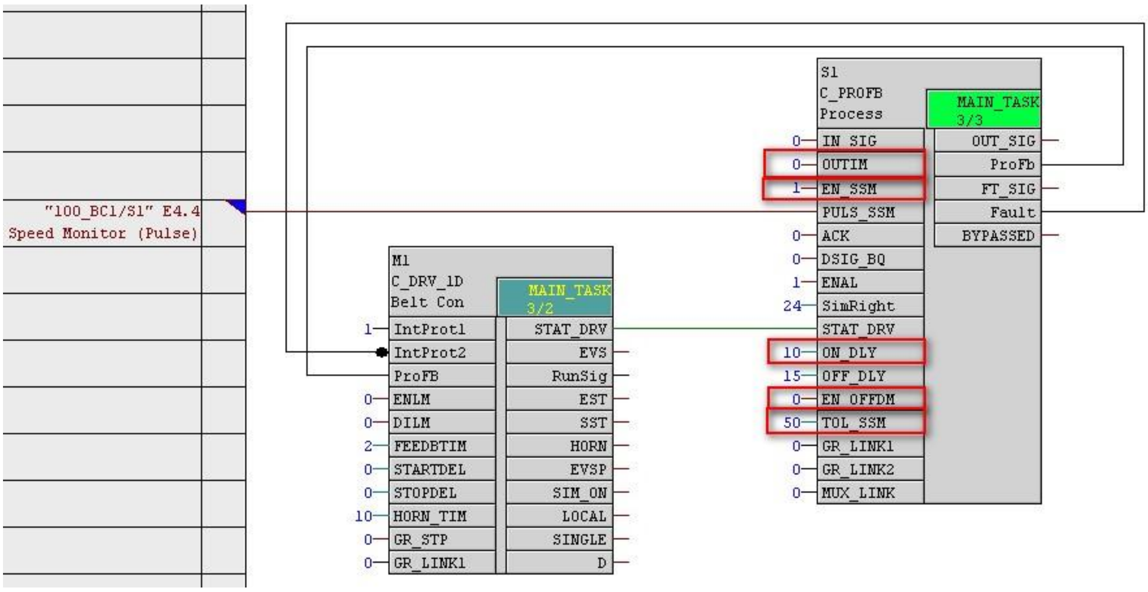
1. The process feedback is a continuous signal. The signal must be connected to interface IN_SIG and parameter EN_SSM must have 0-Signal.
   The feedback supervision time during start-up ON_DEL is set to 10 seconds and parameter OUTIM is configured in a way that outputs EVS and RunSig are immediately set when the process feedback is true.

   Via parameter EN_OFFDM the supervision during switch-off is activated and parameter OFF_DLY is set to 15 seconds.



2. The process feedback signal is a pulse. The signal must be connected to interface PULS_SSM and parameter EN_SSM must have 1-Signal. The feedback supervision time during start-up ON_DEL is set to 10 seconds and parameter OUTIM is configured in a way that outputs EVS and RunSing are set after the feedback supervision time.

   The tolerance value for the detection of pulses TOL_SSM is preset to 50 cycles, which means if after 50 OB1 cycles no signal change at input PULS_SSM is detected, the motor is switched off with 'Process feedback fault'. The supervision during switch-off is deactivated.



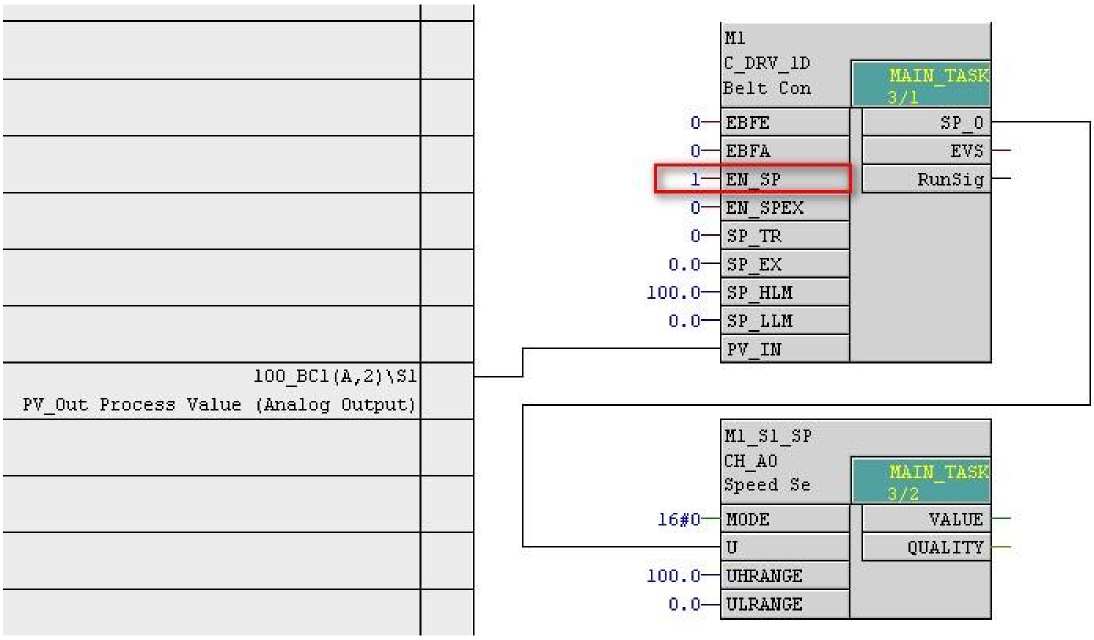
Set point input

A set point, as e. g. the speed can directly be entered via the drive faceplate and will be transmitted to output SP_O of the drive block. From there the value can be connected to an analog output or to another block.

In order to display the set point input field in the drive faceplate the function must be enabled via interface EN_SP.

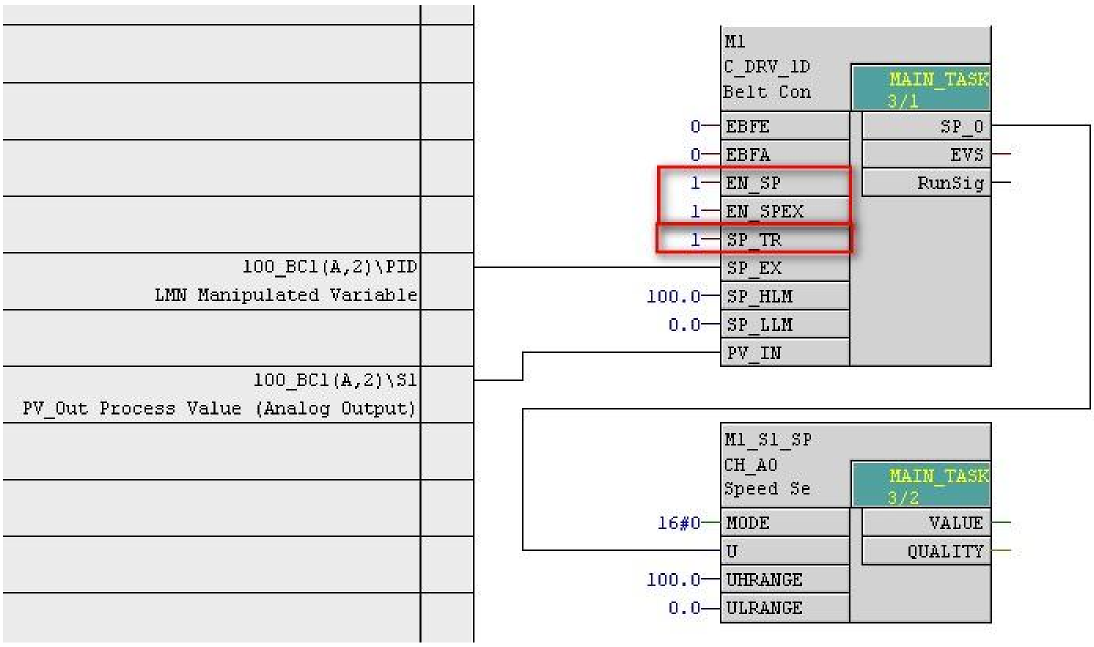
The set point limits are configured via parameter SP_HLM and SP_LLM. The default unit 'rpm' can be modified via signal attributes.

The actual value (e.g. actual speed) must be connected to structure input PV_IN.



If the set point is given by a program, e. g. by a primary controller, this function must be enabled via interface EN_SPEX and the external set point must be connected to input SP_EX.

For set point tracking SP_TR must have 1-Signal.

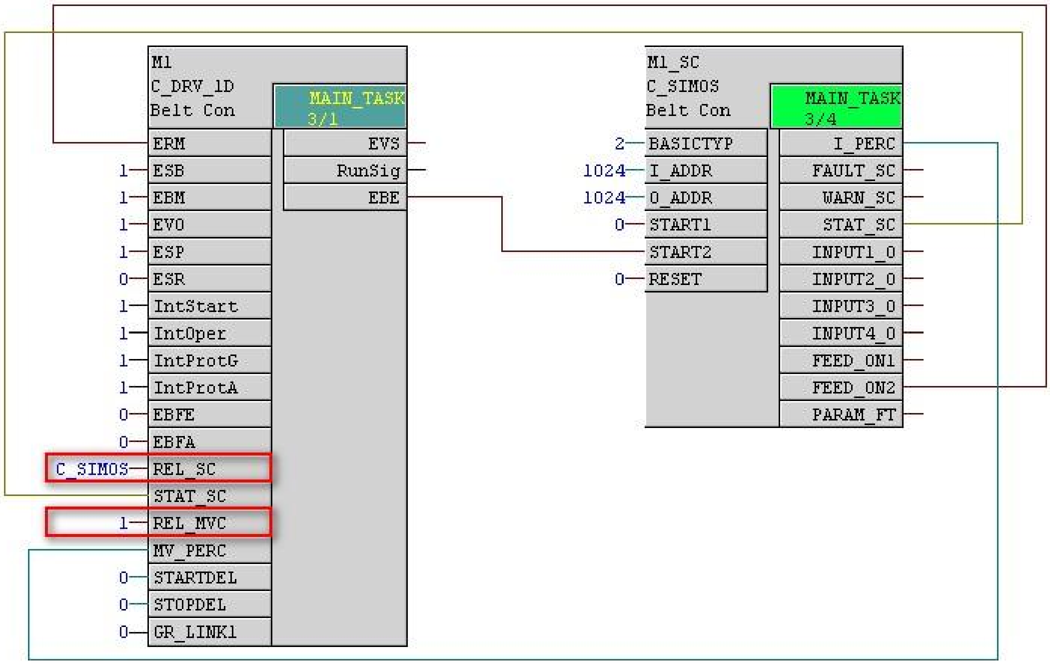


Drive with SIMOCODE

The SIMOCODE Adapter block C_SIMOS reads the cyclic data from SIMOCODE transmits this information to the drive. The start commands from the drive are transmitted to the SIMOCODE. Block C_SIMOS can be used for unidirectional drives, bi-directional drives, valves and dampers.

Block C_SIMOS and the corresponding faceplate are only templates and may have to be adapted. The block is ‘open’ (not protected) and can be modified by the user.

Connections between drive block and C_SIMOS:



The name of the SIMOCOE Adapter must be similar to the drive + ’_SC’. In the example above the motor block is called ’M1’ and the Adapter block is called ’M1_SC’.

Parameter REL_SC must contain the block type for the SIMOCODE Adapter, in this case C_SIMOS.

The drive gives the start command (output EBE) to the C_SIMOS (input START2).

The C_SIMOS gives the feedback (output FEED_ON2) to the drive (input ERM).

The C_SIMOS transmits the status (output STAT_SC) to the drive (STAT_SC). The status contains the information ’General Fault’ and ’General Warning’, which is displayed in the diagnostic window of the drive block.

If the motor current in % shall be displayed in the drive faceplate, the function must be enabled via parameter REL_MVC and output I_PERC of C_SIMOS must be connected to input MV_PERC of the drive.

At the SIMOCODE Adapter block you have to configure the Basic type and the start address for inputs and outputs. Make sure that the inputs and outputs are in the process image!

Additional connections:

The 4 free configurable outputs INPUT1_O, INPUT2_O, INPUT3_O and INPUT4_O of the C_SIMOS are not predefined and can be used in order to transmit additional information (e. g. electrical availability or overload) from the SIMOCODE block to the drive.

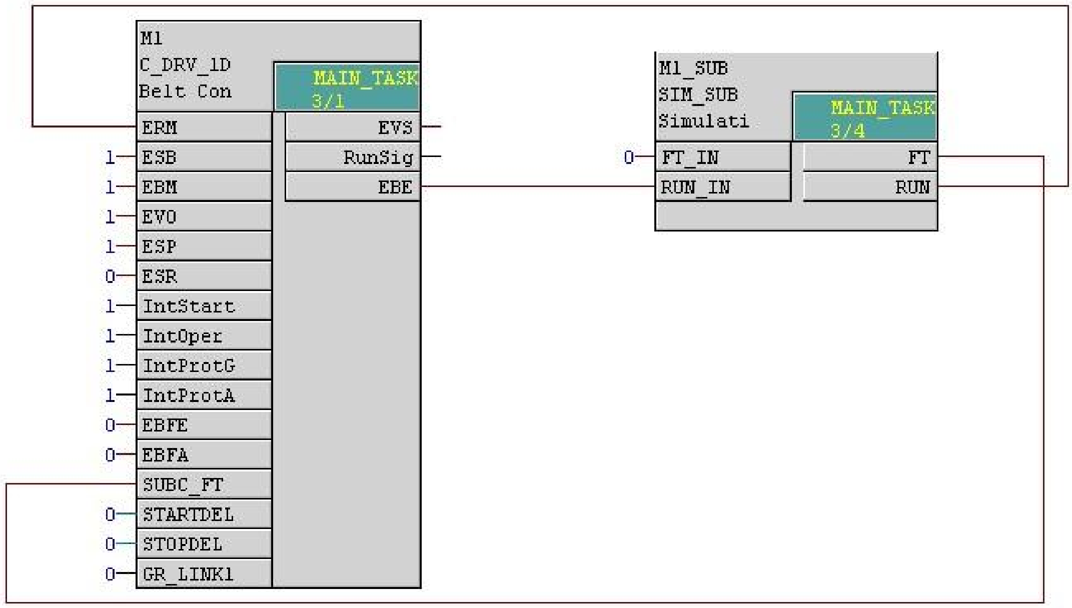
Input RESET of C_SIMOS can be used in order to acknowledge the SIMOCODE fault via program.

Drives with Subcontrol function

For complex functions like weigh feeders, filters etc. you may get blocks and faceplates from sub-suppliers. These functions often have a completely different operator interface and cannot directly be connected to the Cemat group without using an adapter block.

We recommend using a unidirectional drive block in order to control the function. The advantage is that you can use the drive interfaces for process interlocks, change of operation modes, process parameters as usual, and that the operation is identical to the rest of the system.

Subcontrol Function controlled by a C_DRV_1D block



The Subcontrol does not have its own block icon. If required, the subcontrol faceplate can be opened via the drive faceplate.

The block icon of the drive must have special attributes, containing the information about the subcontrol name and type.

Additional Link Attributes for Subcontrol Functions (only for C_DRV_1D, Index 150):

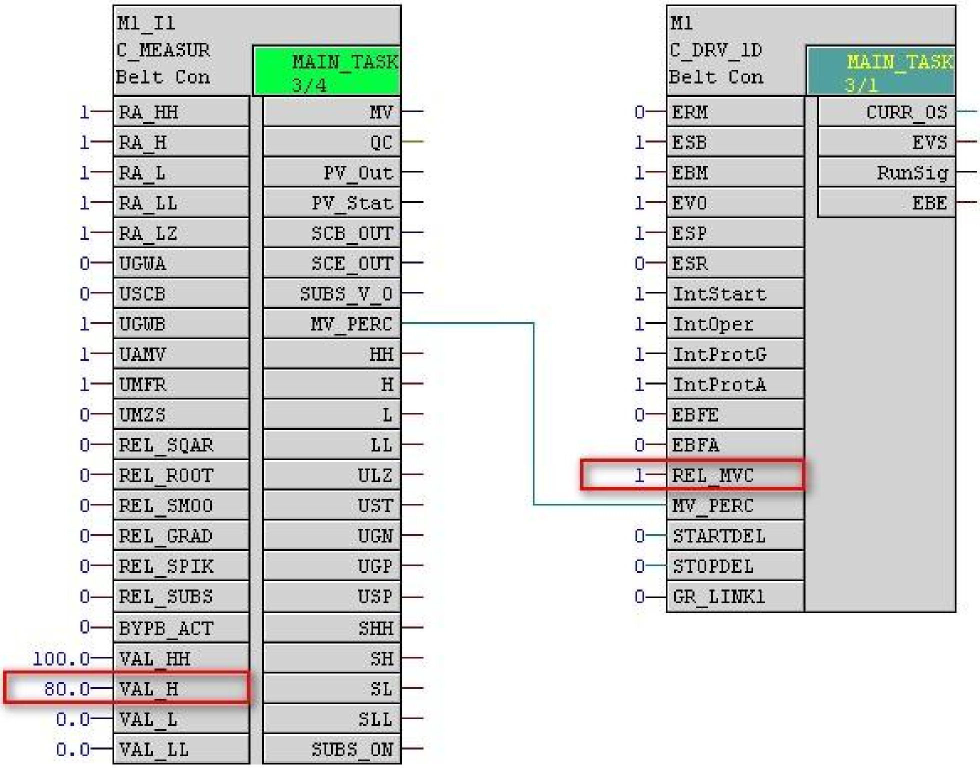
SubText	is the button Text, displayed in the motor faceplate
SubTag	is the TAG (Tagname) of the Subcontrol
SubType	is the S7 Structure name (Block type)

Display of an analog value in % (e. g. Motor current)

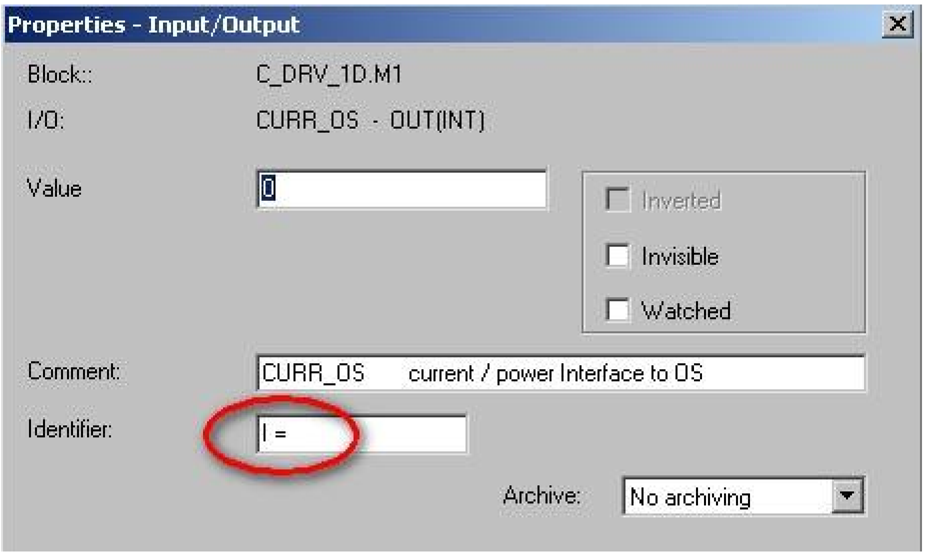
In the drive faceplate a percentage value of a measure (e. g. for motor current or power) can be displayed.

To enable the function the input REL_MVC of the drive must be set to 1-Signal and input MV_PERC of the drive must be connected with output MV_PERC of the measured value.

Caution: The output MV_PERC of the measured value is calculated based on the upper limit 1, which means VAL_H corresponds to 100%.



If you want to display a power instead value of a current value, you have to modify the description at parameter CURR_OS:

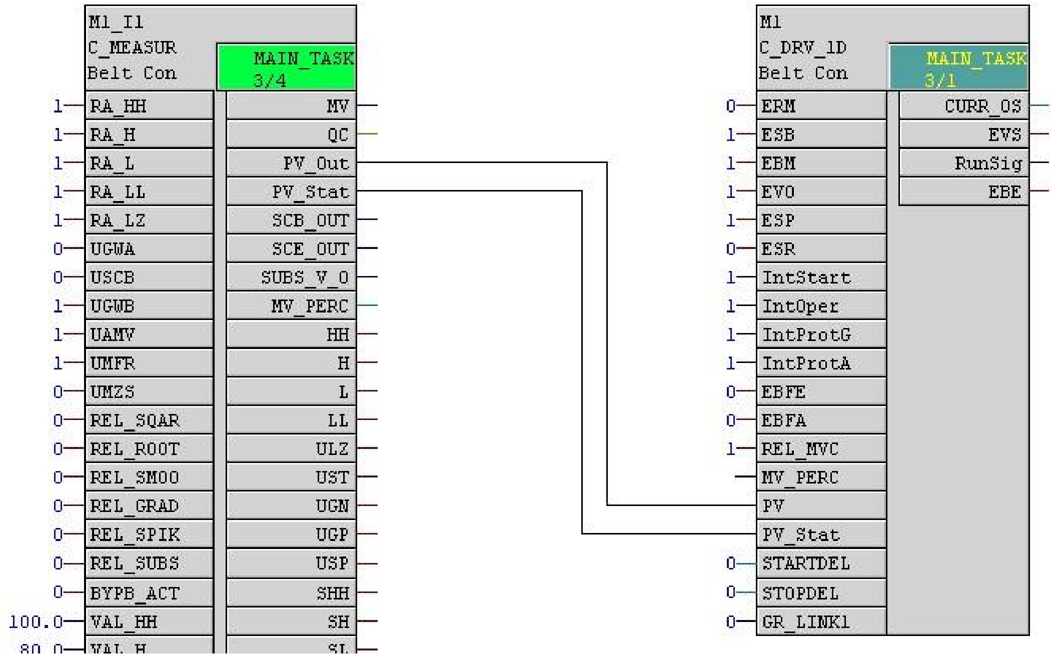


Display analog value

In the drive faceplate an analog value including its status information (limit exceeded, simulation) can be displayed.

The output PV_Out of the measured value must be connected to input PV of the drive and the output PV_Stat of the measured value to input PV_Stat of the drive.

The structure connections allow the direct faceplate call of the measured value from the drive faceplate.



If you want to link more than one measured values to the drive and select one of those for display in the drive faceplate, you have to use the function C_ANASEL. This block is described under additional blocks.

Star-Delta Starting; Slip ring Rotor with Starter Motor

For drives with star-delta starting or slip ring rotor with starter motor, the contactor feedback ERM may be created after a big delay (20-30 seconds).

For these drives the feedback time in seconds must be adapted at process parameter FEEDBTIM.

Note: In the local operation (local control using the AS), the Local-On push-button must remain depressed until the drive runs fully (ERM=1).
The ERM signal provides the criteria for maintaining the On command. If the On command was stored immediately and the drive for some reason did not start, then dangerous operational states could arise, because the motor could start subsequently without the push-button being activated.

Limit positions of the Damper

Limit switches must be wired **directly** in the contactor control circuit and so perform a **hardware** disconnection of the drive.

The signals passed to SIMATIC must not be used for the disconnection, but are required for interlocking and annunciation tasks.

This rule has to be followed for all positioning actions and end limit switches.

Operation modes and control of the damper

Like all drive blocks, also the damper has the operation modes **Automatic mode**, **Single-start mode** and **Local mode** in which the damper is either controlled via automatic program (group or route), operator faceplate or local switch from one end limit to the other. Only a pulse is required to start the drive, and the motor will be switched off if the damper has reached the limit position.

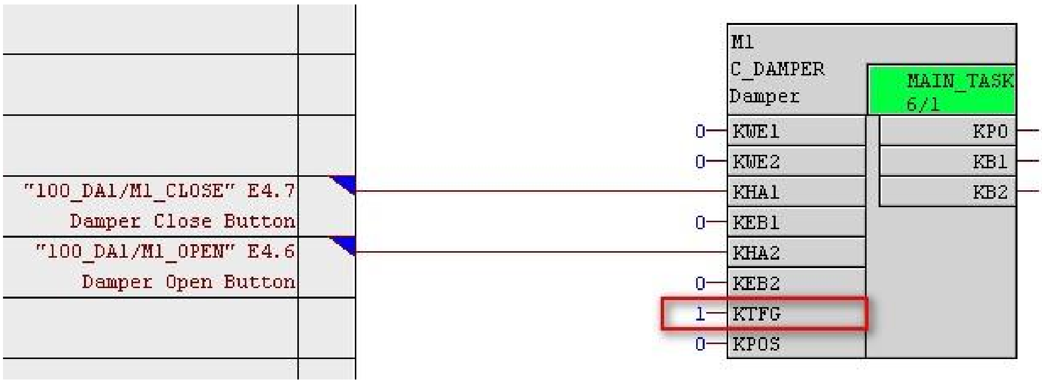
Beside this the damper can be controlled into a certain intermediate position. This can be achieved via **Inching mode** or **Positioning mode**.

Caution: In all Operation modes (Automatic mode, Single-start mode, Local mode, Inching mode or Positioning mode) the damper is controlled via two binary outputs.

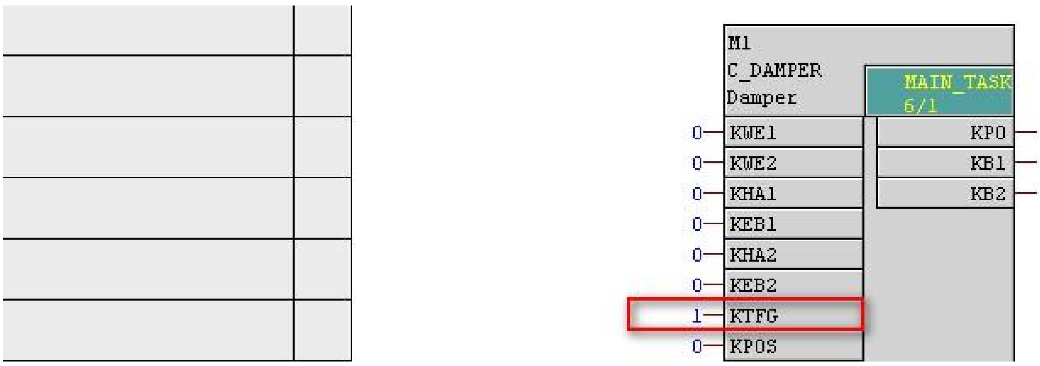
Inching mode is possible via conventional control desk buttons or via the operator faceplate. In case of control desk buttons the damper is opened/ closed as long as there is a 1-Signal at the corresponding input. In case of Inching mode via HMI the damper is opened and closed step-wise via faceplate buttons.

For the inching mode a special release function is required (KTFG = 1) which disables the ON commands for automatic mode.

a) Inching via conventional control desk buttons



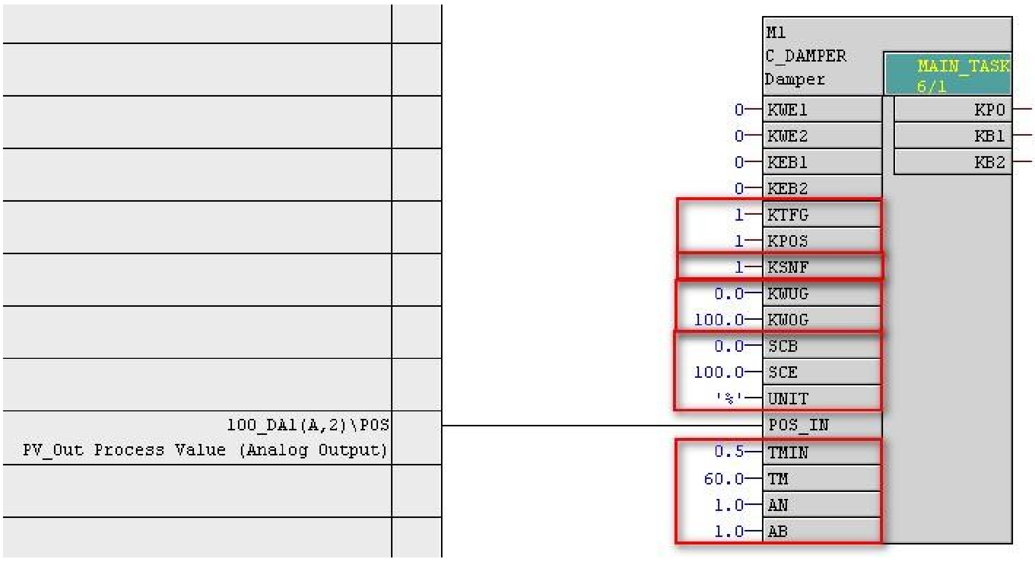
b) Inching via HMI



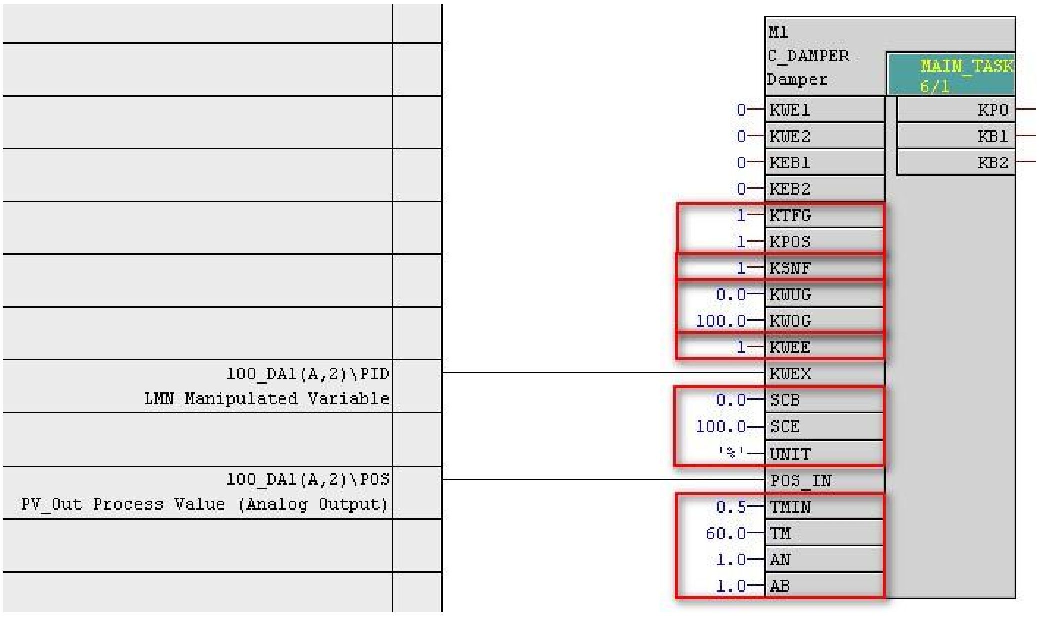
The **Positioning mode** allows entering a set point (as external set point or via damper faceplate) and the damper is opened or closed until the actual position matches with the set point. In this case a position feedback is always required.

- The release of the positioning mode is achieved through a 1-Signal at interface KPOS and KTFG
- Output ‘PV_Out‘ of the measured value for damper position must be connected to interface POS_IN of the damper.
- Via parameter SCB, SCE and UNIT Scale beginning, Scale end and Dimension <u>for the Set point</u> can be defined. The actual value (damper position) is always in %.
- Set point tracking can be enabled via parameter KSNF.
- Via parameter KWUG and KWOG the set point limits can be defined.
- You also have to parameterize the minimum pulse length TMIN, the damper runtime TM, and
..for a Hysteresis function the response threshold AN and the dropout threshold AB.

a) Positioning via operator faceplate:

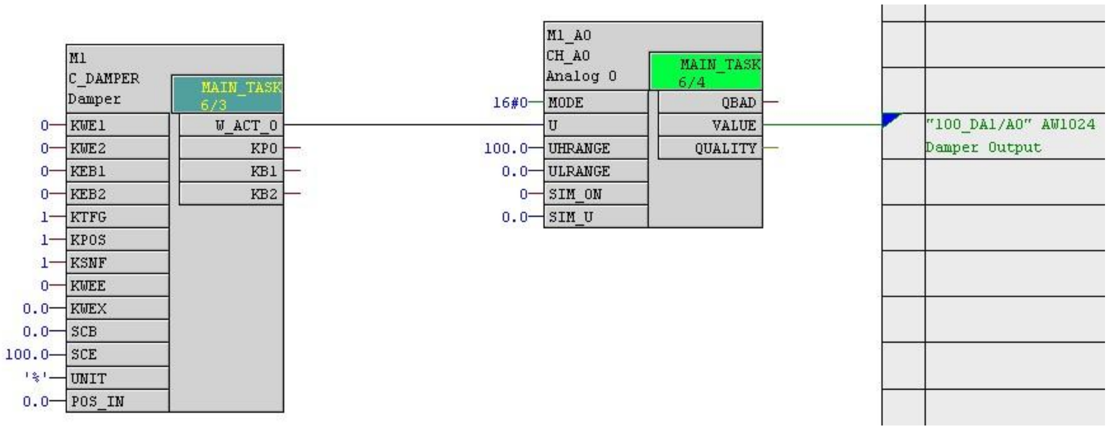


b) Positioning via external Set point (e. g. PID-Controller)
Via interface KWEE the external Set point is enabled and via interface KWEX the set point value is transmitted.



Analog actuators

If the damper/actuator needs an analog signal for the control (like e. g. analog actuators from AUMA), this can be achieved as follows:

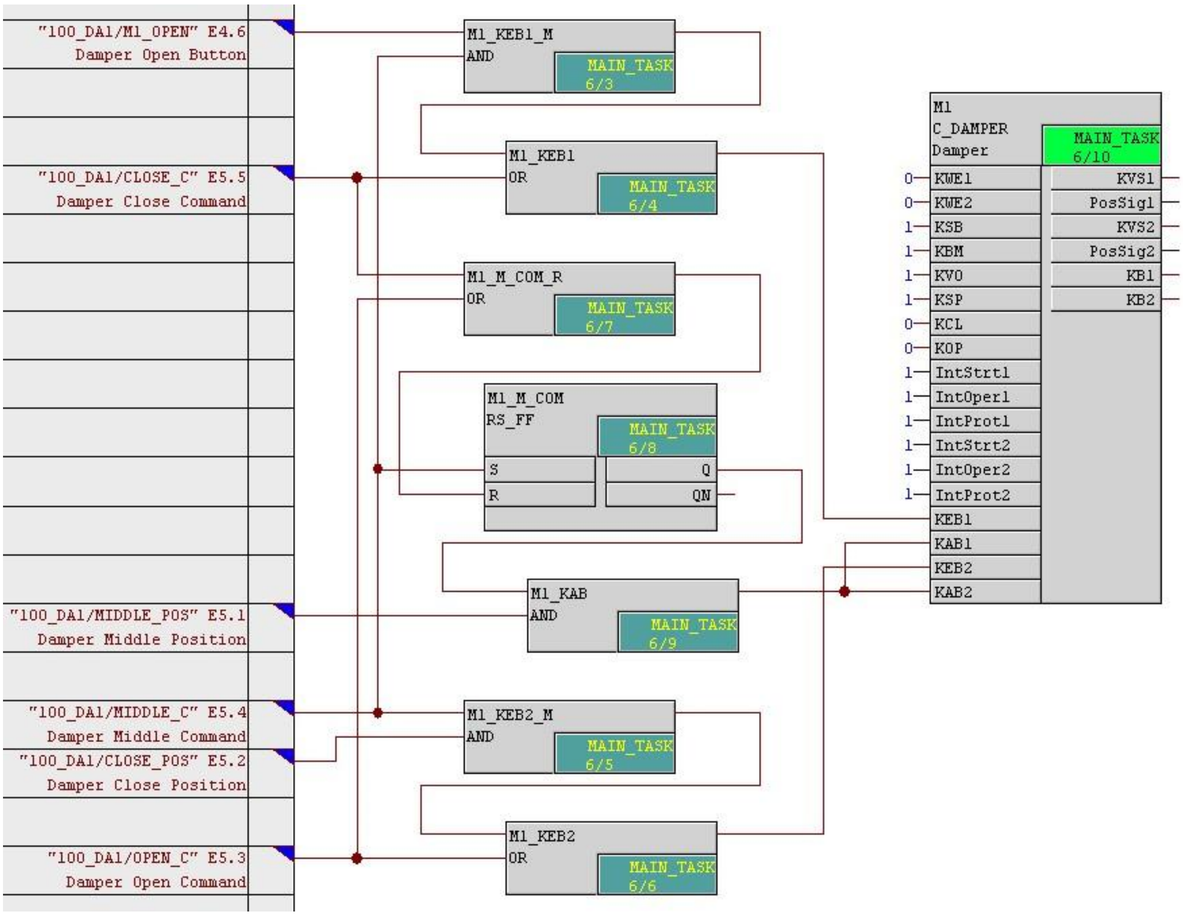


Caution: These actuators can exclusively be controlled in positioning mode, because in all other operation modes the output W_ACT_O is not actualized.
In local mode the analog output must be controlled directly (not via PLC)!

Damper with middle position

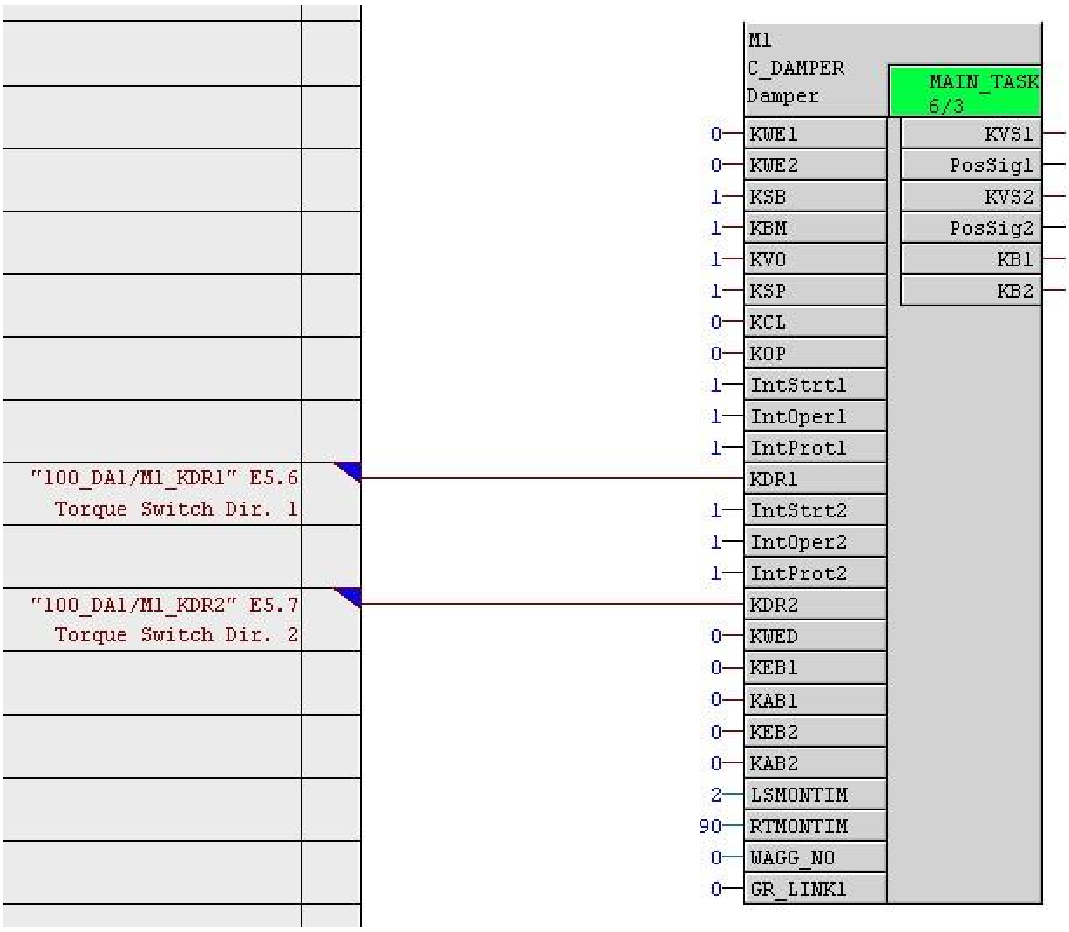
The damper block includes the supervision of 2 limit positions. Limit position 1 = Closed, Limit position 2 = Open. In some cases the supervision of 3 limit positions is needed (Open position – Middle position – Closed position).

The following example shows a solution for a damper with middle position.



Damper with torque switch

In order to stop the damper in case of a torque switch fault, the torque switch signals must be connected to interface KDR1/KDR2. An alarm message “Torque switch” will be created.



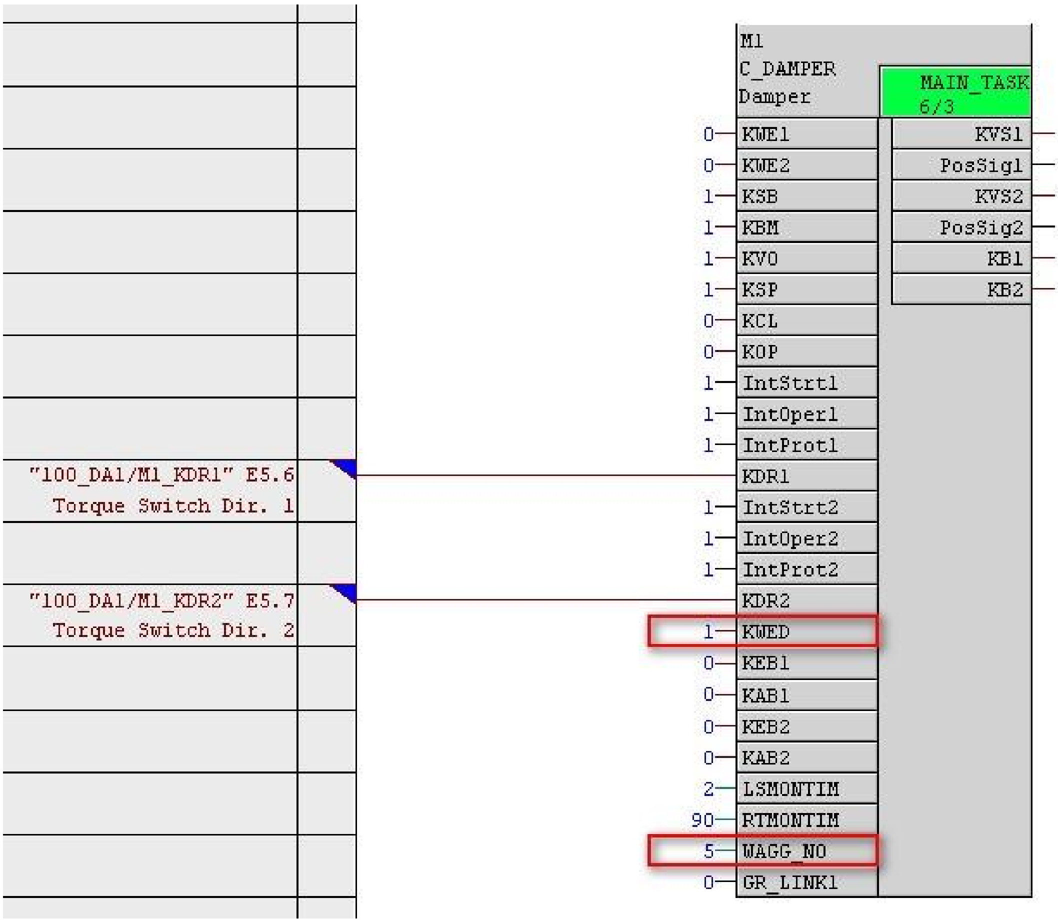
Wagging function

If the wagging function is enabled (KWED = 1-Signal), in case of a torque switch trip the damper will be controlled back to the old position. After this it will try again for the controlled position. This behaviour is called ‘wagging’.

The number of trials (wagging number) can be configured at process parameter WAGG_NO.

If after the maximum number of trials the selected limit position is still not reached, an alarm message is given for ’Mechanical fault’.

Example for a damper with torque switches, in case the wagging function is enabled:



Fan and Damper

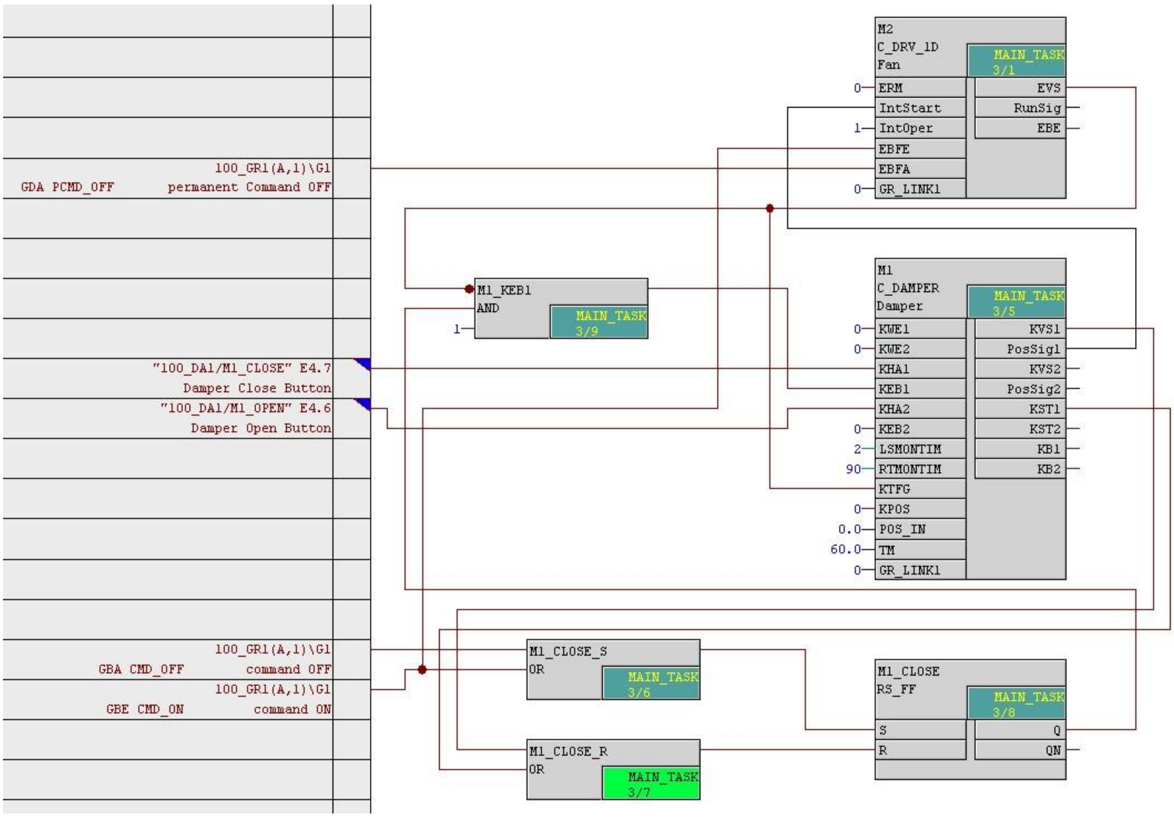
The big fans can only be started of the damper is closed. If the process allows, it makes sense to close the damper automatically when the group is stopped.

The following example shows how this task can be solved. Also after an eventual damper fault the damper can be closed through group stop, or it will be closed during group start.

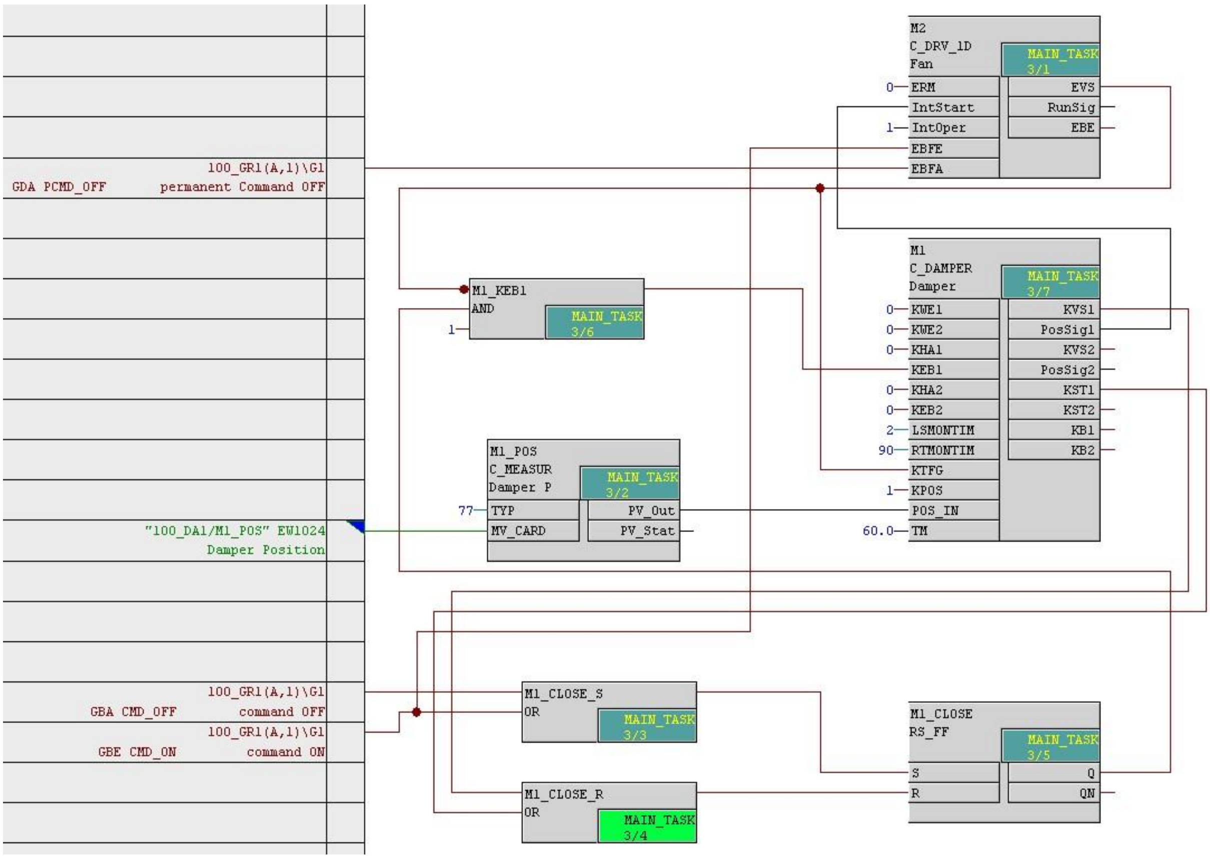
In example a) the positioning of the damper is carried out via the inching commands of the damper block. This solution is suitable for conventional control desks or local control panels.

In example b) the positioner function is used for the same task. This allows entering a numeric set point via HMI.

a) Connections between fan and damper in case of positioning through control desk buttons:



b) Connections between fan and damper positioner

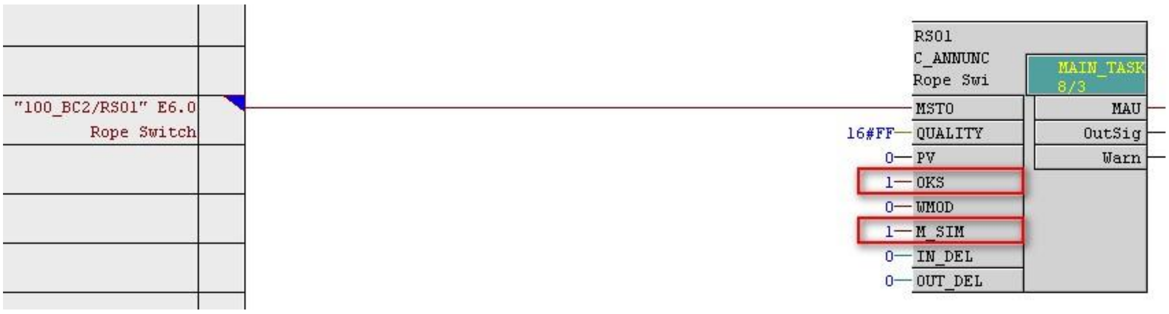


Annunciation blocks

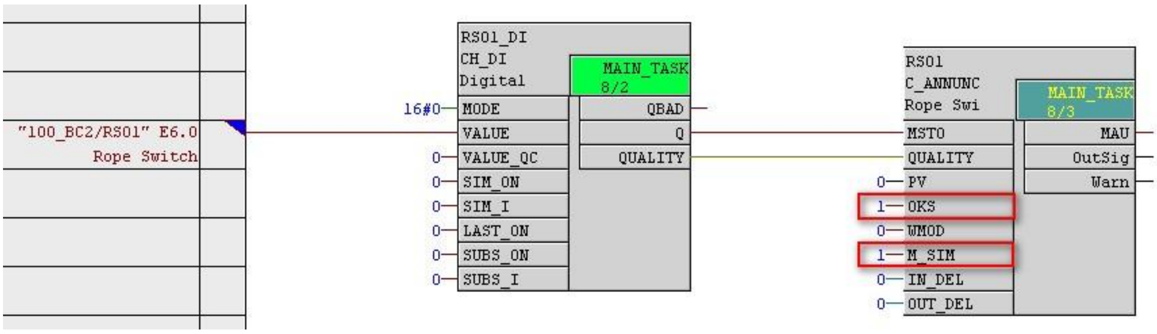
Annunciation blocks are used to for the supervision of binary process signals. The signal status is displayed in the process picture (ok, fault or warning) and in the summarizing indication of the group. In case of a fault a message (fault message or warning message) can be created in the alarm line.

Depending on the format of the input to be evaluated, the annunciation block has different interfaces.

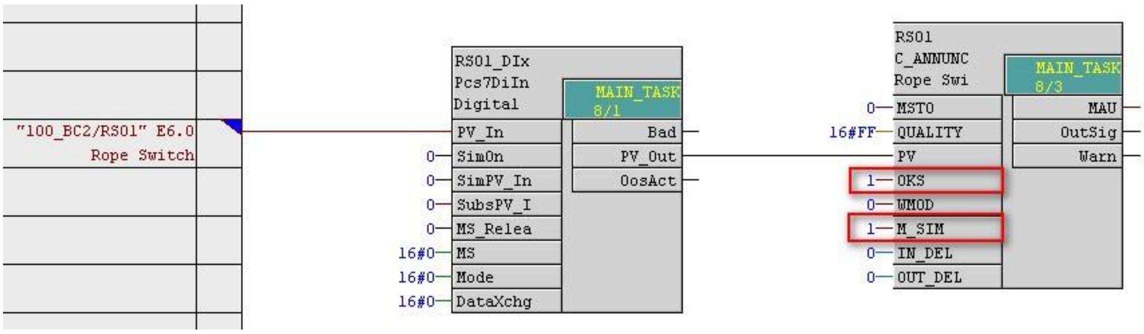
MST0 for the evaluation of a Boolean signal



MST0 + QUALITY for the connection of channel driver block CH_DI



PV for the evaluation of a structure input (e. g. from APL channel driver Pcs7DiIn)



Parameter OKS decides how the signal is evaluated. The status of OKS corresponds to status 'input signal ok'. As most of the process signals are fail-safe, which means in good condition they are "1", we recommend to set parameter OKS to 1-Signal in order to avoid the negation of the input.

The simulation value M_SIM should be similar to the OK-Signal.

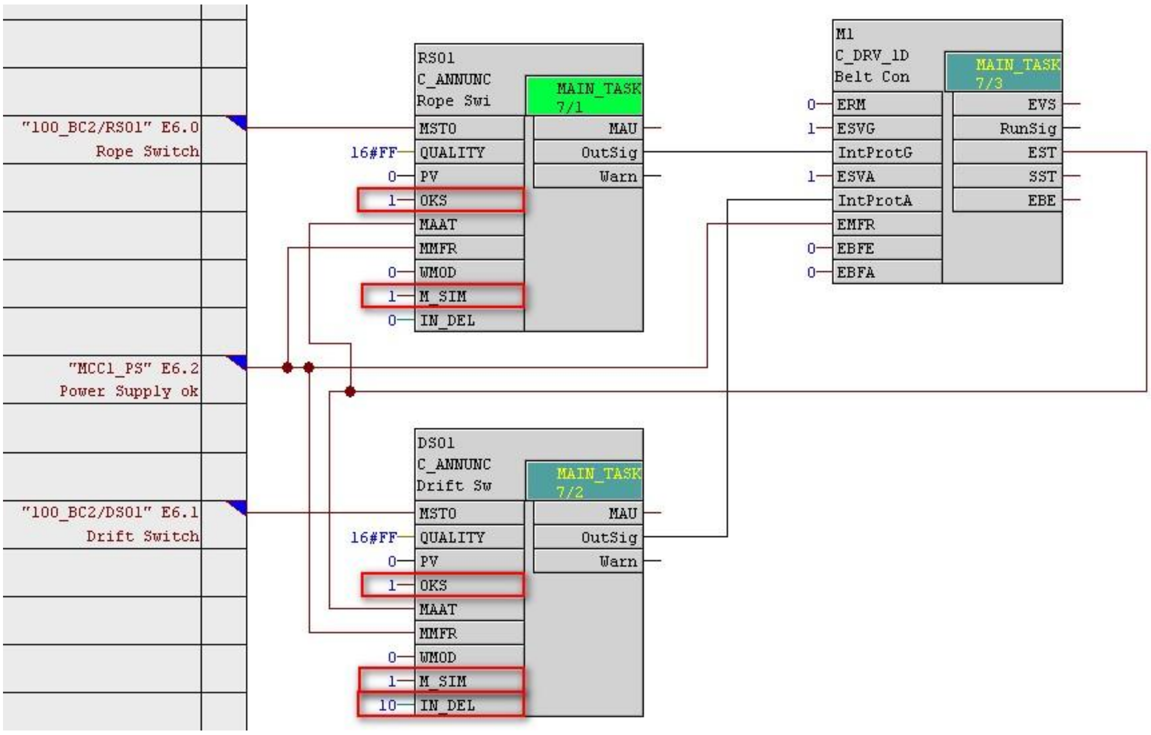
Via the process parameters IN_DEL and OUT_DEL incoming and outgoing faults can be delayed.

Parameter WMOD decides if a fault or a warning is displayed. (WMOD = 1-Signal for warnings).

The following pages show different applications for annunciation blocks:

Drive fault annunciations

All protection interlocks which lead to a drive stop (e. g. pull rope switch, belt drift switch, bearing temperature) must create an alarm message at this moment.
The alarm message shows the operator which condition caused the drive stop and only in case of a present alarm the drive fault can be acknowledged with the acknowledge button in the alarm line.

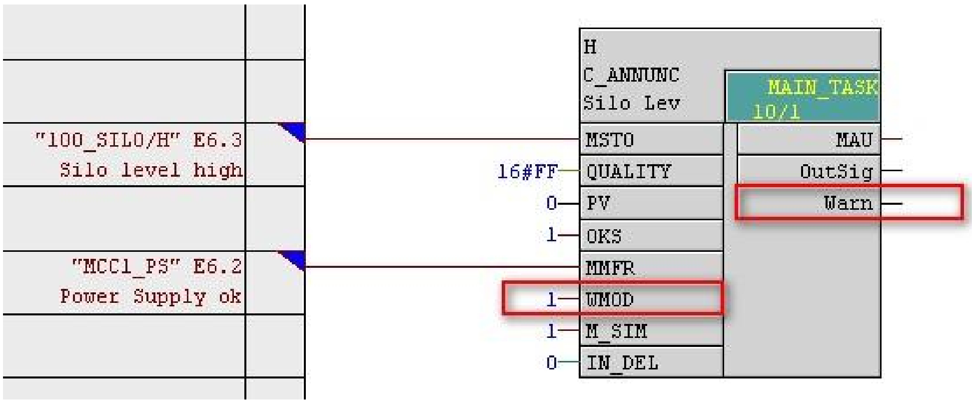


- In the example above the binary signals are directly read from the process image. For this purpose interface MST0 is used. Both signals are fail-safe (in good condition ’1’); therefore OKS and M_SIM are set to 1-Signal.
- The message for the pull rope switch must be created without delay (process parameter IN_DEL = 0). The message for the belt drift switch has to be delayed for 10 seconds (Process parameter IN_DEL = 10).
- In order to create the message at the same time as the drive stop, the output of the annunciation block (MAU or OutSig) bust be used for the connection to the protection interlock. It must be distinguished between protection interlocks which are always effective (ESVG or IntProtG) and protection interlocks which are effective only in automatic or single-start mode. (ESVA or IntProtA).
- In order to trigger the alarm with each drive fault (even if the operator tries to start an already faulty motor) the alarm activation MAAT of the annunciation block must be connected with the output ‘dynamic fault’ (EST, VST, KST) of the drive block.
- The annunciation release of drive block and annunciation block must be connected with the same signal for ‘control power ok’.

Warning messages

Via parameter WMOD is defined if in case of a bad input signal a fault or a warning is indicated. (WMOD = 1-Signal for warning).

If the signal is used for interlocks with other blocks, block output 'Warn' must be used, because MAU and OutSig are always 'good'.



Another example you find under "Interlocking Annunciation".

Caution: Output 'Warn' does not depend on OKS. In case of a fault output 'Warn' has 1-Signal!

Alarm interlocking for the annunciation block

For drive blocks it is assured that a fault message is only created if the drive is “active”, which means

a) if a running drive stops with fault,

b) if a start command is given for an already faulty drive and

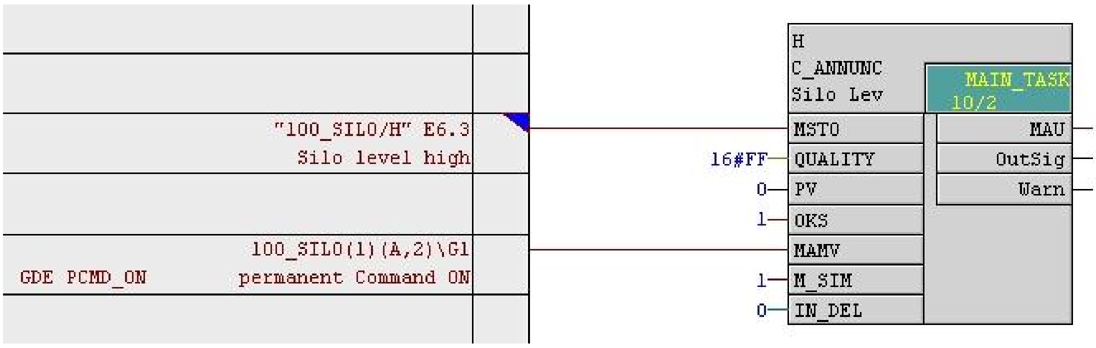
c) if the drive is in standby mode and a fault occurs.

For the annunciation blocks (C_ANNUNC) you have to insure that only plausible alarm messages are created. In case of drive fault annunciations this is already guaranteed through the connection of the alarm activation (MAAT).

For the annunciation of process signals which do not belong to any drive, the alarm interlocking MAMV must connected in order to create only plausible messages. Which signal is used for the alarm interlocking depends on the technological function of your application. Criteria for the plausibility of a message could be group running completely, drive running, group status etc.

Example:

The alarm message must only be created if the group was started (and not stopped).

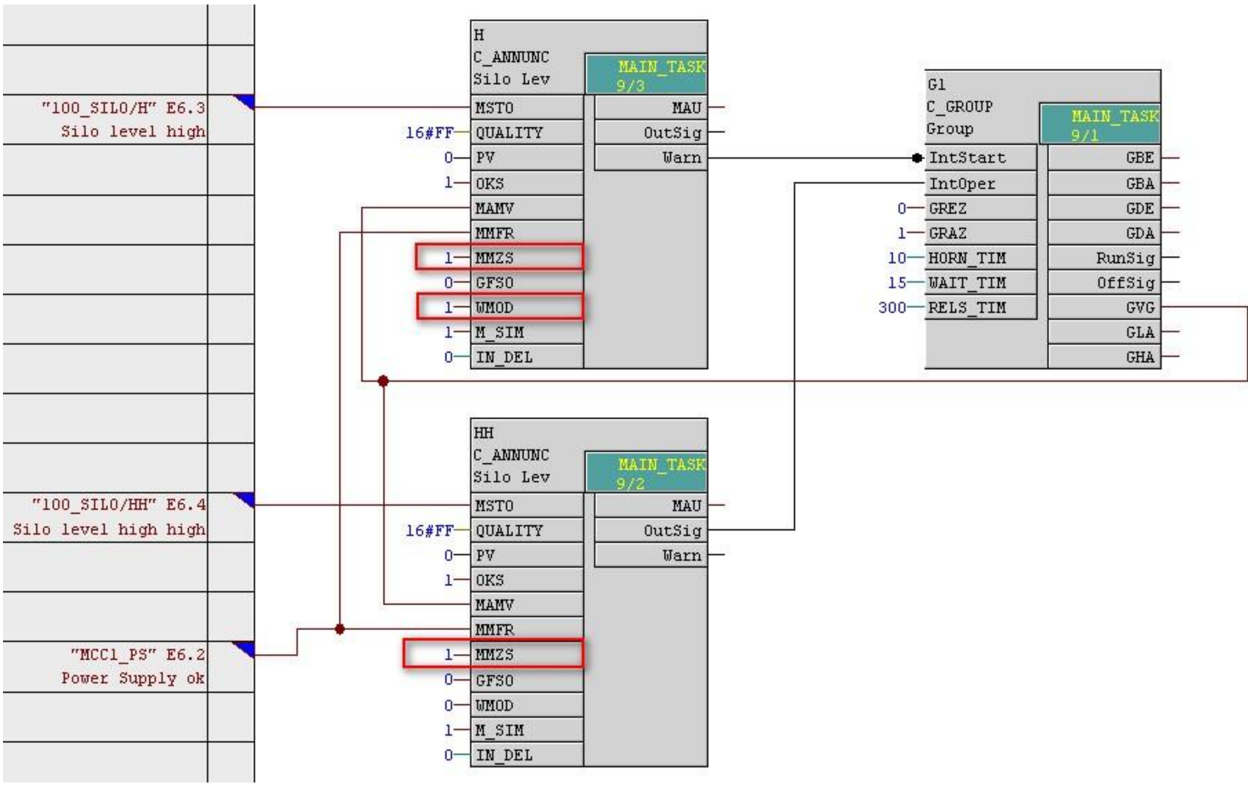


Interlocking annunciation (Silo levels)

Annunciation blocks are not only used for the indication of faults but also for interlocking conditions. In the following example the silo levels “High“ and “Maximum“ are used as interlocking conditions for the group.

At level “High“ only the group start shall be inhibited. If the group is running and level “High“ is reached, a warning message is created but the group will continue to run.

Level “Maximum“ must inhibit the group start and must also switch off the group together with a red (fault) message.

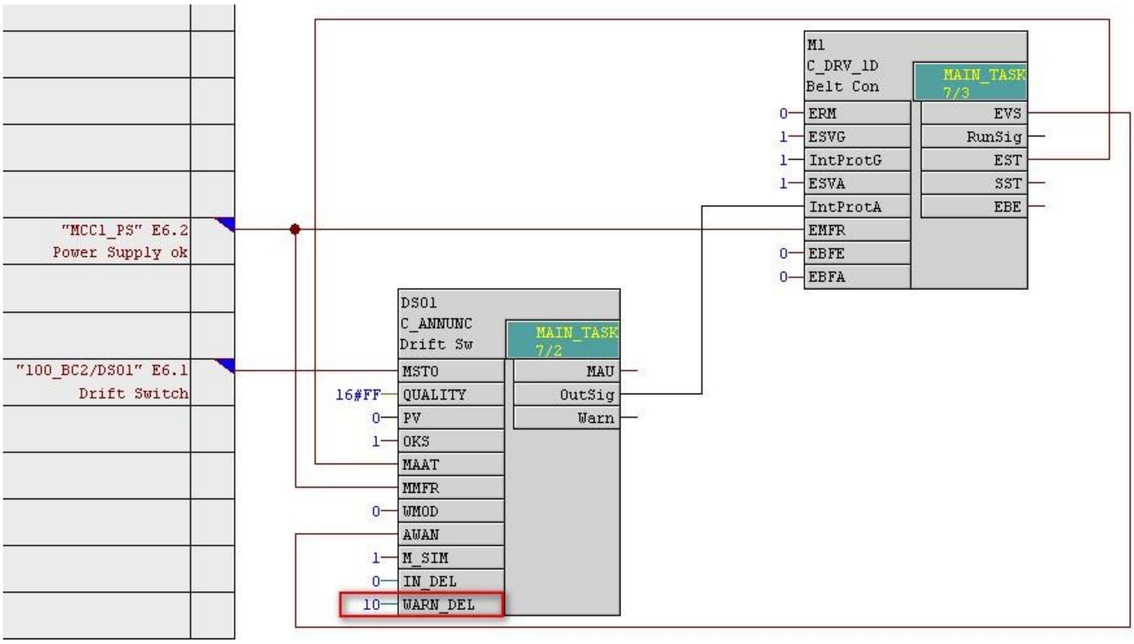


- Via connection of WMOD with 1-Signal, in case of level “High“ a warning message is created instead of a fault message. Output “Warn“ is used as start interlocking of the group.
  Caution: Signal must be inverted (does not depend on OKS).
- The connection of MMZS with 1-Signal effects that the fault or warning is not included in the summarizing indication of the group. After all, the silo levels are no faults or warnings, they are only interlocking conditions.
- If GFSO is connected with 1-Signal the interlocking would be excluded form group status call as well. However, this behaviour is not wanted, because the interlocking should appear in the status call as well.
- Through connection of the alarm interlocking MAMV with signal GVG of the group, the message is only created if the group was started. If the group is was not started only a static indication is given (without message).
- The annunciation release must be connected with the signal for ‘control power ok’.

Two-level alarm (first warning then switch-off)

In the following example, for the same process signal (belt drift switch) first a warning message is created and after a time delay a fault message is created.

The warning does not switch off the motor. Only with the fault message the motor is switched off via protection interlock.



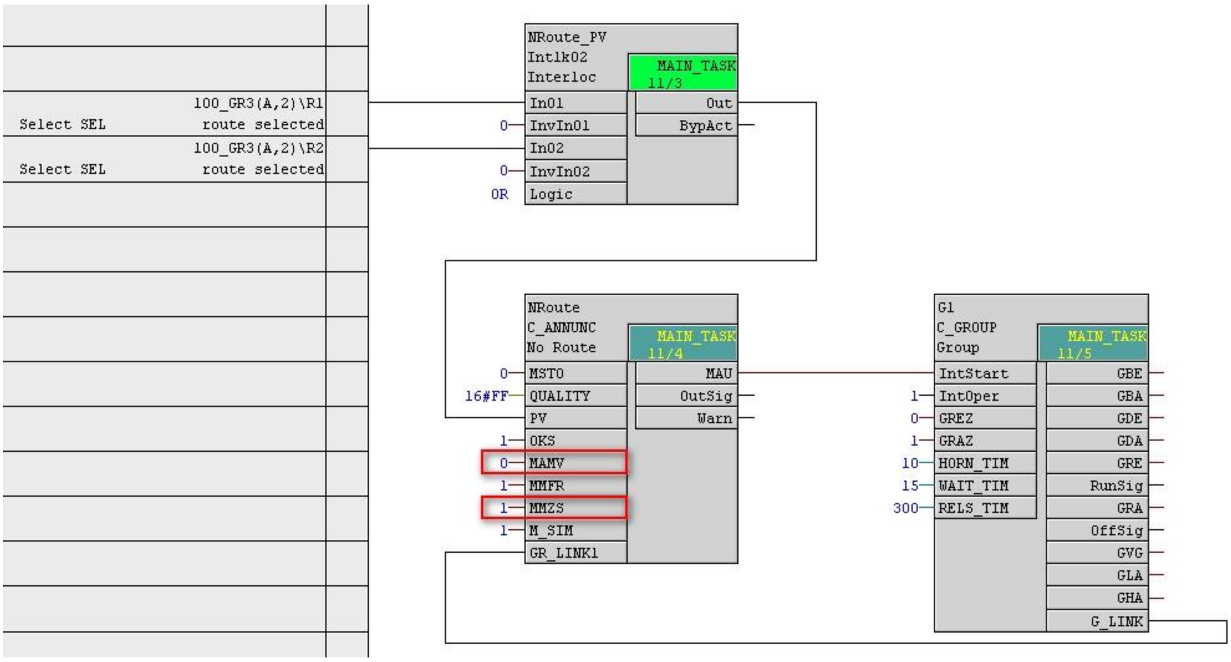
- As parameter IN_DEL has 0-signal, the warning message is created without delay. However, a time delay for the warning is also possible.
- Via parameter WARN_DEL the time delay between warning message and fault message is configured.
- Signal OutSig is not influenced by the warning and switches off the drive only in case of a fault.
- Input AWAN activates the two-level alarm and must be connected with the running signal of the motor.
  Only for a running drive (EVS = 1-Signal) the belt drift fault leads to a two-level alarm.
  In case of a present belt drift fault at the moment of a drive start, the fault message is created immediately.

Caution: In project version 004 the running signal EVS can not be used for AWAN because in single-start mode the running signal is not created. You must use a logic instead: Contactor feedback ERM & not LOCAL.

Annunciation for Status call

Annunciation blocks can also be used in order to visualize software interlocks in the status call of the group.

In the following example, before starting the group it has to be insured that at least one route is selected. In order to visualize the interlocking condition in the status call, an annunciation block is programmed.



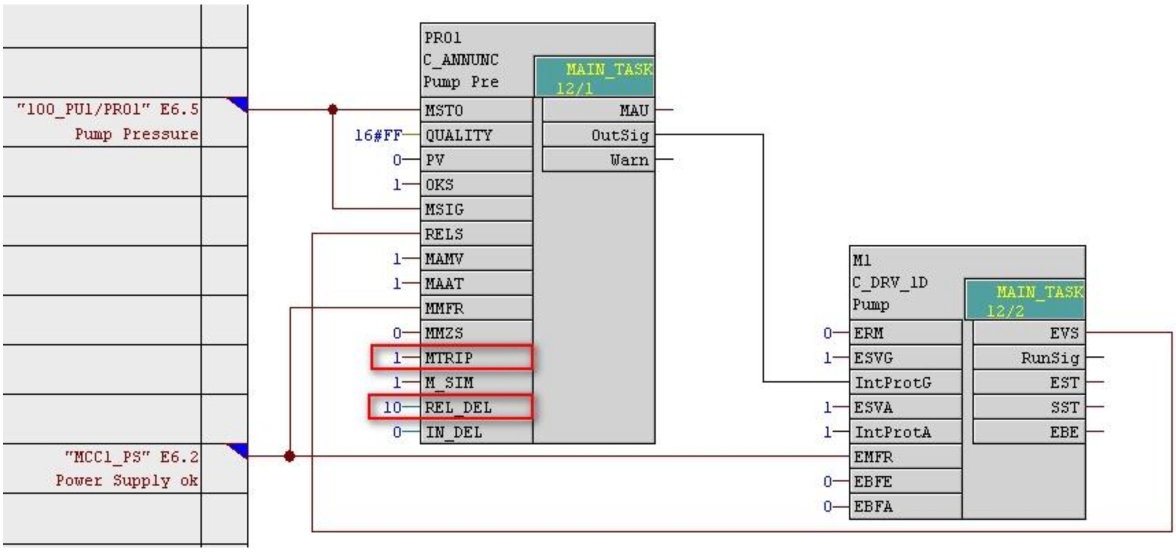
- Interface MMZS is connected with 1-Signal in order to exclude the annunciation block in the summarizing indication of the group.
- Interface MAMV is connected with 0-Signal, this kind of annunciation block must never create an alarm.
- This annunciation block does not need a block icon. Deselect the option in the block properties.

Release supervision

Certain process signals must only be evaluated if the associated device which creates or influences this signal is running.

The following example shows a pressure supervision which should only be active if the associated pump is in operation.

If 10 seconds after the pump start the pressure is not yet built or in case of pressure loss during operation, the pump will be stopped immediately (via protection interlock).



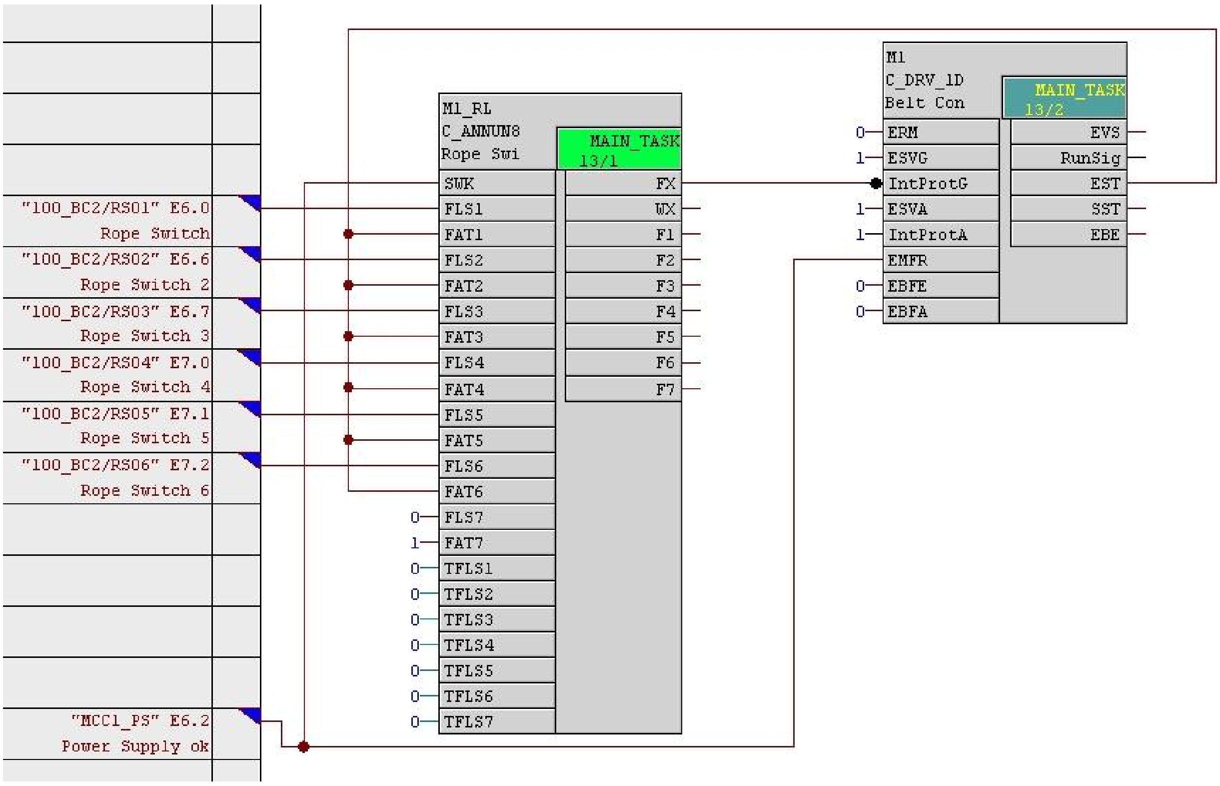
- The supervision gets enabled through connection of interface RELS with the running signal EVS of the drive. The time delay of 10 seconds for the supervision is configured at parameter REL_DEL.
- If the input signal which is connected to MST0 is additionally connected to MSIG, the block icon in the process picture shows green colour as soon as the pressure is built.
- With 1-Signal at interface MTRIP you achieve that after the stop the fault is memorized until acknowledgement. If MTRIP has 0-Signal, immediately after the drive stop the status of the annunciation block is ‘good’ again.

Engineering of an ANNUN8 block

Instead of the annunciation block C_ANNUNC the annunciation block C_ANNUN8 can be used.

Block C_ANNUN8 is capable to create 7 fault messages or warning message and can be used if you want to show the status of these signals in a common block icon. The block icon only has a summarizing indication and always shows the worst status.

Example: Annunciations of 6 Rope switches for a long belt conveyor.



- Via connection of the protection interlocking IntProtG with the summarizing fault FX of the annunciation block (negation required) the motor will be switched off if one of the ropes is pulled.
- Through connection of the alarm activation FAT1 – FAT7 with the dynamic fault of the drive, the fault message is only created if the motor is stopped by fault.
- Don't forget to enter the individual fault texts via property 'identifier' at parameter FLS1 – FLS7. The texts are used in the diagnostic window and in the status call function. The maximum text length is 16 characters.
- The annunciation release SWK must be connected with the signal for 'control power ok'.

Measured value

Read in a measured value

Measured value blocks are used for the supervision of analog process signals. The value and the signal status are displayed in the process picture and in the summarizing indication of the group. In case of a fault a message (fault message or warning message) will be created in the alarm line.

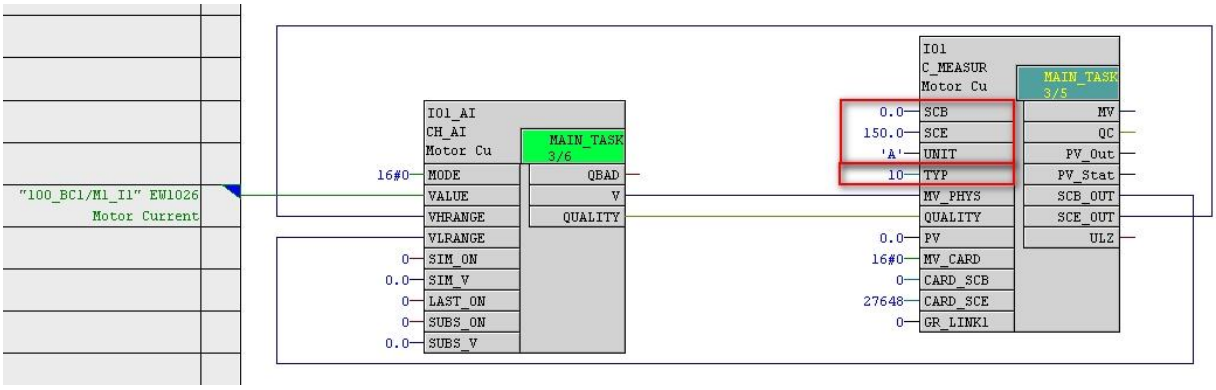
Depending on the format of the input to be evaluated, the measured value block has different interfaces. Parameter TYP defines which input is evaluated:

TYP = 77 Reading a card value from input MV_CARD
TYP = 10 Reading a REAL value form input MV_PHYS
TYP = 20 Reading a REAL structure from input PV

Reading a card value (WORD Format)

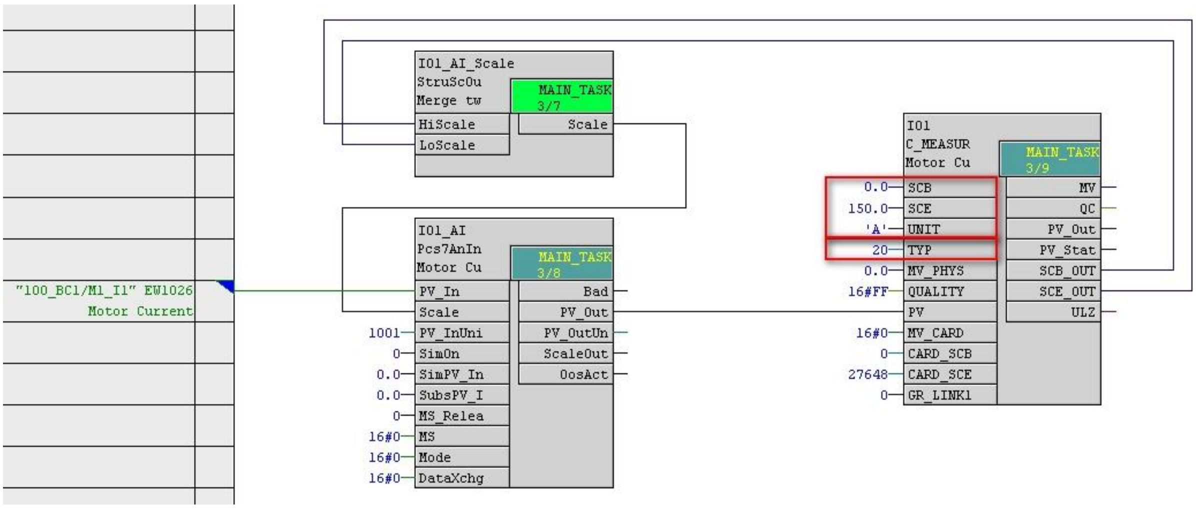
- With TYP = 77 reading from MV_CARD is selected
- The analog input must be connected to input MV_CARD of the C_MEASUR.
- Scale beginning, scale end and dimension must be defined at parameter SCB, SCE and UNIT.
- In this case the measured value block must do the conversion of the card value into the physical value. Scale beginning and scale end of the card value have to be defined at parameter CARD_SCB and CARD_SCE.

Reading a physical value from the channel driver block CH_AI



- With TYP = 10 reading from MV_PHYS is selected
- The output V of the channel driver block must be connected to input MV_PHYS of the C_MEASUR.
- Output QUALITY of the channel driver block must be connected to input QUALITY of the C_MEASUR.
- Scale beginning, scale end and dimension must be defined at parameter SCB, SCE and UNIT.
- In order to use similar scaling for measured value and driver block, you should connect the outputs SCB_OUT and SCE_OUT of the measured value to the inputs VLRANGE and VHRANGE of the driver block.
  Caution: In case of thermo elements (PT100) don't connect!!

Reading a physical value in structure format (e. g. from APL driver block Pcs7AnIn)



- With TYP = 20 reading from PV is selected
- The structure output PV_Out of the channel driver block must be connected to structure input PV of the C_MEASUR.
- Scale beginning, scale end and dimension must be defined at parameter SCB, SCE and UNIT.
- In order to use similar scaling for measured value and driver block, you should connect the outputs SCB_OUT and SCE_OUT to structure converter StruScOu which provides an output ‘Scale’ for connection to the driver block.

Measured values, Calculated Values

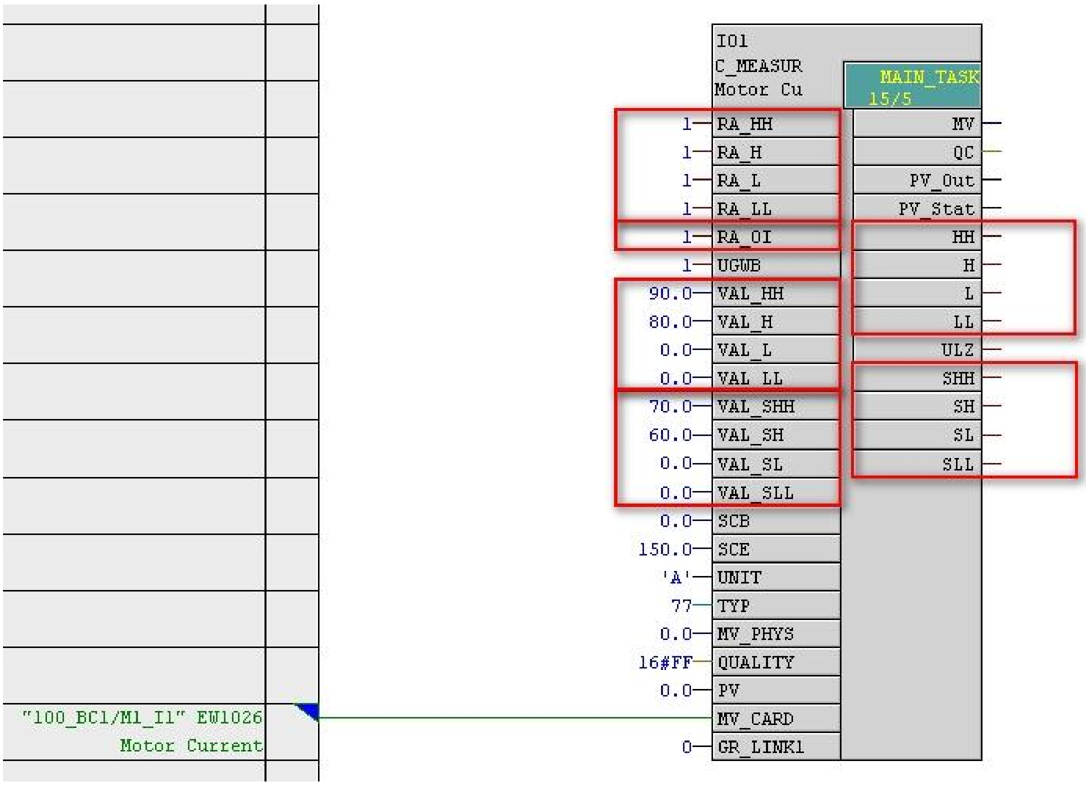
Certain values are calculated during the plant configuration, e. g. total amounts from sub-amounts, temperature or pressure average values.

Similar to analog signals coming from the periphery, also for the calculated values a measured value block should be programmed. The measured value block simplifies the integration in the HMI system.

To improve the readability, the calculation should be performed before the invocation of the measured value block.

Measured values, Limit Value Supervision

The measured value block has 8 limit values. The violation of limit values VAL_HH, VAL_H, VAL_L and VAL_LL leads to a message, the limit values VAL_SHH, VAL_H, VAL_L and VAL_LL are only switching limits and don't create messages.



- If UGWB = 0-Signal the limit supervision is completely blocked.
- In case of a limit violation the corresponding output HH, H, L, LL, SHH; SH, SL or SLL is set.
  H and L lead to a warning message and HH and LL to a fault message.
- If parameter RA_HH, RA_H, RA_L or RA_LL are connected with 0-Signal the corresponding limit behaves like a switching limit, the fault bits are not set and no message is created. The outputs HH, H, L and LL are still set.

  If you want to inhibit the outputs HH, H, L and LL in the above mentioned situation as well, you have to connect interface RA_OI with 0-Signal.

Alarm interlocking for the Measured Value block

For drive blocks it is assured that a fault message is only created if the drive is “active”, which means

a) if a running drive stops with fault,

b) if a start command is given for an already faulty drive and

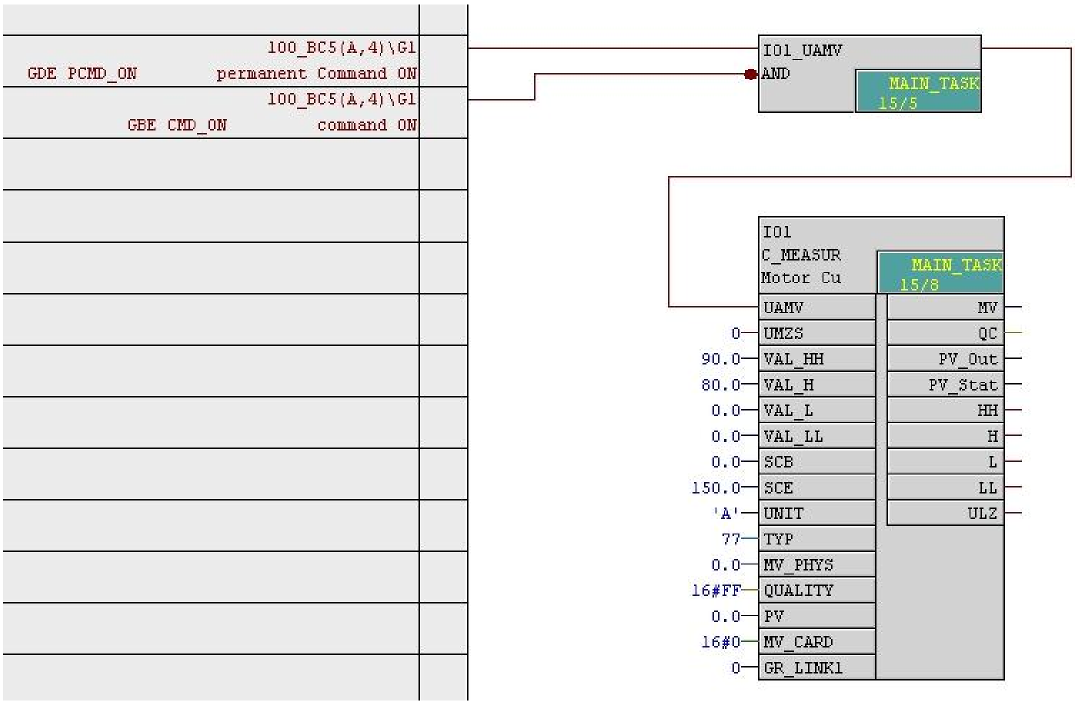
c) if the drive is in standby mode and a fault occurs.

For measured value blocks (C_MEASUR) you have to insure that the alarm message is only created if the corresponding plant section is running and the message is plausible.

The alarm interlocking UAMV has to be programmed individually, dependent on the technological function. Criteria for the plausibility of messages could be group running completely, drive running, group status etc.

Example:

The message for the limit supervisions should only be created if the group is completely started.

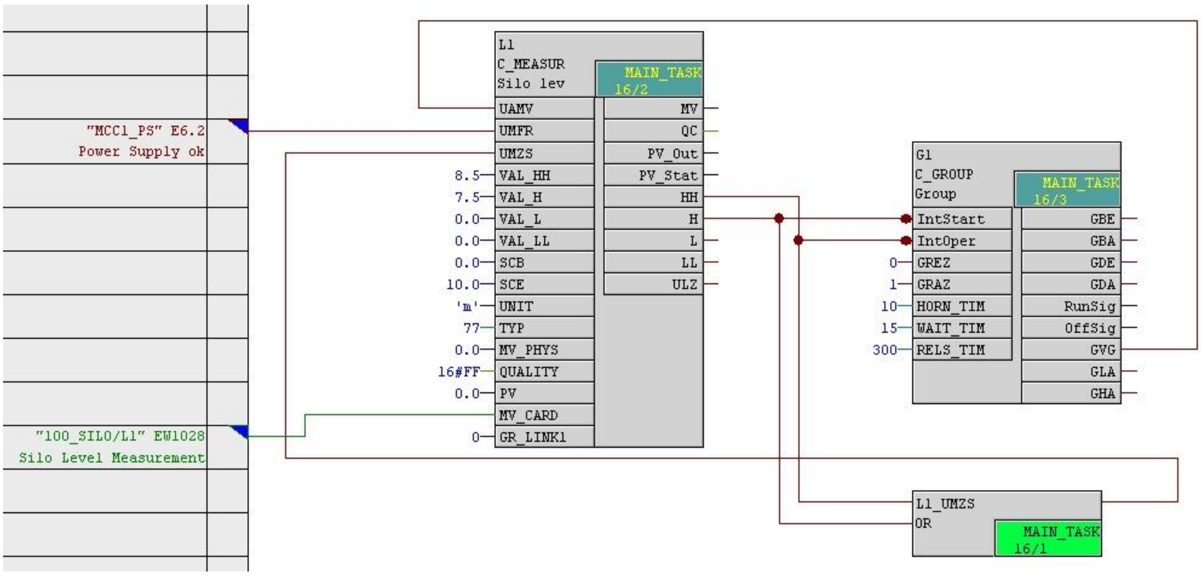


Using limit violation bits as interlocking condition

In the following example the silo levels are used as interlocking condition for the group.

The violation of upper limit 1 inhibits the group start and leads to a warning message.

The violation of upper limit 2 stops the running group and leads to a fault message.



If the limit bits are used for start interlock or operation interlock of the group the following aspects have to be considered:

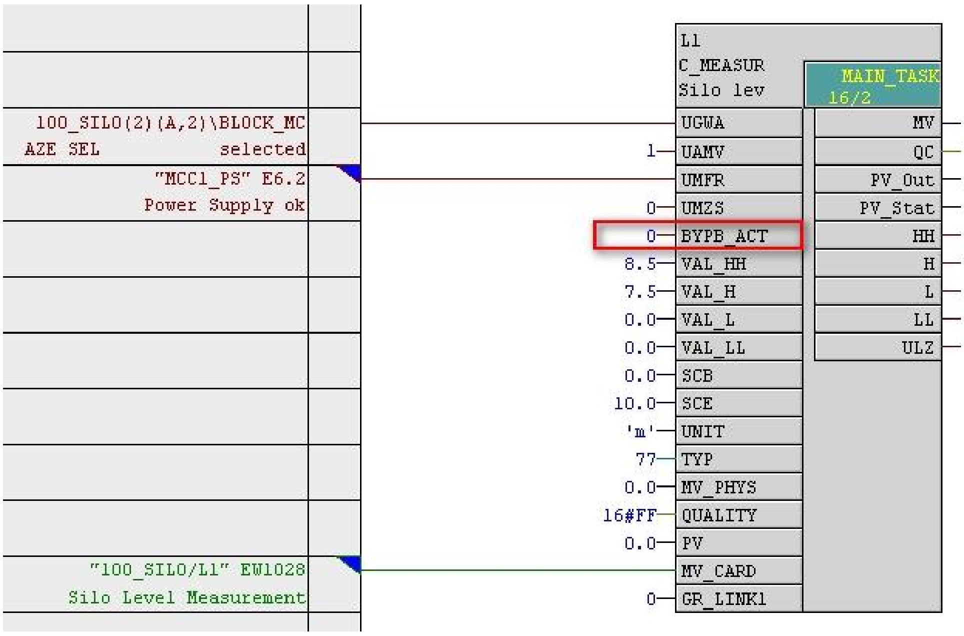
- Upper limit 1 and upper limit 2 are interlocks and should not be included in the summarizing indications of the group block icon. Therefore UMZS has to be connected with the limit bits.

  If UMZS is connected with a continuous 1-Signal (LOG1) also Life Zero fault (bad quality) is excluded from the summarizing indication.
- By connecting the alarm interlocking UAMV with Signal GVG of the group, the measured value block only creates alarm messages when the group is started.
- Via status call function the operator can check for existing interlocks at any time (also if the group has not been started).

Block Measured value, Bypass, Service mode

There are two possibilities to block the measured value:

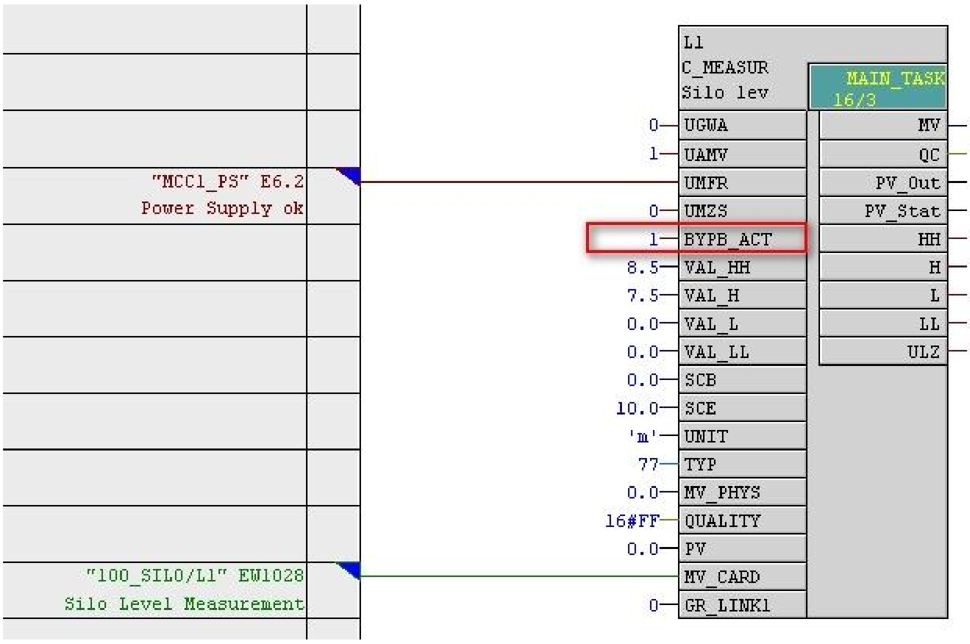
a) The bypass function is not activated:



In this case the measured value is blocked with a 1-Signal at interface UGWA (for example while the calibration is running).

If UGWA = 1-Signal the measured value is not read in anymore and the last value is frozen. The limit bits are frozen as well.

b) The bypass function is activated:



In this case the bypass is carried out via HMI (Service mode).

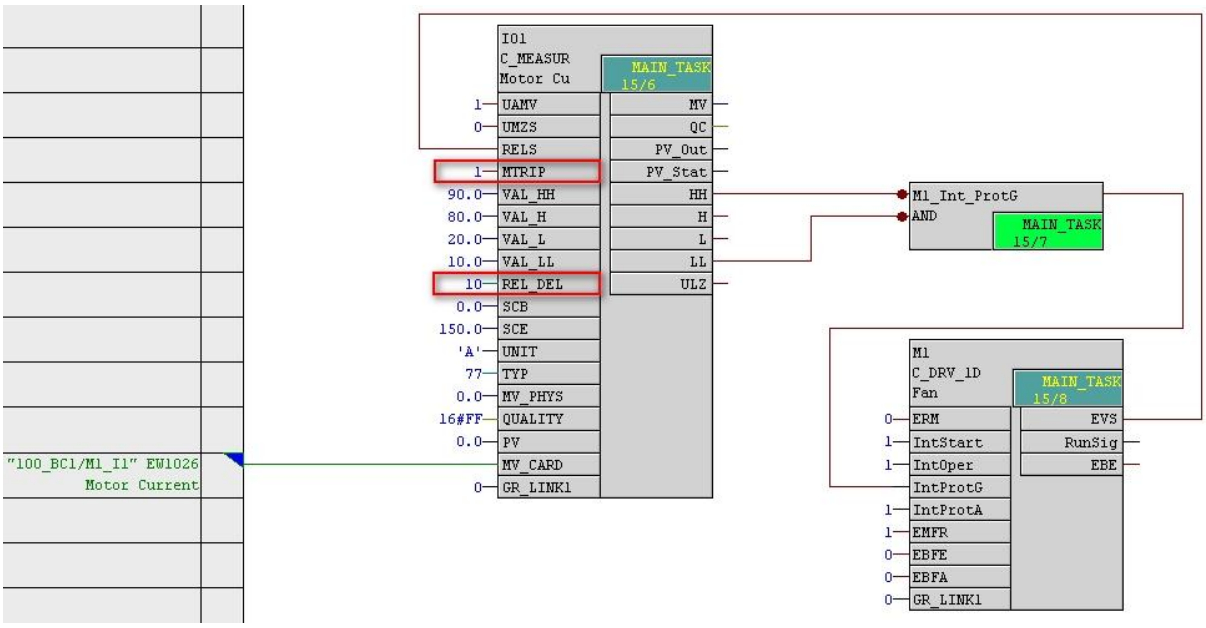
In the Service mode the measured value is still read but all limit bits are forced to “0“.

Release limit supervision

For some analog values the limit supervision must only be enabled if the associated device is running.

The following example shows the supervision of a motor current. Violation of lower limit 2 as well as violation of upper limit 2 shall stop the drive by protection interlocking.

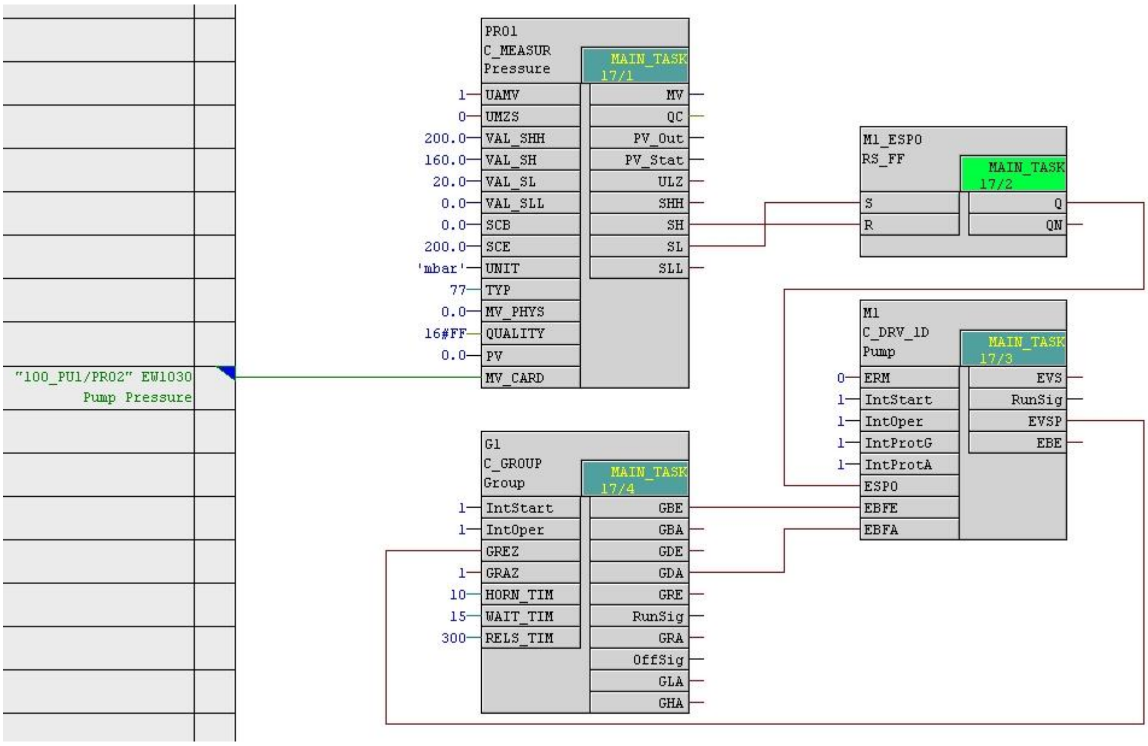
The evaluation should begin 10 seconds after the drive start.



- The limit supervision is enabled via connection of interface RELS with the running signal EVS of the corresponding drive. The time delay of 10 seconds is configured at parameter REL_DEL.
- With 1-Signal at interface MTRIP you achieve that after the drive stop the fault is memorized until acknowledgement. If MTRIP has 0-Signal, immediately after the drive stop the status of the measured value block is ‘good’ again.

Pressure control

A pump shall be switched on if the minimum pressure is reached and switched off with maximum pressure.



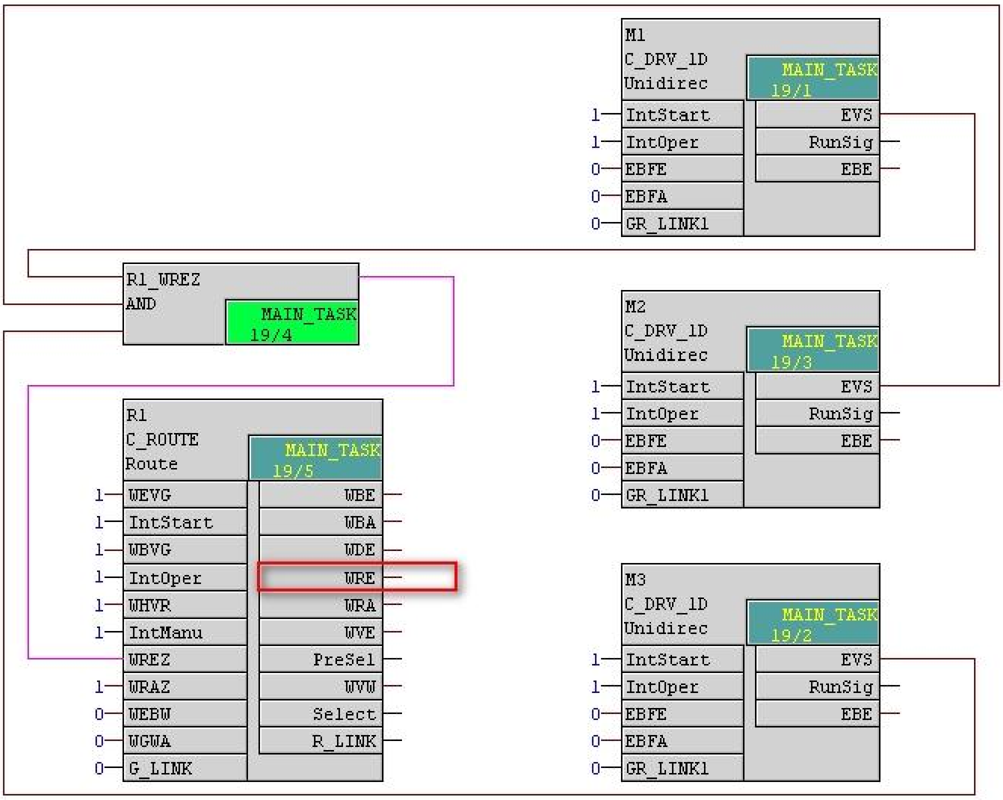
- The switching limit ‘SL‘ is used for switching ON and the switching limit ‘SH‘ for switching OFF.
- Start and stop of the pump is carried out via interface ESPO.
- For the feedback to the group EVSP is used instead of EVS.

Route functions

Feedback ON (all drives running)

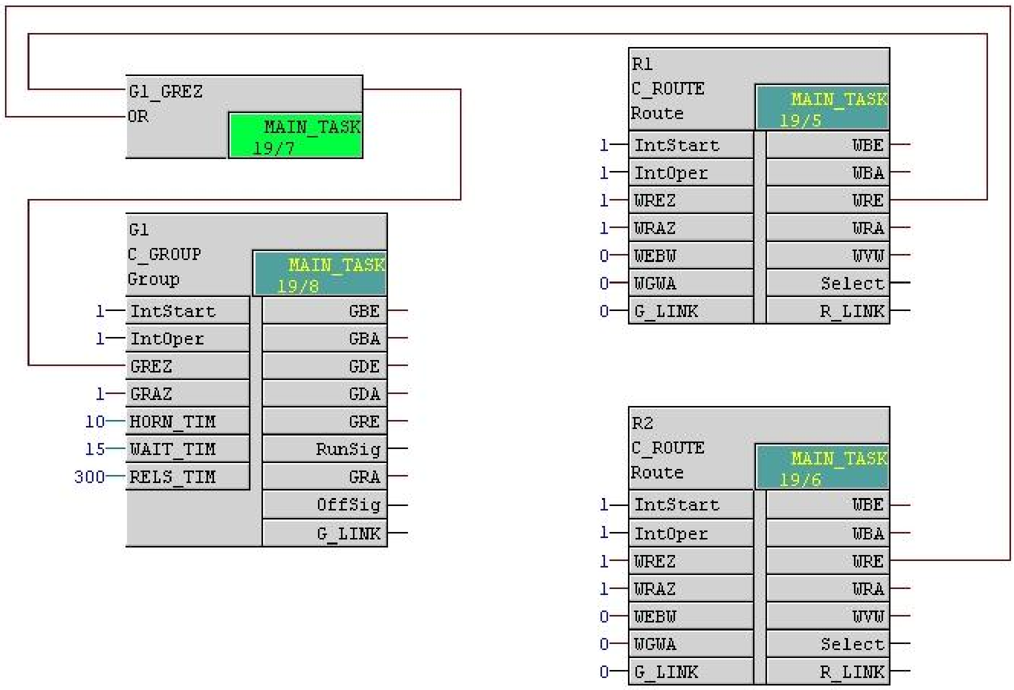
Via the route feedback ON (interface WREZ) the route is informed that the start-up has been completed and all drives are running.

Depending on the application this can be an AND-Function with the running signal of all drives, or the last drive of a start sequence.



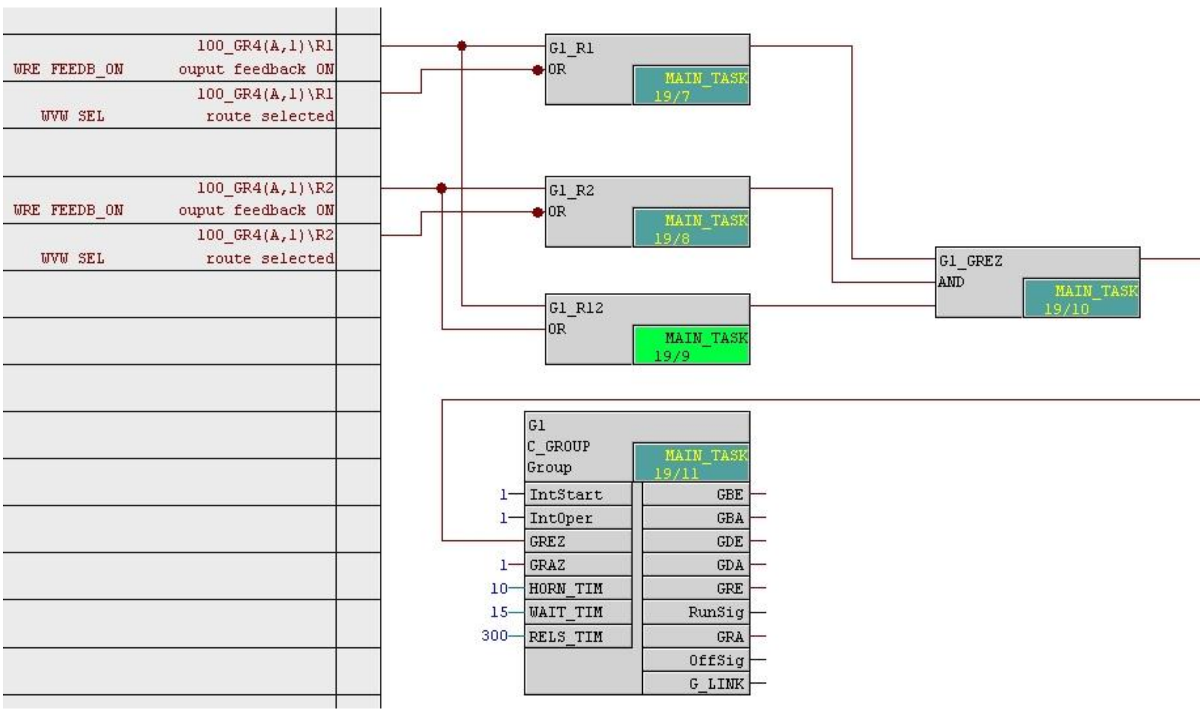
With 1-Signal at input WREZ, for a selected route the binary output WRE is set, which is used as feedback to the group.

a) Feedback to the group if one out of two routes must be running:



b) Feedback to the group if both routes may run at the same time:

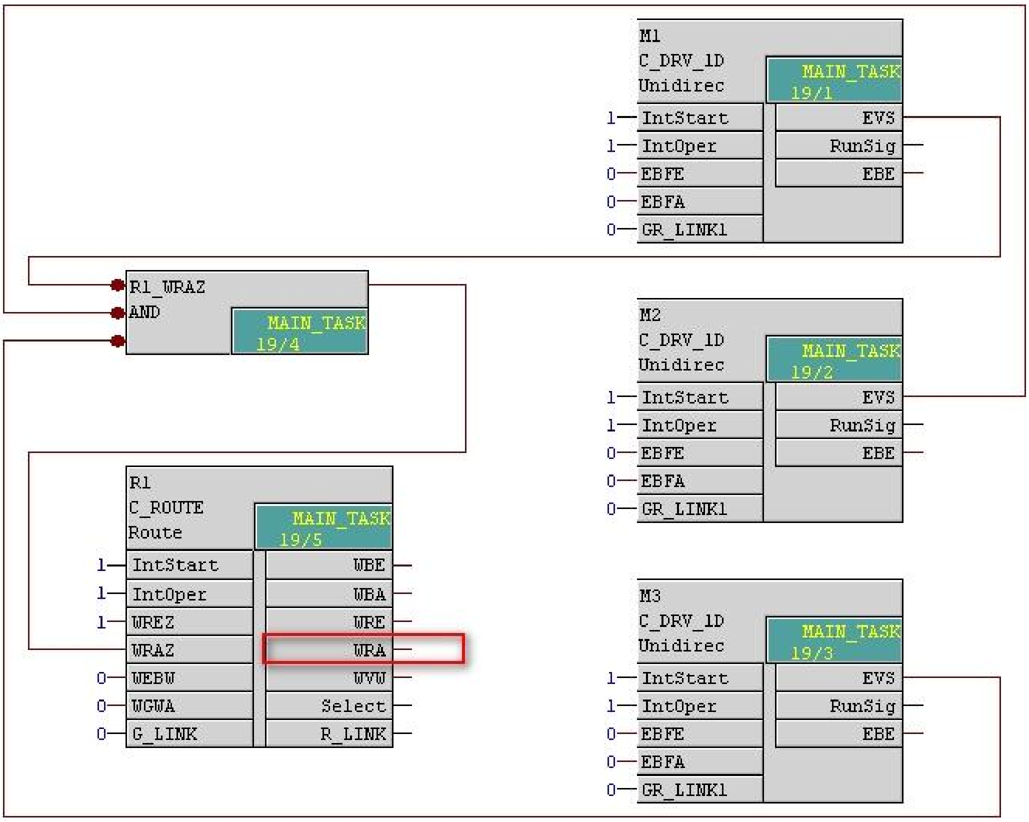
If one route is already running and a second group shall be started additionally, you have to take care that with the selection of the new route the group feedback gets “0“. Otherwise restart of the group is not possible.



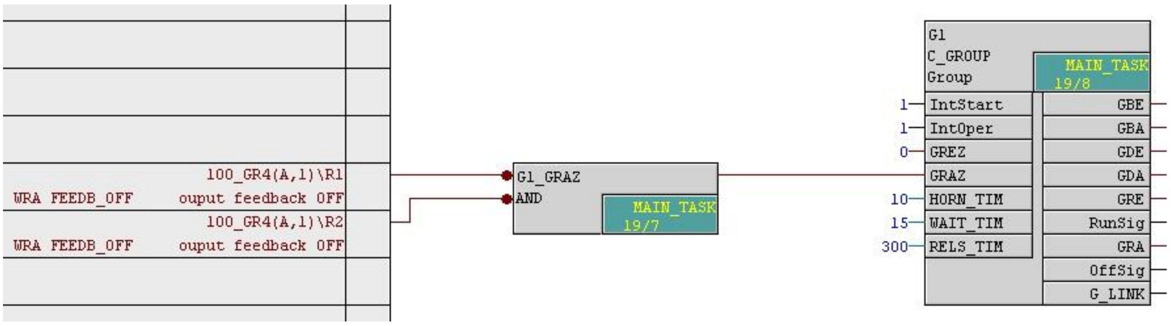
Feedback OFF (all drives stopped)

Via route feedback OFF (interface WRAZ) the route is informed that the stop sequence has been completed and all drives are stopped.

Depending on the application, this can be an AND-Function with the inverted running signal of all drives or with the last drive of a stop sequence.

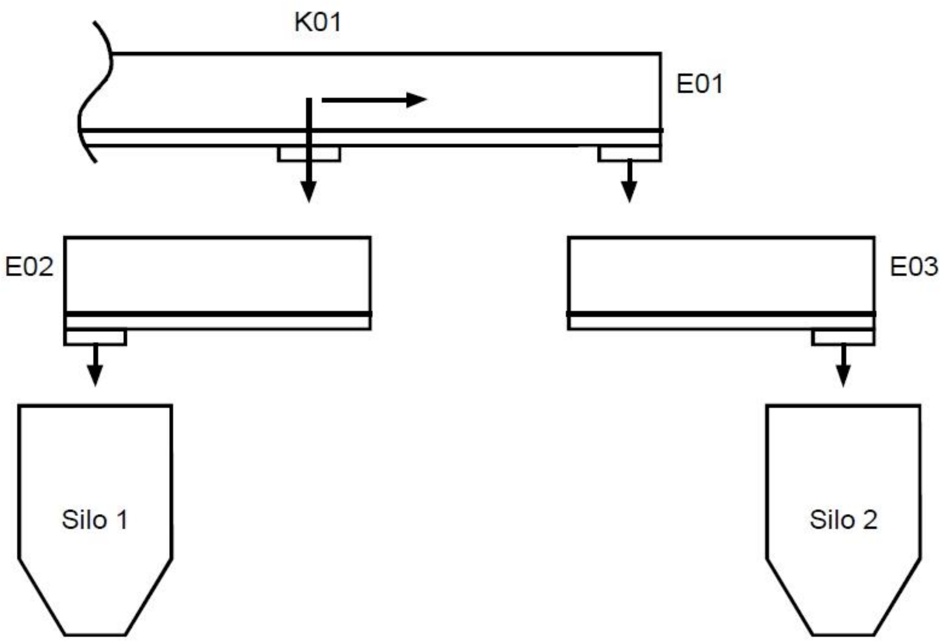


With 1-Signal at input WRAZ the binary output WRA which is used for the feedback to the group is reset.



⚠ The route feedback OFF (Signal WRA) gets "0" if the route is not selected or completely stopped. For the connection of the group feedback OFF the output WRA must therefore be inverted!

Uninterrupted route change-over



Task:

The change for charging into Silo1 or Silo 2 has to be without interrupt, i.e., the feeding drives should not be stopped during the switching operation.

Both air slides E02 and E03 must run while the damper is being moved. The part of the old transport direction that is no longer required is switched off only when the new direction has been traversed completely.

The following connection example provides a solution for a transport group with two routes:

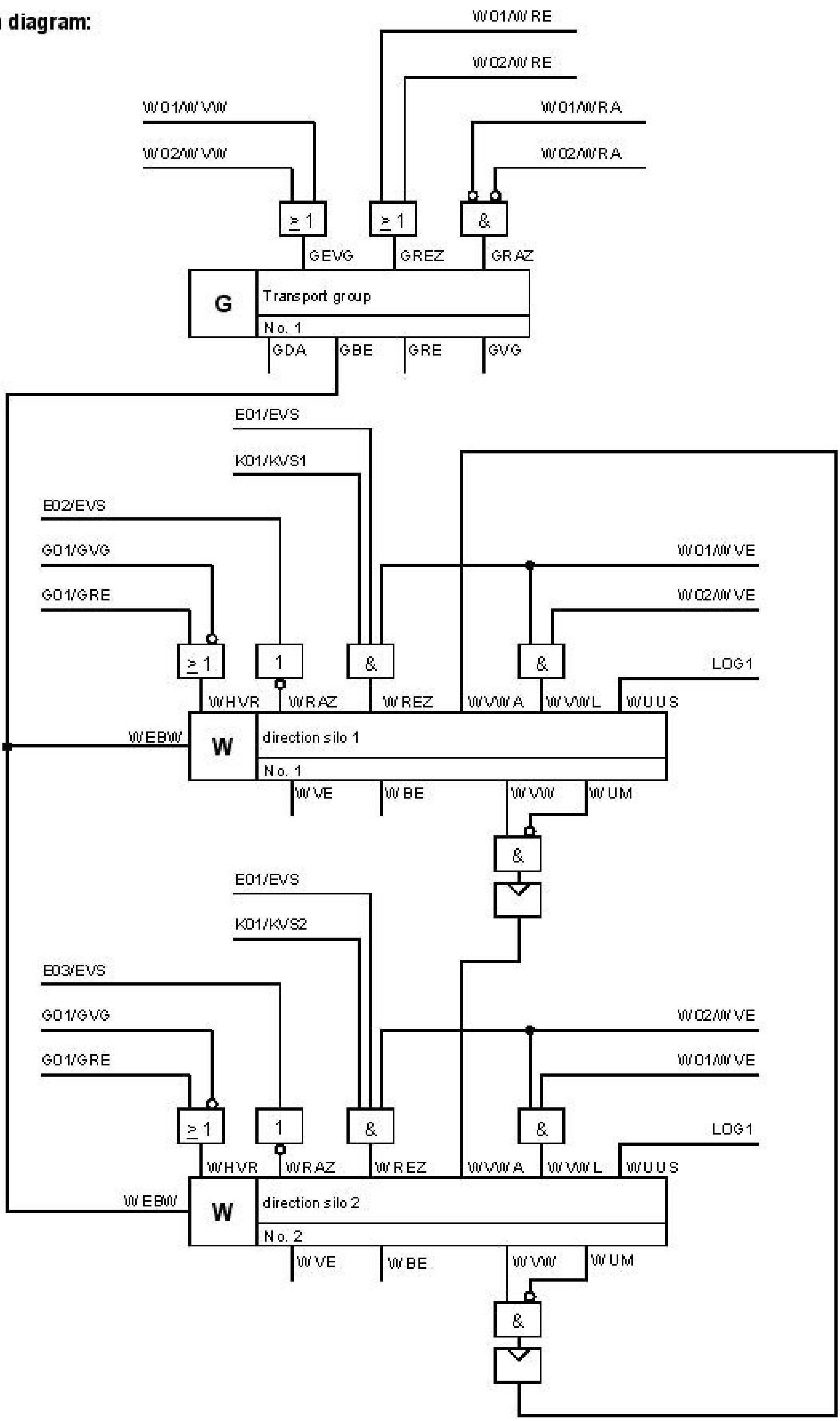
The air slides E02 and E03 and the damper position K1 (direction silo 1) belong to route 1.
The air slides E01 and E03 and the damper position K2 (direction silo 2) belong to route 2.

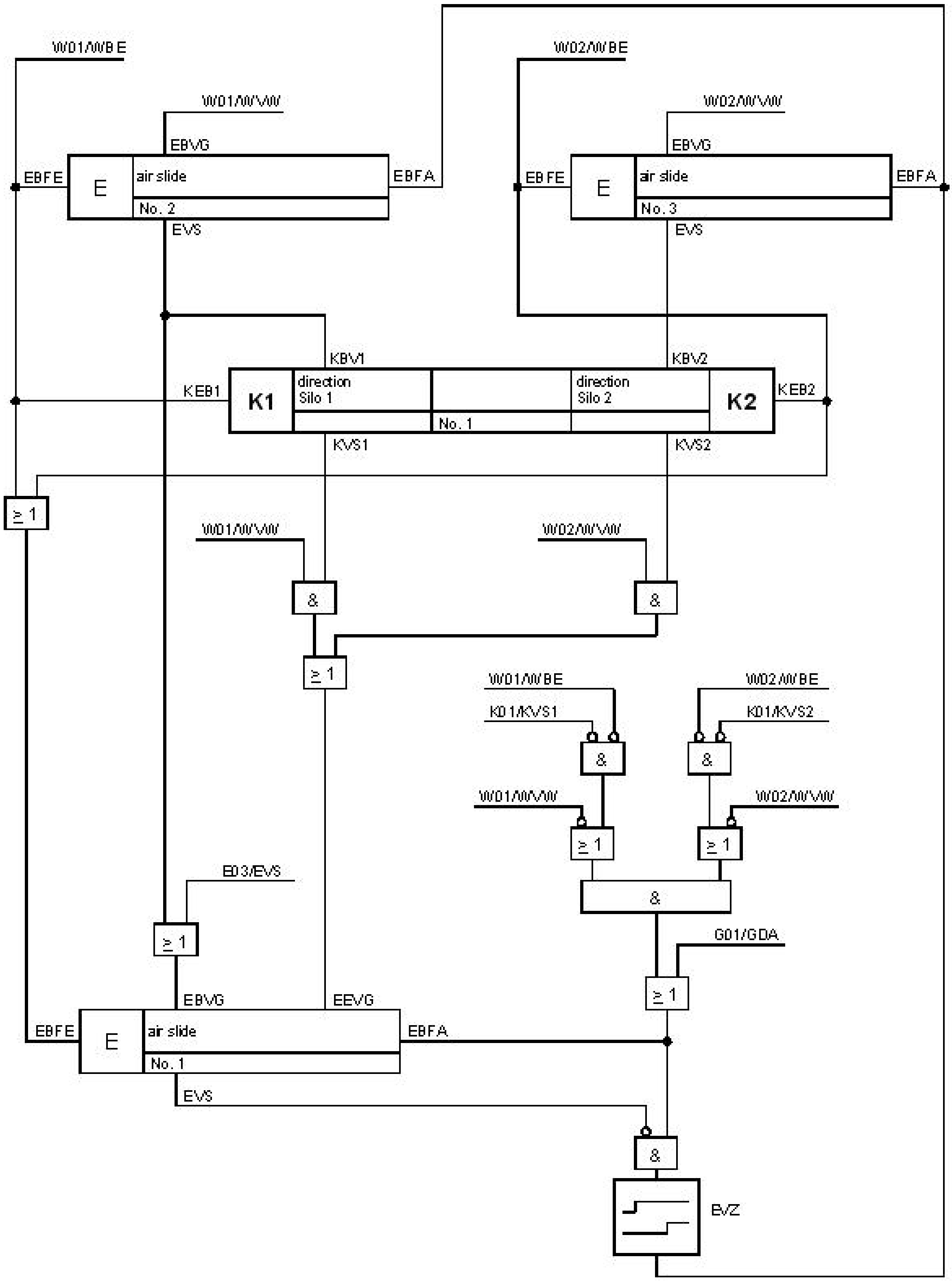
Operation:

During the switching, the new route must be selected first and then the group started.

Whether the route selection and the group start takes place manually or automatically is not relevant.

Function diagram:




W01/WBE
W01/WVW
EBVG
EBFE
E
air slide
No. 2
EBFA
EVS
W02/WBE
W02/WVW
EBVG
EBFE
E
air slide
No. 3
EBFA
EVS
KBV1
KBV2
KEB1
K1
direction
Silo 1
No. 1
direction
Silo 2
K2
KEB2
KVS1
KVS2
≥ 1
W01/WVW
W02/WVW
&
&
≥ 1
W01/WBE
W02/WBE
K01/KVS1
K01/KVS2
&
&
W01/WVW
W02/WVW
≥ 1
≥ 1
&
E03/EVS
≥ 1
G01/GDA
≥ 1
EBVG
EEVG
EBFE
E
air slide
No. 1
EBFA
EVS
&
EVZ

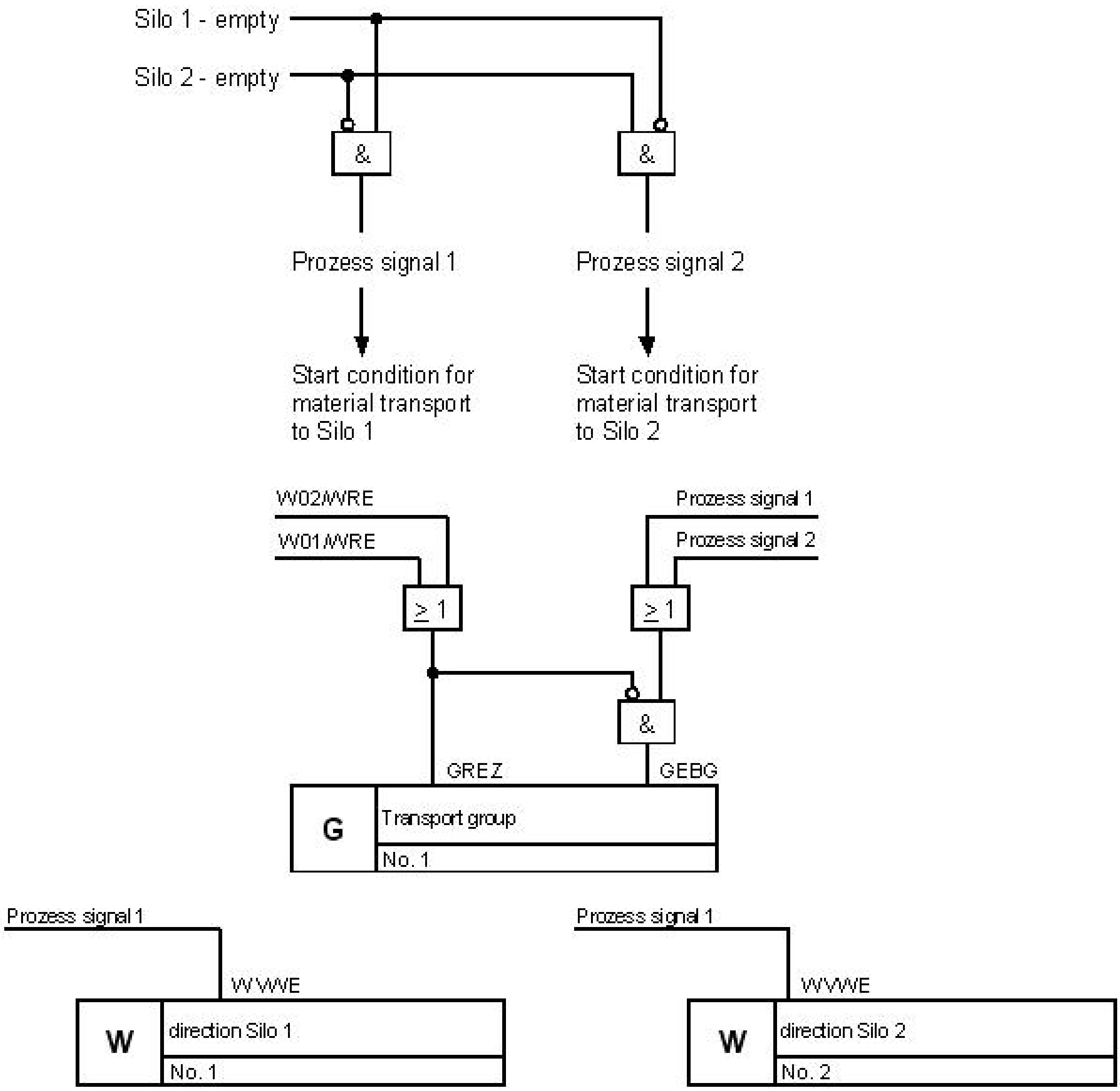
The connection example becomes clearer by looking at the individual conditions that must be fulfilled for a change of routes without interrupt.

- The route change can only take place when the group is stationary or runs completely
  ⇒ refer to WHVR
- If a route was pre-selected and the pre-selection of another route is added, then the own pre-selection must be removed.
  ⇒ refer to WVWL and WUUS
- In case of a route change without interrupt the group has to be started while it is completely running (because the drives of the “old” route are still running).
  To permit the group to give the start command again, the “group feedback On / route feedback On“ must be removed. This is done by linking the route pre-selection WVE with the drive feedbacks.
  ⇒ refer to WREZ
- The old route is deselected automatically once the newly started route runs completely.
  ⇒ refer to WVWA
- The de-selection of the “old” route stops the drives that are no longer required.
  ⇒ refer to EBVG from E02 and E03
- The disconnection of the complete group is performed only for group stop (use GDA signal) or for a fault if the switching has not been performed within a specified time.
  ⇒ refer to EBFA from E01

Automatic Route change

In this example the route change and start is not performed manually but with a process signal (e.g., silo filling level). The process signal (start condition) **must be unique**. If necessary, use an interlock to ensure that only one signal is pending!

Example:



The process signal automatically pre-selects the route.
⇒ Refer to WVWE

Note: The automatic group-On command must be supplied only when the GREZ signal of the group becomes zero (otherwise no start is possible).
⇒ refer to GEBG.

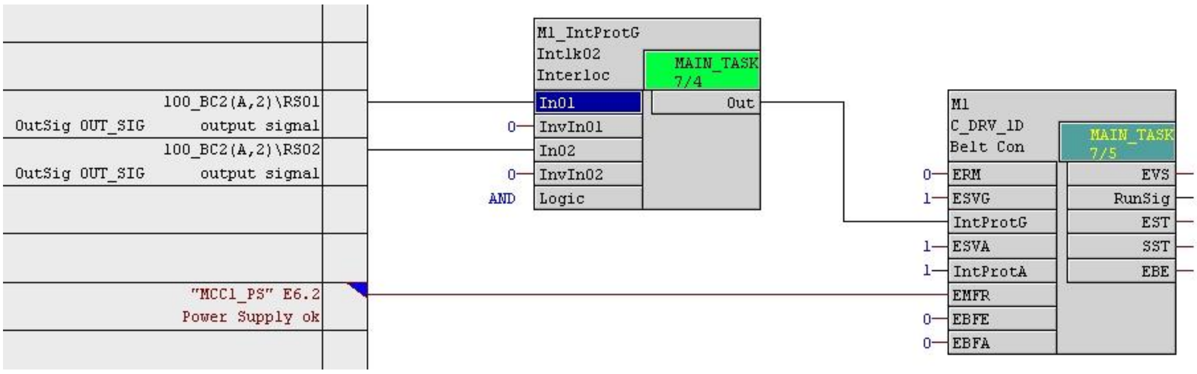
Engineering Examples for additional blocks

Display of interlocking conditions with interlock blocks

Via interlock blocks interlocking conditions can directly be called from the diagnostic window of the Cemat block.

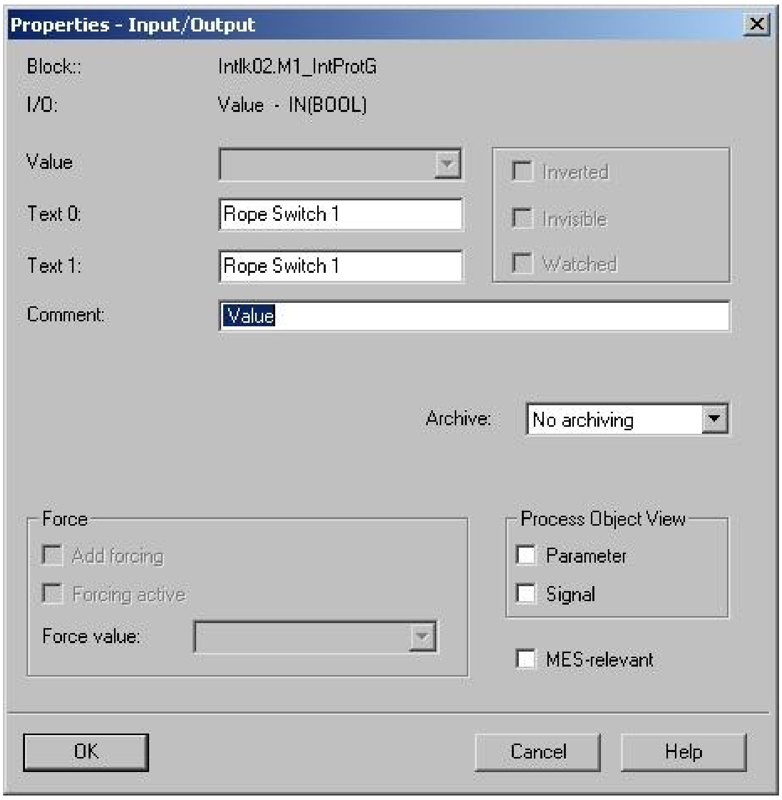
Using structure interlocks the faceplate of the interlock block can be opened directly from the faceplate of the Cemat block. (It is also possible to open a connected Cemat block instead.)

a) Structure interlocks Intlk02, Intlk04, Intlk08 and Intlk16

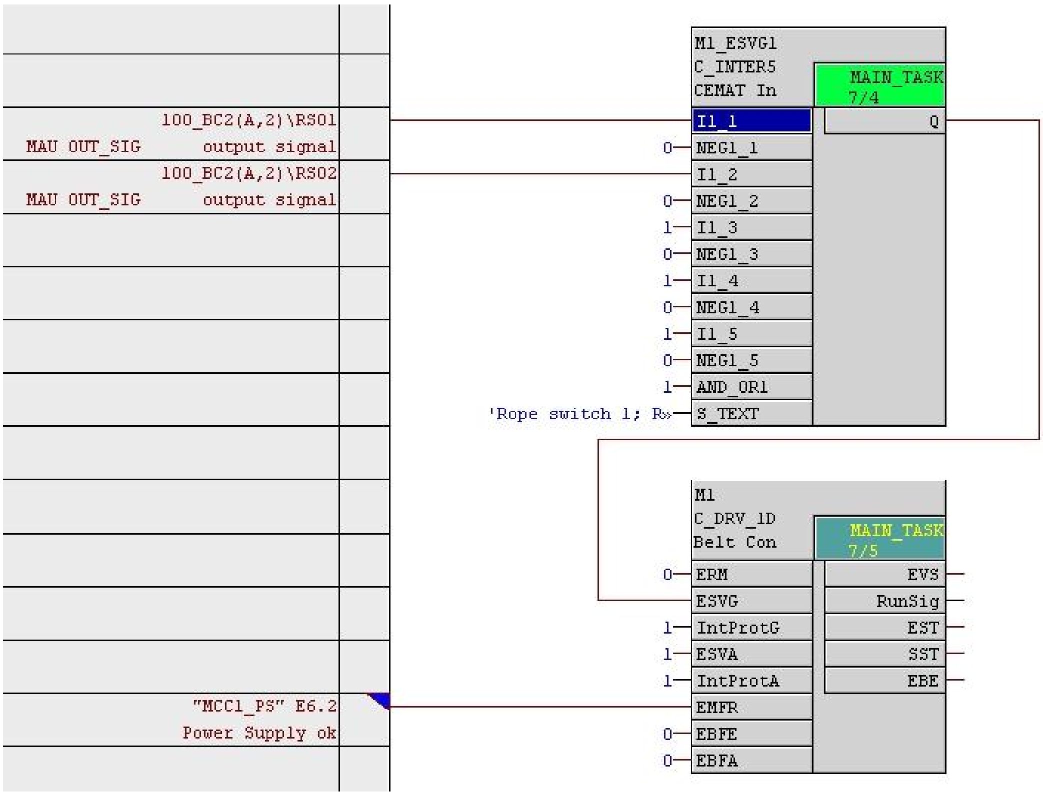


The name of the Interlock block is not relevant.
Exception: don't use the naming conventions for binary interlock blocks (in this case do not use M1_IntProtG1), otherwise the interlock faceplate will be opened two times.

In the faceplate of the interlock block at each input the name of the connected block is displayed automatically, here e. g. 100_BC2/RS01. For additional information you can enter a text at the corresponding input under Object properties → Text 0 and Text 1 (maximum 16 characters).



b) Binary interlock block C_INTER5 (can only be called from diagnostic window)

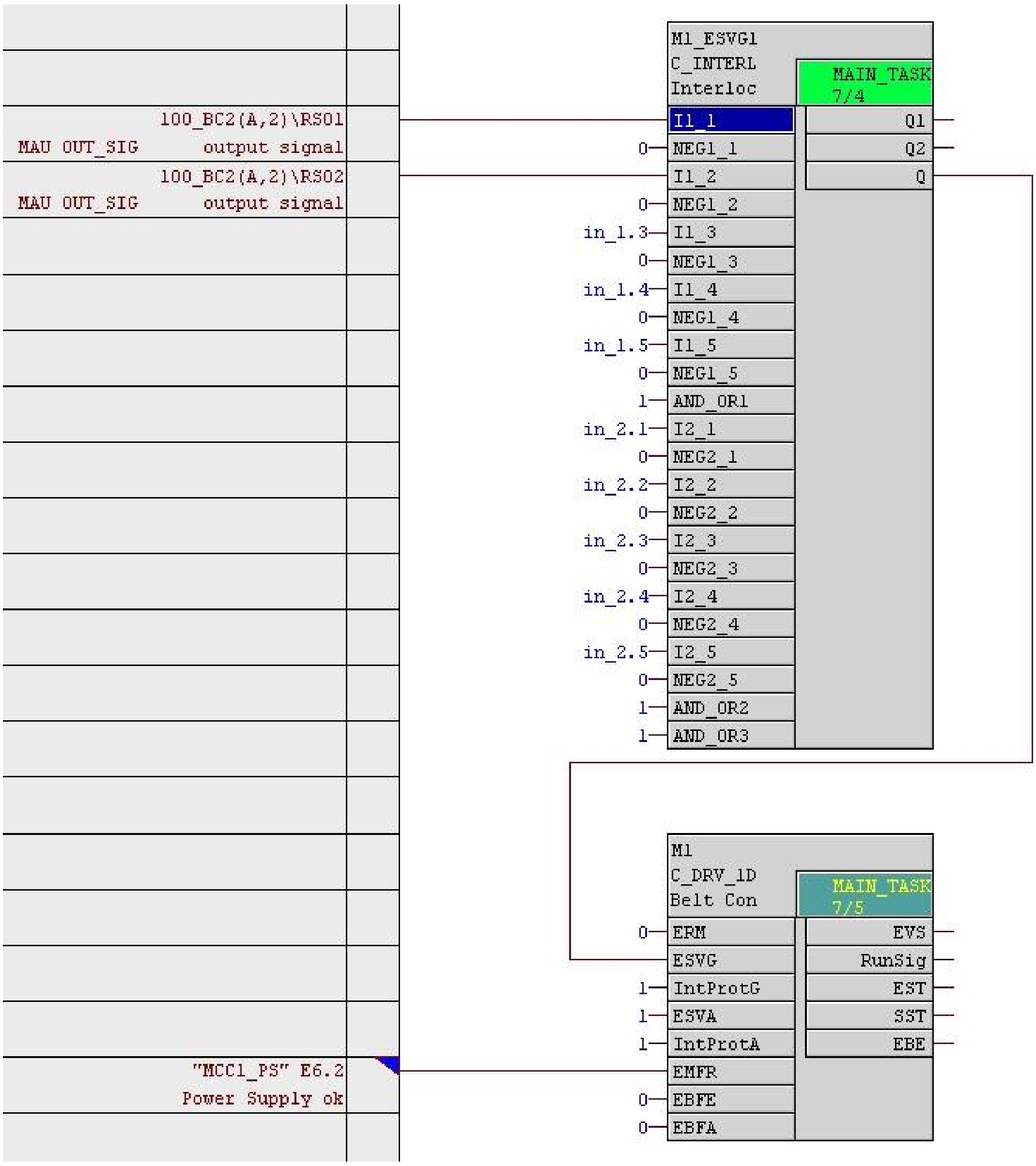


C_INTER5 is called via the variable name. The interlock block must therefore be named as follows:
Name of the Cemat block + Underline + Name of the interface + Index (1/2/3).

You can connect 3 blocks of type C_INTER5 to each interface.

The text for the inputs is configured via parameter S_TEXT (5 texts a 16 characters, separated by semicolon).

c) Binary interlock block C_INTERL (can only be called from diagnostic window)



C_INTER5 is called via the variable name. The interlock block must therefore be named as follows:
Name of the Cemat block + Underline + Name of the interface + Index (1/2/3).

You can connect 3 blocks of type C_INTERL to each interface.

The text for the inputs is configured via parameter S_TEXT (5 texts a 16 characters, separated by semicolon).

The texts for the inputs are defined via object properties of the corresponding input (I1_1 to I1_5 and I2_1 to I2_5) under Text 1 (text length max. 16 characters).

Caution: A mix of C_INTER5 and C_INTERL at the same interface is not permitted.

Query settings within the AS

Via the function "Object Browser" the operator can search for simulations and operation mode settings within an AS. See description 08_OS_Engineering_009.pdf under “General Cemat Functions“.

The function for left mouse click must be adapted (enter number of the AS):

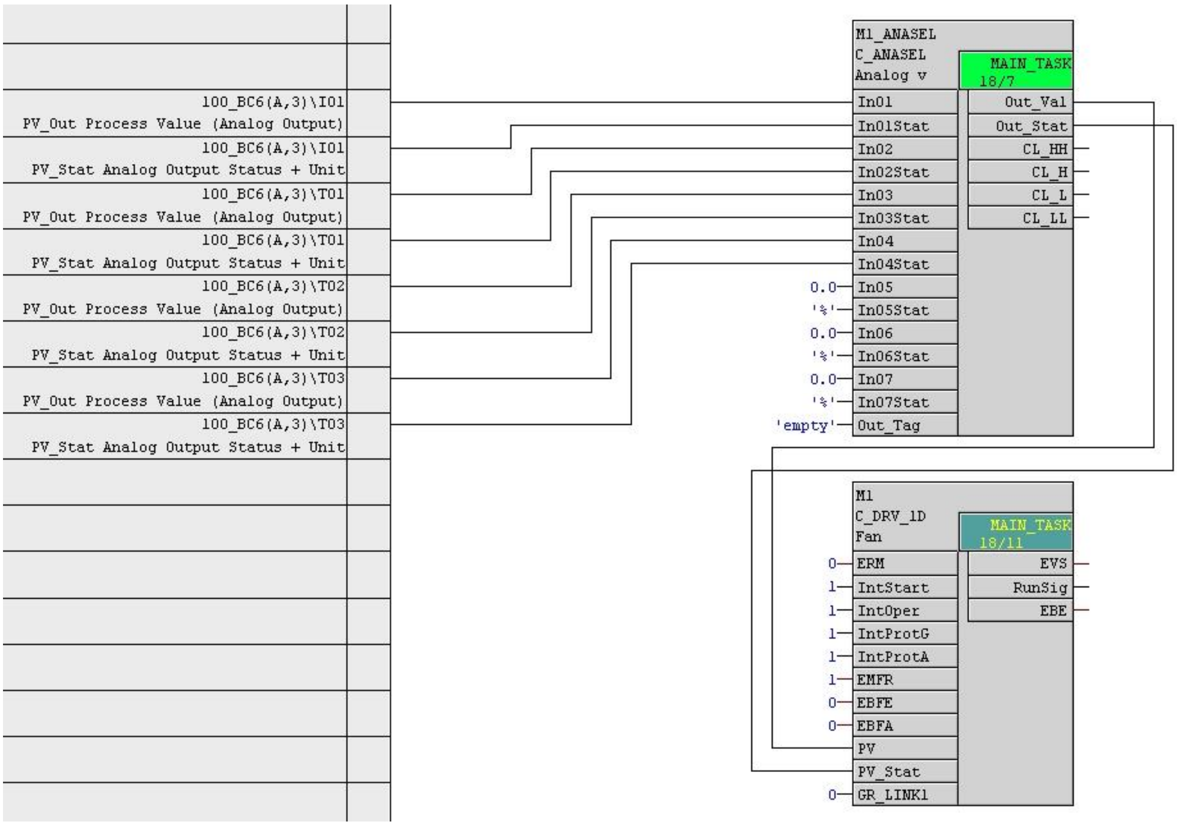
```
long lPlcNo = 1;          //The number of the PLC. (e.g.: 1 ==> SYSPLC01; 12 ==> SYSPLC12; ...);
```


Analog Value Selection

Via block C_ANASEL up to 16 measured values can be linked to a drive C_DRV_1D, C_DRV_2D or C_DAMPER.

Via the drive faceplate the faceplate of the C_ANASEL block can be opened, and from there you can select the desired measured value.

In the drive faceplate you will see the selected measured value + Unit as well as the status (limit violation, simulation, quality).



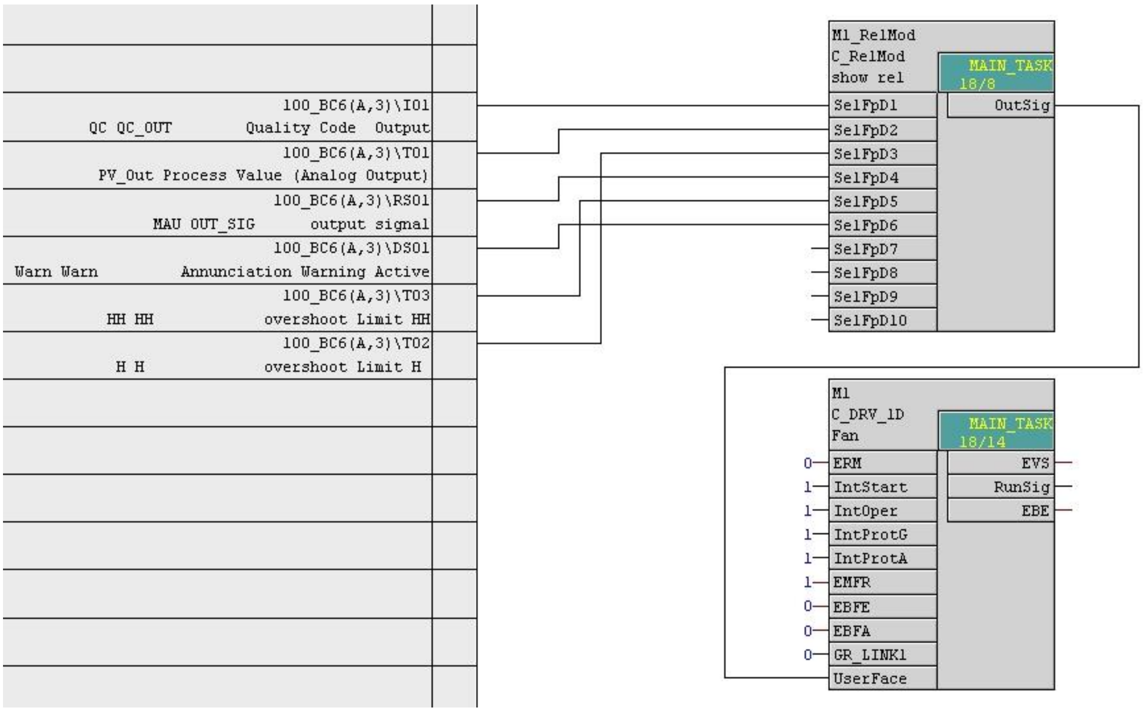
- You have to connect output PV_Out of the measured value to input Inxx of the C_ANASEL and output PV_Stat of the measured value to input InxxStat of the C_ANASEL.
- The output Out_Val of the C_ANASEL must be connected to input PV of the drive and the output Out_Stat of the C_ANASEL to input PV_Stat of the drive.

Show related objects

Via interface UserFace of the Cemat blocks any block can be linked to the Cemat block and its faceplate can then be opened through the Cemat block. This possibility exists for C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE C_GROUP, C_ROUTE, C_SELECT and C_ANASEL.

The link has no influence in the control function, it has only for visualization purpose.

In the following example the block C_RelMod is connected to a unidirectional drive. The block C_RelMod has 20 inputs over which further blocks can be assigned to the drive.



- Connect any output of the desired block to input SelFpDx of C_RelMod.
- Connect output OutSig of C_RelMod to input UserFace of the Cemat block.

AS Engineering

Content

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\06_AS_Engineering_009.doc

General

The project settings which have to be carried out during the installation of a new PCS 7 project are no longer described here. These steps you can find in chapter 3 of this manual (03_PCS7-Project).

Chapter AS-Engineering explains the general settings in the System chart, and all required steps for the application programming in the CFC.

Before you start with the application programming have to create a symbols file.

Creating Templates (Process Tag types) in the master library avoids repetition of the same settings in all instances and ensures consistency of the program.

System Chart SYSPLCxx

During the creation of the PCS 7 Project the System Chart has been copied from the Project Library into the Chart folder of the AS and renamed according to the PLC Number:

SYSPLC01 for AS1

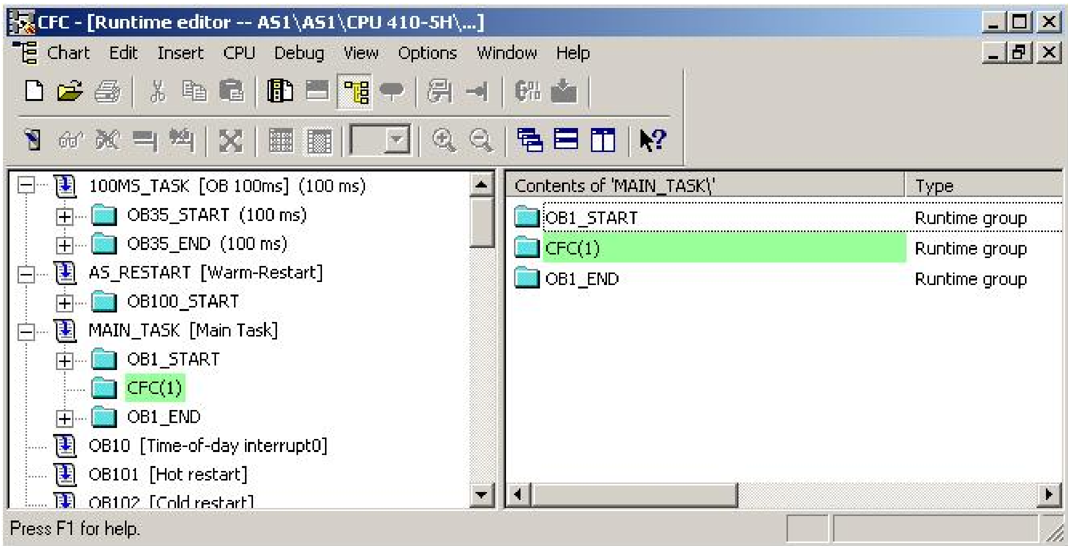
SYSPLC02 for AS2

SYSPLC03 for AS3

….

Note: The System Chart must be installed in the Plant Hierarchy, which is assigned to an OS Project; otherwise no Tags from System Chart are created in the Tag Management.
Renaming the System Chart different than SYSPLCxx is not permitted and will not work.

Make sure that the blocks from system chart are installed at the correct position in the Runtime sequence, which should be as follows:



The Cemat blocks need to be installed in OB1, between Runtime group OB1_START and OB1_END.

The System Chart contains a lot of general settings for AS Program, definitions which are valid for all Cemat blocks. These are e. g.:

- The Acknowledgement mode
- The Warning mode
- The PLC number
- The Sequence Test Mode

Acknowledgement mode

Beside the message in the alarm line, the Cemat block as well needs acknowledgement, in order to reset the dynamic fault.

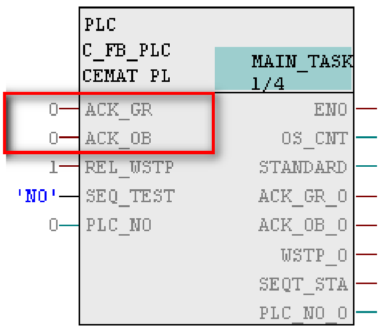
The **messages in the alarm line** can only be acknowledged one by one,

- either via the acknowledge button in the alarm line (only the visible alarms will be acknowledged) 
- or via the acknowledge button in the Object Faceplate. 

For the **fault acknowledgement in the AS** (dynamic fault of the Cemat blocks) different options exist. It may be carried out

- together with the acknowledgement of the alarm line or via acknowledgement button in the object faceplate.
- for the complete AS, group-wise or for the individual object.
- via HMI or by program

The general configuration of the fault acknowledgement from HMI is configured in the System chart SYSPLCxx at block C_FB_PLC, interfaces *ACK_GR* and *ACK_OB.* The settings are valid for the complete PLC and the default setting is "AS-wide acknowledgement".



AS-wide acknowledgement (default):

With the acknowledgement of the message in the alarm line, an acknowledgement command is sent to the AS which produced the message, and all blocks within this AS are acknowledged at the same time.

Via the acknowledgement button of any Cemat Object Faceplate an acknowledgement command is sent to the corresponding AS, and all blocks within this AS are acknowledged at the same time.

Parameter Setting on C_FB_PLC

ACK_GR = 0
ACK_OB = 0

Group-wise acknowledgement

The acknowledgement of the message in the alarm line is independent of the fault acknowledgement in the block.

For fault acknowledgement in the AS the Operator must press the Acknowledgement button in the group faceplate. The acknowledgement command is sent only to the group. From the group it is transmitted to all connected objects.

In order to acknowledge the objects by the corresponding group you have to link the output ACK of the group with the interfaces EQIT, VQIT, KQT1 etc. of all objects belonging to the group.

Acknowledgement of individual Objects via its faceplate is possible as well.

Parameter Setting on C_FB_PLC

ACK_GR = 1
ACK_OB = 0

Object-wise acknowledgement

The acknowledgement of the message in the alarm line leads to the acknowledgement of the block which generated the alarm.

In the mode the acknowledgement of individual Objects via its faceplate is possible as well.

Parameter Setting on C_FB_PLC

ACK_GR = 0
ACK_OB = 1

Caution: In case of protection interlock of a drive, the message is created by an annunciation block or measure, not by the drive itself. In this case the Acknowledgement of the alarm line does not acknowledge the drive fault. The drive faceplate must be opened and Acknowledge Button must be pressed there.

Fault acknowledgement by AS Program

All Cemat Objects have an interface xQIT for individual fault acknowledgement by AS program. This interface is used in case of group-wise acknowledgement, but in can also be used wherever acknowledgement by AS program is needed.
Interface xQIT works parallel to the acknowledgement command from HMI.

Parameter Setting on C_FB_PLC

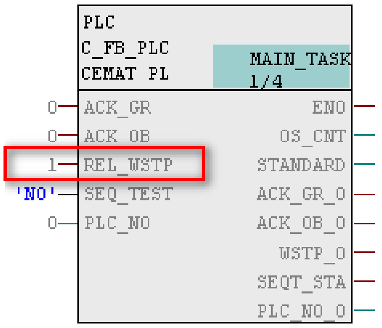
Irrelevant

Warning mode

In the system chart SYSPLCnn at block C_FB_PLC the reaction of the Group in case of a warning during start-up can be configured.

If Parameter *REL_WSTP* = 1	the (dynamic) warnings interrupt the group start
If Parameter *REL_WSTP* = 0	the (dynamic) warnings don't interrupt, the group start will continue.

Since the function exists only from Cemat V7.1, in the default settings for the system chart SYSPLC00 Parameter *REL_WSTP* has 1-Signal (because of the compatibility to previous versions).



Note: (Dynamic) faults in any object of the group always interrupt the group start!

Holcim Standard: Parameter REL_WSTP at block C_FB_PLC must be connected with 0-Signal, otherwise in case of a warning message the group start gets interrupted.

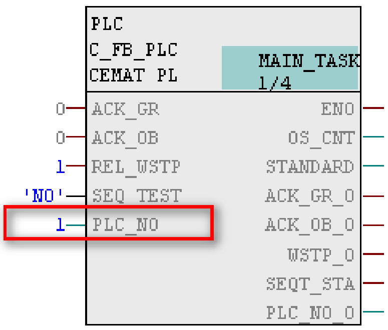
PLC Number

Parameter PLC_NO of block C_FB_PLC is only needed in case of object links to a group in a different AS (see object description of C_GROUP).

In this case parameter *PLC_NO* must be set to the same number as the System chart:

Example:

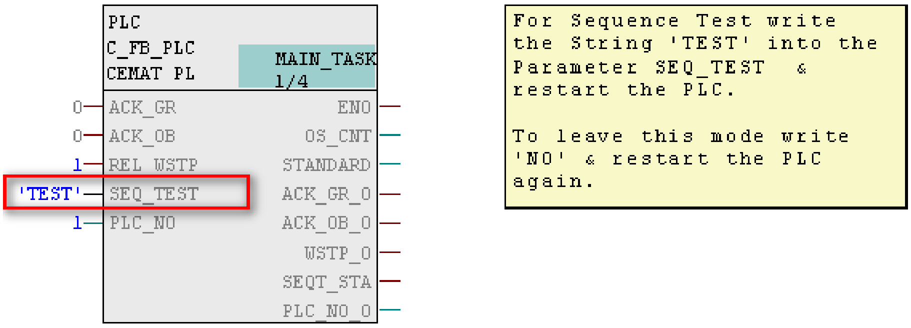
In the chart SYSPLC01 set a "1" on PLC_NO.
In the chart SYSPLC02 set a "2" on PLC_NO.
In the chart SYSPLC03 set a "3" on PLC_NO.
etc.



Sequence-Test

In the Sequence test mode the program can be tested without Hardware inputs and outputs. It is a pure simulation mode and can only be activated or deactivated with a restart of the AS.

Start and Stop of the Sequence test mode is carried out in system chart, block C_FB_PLC, via Parameter SEQ_TEST.



To start the sequence test mode, enter string ‘TEST’ on input parameter SEQ_TEST and restart the AS.

To leave the sequence test mode, enter string ‘NO’ on input parameter SEQ_TEST and restart the AS.

In sequence test mode, for all Drive and Annunciation blocks the output SIM_ON is set. If module drivers are used, the output SIM_ON can be connected to input SIM_ON of the driver block in order to use the Simulation Value. See also “Using driver blocks with CEMAT”.

In case of the Motor, the feedback of the contactor and eventually a speed monitor are simulated. In case of damper or valve, the limit switches are simulated.

All further hardware inputs are still active and must be simulated via test program at the beginning of OB1 Cycle.
If module drivers are used, the output SIM_ON of the CEMAT block can be connected to input SIM_ON of the driver block in order to use the simulation value.

Conventional control desks

Some clients still like to use conventional control desks for certain applications. In this case block C_PUSHB must be called. C_PUSHB has input parameters for Release, Fault Acknowledgement, Horn Acknowledgement and Lamp Test, as well as an output which can be connected to a horn.

Note: The inputs and outputs connected to C_PUSHB are always effective for the complete AS. If Fault Acknowledgement, Lamp Test or Release Functions should work only for specific CEMAT objects, the interfaces at C_PUSHB cannot be used. You have to connect the signals to the xQIT, xLPZ and xFGS interface of the CEMAT Object itself.

Module Parameters of C_PUSHB

Input parameters

FGS **Release Button** **Basic State: 0-Signal**

Format BOOL

The Release Button must be pressed together with Group Start, Group Stop, Route Selection etc. (2-Hand-Operation).

QT **Acknowledge Faults** **Basic State: 0-Signal**

Format BOOL

With a positive Edge on QT all dynamic faults in the AS are acknowledged.

QT_H **Acknowledge Horn** **Basic State: 0-Signal**

Format BOOL

With a positive Edge on QT_H the horn gets switched off.

LP **Lamp Test** **Basic State: 0-Signal**

Format BOOL

Parameter LP is used to parameterize a Lamp Test button for Running/Fault Lamps of the drives, Annunciation lamps, Group status indications and Route selection lamps.

THUP **Horn time in Seconds** **Basic State: 120**

Format INTEGER

After this time the horn is switched off.

Output Parameters

HORN **Horn**

Format BOOL

The output can be used for an acoustical alarm in the control room.

CEMAT Objects

Drive functions

C_DRV_1D	Unidirectional drive
C_DRV_2D	Bi-directional
C_DAMPER	Damper
C_VALVE	Valve
C_SIMOS	Adapter for SIMOCODE (with Curve Display)

Annunciation functions

C_ANNUNC	Annunciation module
C_ANNUN8	Annunciation module with 8 Alarms
C_MEASUR	Measuring value
C_ANASEL	Analog value selection
C_STORAG	Storage module
C_STO_MA	Storage module (Master for Multichamber)

Control/Supervision

C_GROUP	Group module
C_MUX	Additional block for group/route
C_ROUTE	Route module
C_SELECT	Selection module

Controller functions

CTRL_PID	PID-Controller
C_PID3	PID-Controller with 3 Parameter tables
CTRL_S	Step Controller
RATIO_P	Ratio Controller
C_POLY3	Polygon Module

Caution: CEMAT V7 has no special block for analogue output function. You have to use the PCS 7 driver block CH_AO.

Silo pilot function

C_SILOP	Silo pilot

Information system

C_RUNNT	Runtime supervision (MIS)
C_COUNT	Counter block (MIS)
C_MEAS_I	Measuring value integration (MIS)

Special functions

C_ADAPT	Adapter block to include non-CEMAT modules into group supervision.
C_PUSHB	Parameterization of a conventional Control Desk with Push Buttons and Lamps
C_RelMod	Show related Modules

The detailed description of the CEMAT Functions you find in the object description an in the online help. All other blocks from the PCS 7 standard library can be used as well.

Creating a Master Library

Message configuration

The messages are configured with the Function blocks and normally don't need to be changed; except if the customer wants to have different texts, e. g. adapted to his tagging system.

In this case the adaptations for the message configuration have to be done at the function block itself, in the block library.

For the status call function the message texts must be defined in the object specific Config files, section [Fault]. If the message configuration of a block is changed, the Config file of the corresponding block needs to be adapted as well.

Caution: Make sure that you copy the modified block to the program folders of all PLCs and be careful with updates:
- Whenever you get an updated block version the modifications in the message configuration have to be carried out again.
- Make sure that you configure the same texts in the message definition and in the Config files. The config files must be copied to all OS Stations.
- Take a backup of the Config files because they also get overwritten in case of an update.

Templates (Process Tag Types)

During creation of a new PCS 7 Project for Cemat a master Library has been created. Beside the blocks, the Symbols and the System Chart the master library will contain the process tag types.

A set of Process Tag Types you find in Library ILS_CEM, S7 Program TYPICALS. These Process Tag Types fit to Project Standard 000 and must be seen as Examples.

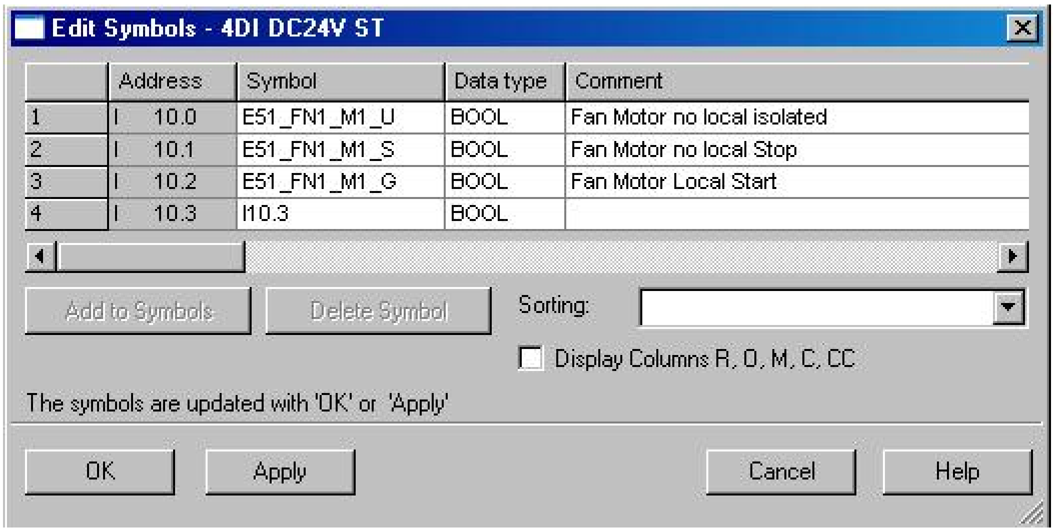
The adaptations may comprise the following functions:

- Other enable functions in the CFC
- Text variables for HMI display
- Message texts (e. g. for the drives)
- Standardized logic or connections for certain functionality

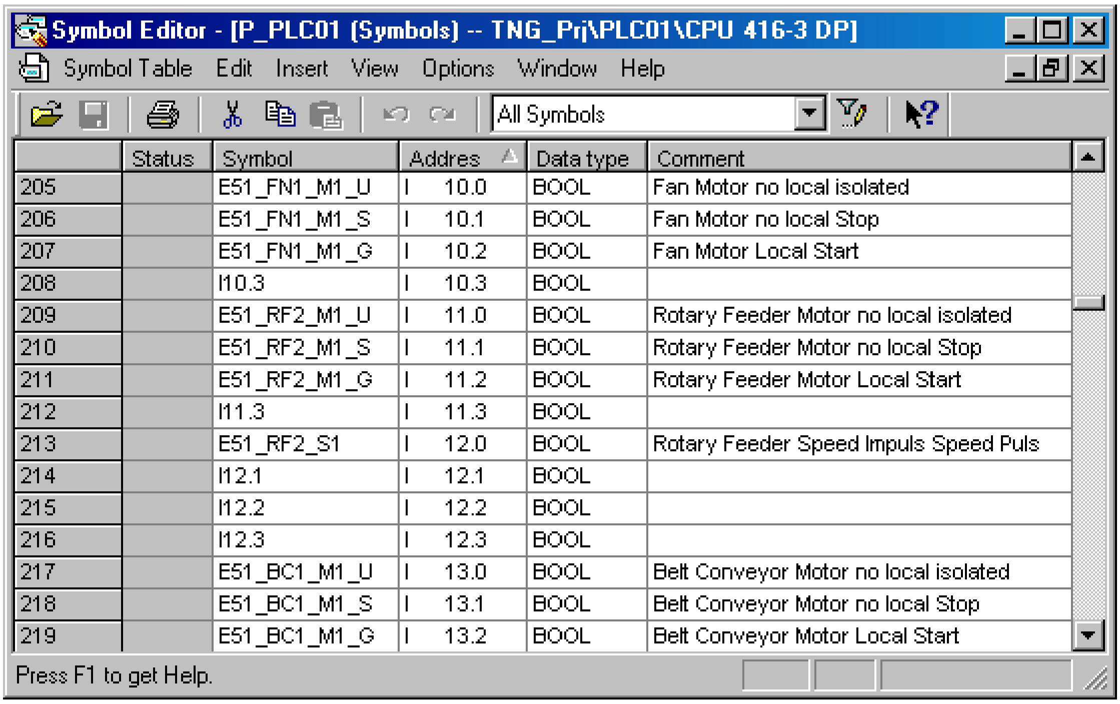
Recommendation: If an Engineering Tool is used, process tag types are anyway a <u>must</u>, but also for manual Engineering it is highly recommended to use template charts, otherwise the same project settings have to be carried out for each instance of the block.

Creating the Signal list (Symbols)

The hardware signals can be entered directly in the hardware configuration.



They will be added to the Symbols list and can further on be used in CFC.



There is also the possibility to import the hardware signal list from an excel file.

Creating the Application Program with CFC

How to add a new chart and insert a block

The program logic is written in the Continuous Function Charts (CFC). New CFCs can either be created manually or generated via Import/Export Assistant.

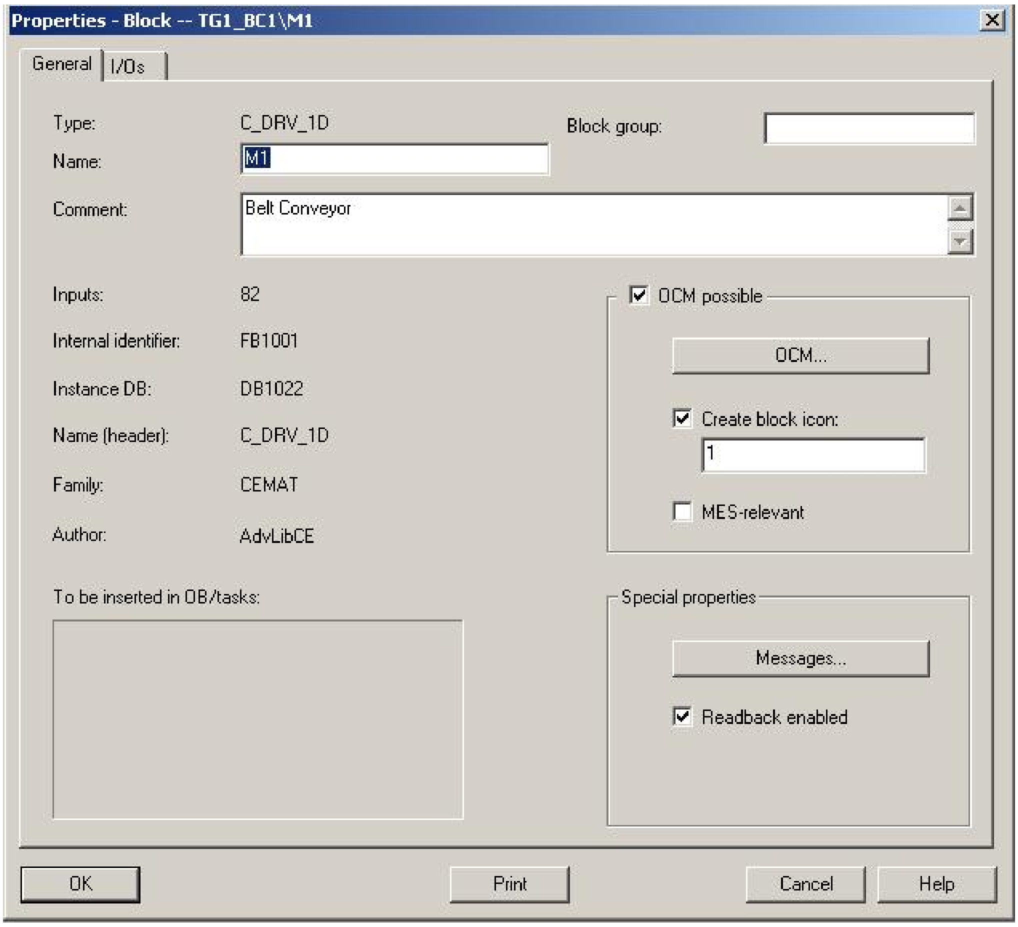
Here only the manual creation is described:

- To insert new charts use Plant view or Process object view. Later the charts can be edited from all views of the SIMATIC Manager.
- The easiest way to create a new chart is to *copy* one of the template charts from the master library and paste it to the proper position in the plant hierarchy. After this you have to rename this chart according to your tagging system. The chart name (e.g. 345_BC1) is the first part of the tagname.

  Advantage when copying: all blocks already have the correct default settings and internal connections. You just have to delete the blocks you don't need and modify the functions according to your functional descriptions.
- You may also create a chart by yourself and name this chart according to your tagging system and then drag and drop the blocks (e. g. from block folder CEMAT of block folder CONTROL). All the required settings and connections must then be done manually.

Block Properties

Double-click on the block in order to open the properties dialog.



Carry out the following settings in the property window:

Name:
The name (instance name, e.g. M1) is the second part of the tagname of the motor. If the Hierarchy folder name is not part of the tagname (project setting), the tagname of the motor consists of the chart name and the instance name. In the above mentioned example this would be 345_BC1/M1.
Make sure that the complete tagname does not exceed 20 characters.

Comment:
The comment contains an explanation of the motor, e. g. “Belt Conveyor”. This text is later on available in WinCC as internal variable #comment and will be displayed in the faceplate of the motor, in the status call and in the object list.
The maximum length of the comment is 40 characters.

Under “OCM possible” you find the settings regarding WinCC:

Create Block Icon:
Select the option “Create block icon” if the block icon should be created automatically. In this case the block icon which corresponds to the Index entered below will be chosen from the template picture @PCS7Typicalsx.pdl.

Save the Object Properties with OK.

How to connect and parameterize blocks

Each block has a list of input and output parameters, which can be connected either with a signal (from the symbols file or an output of another block) or parameterized with a value. Due to its utilization the inputs and outputs have different attributes which can be seen in the object properties in tab "I/Os".

In Cemat we distinguish between different parameter types:

Hardware Inputs and Outputs

These are the interfaces which are connected to the inputs and outputs of the periphery, as e. g. contactor feedback, electrical readiness, bimetal, on command of the motor or the analog input of a measure. Normally a signal from the symbols file is assigned to this interface.
The Hardware inputs and outputs are of type "Signal". The assignment of the periphery signal can either be carried out in the CFC or in the process object view in tab "Signals".

Input Interfaces

The so-called interface flags are used for starting, stopping, changing operation modes or interlocking the blocks. They are mostly connected to an output of another Cemat block or to some logic.
The interface flags are of type "Parameter". The linking of the interface flags can either be done in the CFC or in the process object view in tab "Parameters".

New: Since Cemat V7.1 the interlock interfaces (start interlock, operating interlock, protection interlock etc.) and some of the block outputs (drive running, route selected etc.) are available as structure inputs.
Instead of a binary connection, the link can be made via structure connection. Beside the binary information the structure contains the signal status, which is displayed in the diagnosis window. An additional advantage of the structure connection is that in the operation system you can directly jump from one object faceplate to its predecessor or to the following object. Using this method you can follow a missing interlocking condition very fast to its original cause.

The binary interfaces for interlocking conditions can still be used, if someone doesn't want to use the new functions (e. g. in case of a migration). It is also possible to use both inputs (binary and structure) parallel.

Releases

Some functions of the Cemat blocks must be enabled or disabled according to the demand. Some release functions can be carried out by the program others only from the Operator Station (both is not possible).
The Release bits which can be connected in the CFC are of type "Parameter". The connection can be programmed in the CFC or in the process object view in tab "Parameters".

Links

Each drive, annunciation block and measure must be connected to the group via the so-called group link. The blocks can directly be connected to the group or indirectly via route module.
The links are of type "Parameter" and can be connected in the CFC or in the process object view in tab "Parameters".

Process Values

Limit values, supervision or delay times are called the process values of the blocks. You can retain the default values or if required you can adapt it according to the need of your application. The adaptation can be carried out either in the CFC or from the Operator Station. We recommend to adapt the values as far as possible during the engineering and to leave the fine tuning for the commissioning.
The process values are of type "Parameter". The parameterization can be carried out in the CFC or in the process object view in tab "Parameters".

Output Interfaces

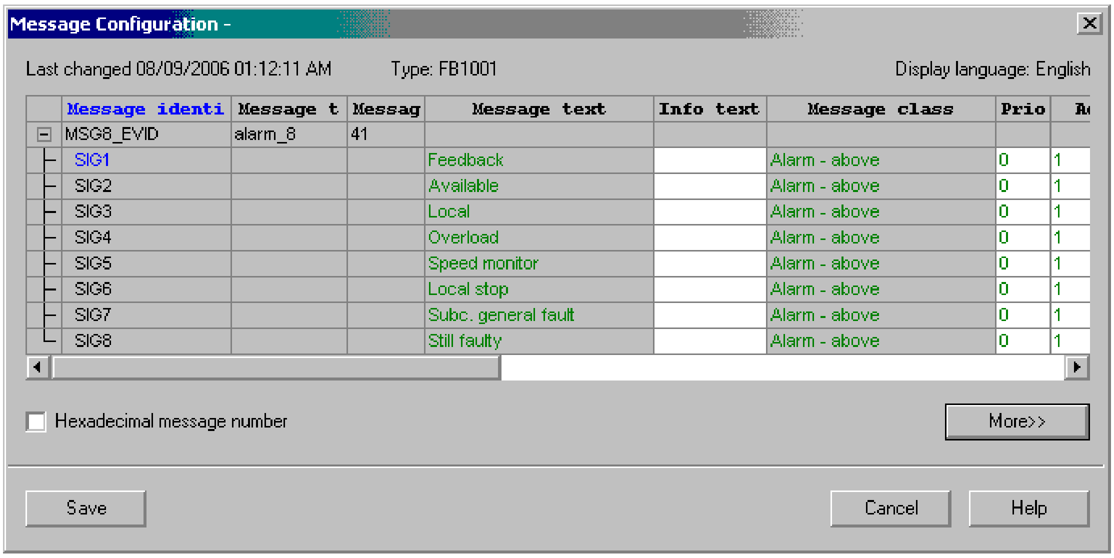
The most important information of the blocks are transferred to block outputs and therefore available for the connection to other blocks.

New: Some of the outputs (drive running, route selected etc.) are additionally available as structure output. See Input Interfaces.

A detailed functional description of the blocks you will find in the reference manual. There is a separate chapter for each object type.

Message configuration and customizing messages

Each Cemat block uses one ore more ALARM_8 block or ALARM_8P block, whose signals SIG1 to SIG8 are assigned a particular function. The message texts (Event texts in WinCC) and the Message classes are standard for the most blocks and therefore locked in the function block. An adaptation per instance is not required.



Note: The message texts are defined at the function block and can be modified at the function block itself. For any modification please note the following:

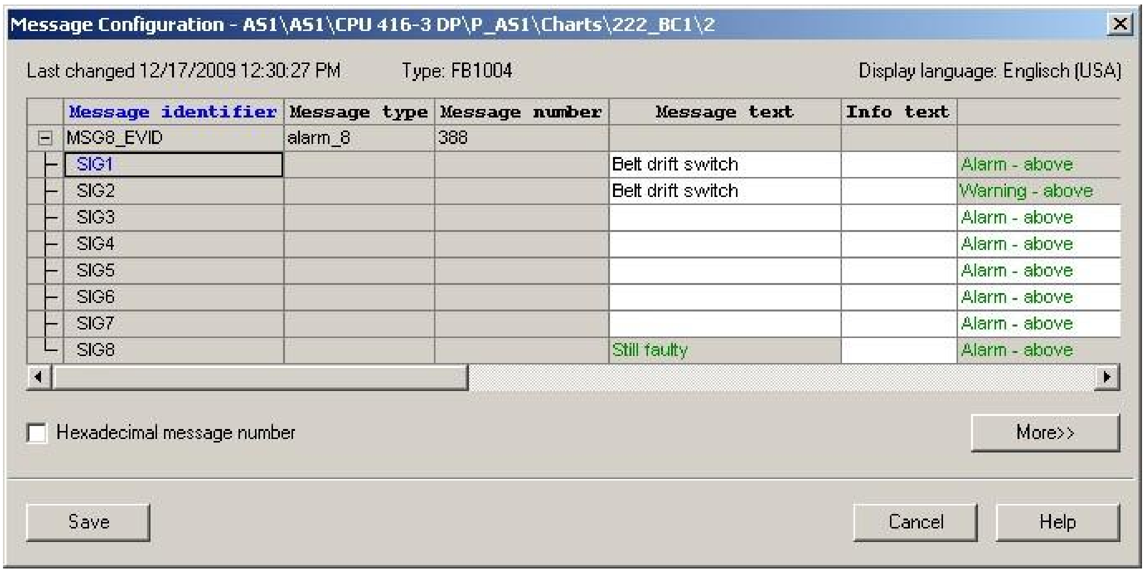
1. If you carry out an update for the blocks the message definitions get overwritten. In this case you have to define the message texts again.
2. For the status call the message texts are defined in the object specific Config files under [Fault].
   Make sure that you use the similar texts in the message definition and in the Config files. Take a backup of the Config files because they also get overwritten in case of an update.

Example: Message text definition in the Config File of a Unidirectional Drive:

```
[Fault]
;Visible, Attribut,Comment,Bit,Fault Class
1,ESS,Feedback,9,M
1,ESB,Available,10,E
1,EVO,Local,11,P
1,EBM,Overload,12,M
1,ESD,Speed monitor,13,M
1,LST,Local Stop,15,E
1,SCF,Simocode Fault.,8,E
1,SUB,Subc.general fault,32,E
```

In case of the annunciation blocks C_ANNUNC and C_ANNUN8 there is no particular message text definition. The user must insert an individual text (max. 16 characters).

Message text definition in block C_ANNUNC:



Note: You need to configure either the message text for the Warning (Warning – above) or the text for the Error message (Alarm – above) according of Parameter WMOD of the annunciation block.
In case of a two-level alarm you have to use similar texts for Warning and Error Message.

As in each annunciation block C_ANNUNC we have individual message texts, it is not possible to predefine the message texts fort he status call in the Config file.

Here you have two options:

1. Display of an individual fault text in the status call:

   In this case a wild card "?" must be entered in the Config file (default setting). The individual message text must then be defined in the CFC at block parameter IN_DEL:

   Settings in the Config File of the annunciation block C_ANNUNC:

```
[Fault]
;Visible, Attribut,Comment,Bit,Fault Class
1,MS0,?,1,M
```

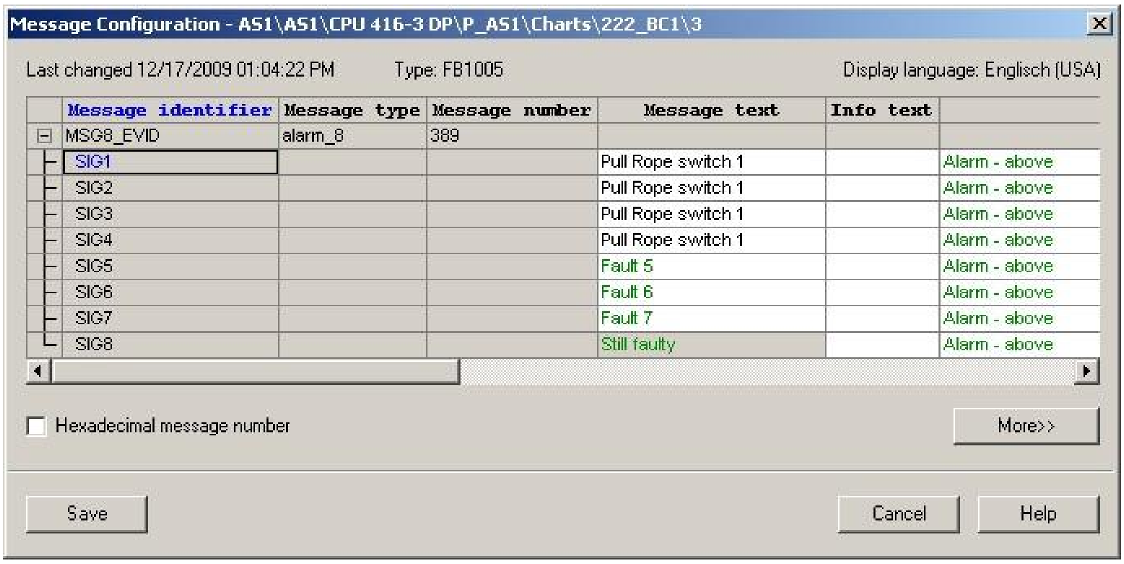
   To enter the message text in the CFC, select parameter IN_DEL of block C_ANNUNC and go to Object properties. The text must be entered under "Identifier" (max. 16 characters).
   After the OS Compile this text is available as an internal variable in the tag management of the OS.

2. Display of a standard fault text in the status call (similar for all annunciation blocks):

   In this case a fix text must be entered in the Config file:

```
[Fault]
;Visible, Attribut,Comment,Bit,Fault Class
1,MS0,Fault,1,M
```

Message text definition in block C_ANNUN8:



Also in the annunciation block C_ANNUN8 individual message texts can be defined, which means it is not possible to predefine the message texts for the status call in the Config file.

Also here you have two options:

1. Display of an individual fault text in the status call:

   In this case a wild card "?" must be entered in the Config file (default setting). The individual message text must be defined in the CFC at block parameter FLS1 – FLS7 under "Text 1":

   Settings in the Config File of the annunciation block C_ANNUN8:

```
[Fault]
;Number, Attribut,Comment,Bit,Fault Class
1,MS1,?,3,M
2,MS2,?,4,M
3,MS3,?,5,M
4,MS4,?,6,M
5,MS5,?,7,M
6,MS6,?,8,M
7,MS7,?,9,M
```

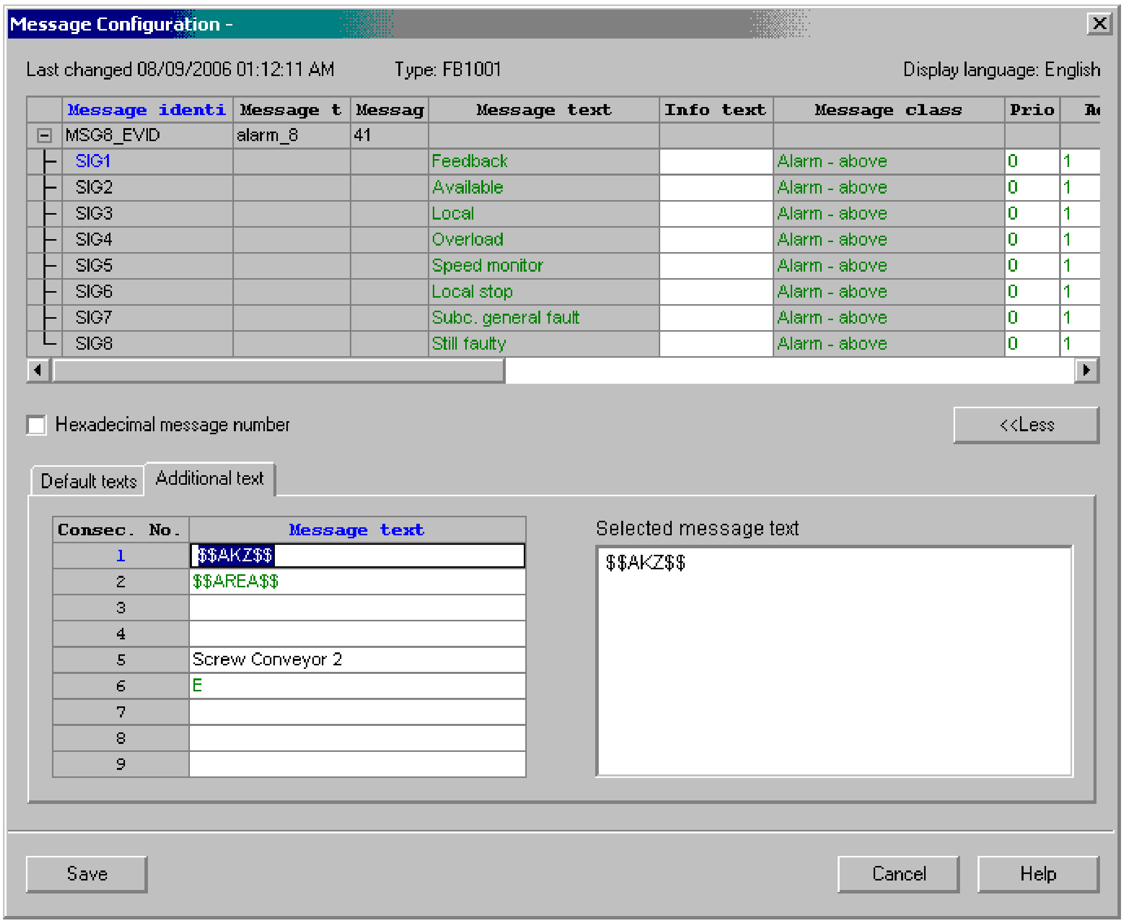
   To enter the message text in the CFC, select parameter FLS1 – FLS7 of block C_ANNUN8 and go to Object properties. The text must be entered under "Text 1" (max. 16 characters).
   After the OS Compile this text is available as an internal variable in the tag management of the OS.

2. Display of a standard fault text in the status call (similar for all annunciation blocks):

   In this case a fix text must be entered in the Config file:

```
[Fault]
;Number, Attribut,Comment,Bit,Fault Class
1,MS1,Fault 1,3,M
2,MS2,Fault 2,4,M
3,MS3,Fault 3,5,M
4,MS4,Fault 4,6,M
5,MS5,Fault 5,7,M
6,MS6,Fault 6,8,M
7,MS7,Fault 7,9,M
```

Another special characteristic in Cemat is the representation of the block comment as a separate text field in the message line. To enable this you must provide Additional text 5 (in WinCC this is "Free Text 1") with the block comment.



This can be carried out in the CFC (under Object properties → Special properties → Messages… → More → Tab "Additional Texts" and there Consec. No. 5) or you may use the Process Object View, tab "Messages" and copy the content of column "Block comment" into column "Free Text 1".

Caution: If the Process Object View, tab "Messages" does not contain the column "Free Text 1", you have to change the settings to make it visible.
In the SIMATIC Manager under Options → Customize.... → Columns → Process Object view → Messages select the corresponding column.

A detailed function description you will find in the Reference Manual. Each object type is described in a separate chapter.

Text definitions for the Cemat Faceplates

The Faceplates of some of the Cemat Objects contain Text variables which have to be defined in the CFC. For Performance reasons there are different methods for the engineering of the individual texts. The following list shows the different text variables and the engineering of each:

The physical unit of the measuring value C_MEASUR

In difference to the PCS 7 Standard Libraries the physical unit of the measuring value block C_MEASUR is configured at the block parameter UNIT.

The variable UNIT is part of the Tag Management of the OS and can be modified online.

The Texts of the Cemat Interlock block C_INTERL

At the block C_INTERL the texts are configured via block parameter I1_1 to I1_5 and I2_1 to I2_5. In the object properties of the corresponding parameter you have to enter a string of up to 16 characters into field "Text 1".

After the OS Compile the texts are available as internal variables in the Tag Management of the OS. A modification of the texts requires a new OS Compile.

The Texts of the Cemat Interlock block C_INTER5

At the block C_INTER5 the texts are configured via block parameter S_TEXT. The maximum length of the String Variable is 80 characters (up to 5 text strings of maximum 16 characters must be separated by Semicolon).

The variable S_TEXT is available in the Tag Management of the OS and can be modified online.

The individual faults in the diagnosis picture of C_ANNUN8

In the diagnosis picture of the ANNUN8 block the single faults are named with "Fault 1" to "Fault 7". Instead of this default text, for each fault type an individual text can be displayed. The texts are configured in the CFC via block parameters FLS1 to FLS7. In the Object properties of the corresponding parameter you can enter a text (up to 16 characters) into field "Text 1".

After the OS Compile the texts are available as internal variables in the Tag Management of the OS. A modification of the texts requires a new OS Compile.

The Message Text for Status Call function for C_ANNUNC and C_ANNUN8

As it is already mentioned in the last chapter, the message texts for block C_ANNUNC and C_ANNUN8 have to be configured individually according to the function.
In order to show the same text also in the status call function under "Fault Type", you have to enter the text for block C_ANNUNC into the object properties of parameter IN_DEL under "Identifier".
For the block C_ANNUN8 the text must be entered in object properties of parameter FLS1 to FLS7 under "Text 1". In both cases the text is limited to 16 characters.

After the OS Compile the text is available as an internal variable in the Tag Management of the OS. A modification of the texts requires a new OS Compile.

The bar description for motor current in the drive faceplate

If a measuring value is assigned via input MV_PERC to the motor, the faceplate of the drive shows a bar with the percentage value of the measure and the measuring value in %.
The description of the measure depends on the measure itself (normally current of power measurement) and must therefore be configured at the drive block. The parameterization is carried out in the object properties of parameter CORR_OS under "Identifier".

After the OS Compile the text is available as an internal variable in the Tag Management of the OS. A modification of the texts requires a new OS Compile.

The Unit for Set Point and Actual Value of a VDS drive

For Variable Speed drives, the Setpoint can directly be entered via the Faceplate of the drive block C_DRV_1D and the actual speed is indicated there as well.
In order to show the correct Unit in the faceplate (default is "rpm") you have to modify in the CFC the property "Unit" of bock parameters SP_IN, SP_EX.Value and SP_O.Value the unit for PV_IN.Value must be entered under "Identifier"!

After the OS Compile the text is available as an internal variable in the Tag Management of the OS. A modification of the texts requires a new OS Compile.

Annunciation Release

In all Cemat blocks through plausibility logic it is assured that in case of several similar faults only the "perpetrator" creates an alarm message.
For example, in case of a protection interlock of a motor, no alarm is created by the motor block, it is created by the associated annunciation module.

In order not to create an alarm flush in case of loss of control power, each Cemat block has an interface "Annunciation Release" xMFR, which has to be connected with the control power signal.

In case of a periphery failure (such as bus failure, card failure, MCC failure) this signal must get "0". Now the Cemat block does not create further messages (no incoming, no outgoing messages).

For the annunciation of the control power failure itself an additional annunciation block has to be programmed.

The annunciation release is one of the most important interfaces in order to achieve a smooth operation. The project engineer must assure that only the perpetrators of a fault create alarm messages.

Invisible Module Parameters

In the Object description you will find a list of the Parameters for all Objects as well as a detailed description of the programming rules.

For the CEMAT Modules by default only the module parameters which are usually used in most of the plants are set to visible. This means, if you drop a CEMAT Module to your chart, you will not see all the available parameters.

The invisible parameters can be switched to visible if required. This can be done generally at the FB itself (change of the Attribute S7_visible to ‚true') or in the CFC for each instance.

Note: With a Standard Update your attribute changes at the function block will be lost and must be performed again if required. The instances keep the original Settings.

Display of Motor current (or Power) in the drive faceplate

The following interfaces of C_DRV_1D and DRV_2D allow the display of the motor current in % in the drive faceplate. In the default settings they are switched to invisible but they can be changed to visible if required.

Element	Bedeutung	Typ	Vorbe-setzung	Art	Attr.	B&B	zulässige Werte
REL_MVC	enable display of motor current	BOOL	0	I	U		
MV_PERC	Motor current from C_MEASUR	POINTER	0	I	U		

Display of an additional measuring value in the drive faceplate

The following interfaces are used for the link to a measuring value of to an Analog Selection block. This permits the display of an additional measure in the drive faceplate and the jump to the connected block. In the default settings the interfaces are invisible but they can be changed to visible if required.

Element	Bedeutung	Typ	Vorbe-setzung	Art	Attr.	B&B	zulässige Werte
PV	Process value input (general use)	STRUCT		I	U		
PV.Value	Value	REAL	0.0	I	U	+	
PV.ST	Signal Status	BYTE	16#FF	I	U		
PV_Stat	Process value status + unit	STRUCT		I	U		
PV_Stat.UNIT	Unit	STRING [8]	%	I	U	+	
PV_Stat.STATUS	Status	DWORD	16#00	I	U	+	

Variable Speed Drives

The following interfaces of the C_DRV_1D belong to the Variable Speed Drive function. In the default settings they are switched to invisible but they can be changed to visible if required.

Element	Meaning	Format	Default	Typ e	Attr.	HMI	Permitted Values
EN_SP	Enable setpoint function	BOOL	0	I	U		
EN_SPEX	Enable external setpoint	BOOL	0	I	U		
SP_TR	Setpoint tracking	BOOL	0	I	U		
SP_IN	Setpoint from OS	REAL	0.0	I	U	+	
SP_EX	External Setpoint	STRUCT		I	U		
SP_EX.Value	Value	REAL	0.0	I	U	+	
SP_EX.ST	Signal Status	BYTE	16#FF	I	U		
SP_HLM	Setpoint high limit	REAL	0.0	I	U	+	
SP_LLM	Setpoint low limit	REAL	0.0	I	U	+	
PV_IN	Process value input for setpoint function	STRUCT		I	U		
PV_IN.Value	Value	REAL	0.0	I	U	+	
PV_IN.ST	Signal Status	BYTE	16#FF	I	U		
SP_O	Setpoint Output	STRUCT		O	U		
SP_O.Value	Value	REAL	0.0	O	U	+	
SP_O.ST	Signal Status	BYTE	16#80	O	U		

Subcontrol Function

The following interface of C_DRV_1D is needed in case of Subcontrol Functions. In the default settings it is switched to invisible but they can be changed to visible if required.

Element	Bedeutung	Typ	Vorbe-setzung	Art	Attr.	B&B	zulässige Werte
SUBC_FT	General fault Subcontrol	BOOL	0	I	U		

Positioner function

The following interfaces of the C_DAMPER belong to the positioner function. In the default settings they are switched to invisible but they can be changed to visible if required.

Element	Meaning	Format	Default	Type	Attr.	HMI	Permitted Values
KPOS	Positioner	BOOL	0	I	U		
KSNF	Setpoint tracking	BOOL	1	I	U		
W_OS	Setpoint of OS (KWCO)	REAL	0.0	I	U	+	
KWUG	Setpoint lower limit	REAL	0.0	I	U	+	
KWOG	Setpoint upper limit	REAL	100.0	I	U	+	
KWEE	External setpoint active	BOOL	0	I	U		
KWEX	External setpoint	STRUCT		I	U		
KWEX.Value	Value	REAL	0.0	I	U	+	
KWEX.ST	Signal Status	BYTE	16#FF	I	U		
SCB	Scale beginning	REAL	0.0	I	U	+	
SCE	Scale end	REAL	100.0	I	U	+	
UNIT	Unit	STRING [8]	‚%'	I	U	+	
POS_IN	Position value 0-100	STRUCT		I	U		
POS_IN.Value	Value	REAL	0.0	I	U		
POS_IN.ST	Signal Status	BYTE	16#FF	I	U		
POS_LZ	Live-zero for position	BOOL	0	I	U		
TMIN	Min. pulse length	REAL	0.5	I	U	+	
TM	Actuator run-time	REAL	60.0	I	U	+	
AN	Switch on of the dead zone	REAL	1.0	I	U	+	
AB	Switch off of the dead zone	REAL	1.0	I	U	+	
X_POS_OS	Damper position display	STRUCT		O	U		
X_POS_OS. Value	Value	REAL	0.0	I	U	+	
X_POS_OS. ST	Signal Status	BYTE	16#80	I	U		
KPO	Positioner ON	BOOL	0	O	U		

Drives with SIMOCODE

The following interfaces of the drive blocks are needed in case of a connection to a SIMOCODE Adapter block. In the default settings it is switched to invisible but they can be changed to visible if required.

Element	Bedeutung	Typ	Vorbe-setzung	Art	Attr.	B&B	zulässige Werte
REL_SC	Enable SIMOCODE	BOOL	0	I	U	+	
STAT_SC	Status SIMOCODE	BYTE	16#00	I	U		

Control desk interfaces at the CEMAT Objects

The following interfaces will only be required if a conventional control desk with pushbuttons and lamps is used. In the default settings they are switched to invisible but they can be changed to visible if necessary.

Function	DRV_1D	DRV_2D	DAMPER	VALVE	ANNUNC	MEASUR	ROUTE	GROUP
Lamp test	ELPZ	ELPZ	KLP1	VLPZ	MLPZ		WLPZ	GLPZ
Acknowledge	EQIT	EQIT	KQT1	VQIT	MQIT	UQIT		GQIT
Release signal								GFGS
Pushbutton release							WPTS	GPTS
Pushbuttons							WVT	GTA GTE
Lamps	ELS	ELS1 ELS2	KL1 KL2	VL1 VL2	MLA		WVL	GZV GZS GZB

Using driver blocks with CEMAT

Driver blocks are not necessary for Cemat. As before, the block inputs and outputs can be directly connected to the periphery (process image). The only exceptions are the Analog outputs (e. g. from a PID controller), where the calculation from REAL value to card value is carried out by the driver block CH_AO.

Some customers use driver blocks for Analog Inputs as well. (The driver block detects the Card type and converts the Card Value into REAL format.)

Caution: Through the driver wizard a lot of additional charts and blocks are generated which require additional resources (Communication jobs). This can especially be critical if ET200S periphery is used.

From Cemat V6, connecting the Cemat blocks to the driver blocks is possible. If you wish to use driver blocks in your project, please pay attention to the following engineering rules:

Driver functions and drive blocks

The parameterization of the driver bocks can exclusively be carried out in CFC, at the driver block itself. Only in the CFC the simulation value can be set and the behavior in case of module fault (substitution value or last valid value) can be parameterized.

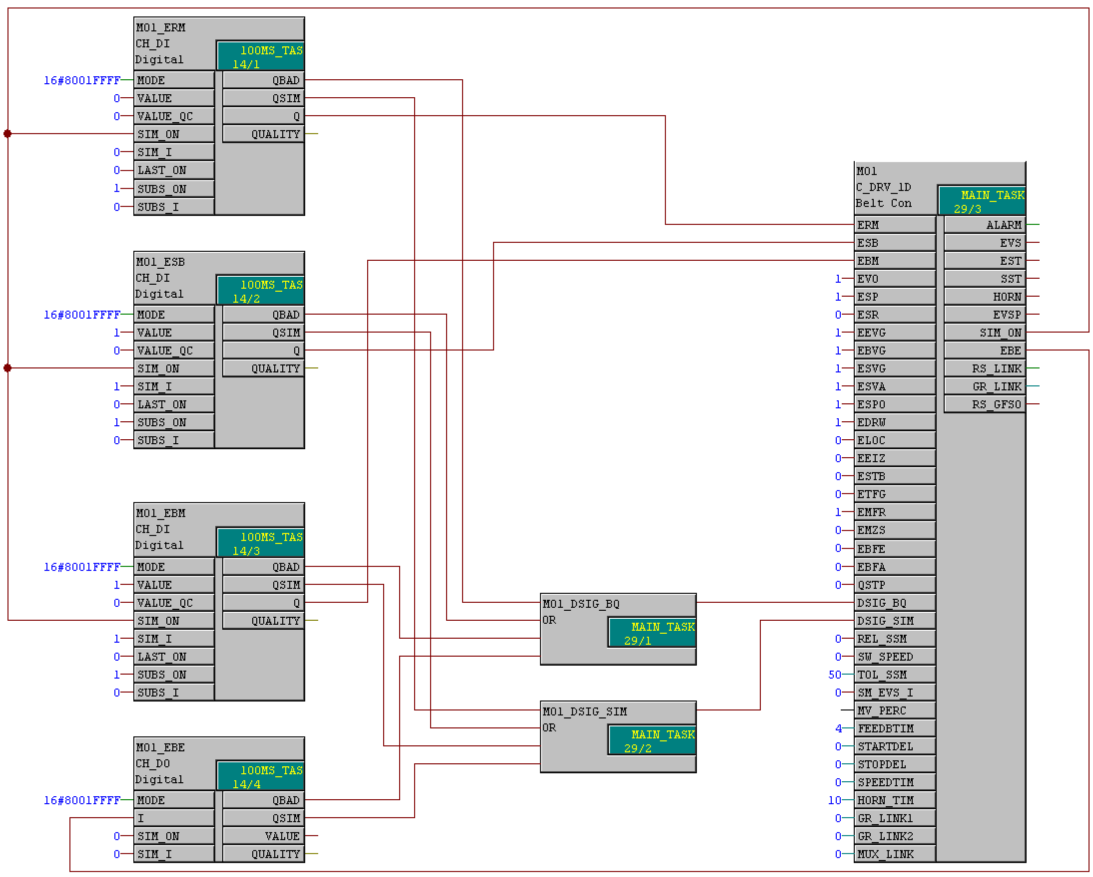
Connecting output SIM_ON of the CEMAT drive block to the input SIM_ON of the Driver block, in sequence test mode the driver blocks will be automatically switched to simulation mode.

Display of "Bad Quality" and "Simulation" at the drive block icon and faceplate:

- If the driver block recognizes a fault the output QBAD will be set. In order to indicate this status in the drive block icon and faceplate, you have to connect output QBAD of all related driver blocks with an OR-Function to interface DSIG_BQ of the drive.
- If the driver block is switched to Simulation the output QSIM will be set. In order to indicate this status in the drive block icon and faceplate, you have to connect output QSIM of all related driver blocks with an OR-Function to interface DSIG_SIM of the drive.

⚠ **Caution:** The Display of "Bad Quality" and "Simulation" in the block icon of the motor requires special attributes of the block icons. See OS Engineering.

Example:



Driver functions and Annunciation blocks

The parameterization of the driver bocks can exclusively be carried out in CFC, at the driver block itself. Only in the CFC the simulation value can be set and the behavior in case of module fault (substitution value, last valid value or invalid value) can be parameterized.

Simulation:

Block C_ANNUNC already includes a simulation function, which can be enabled via the operator station. The simulation value must be configured in the CFC at block C_ANNUNC via parameter M_SIM. In case of a simulation the status is displayed in the block icon and in the Faceplate and the block is entered into the list of simulated objects.
Changing the AS into Sequence test mode automatically switches all annunciation blocks into simulation.

In case of a simulation via the driver block, only the quality code shows that it is a simulated value. There is no entry into the list of simulated objects and the Simulation can not be enabled via the faceplate of the annunciation block. For this reason you should use the simulation at the annunciation block rather than the simulation at the driver block.

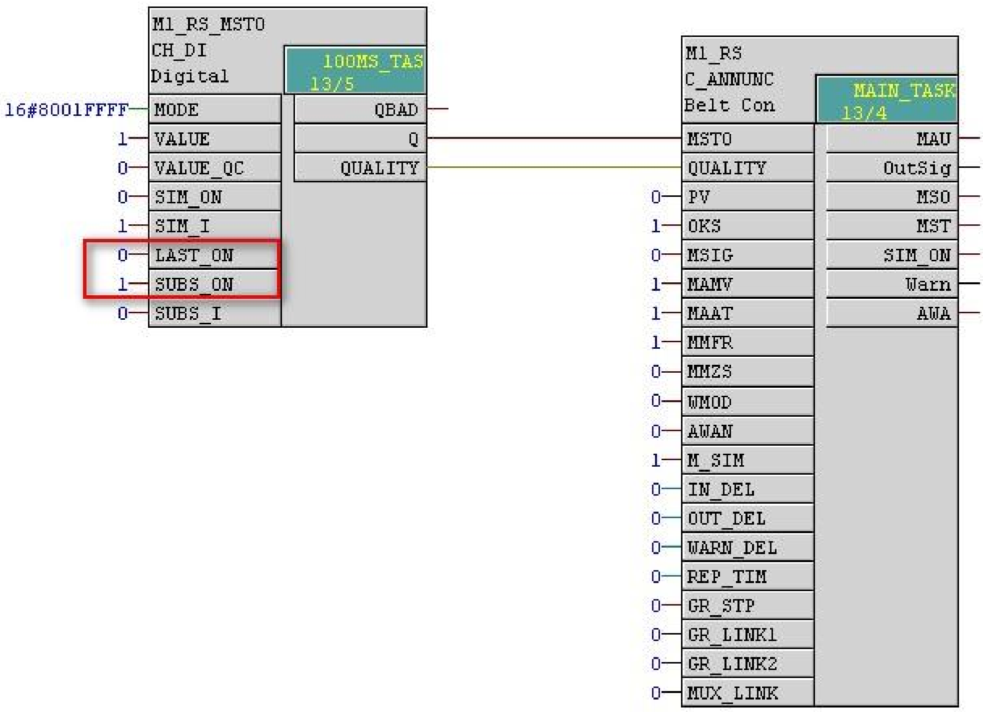
Configuration of the behavior in case of "Bad Quality":

For the configuration of the driver blocks in PCS 7 V7, the following options exist:

SUBS_ON = 1 and LAST_ON = 0	Substitution value (Quality Code = 16#48)
SUBS_ON = 0 and LAST_ON = 1	Last valid value (Quality Code = 16#44)
SUBS_ON = 0 and LAST_ON = 0	Invalid value (Quality Code = 16#00)
SUBS_ON = 1 and LAST_ON = 1	Invalid value (Quality Code = 16#00)

In case of Bad Quality the Quality Code is displayed in the Diagnosis picture of the Annunciation block.

Example:



Driver functions and Measuring values

With the appropriate connections between driver block and Measuring value block, the parameterization of the driver block is possible online (via faceplate of the Measuring value). From the Diagnostic Picture the Simulation value and the Substitution value can be entered and via Enable functions the Simulation can be switched on/off and the behavior in case of a module failure (substitution value or last valid value) can be parameterized.

To enable this, the following connections between Measuring value block and driver block are required:

To read in a physical value the Measuring value type (parameter TYP) must be 10.

The output Q of the driver block has to be connected to input MV_PHYS of the measuring value.

The output QUALITY of the driver block has to be connected to input QUALITY of the measuring value. This enables the display of the module status “Bad Quality” or “Simulation” in the block icon of the measure and in the faceplate.

Caution: The Display of the Module status in the block icon of the measure requires special attributes of the block icons. See OS Engineering.

Simulation:

Block C_MEASUR already includes a simulation function, which can be enabled via the operator station. The simulation value must be configured in the CFC at block C_MEASUR via parameter SIM_VAL. In case of a simulation the status is displayed in the block icon and in the Faceplate and the block is entered into the list of simulated objects.
Changing the AS into Sequence test mode automatically switches all measuring value blocks into simulation.

In case of a simulation via the driver block, only the quality code shows that it is a simulated value. There is no entry into the list of simulated objects and the Simulation can not be enabled via the faceplate of the measuring value block. For this reason you should use the simulation at the measuring value block rather than the simulation at the driver block.

Configuration of the behavior in case of "Bad Quality":

For the configuration of the driver blocks in PCS 7 V7, the following options exist:

SUBS_ON = 1 and LAST_ON = 0	Substitution value (Quality Code = 16#48)
SUBS_ON = 0 and LAST_ON = 1	Last valid value (Quality Code = 16#44)
SUBS_ON = 0 and LAST_ON = 0	Invalid value (Quality Code = 16#00)
SUBS_ON = 1 and LAST_ON = 1	Invalid value (Quality Code = 16#00)

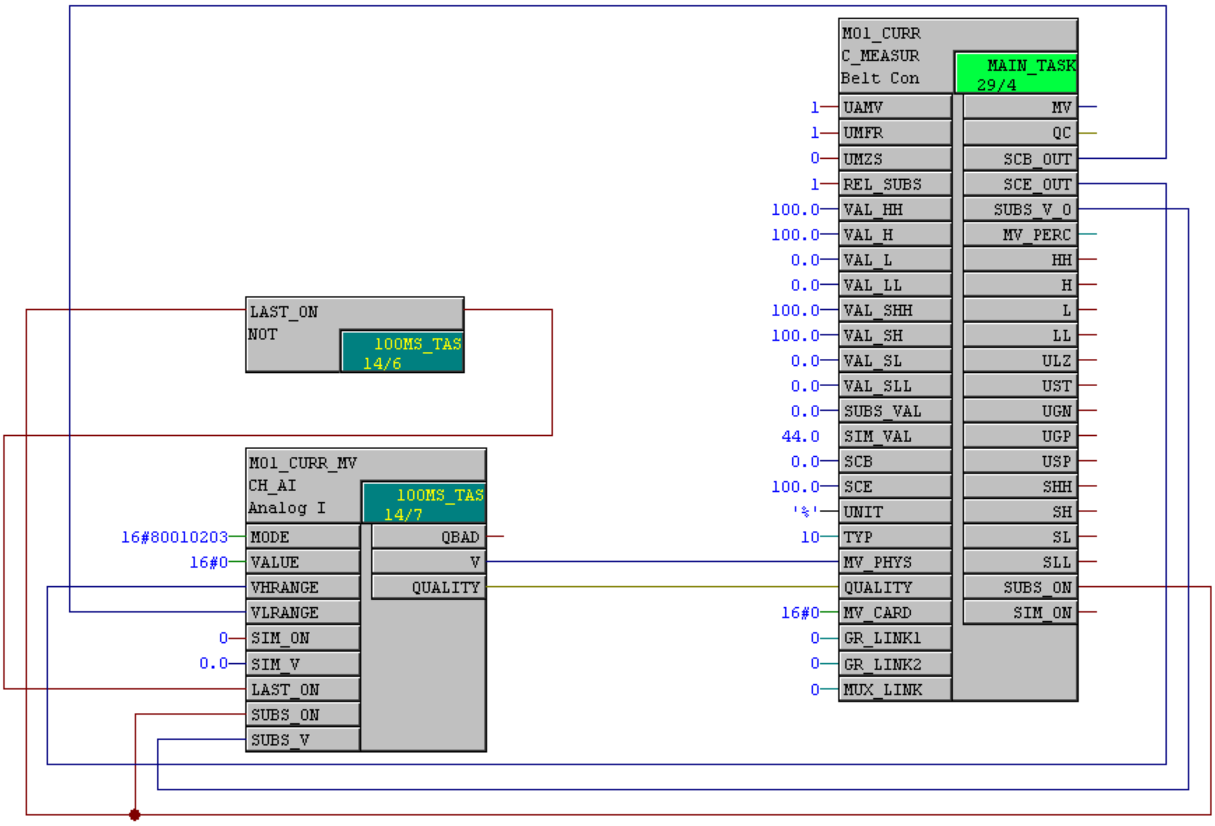
- The Cemat Measure block has a Process Parameter for Substitution value, in order to show this value in the diagnosis picture. If you chose "Substitution value", you have to set REL_SUBS to 1-Signal and enter the Substitution value to SUBS_VAL.
  In order to transmit this information to the PCS 7 driver block, connect output SUBS_V_O of the measure to input SUBS_V of the driver block.
  To enable the function at the driver block, connect output SUBS_ON of the measure to input SUBS_ON of the driver block and the inverted information to input LAST_ON of the driver block.
- In order to use the "Last valid value", set input REL_SUBS to 0-Signal and connect output SUBS_ON of the measure to input SUBS_ON of the driver block and the inverted information to input LAST_ON of the driver block.
- For option "Invalid Value", set input REL_SUBS to 0-Signal and at the driver block you may set signals LAST_ON and SUBS_ON either both to 1-Signal or both to 0-Signal.

In case of Bad Quality the Quality Code is displayed in the Diagnosis picture of the Measure.

In order to enter the Measuring range only once, the outputs SCB_OUT and SCE_OUT of the Measure can be connected to inputs VLRANGE and VHRANGE of the driver block.

Caution: For PT100 VLRANGE and VHRANGE must not be connected!!!

Example:



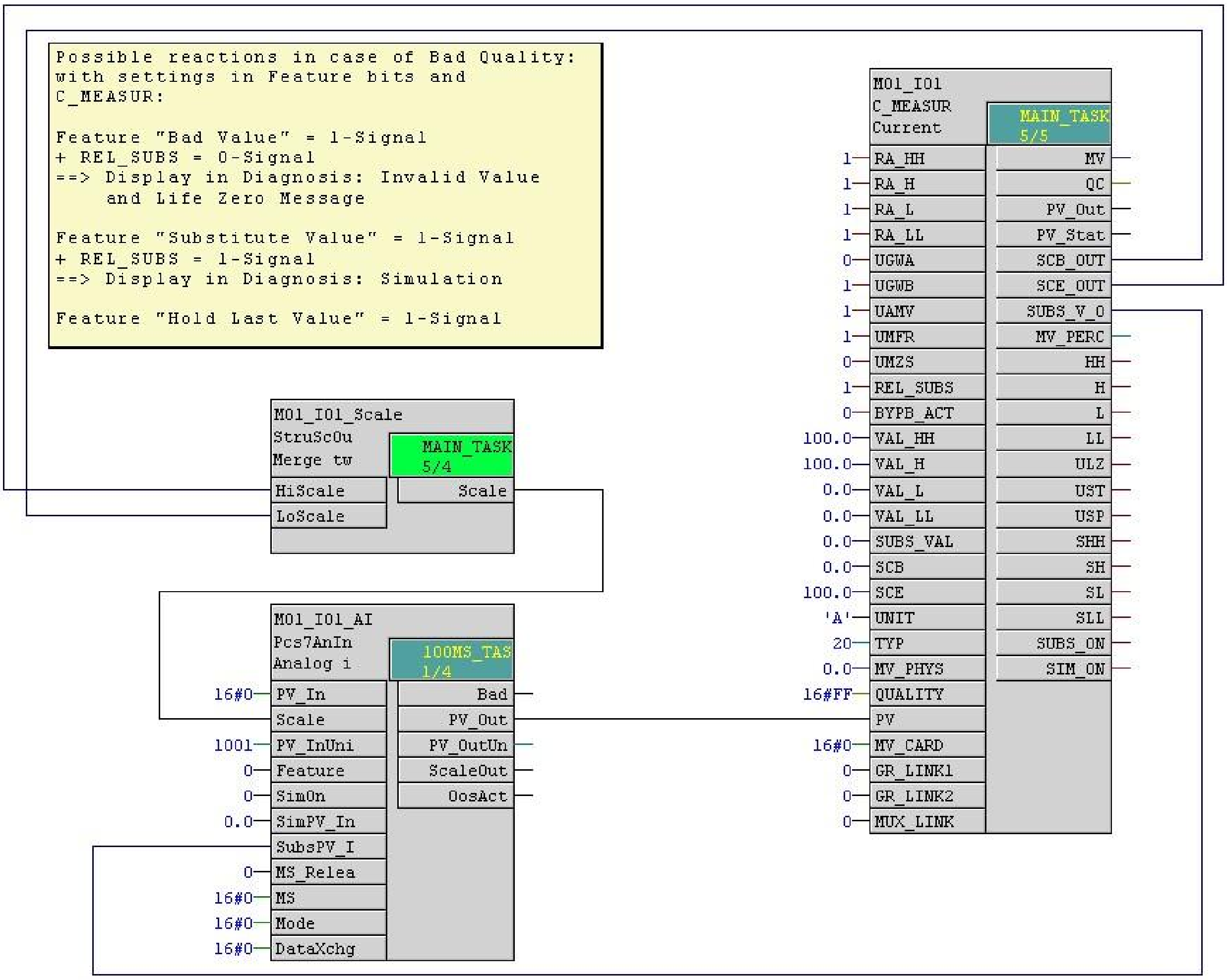
APL Driver block and Measuring value

With TYP = 20 the measuring value block can read and evaluate the analog value form the APL driver block Pcs7AnIn.

Via the output structure PV_Out of Pcs7AnIn the value and the status is transmitted.

As Scale beginning and Scale end is already configured at the C_MEASUR, we recommend connecting the outputs SCB_OUT and SCE_OUT. However, a structure converter must be used. (See example below.)

Caution: Structure input Scale of driver block Pcs7AnIn must not be connected in case of PT100 and in this case Scale beginning and Scale end must be set to 0!



The behavior in case of Bad Quality is configured via Feature word. This may be either 'Bad value', 'Substitute value' or 'Hold last value'.
Please parameterize the C_MEASUR accordingly.

Caution: In the APL blocks the evaluation and display of card failures differs from the driver blocks of the standard PCS 7 library. See comparison next page.

Operation status	Standard PCS 7 Driver	APL Driver
Valid value	80 (valid value)	80 (valid value)
Simulation	60 (Simulation)	60 (Simulation)
Substitute value	48 (Substitute value)	60 (Simulation)
Hold last value	44 (Last valid value	60 (Simulation)
Invalid value	00 (Invalid value)	00 (Invalid value)

Settings at the block and Result:

Settings for Bad Value:
Feature bit 'Bad Value' = 1 and REL_SUBS = 0
In case of a card failure, the diagnosis picture of C_MEASUR shows 'Invalid Value" and the C_MEASUR creates a message for "Bad Quality".

Settings for Substitute value:
Feature bit 'Substitute value' = 1 and REL_SUBS = 1
In case of a card failure, the diagnosis picture of C_MEASUR shows 'Simulation" and the C_MEASUR creates no message.

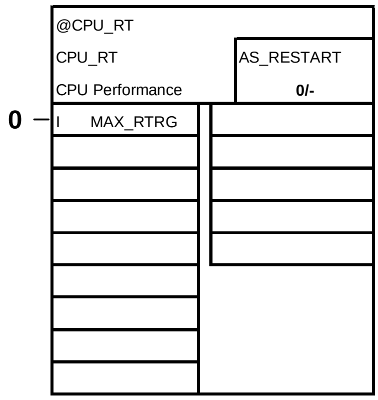
Settings for Hold last value:
Feature bit 'Hold last value' = 1 and REL_SUBS = 0
In case of a card failure, the diagnosis picture of C_MEASUR shows 'Simulation" and the C_MEASUR creates no message.

Function block CPU_RT

During the generation of the driver blocks, independent on CPU and the FW of the CPU, automatically the chart @CPU_RT is generated. This chart contains the block CPU_RT "Performance data and Stop Avoidance" which are called in OB100, OB1, and in all OB3x as well as OB8x.

The CPU_RT determines the runtime of the individual OBs and their participation in the cycle time. It leads to stop avoidance if the CPU is overloaded (OB80, Cycle time exceeded). Through "load shedding" the CPU can be prevented from becoming inoperable. The load shedding is achieved by interrupting the cyclic levels and is an emergency operation.

Caution: For Cemat this behavior is not desirable! Therefore parameter MAX_RTRG has to be changed to "0", in order to **switch off** the function "Fault avoidance in case of overload".



Definition of the run sequence

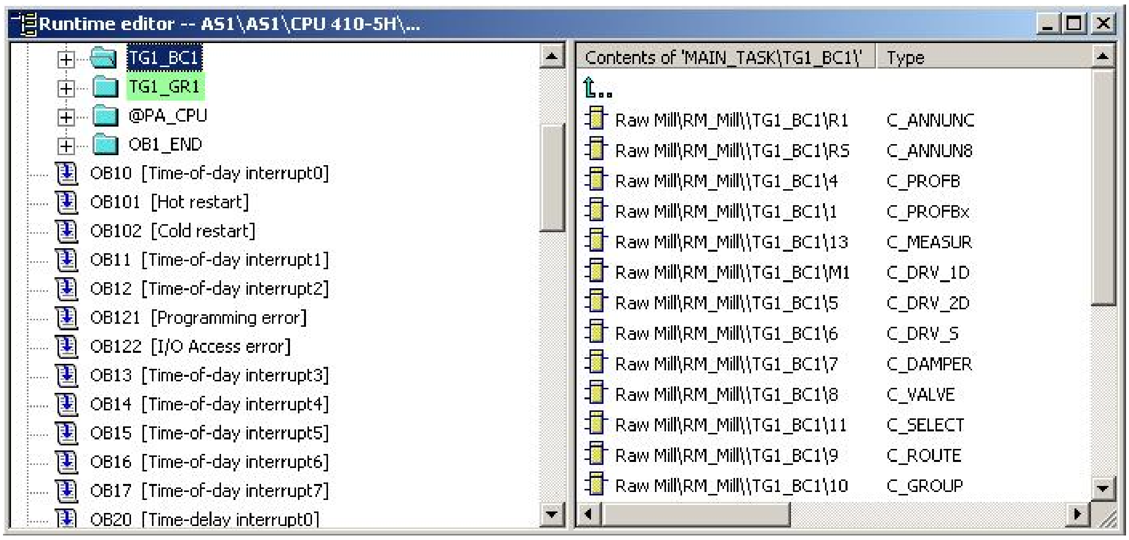
Make sure that the blocks are called at the right order in the runtime sequence (see object descriptions).

The CEMAT blocks **must be called** in (OB1 = MAIN_TASK) between the runtime groups OB1_START and OB1_END.

The Runtime Sequence is:
1. first : MUX
2. then : Annunciations and Drives
3. next : Corresponding Routes
4. finally : Corresponding Groups

All modules in the CFC will be assigned to a (default) OB. The module which was defined last will automatically be the predecessor for the installation of the next module.

The so-called run-time editor allows the definition of Run-time groups and the modification of the run sequence according to the desired structure of the program.



Note: The runtime sequence is especially important in case of C_MUX modules and in case of cascaded C_MUX, or if one C_MUX is programmed and used for different Cemat blocks (e. g. for a drive and all allocated Annunciations and Measures).
It must be assured that between the call of the C_MUX and the Cemat Objects where it is connected to, no other C_MUX block is called!

Check of the run sequence and the group / route links

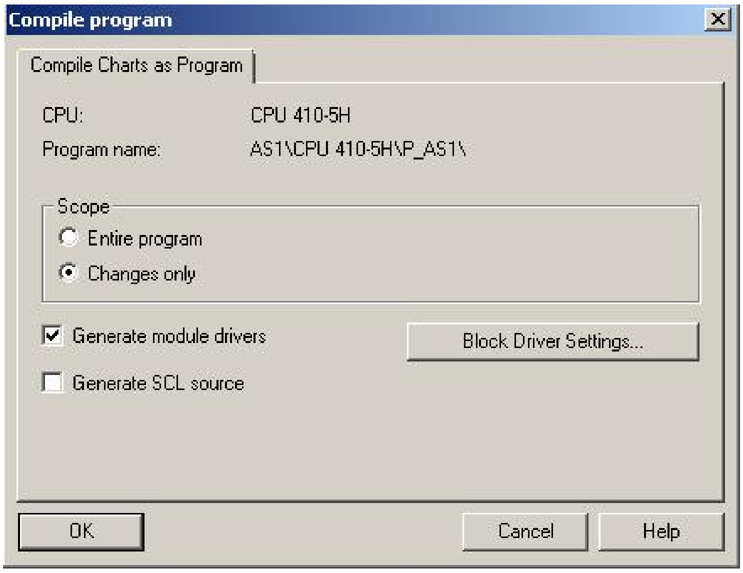
With the Excel tool "CEMAT_CheckTool.xlsm" you can check your Group links and your runtime sequence. E. g. you call a drive block before his related C_MUX or you call a drive block after his related C_GROUP block. The Excel tool will detect this engineering fault and show this kind of engineering faults in a list.

The Excel tool "CEMAT_CheckTool.xlsm" you will find in the directory D:\CEMAT_CS\TOOLS.

The user manual for the tool you will find in the chapter 10_Checkliste_009.pdf.

Compile CFC

Before you can download the program into the AS it must be compiled. It will be transferred into a machine language (SCL). During compilation the program will be checked for errors and warnings. You can compile the complete program or only the changes.

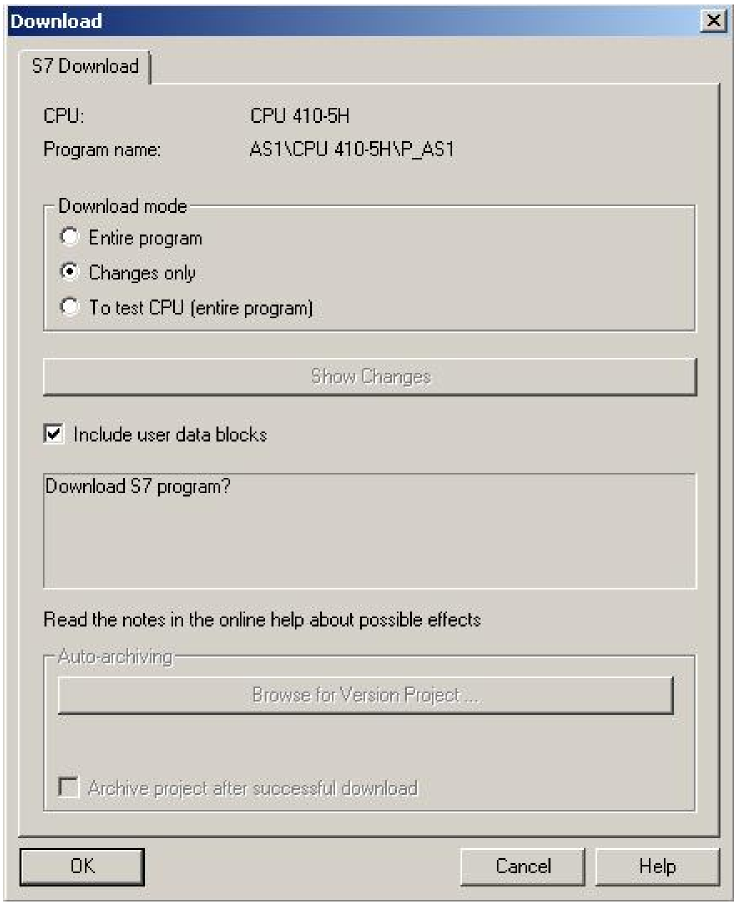


During the generation of the driver blocks the chart @CPU_RT with block CPU_RT is generated.
The first time the parameter MAX_RTRG of block CPU_RT is preset with value "50" and it has to be changed to "0"!

Download to the AS

First the hardware definitions and the network configuration have to be downloaded into the AS (if this was not done before).

The S7 Program has to be downloaded from the CFC. PCS 7 takes care of the right order of the loading procedure. Do not download the blocks! You will be asked for a complete download or to download only the changes. Changes in the interfaces of a function or function block will require a download of the complete program. This means AS stop.



OS Compile

With the OS Compile the following steps are carried out:

- From the CFC blocks all Variables with Attribute 'Operator Control and Monitoring' are transferred into the Tag Management of WinCC. The Messages, defined in the CFC blocks are transferred to the Alarm Logging.

  In the background the Block Icons are generated and/or actualized. In order to prevent the automatic generation you have to deselect the option in the Object properties of the process picture.

  Also in the background the archive variables are generated (if any of the Block parameters is selected for "Archiving" or “Long-term archiving”).

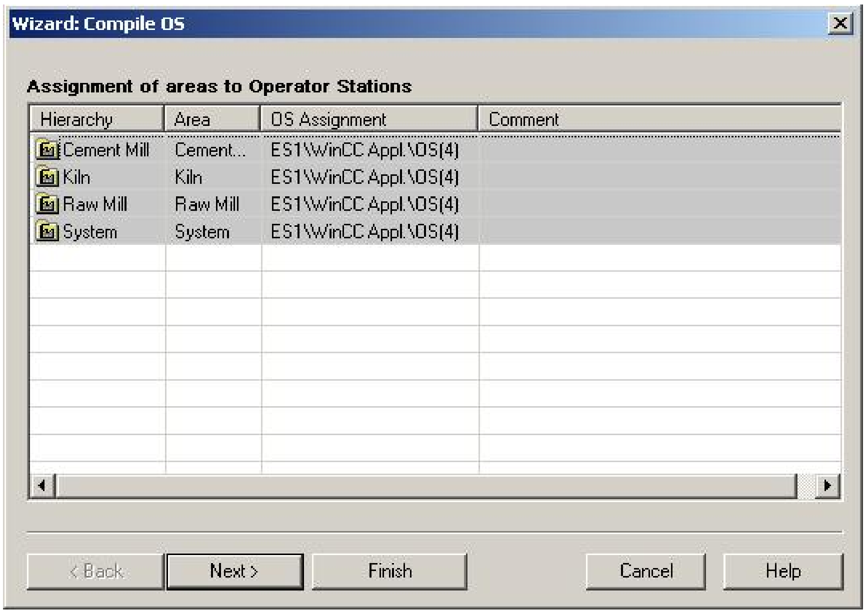
Caution: In case of automatic archive creation, all archive tags are created in the so-called "SystemArchive". Archive splitting (creating area specific archives) is not possible in this case!
See OS Engineering under "Adding an archive tag to Process Value Archive.

- The SFC Visualization is generated. (This is not needed on Cemat.)
- Based on the Plant view of the SIMATIC Manager the Picture Tree is created and/or actualized. This is essential because in this step also the Group Display get actualized.

Start the OS Compile from the SIMATIC through selection of the OS with right mouse button and option '*Compile*'.

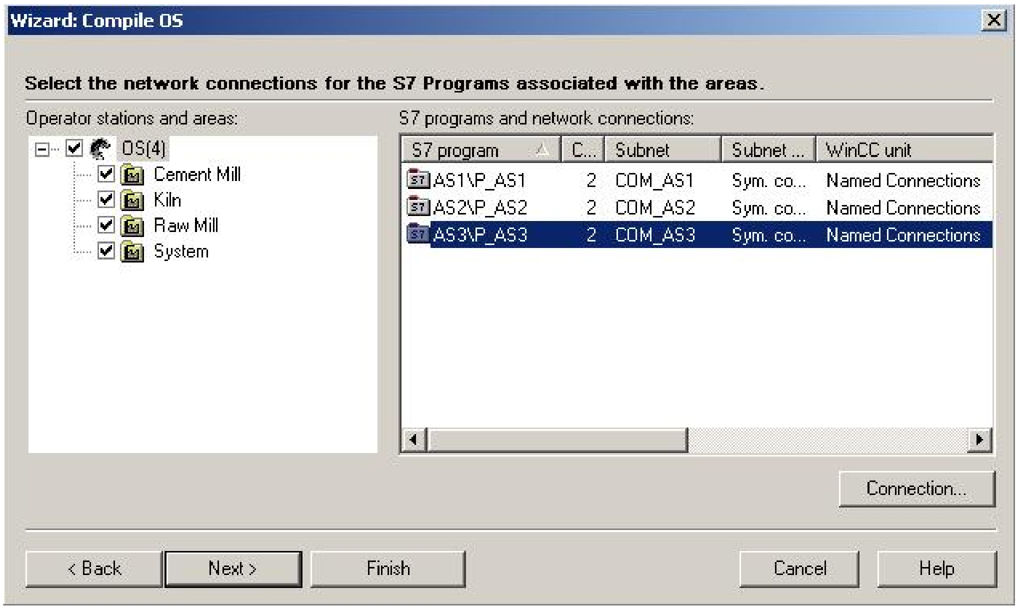
In the Introduction Window say '*Next >*'

In the following window you see the assignment of the areas to the operation station:

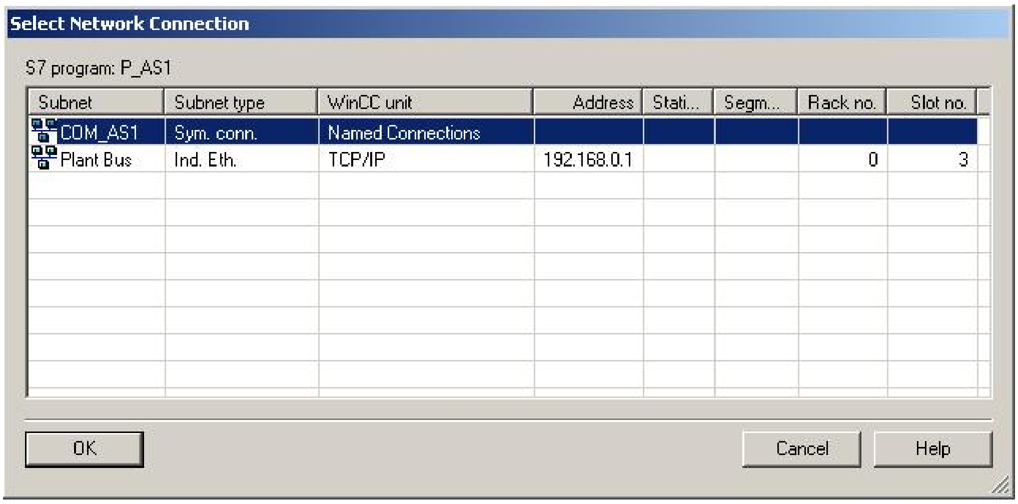


Continue with '*Next >*'

In the following window you can select the network connections for the S7-Programs associated with the areas.

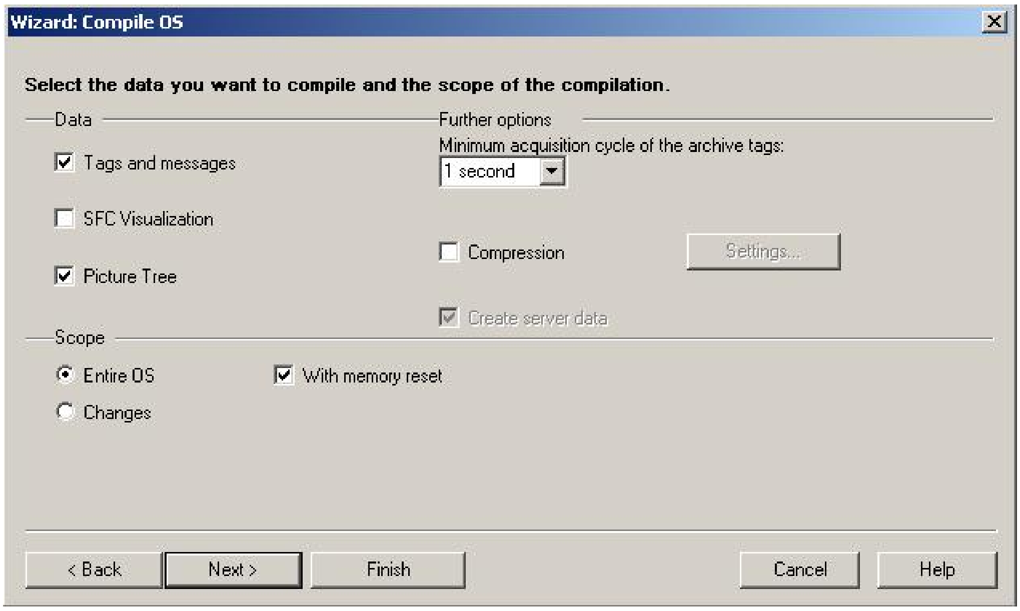


Select each AS and press '*Connection…*'



Select the network connection you want to use for the communication to WinCC.
In PCS 7 V7 for a real AS always use *'Named connections'.* In case of PLCSIM you have to use MPI.

Finally select the transfer data and the scope of compilation.



Now press *'Finish'* to start the transfer.

Project-Download

After new plant objects or pictures were added the Servers and the Client have to be updated as well. This requires a Project Download to the OS PCs.

If there was no structural change and as long as an OS Compile for changes is possible the OS Stations can remain in Runtime mode during the Project Download. After a complete OS compilation the destination Project has to be deactivated during the Download procedure.

Note: The modifications in the OS-Projects for the Server, Standby-Server, and Terminals are never carried out directly on the OS Stations. Any modification has to be carried out on the Engineering station and then being downloaded are different OS Stations.

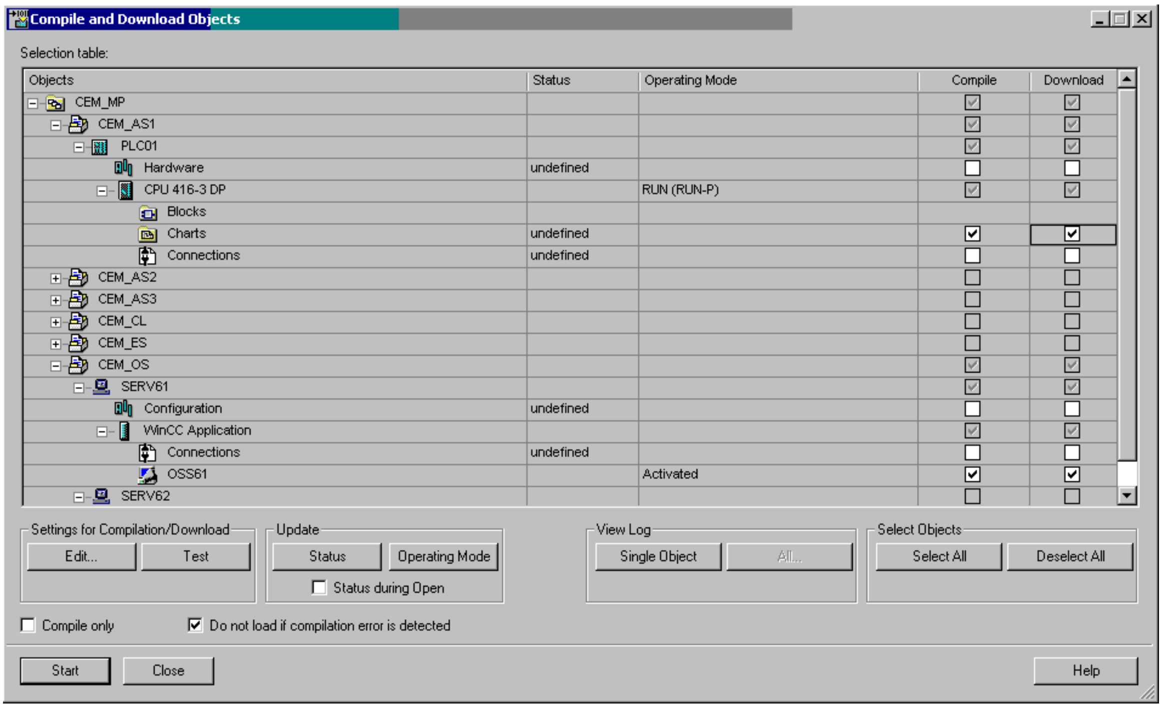
The update procedure can be performed manually (step by step) or automatically using the "Compile and Download Objects" Dialog in the SIMATIC Manager.

Manual update procedure for changes (The AS is running and both Servers are activated):

1. Compilation for changes of the AS-Program (from CFC)
2. Download for changes of the CFC
3. Compile the OS Project(s)
   The compilation of the OS must be carried out for each Server and single station.
   At the end of the OS Compile procedure the Server Data is automatically updated by the system and the updated package is automatically loaded into the Client Projects.
4. For the Download for changes in a Server Project, select the OS in the SIMATIC Manager and press the Download Button. The Download for changes is carried out first to the Standby-Server. The 'Successful completed' message must be confirmed. After that the Download is carried out to the primary Server. The Clients get automatically actualized as well (the new package is loaded).
5. For the download into the Single User Station use the same procedure.

The Function "Compile and Download Objects" allows to select the above mentioned Steps in a Selection List and to start it with one Button.

Compile and Download Objects:



Note: Even if only the OS Project for Server is selected for Compilation and Download, the Standby-Server is getting updated as well. Update for the Client is not required.

Structural changes (e. g. the format of a variable was modified for an existing object or a variable was added) result in an interface conflict and require the stop of the Runtime System during the Project Download. This is still no problem because of redundant Servers but the Download Procedure is slightly different:

1. To actualize the Tag Management an OS Compile has to be performed.
   The OS Compile must be carried out into the OS-Projects of ES and Server.
   At the end of the OS Compile procedure the Server Data is automatically updated by the system and the updated package is automatically loaded into the Client Projects.
2. After that you must close the OS-Project on the primary Server Station. First close the WinCC Explorer (if it is not already closed) and second the Runtime.
   The Standby-Server will be MASTER, the Clients will be switched to the Standby-Server.

Important: After closing the WinCC Project on the Server Station, you have to wait for at least 2 minutes. During this time the project is still “used”.

3. Now you can carry out the Project-Download from the Engineering Station to the primary Server Station. The Server-Project on the Engineering Station must remain closed. The Download is carried out from the SIMATIC Manager.
4. Wait for the “Download successfully completed” – Message before you continue.
5. After both Servers are running again, the Redundancy Manager starts actualising the Server data. If possible wait until the procedure is finished.
6. Now you can close the OS-Project on the Standby Server Station. First close the WinCC Explorer (if it is not already closed) and second the Runtime.
   The primary Server will be MASTER, the Clients will be switched to the primary Server.

Important: After closing the WinCC Project on the Server Station, you have to wait for at least 2 minutes. During this time the project is still “used”.

7. Now you can carry out the Project-Download from the Engineering Station to the Standby-Server Station. The Download is carried out from the SIMATIC Manager.
8. After the “Download successfully completed” – Message the Project Download is finished!
9. If it is necessary to update the Clients as well (may not always be required), use the same procedure for each Client.

Integrated Asset Management

SIMATIC PCS 7 with the diagnostic and maintenance functions provides the monitoring of the various components of a PCS 7 plant, and display of the status in the process mode. This status is shown with defined symbols, which are the basis for the NAMUR/PNO definition.

When diagnostic and maintenance functions are used, a Maintenance Station must be created.

For detailed information regarding configuration of a Maintenance Station refer to the PCS 7 Configuration Manual Engineering System and to the PCS 7 Configuration Manual Operator Station.

AS-AS Coupling

Content

Edition 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\07_AS-AS_Coupling_009.doc

AS-AS Coupling with PCS7 blocks

General

For the AS-AS coupling to another PCS7 AS please use only the blocks from the Standard Library and from the PCS7 Library V71.

Caution: The blocks C_PLC_SEND and C_PLC_RECEIVE from the CEMAT Library are foreseen for the communication to an AS from older CEMAT Versions (V2, V3 and V4). As with this communication only ISO Transport connection and no S7 Connection is possible, these blocks can not be recommended the communication to another PCS7 AS.

Caution: Since PCS7 V7.1 you have the possibility to create with CFC AS-wide interconnections. This possibility only exists if the program is installed in a time interrupt OB, for OB1 it is not possible.
Please use this possibility very carefully and check with the online help the limits of this kind of communication. Otherwise you could cause performance problems. If you create your communication with PCS7 blocks listed in the following, then you control how many data are sent and when the telegrams are sent.

For the AS – AS communication the following blocks are available:

USEND, URCV, BSEND, BRCV from the Standard Library

or

SEND_BO, REC_BO, SEND_R, REC_R from the PCS7 Library V71.

These blocks use as link a S7 connection.

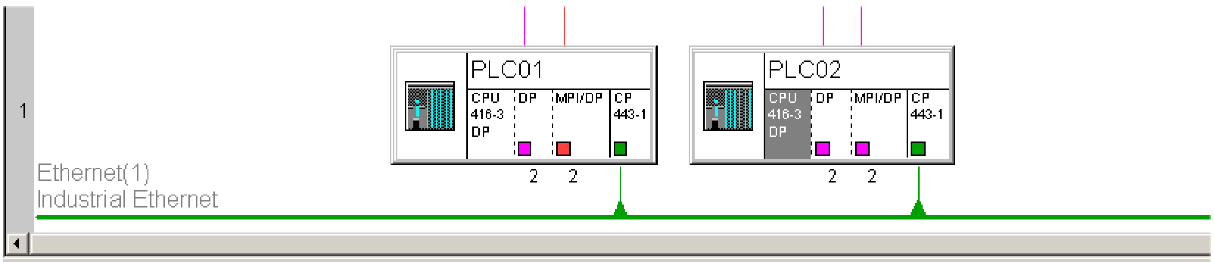
A S7 connection could be used for different telegrams between two AS. The parameter R_ID is for the determination of the different telegrams, which are transported via the same S7 connection.

Project settings in NETPRO:

S7 connection:

E.g. the ID "A2" is in AS PLC01 and

e. g. the ID "A1" is in AS PLC02.



Local ID	Partner ID	Partner	Type	Active connection partner	Subnet
A1	A2	PLC01 / CPU 416-3 DP	S7 connection	No	Ethernet(1) [IE]

Send with SEND_BO

With block SEND_BO up to 128 BOOL values can be sent. If more than 128 BOOL values are needed, the block can be called multiple times. The different orders are identified by Parameter R_ID. In the destination AS a block REC_BO must be called which has the same R_ID as in the corresponding send call.

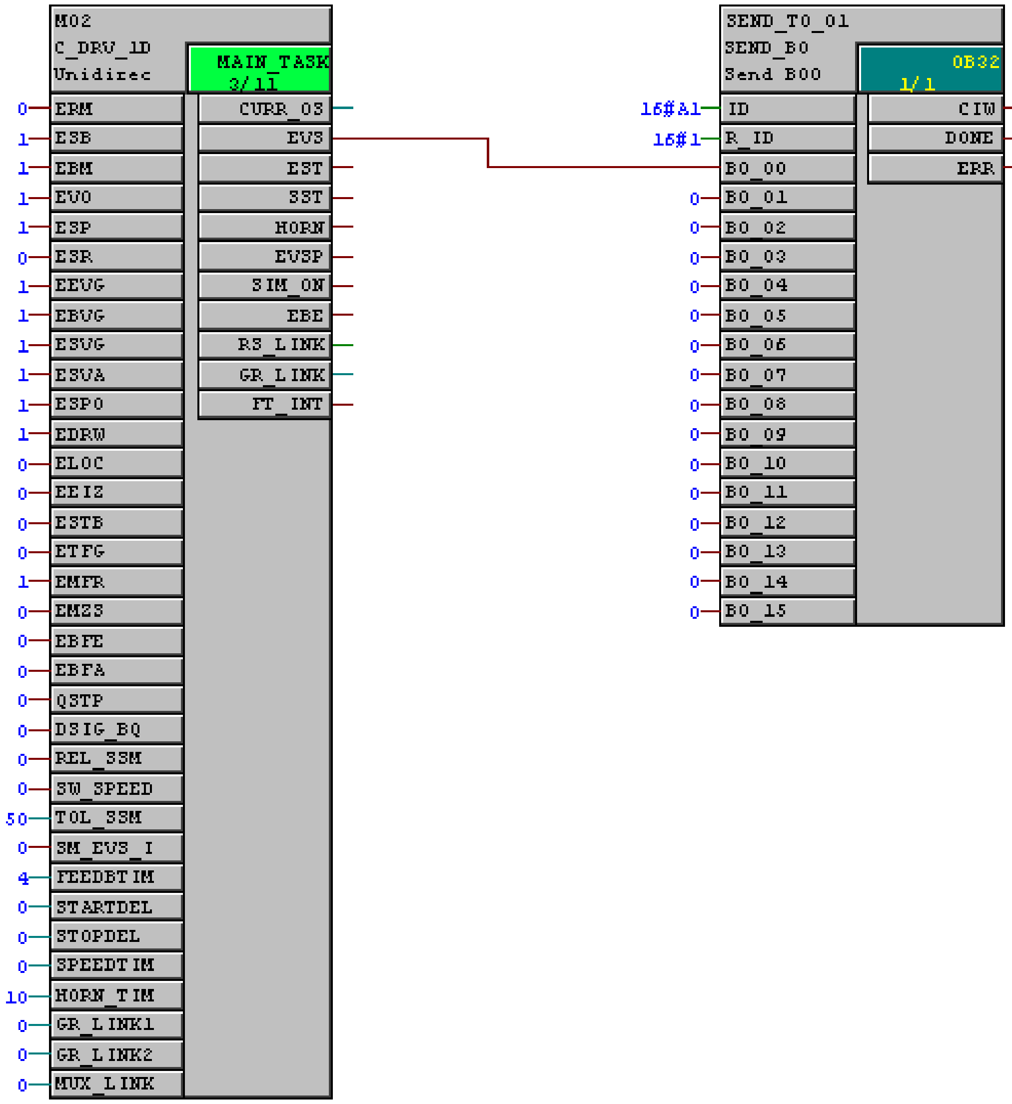
The ID of the S7 connection you find in the connection programming in NETPRO.

Example: PLC02 sends data to PLC01:
According to the NETPRO connections above the Send call in the AS PLC02 must have the ID "A1". The corresponding Receive call in the AS PLC01 must have the ID "A2".

The send job will be triggered by the (invisible) parameter "COM". To trigger the send job each second then there are two possibilities:

1. Keep the parameter "COM" at default value "1" and call the block SEND_BO in OB32 (each second).
2. Change parameter "COM" to "visible" and connect it to a pulse (one pulse each second). The block SEND_BO must be called at least every second (corresponding OB).

See the help function of SEND_BO.



Receive with REC_BO

With block REC_BO up to 128 BOOL values can be received. If more than 128 BOOL values are needed, the block can be called multiple times. The different orders are identified by Parameter R_ID. On the send side a block SEND_BO must be called which has the same R_ID as in the corresponding receive call.

The ID of the S7 connection you find in the connection programming in NETPRO.

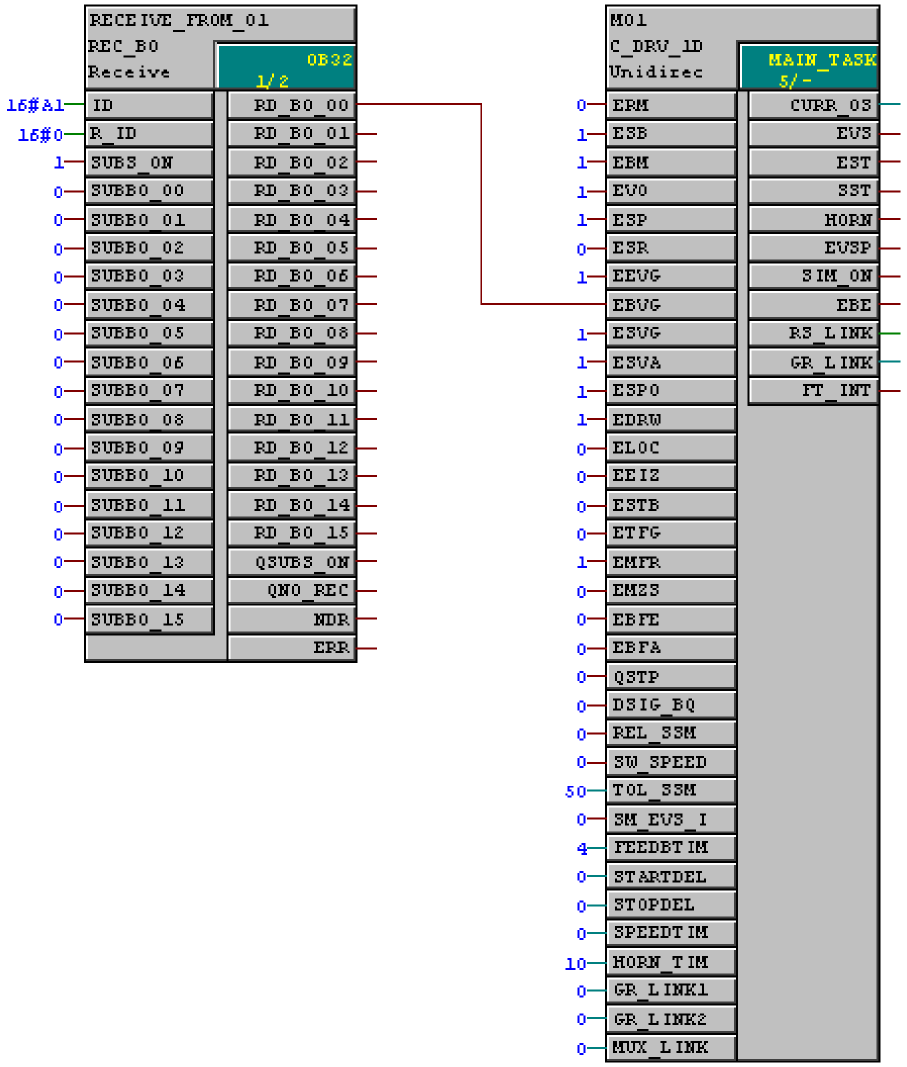
Example: PLC02 receives data from PLC01.
According to the NETPRO connections above the Receive call in the AS PLC02 must have the ID "A1". The corresponding Send call in the AS PLC01 must have the ID "A2".

To receive the data correctly two calls of the REC_BO are necessary. That means if you call the REC_BO every second you will receive data only every 2 seconds. If you need the data every second, then you should call the block REC_BO in the 500 ms OB.

The monitoring can be parameterized through the (invisible) parameter REC_MON. The default value is 3 cycles (calls). If a longer monitoring time is needed, then the value on parameter REC_MON must be increased.

If after the predefined number of block calls no telegram is received, then there is a fault and all values can be overwritten by substitution values.
For each of the 128 BOOL variables a substitution value can be defined. With parameter SUBS_ON you can decide if you want to use substitute values or not.

See the help function of REC_BO.



Send with SEND_R

With block SEND_R up to 32 BOOL values and 32 REAL values can be sent. If more than 32 BOOL values or more than 32 REAL values are needed, the block can be called multiple times. The different orders are identified by Parameter R_ID. In the destination AS a block REC_R must be called which has the same R_ID as in the corresponding send call.

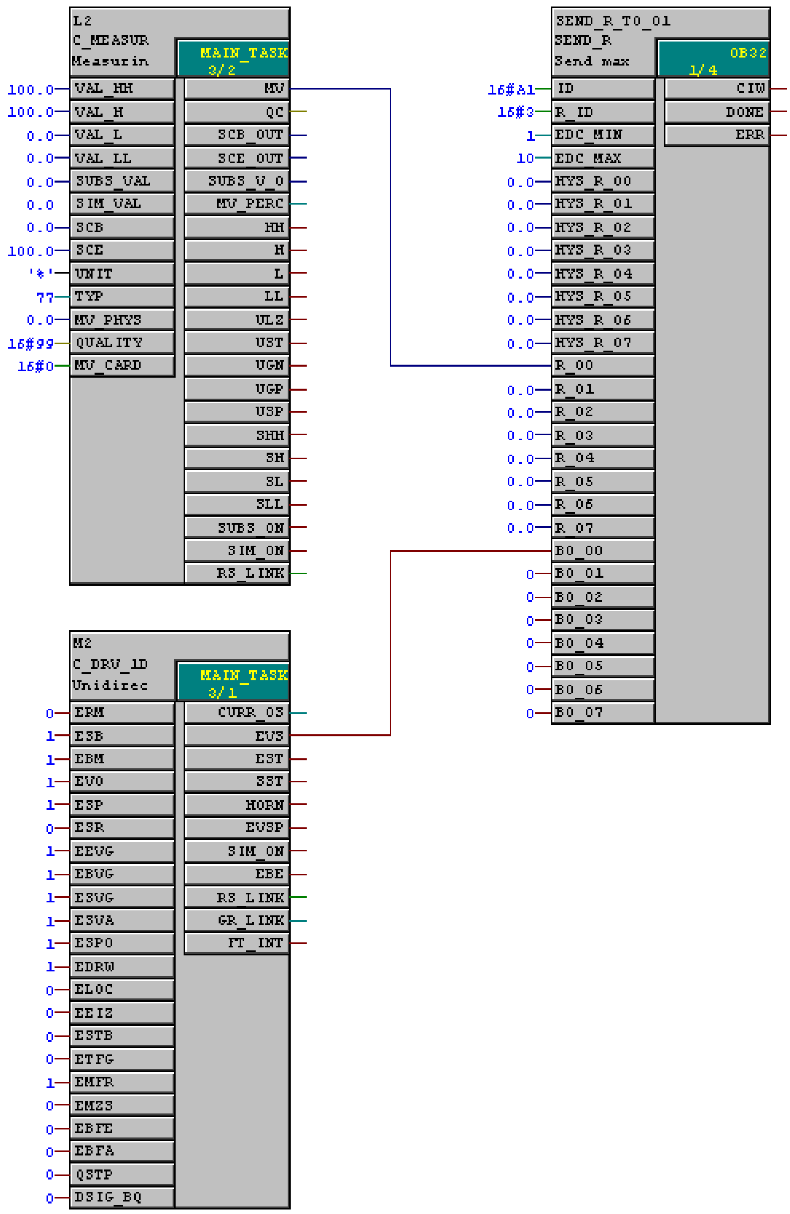
The ID of the S7 connection you find in the connection programming in NETPRO.

Example: PLC02 sends data to PLC01:
According to the NETPRO connections above the Send call in the AS PLC02 must have the ID "A1". The corresponding Receive call in the AS PLC01 must have the ID "A2".

The send job will be triggered by the (invisible) parameter "COM". To trigger the send job each second then there are two possibilities:

1. Keep the parameter "COM" at default value "1" and call the block SEND_R in OB32 (each second).
2. Change parameter "COM" to "visible" and connect it to a pulse (one pulse each second). The block SEND_R must be called at least every second (corresponding OB).

See the help function of SEND_R.



Receive with REC_R

With block REC_R up to 32 BOOL values and 32 REAL values can be received. If more than 32 BOOL values and 32 REAL values are needed, the block can be called multiple times. The different orders are identified by Parameter R_ID. On the send side a block SEND_R must be called which has the same R_ID as in the corresponding receive call.

The ID of the S7 connection you find in the connection programming in NETPRO.

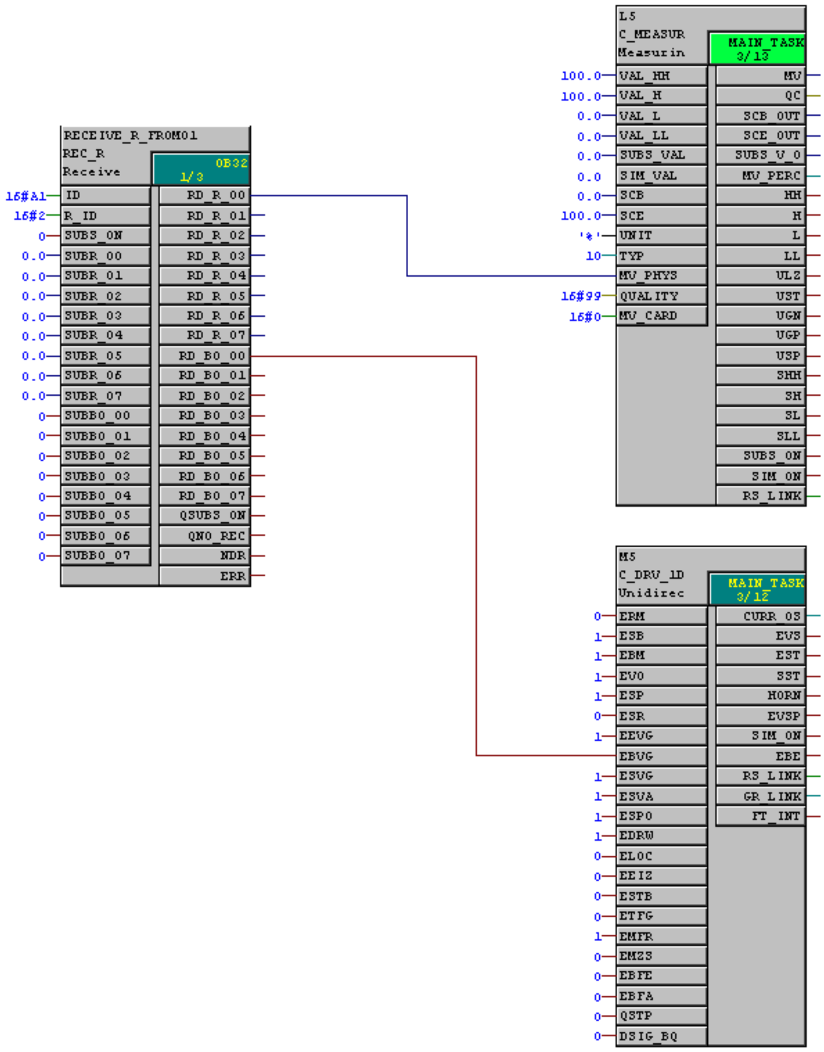
Example: PLC02 receives data from PLC01.
According to the NETPRO connections above the Receive call in the AS PLC02 must have the ID "A1". The corresponding Send call in the AS PLC01 must have the ID "A2".

To receive the data correctly two calls of the REC_R are necessary. That means if you call the REC_R every second you will receive data only every 2 seconds. If you need the data every second, then you should call the block REC_R in the 500 ms OB.

The monitoring can be parameterized through the (invisible) parameter REC_MON. The default value is 3 cycles (calls). If a longer monitoring time is needed, then the value on parameter REC_MON must be increased.

If after the predefined number of block calls no telegram is received, then there is a fault and all values can be overwritten by substitution values.
For each of the 32 BOOL variables and the 32 REAL values a substitution value can be defined. With parameter SUBS_ON you can decide if you want to use substitute values or not.

See the help function of REC_R.



Send with BSEND

How to create your own function block with input parameters of different formats, which are sent using BSEND block.

The Cemat Library contains the two function blocks SEND_CFC_EXAMPLE (FB 1601) and BSEND_CALL (FB 1603). These blocks are an example for the send of any CFC parameters.

Based on the SEND_CFC_EXAMPLE you have to create "your" SEND_TO_AS02 as required: E. g. 20 Boolean, 5 Integer and 8 Real values have to be sent from AS01 to AS02. In this case you have to create a function block with at least 20 Boolean, 5 Integer and 8 Real values as input parameters.
Additionally the block needs two output parameters of the type integer (I_DB and LENGTH) and the following 4 local variables:

```
VAR_TEMP
  DBNR : WORD ;
  RET_VALUE : INT ;
  DBLENGTH : WORD ;
  NO_WRITE : BOOL ;
END_VAR
```

We recommend to creating some spare variables of each type. In this case you can easily send some more data without an interface change (AS stop). Renaming of the parameters later on is no problem (no interface change). Beside these individual input parameters and the two output parameters there should be no further parameter. The function block needs the following STL code:

```
      L     DINO;
      T     #I_DB;                   //SEND-DB into variable typ WORD
      T     #DBNR;
      CALL  "TEST_DB" (
            DB_NUMBER        := #DBNR,
            RET_VAL          := #RET_VALUE,
            DB_LENGTH        := #DBLENGTH,
            WRITE_PROT       := #NO_WRITE);
      L     #DBLENGTH;
      +     -4;                      //- I_DB and LENGHT
      T     #LENGHT;
      BE    ;
```

See example function block FB SEND_CFC_EXAMPLE (FB 1601)!

For each SEND_CFC_EXAMPLE you have to create the appropriate function block RECEIVE_CFC_EXAMPLE for receiving the data with exactly the same parameters. But on the receive side these are output parameters. See chapter with RECEIVE_CFC_EXAMPLE.

The function block BSEND_CALL (FB 1603) is a simple call of the system function block SFB12 BSEND. Up to 65534 byte could be send with the function block BSEND.

Attention: Max. 400 bytes can be transferred in one telegram as a consistent data block. If you have to transfer more data and the consistency is important you have to take care of this at the receive side. See consistency at the receiving!

Block connection

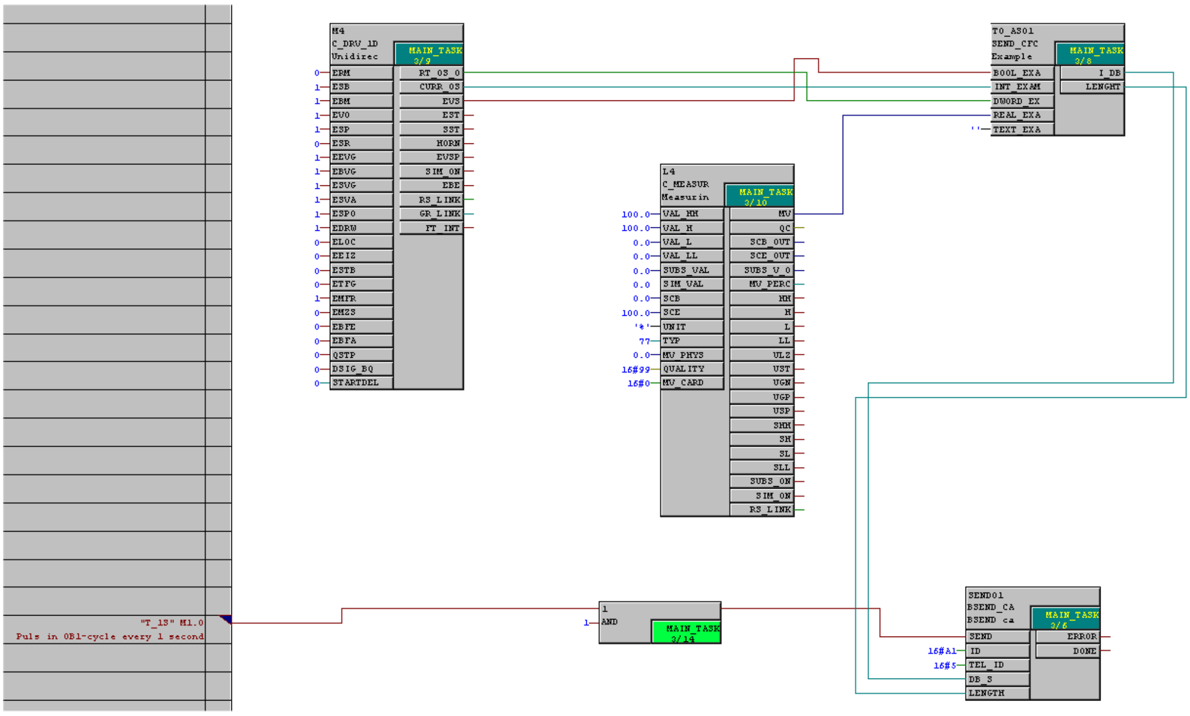
Connect in CFC the parameter I_DB (SEND_CFC_EXAMPLE) with DB_S (BSEND_CALL) and the parameter LENGTH (SEND_CFC_EXAMPLE) with LENGTH (BSEND_CALL). Parameter ID must contain the local ID of the S7 Connection. The Parameter TEL_ID corresponds to the Parameter R_ID of the blocks BSEND, SEND_BO or SEND_R. Thus multiple telegrams for the same S7 connection can be distinguished.

The send order is triggered via the parameter "SEND". If the program has to be sent every second, the parameter "SEND" must be connected accordingly (every second a pulse).

The status and the fault of BSEND can be seen in the output parameters ERROR, DONE and STAT. If more detailed information is required the Example function block BSEND_CALL has to be modified.

See the help function of BSEND.

Example for connections between SEND_CFC_EXAMPLE and BSEND_CALL



Receive with BRCV

How to create your own receive block, using BRCV block, in order to Receive CFC-Parameters of different formats.

The Cemat Library contains the two function blocks RECEIVE_CFC_EXAMPLE (FB 1602) and BRCV_CALL (FB 1604). These blocks are an example for receiving any desired CFC parameters.

Example:
AS02 shall receive 20 Boolean, 5 Integer and 8 Real values from AS01. In this case create based on the REICEIVE_CFC_EXAMPLE (and equivalent to the "SEND_TO_AS02" of the source AS) a function block "REICEIVE_FROM_AS01" with at least 20 Boolean, 5 Integer and 8 Real values as output parameters. Additionally the block needs two output parameters of the type integer (I_DB and LENGTH) and the following local variables:

```
VAR_TEMP
  DBNR : WORD ;
  RET_VALUE : INT ;
  DBLENGTH : WORD ;
  NO_WRITE : BOOL ;
END_VAR
```

We recommend to creating some spare variables of each type. In this case you can easily send some more data without an interface change (AS stop). Renaming of the parameters later on is no problem (no interface change). Beside these individual output parameters and the two output parameters there should be no further parameter. The function block needs the following STL code:

```
L      DINO;
T      #I_DB;               //RECEIVE-DB into variable typ WORD
T      #DBNR;
CALL   "TEST_DB" (
       DB_NUMBER        := #DBNR,
       RET_VAL          := #RET_VALUE,
       DB_LENGTH        := #DBLENGTH,
       WRITE_PROT       := #NO_WRITE);
L      #DBLENGTH;
+      -4;                  //- I_DB and LENGHT
T      #LENGHT;
BE     ;
```

See Example function block RECEIVE_CFC_EXAMPLE!

The function block BRCV_CALL (FB 1604) is a simple call of the system function block SFB13 BRCV. Up to 65534 byte could be send with the function block BSEND.

Attention: Max. 400 bytes can be transferred in one telegram as a consistent data block. If you have to transfer more data and the consistency is important you have to take care of this after receiving. In this case the function block BRCF_CALL (FB1604) must be modified in order to write the data in a receive buffer and to copy the data only after the transfer is successfully completed.
If you don't have enough programming knowledge of STL to write the function yourself, please contact a system integrator.

Block connection

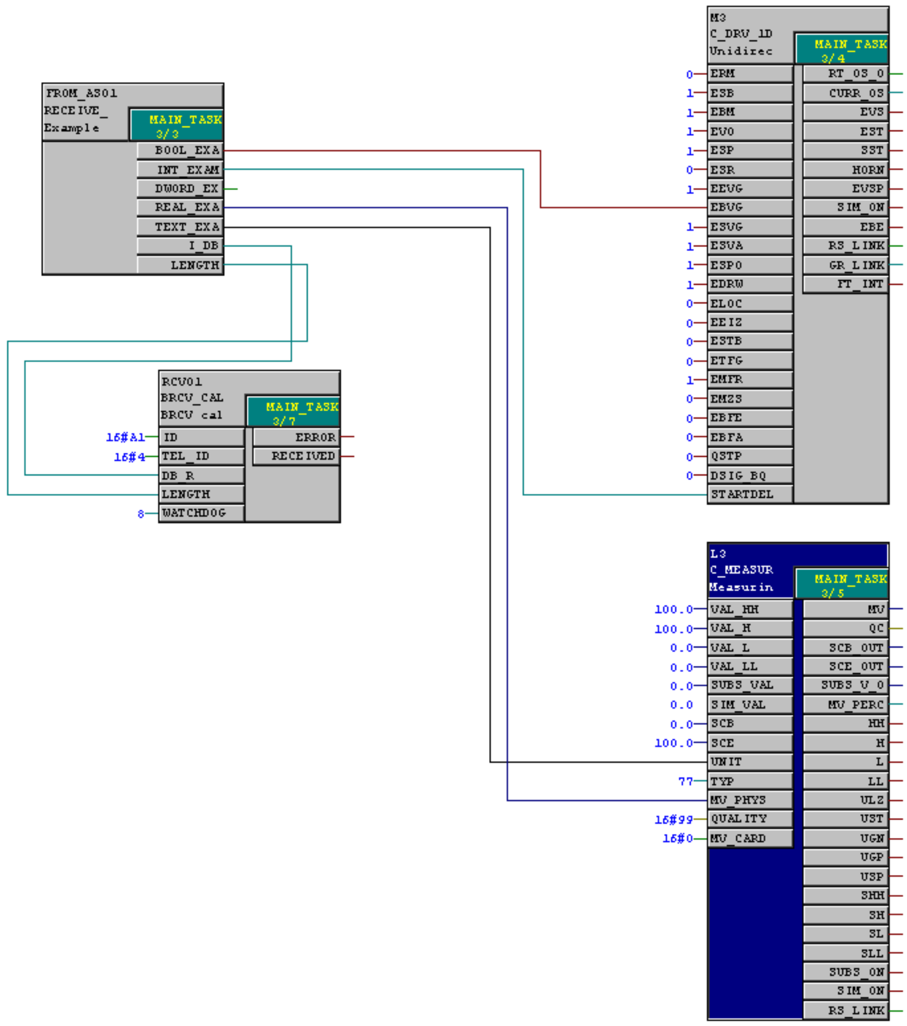
Connect in CFC the parameter I_DB (RECEIVE_CFC_EXAMPLE) with DB_R (BRCV_CALL) and the parameter LENGTH (RECEIVE_CFC_EXAMPLE) with LENGTH (BRCV_CALL). Parameter ID must contain the local ID of the S7 Connection. The Parameter TEL_ID corresponds to the Parameter R_ID of the blocks BRCV, REC_BO or REC_R. Thus multiple telegrams for the same S7 connection can be distinguished.

Through parameter WATCHDOG_SET a supervision time can be set. The default for WATCHDOG_SET is 8 seconds. This means, if after 8 seconds no telegram has been received, the receive data are overwritten with 0. If a different supervision time is required, the value for WATCHDOG-SET must be adapted. .

The status and the fault of BRCV can be seen in the output parameters ERROR, DONE and STAT. If more detailed information is required the Example function block BRCV_CALL has to be modified.

See the help function of BRCV.

Example for connections between RECEIVE_CFC_EXAMPLE and BRCV_CALL



Function block adjustments

The blocks SEND_CFC_EXAMPLE, BSEND_CALL, RECEIVE_CFC_EXAMPLE and BRCV_CALL are only programming examples. The blocks are not know how protected and could be changed for your own requirements. You must consider that on each Cemat-Update the blocks are overwritten. Therefore use for your modified blocks different function block numbers. For each transport connection the must be exist a pair of the blocks SEND_CFC_EXAMPLE and RECEIVE_CFC_EXAMPLE.

AS-AS Coupling to older CEMAT Versions

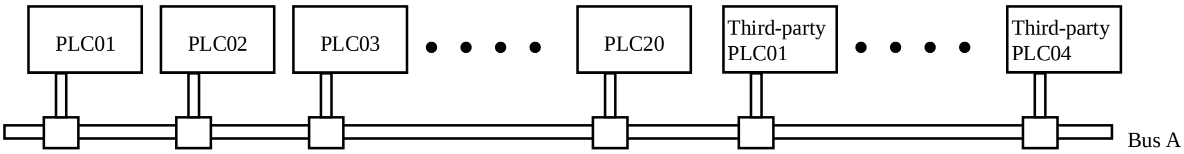
General

Caution: This AS-AS communication from the CEMAT Library should exclusively be used for communication to AS with older CEMAT Generations (V2, V3, V4), because the connection type has to be an ISO transport connection. For the communication between one PCS7 AS to another PCS7 AS, we recommend “S7 connections” and the standard communication blocks from the PCS7 library.
In the following description the expression "AS“ is used for automation system. In the TSAP and in the Variable tables the expression "AG“ is used for automation system. This is necessary to be compatible with older CEMAT versions (V3, V4).

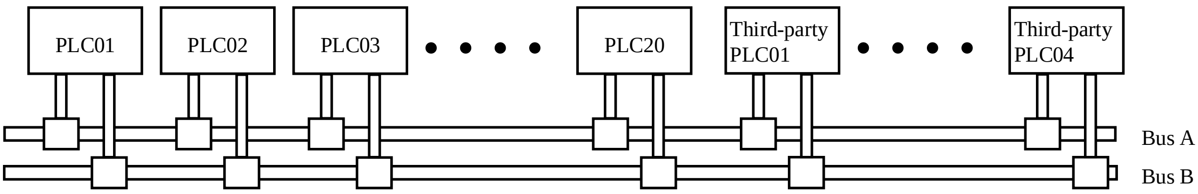
The PLC-PLC Coupling of the Cemat V8.1 Library enables the transmission of user data blocks (DB599 >= DB => DB 400) between SMR-PLCs. The communication is performed via Industrial Ethernet which can be designed as a double bus. The user configures the job parameters in CFC with the blocks C_PLC_SEND = FB1052 and C_PLC_RECEIVE = FB1053. The send job is initiated by setting the job bit TRIGGER and the reception is signalled through a receive bit RECEIVED.

Complete Structure

Single bus:



Double bus:



Performance Data

- 20 SMR-PLCs and 4 third-party PLCs (CEMAT V1.9, V2.0, V3.0 / V4.0 other control system or third-party system).
- Transmission of CEMAT user data blocks (DB599 >= DB => DB 400).
- Max. transmission length = 600 Bytes.
- Initiation of the individual transmissions by setting the initiation bit TRIGGER in the send PLC.
- Signalling of the successful transmissions through a set message bit RECEIVED in the receive PLC.
- Coupling monitoring in case of parameterized coupling request.
- Individual user telegram monitoring for each of the 24 couplings:
  If the monitoring time is exceeded the receive DB can be deleted on request completely or partially.
- Diagnostics interfaces like:
  - Global display for bus A and bus B from the point of view of each individual PLC: Bus is OK.
  - Receipt OK (bus A or bus B) for each parameterized coupling job.
  - Parameterization error for each parameterized coupling job.
  - Status (bus A or bus B) for each parameterized coupling job.

Activation of Coupling

By calling the blocks C_PLC_SEND = FB1052 and C_PLC_RECEIVE = FB1053 in a CFC chart, the user can parameterize his coupling jobs. As soon as a valid coupling job is parameterized the Coupling becomes active. You do not have to do anything else.
To install a coupling job and then to activate it, the best is to proceed as follows:

If the links don't exist, create the links with NETPRO. See chapter "Project network communication" page 11.

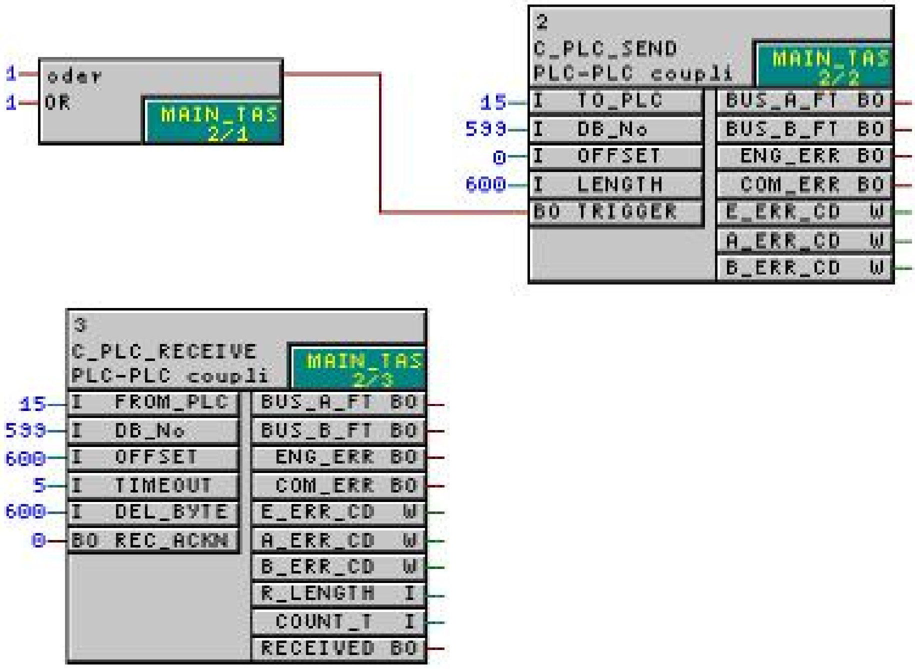
1. Parameterize the send parameters on the function block C_PLC_SEND = FB1052.
2. Parameterize the receive parameters in the corresponding PLC on the function block C_PLC_ RECEIVE = FB1053.
3. Programming the logic for initiation (SET TRIGGER).
4. Compile the CFC chart new and transfer the changes to the PLC. If the links already exits, the transfer is possible during running PLC.

Further possibilities like connection monitoring see page 16.

Interfaces

Initiation Bits and Receive Indicator for Coupling Job

The user must set the corresponding initiation bit to activate the send job = TRIGGER and he can find out through a query of the corresponding receive bit = RECEIVED whether data have arrived. Is the RECEIVED bit used, after query the Bit REC_ACKN must be set. If the REC_ACKN bit is set, the Bit RECEIVED is reset. The initiation and receive bits are located in the parameter set of the FBs C_PLC_SEND and C_PLC_RECEIVE.



Parameters for Coupling Jobs

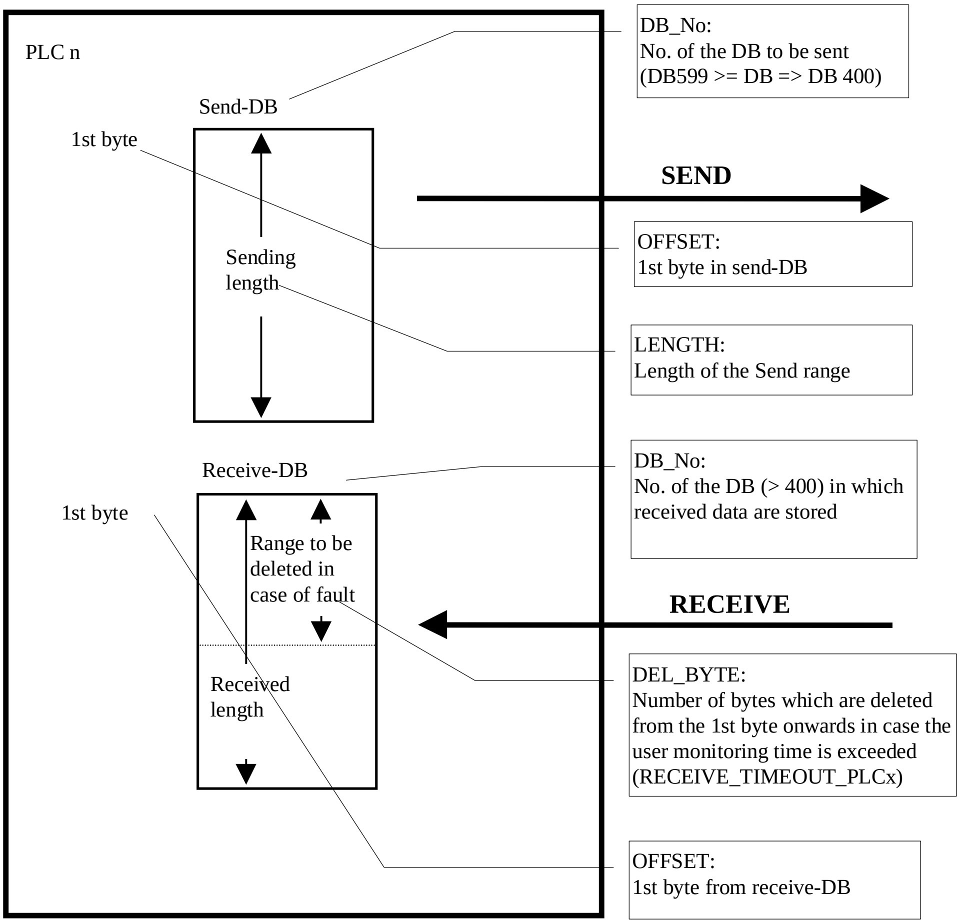
All settings are made in the own PLC. There are no settings (like DB no.) for the partner PLC. The send parameters are made in the sending PLC. The receive parameters are made in the receiving PLC.

Parameters for the send job:

TO_PLC = PLC No. , to which the job is sended (1-24).
DB_No = DB-No. in the sending PLC with the source data (400-599).
OFFSET = start byte no. of the source area (0-65535).
LENGTH = number of bytes of the source area (1-600).

Parameters for the receive job:

FROM_PLC = PLC No.,from where the job is received (1-24).
DB_No = DB-No. in the receiving PLC with the destination data (400-599).
OFFSET = start byte no. of the destination area (0-65535).
TIMEOUT = watch dog time in seconds.
DEL_BYTE = number of bytes which will be deleted, when the watch dog time is elapsed. (start byte is the start byte of the destination area).


PLC n
Send-DB
1st byte
Sending length
DB_No:
No. of the DB to be sent
(DB599 >= DB => DB 400)
SEND
OFFSET:
1st byte in send-DB
LENGTH:
Length of the Send range
Receive-DB
1st byte
Range to be deleted in case of fault
Received length
DB_No:
No. of the DB (> 400) in which received data are stored
RECEIVE
DEL_BYTE:
Number of bytes which are deleted from the 1st byte onwards in case the user monitoring time is exceeded
(RECEIVE_TIMEOUT_PLCx)
OFFSET:
1st byte from receive-DB

Diagnosis Variable for Users

A fault-free coupling is only possible if the parameterization error bits and fault bits have the value 0. Otherwise one must search for the fault and correct it using the additional information.

Diagnosis SEND in the parameter-set of FB C_PLC_SEND:

BUS_A_FT	= Fault during SEND on bus A to PLC x
BUS_B_FT	= Fault during SEND on bus B to PLC x
ENG_ERR	= Parametrize failure on SEND to PLC x
COM_ERR	= dynamic fault during SEND to PLC x
E_ERR_CO	= Error code for engineering failure for SEND to PLC x
A_ERR_CO	= Error code for SEND on bus A to PLC x
B_ERR_CO	= Error code for SEND on bus B to PLC x

Diagnosis RECEIVE in the parameter-set of FB C_PLC_ RECEIVE:

BUS_A_FT	= Fault during RECEIVE on bus A to PLC x
BUS_B_FT	= Fault during RECEIVE on bus B to PLC x
ENG_ERR	= Parametrize failure on RECEIVE to PLC x
COM_ERR	= dynamic fault during RECEIVE to PLC x
E_ERR_CO	= Error code for engineering failure for RECEIVE to PLC x
A_ERR_CO	= Error code for RECEIVE on bus A to PLC x
B_ERR_CO	= Error code for RECEIVE on bus B to PLC x
R_LENGTH	= Received telegram length in bytes
COUNT_T	= telegram counter

Additional informationen for parameterization error in ENG_ERR_CODE:

0	=	No error!
1	=	Illegal DB (only > 400)!
2	=	DB does not exist!
3	=	1st DW is not permitted!
4	=	Send-DB is too short!
5	=	Sending length is not permitted (max. 238 bytes)!
6	=	Receive-DB for block delete during absence of user telegram is too short!
7	=	Block length for block delete during absence of user telegram is not permitted!
8	=	Receive-DB is too short for received length!
9	=	A send-DB is entered in the parameter-set for your own PLC!
10	=	A receive-DB is entered in the parameter-set for your own PLC!

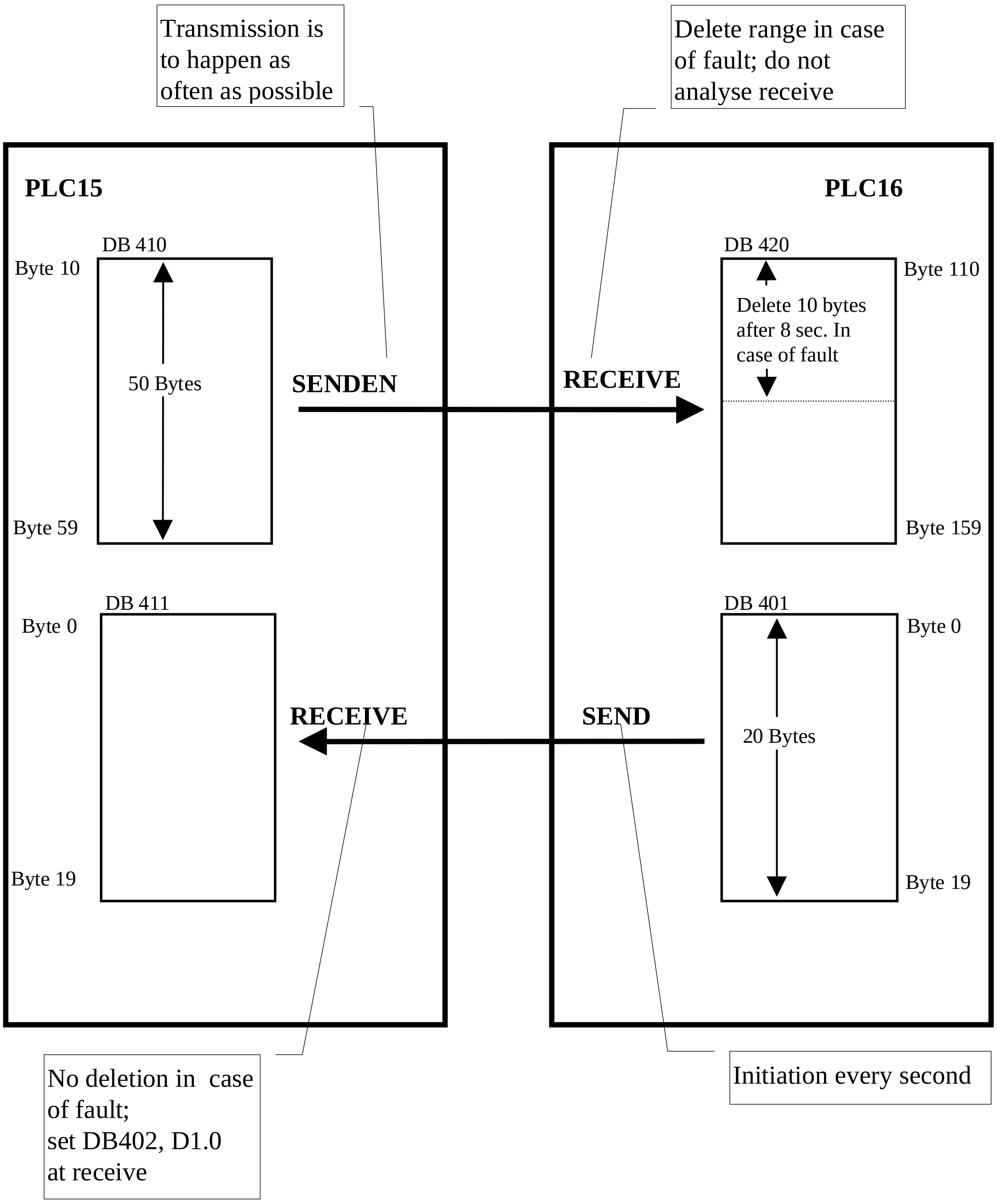
Error codes in case of a communication fault:

"AG_AG_KOMMUNIKATION".VERBINDUNG_x.FEHL_SEND_A	Error code	SEND Bus A
"AG_AG_KOMMUNIKATION".VERBINDUNG_x.FEHL_SEND_A	Error code	SEND Bus B
"AG_AG_KOMMUNIKATION".VERBINDUNG_x.ANZW_REC_A	Error code	RECEIVE Bus A
"AG_AG_KOMMUNIKATION".VERBINDUNG_x.ANZW_REC_A	Error code	RECEIVE Bus B

Additional information regarding error codes is available in the help (mark block and press „F1“) of FC50 (SEND) and FC60 (RECEIVE) or in the NCM S7 manual for Industrial Ethernet.

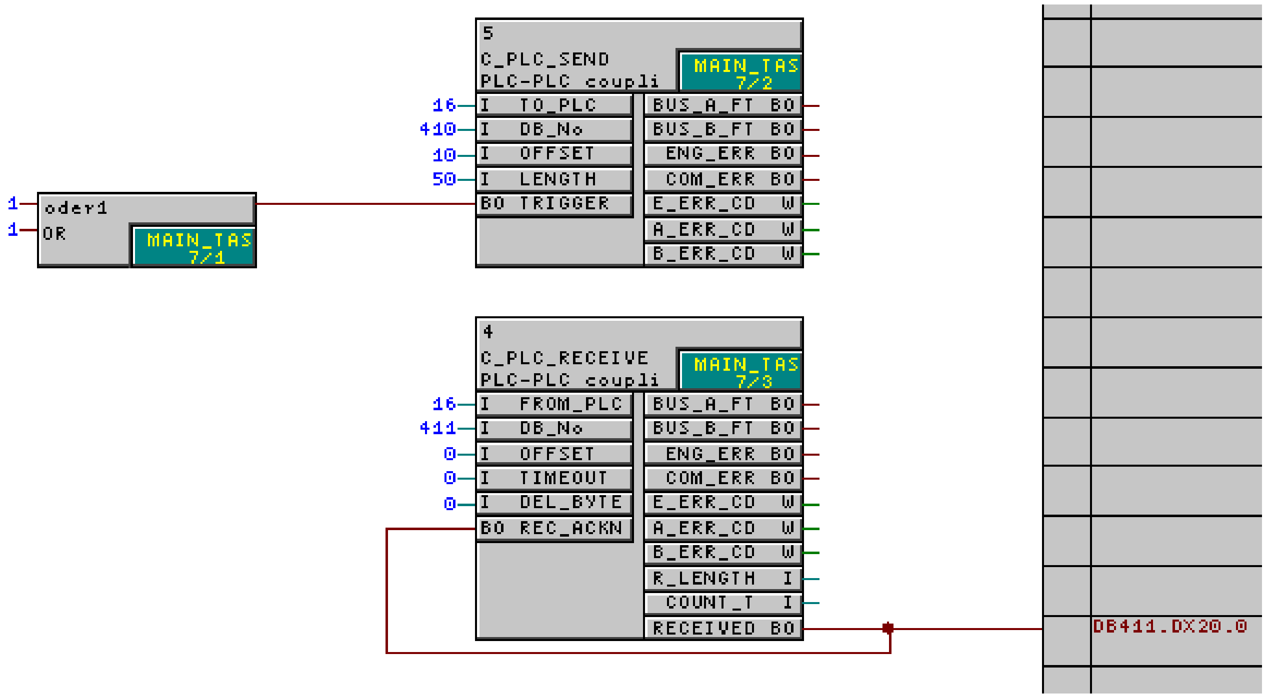
Configuration Example

Exemplary Description

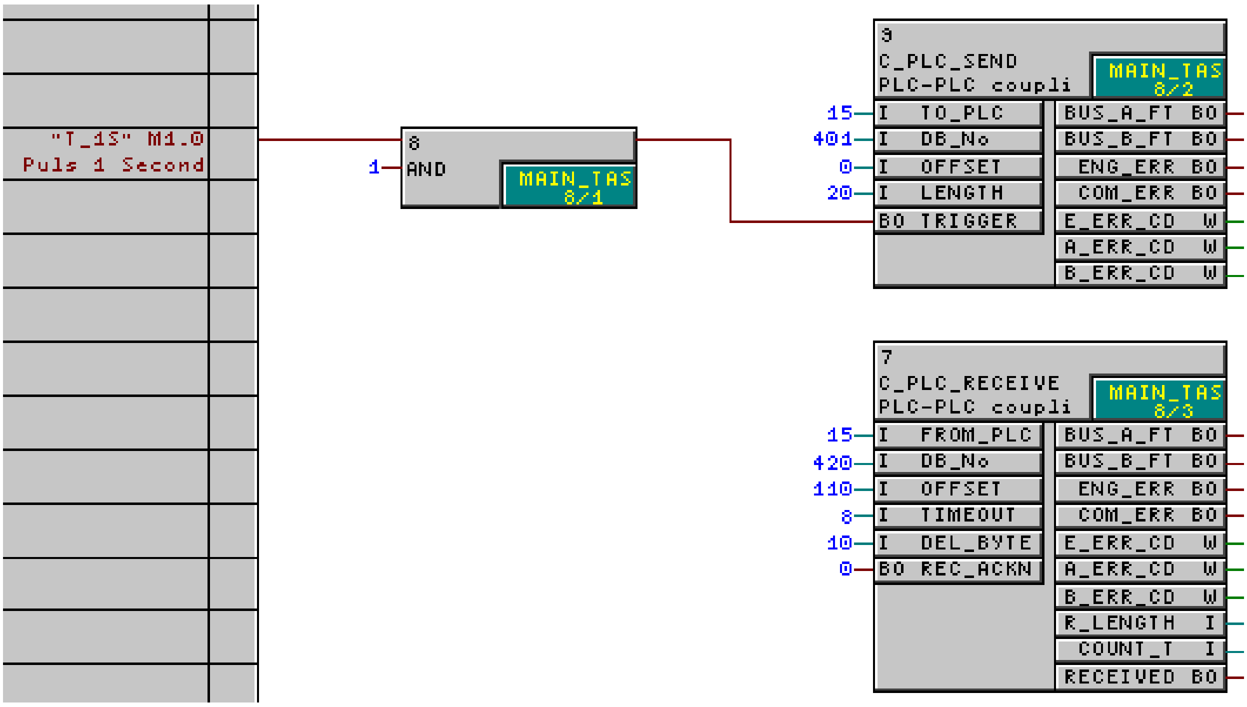


CFC chart for example

PLC 15:

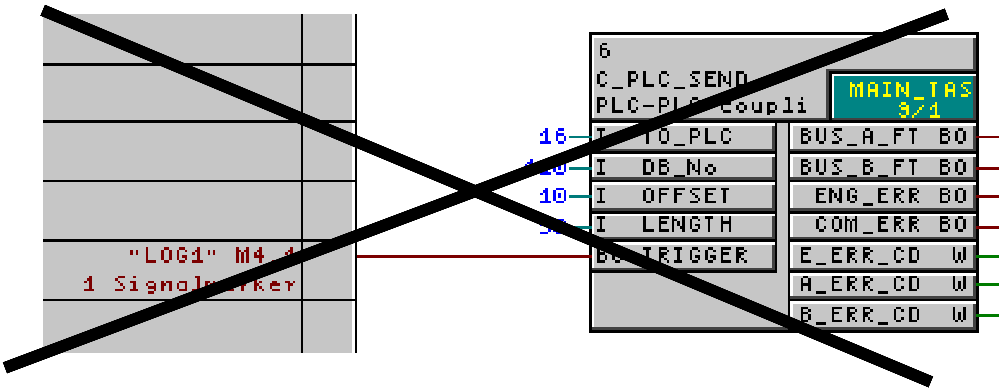


PLC 16:



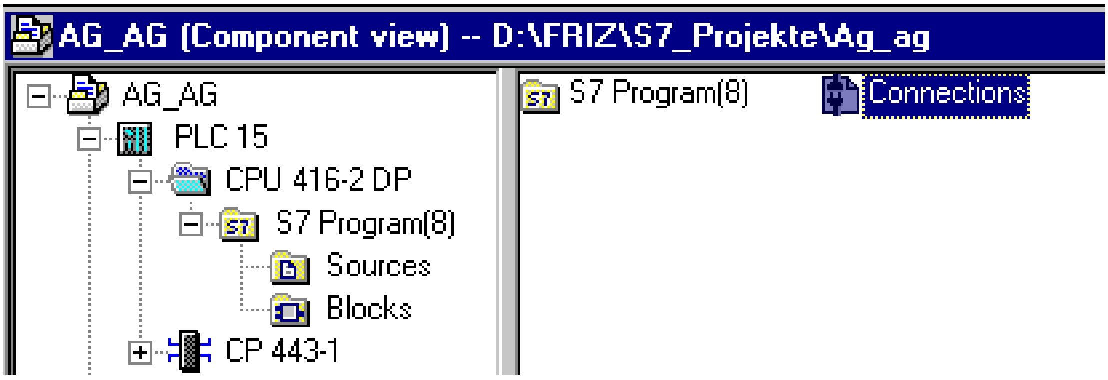
Please be careful during connecting the TRIGGER bit on the SEND block. Please do not use the LOG1 signal.

The LOG1 signal would be reseted after a succesful SEND job.

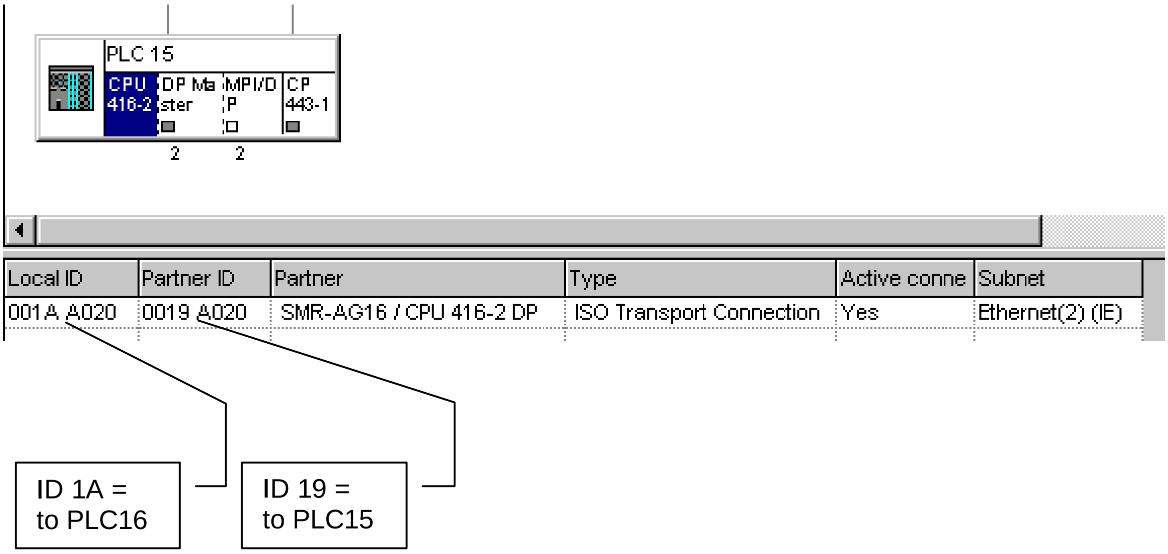


Settings in the Network Communication

Select in the SIMATIC Manager the CPU container and open NETPRO with double click on the connections.

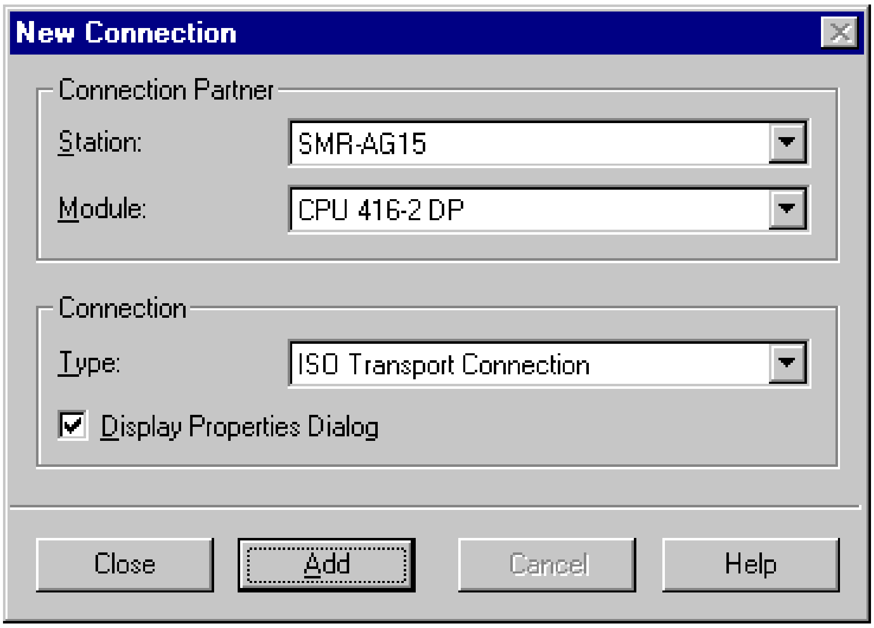


Select the CPU -> you see the connection table:



Connection	Local ID	Partner ID	Active connection setup
PLC15 to PLC16	1A	19	yes
PLC16 to PLC15	19	1A	no

Select the CPU and choose with right mouse button „New connection“:



Connection Partner Station

Select the station to which the connection should be established, in our example the PLC 16.

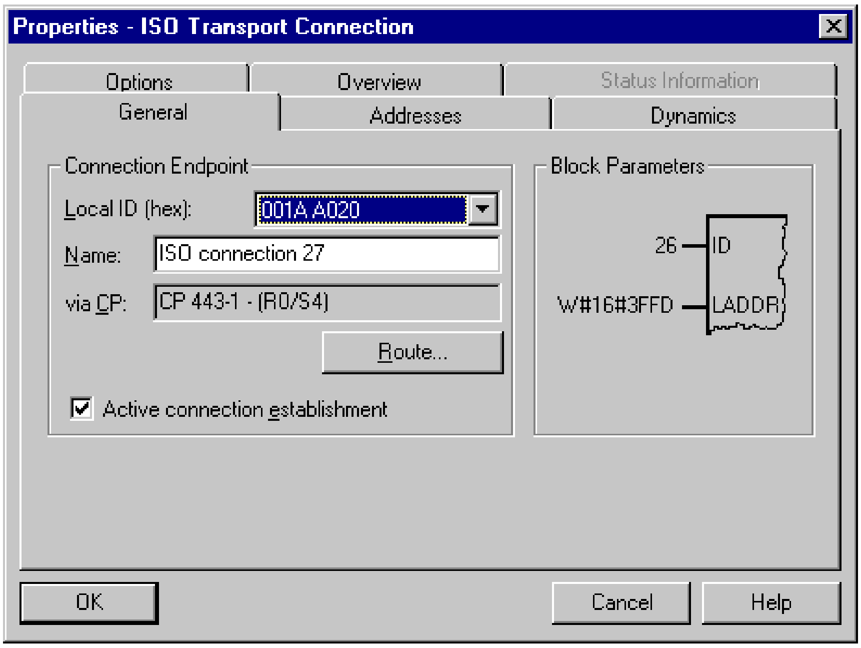
Connection type

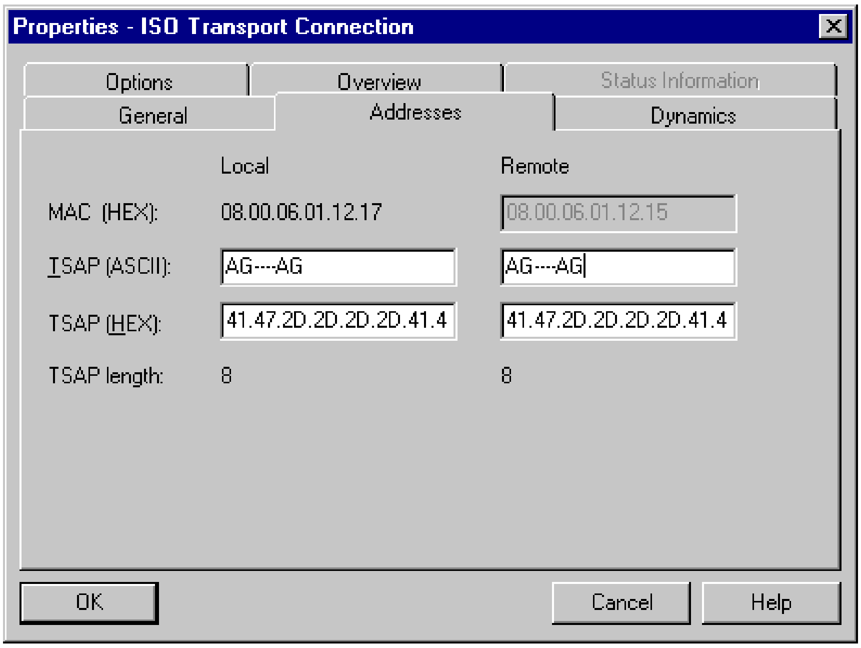
The connetion type is *ISO-Transport-Connection*

(Local) ID

The connection between the CPU program and the connection link is done by the (local) ID. For CEMAT exist the following rule:

ID	Partner PLC
0xB	PLC 1
0xC	PLC 2
0xD	PLC 3
0xE	PLC 4
0xF	PLC 5
0x10	PLC 6
0x11	PLC 7
0x12	PLC 8
0x13	PLC 9
0x14	PLC 10
0x15	PLC 11
0x16	PLC 12
0x17	PLC 13
0x18	PLC 14
0x19	PLC 15
0x1A	PLC 16
0x1B	PLC 17
0x1C	PLC 18
0x1D	PLC 19
0x1E	PLC 20
0x1F	PLC 21
0x20	PLC 22
0x21	PLC 23
0x22	PLC 24





TSAP

For the connection setup is for every connection a local and a remote TSAP necessary. The ethernet adress allone is not enough for the connection setup. There are more than one connections possible betwenn two ethernet cards.

For CEMAT exist the rulel: **local TSAP = remote TSAP**

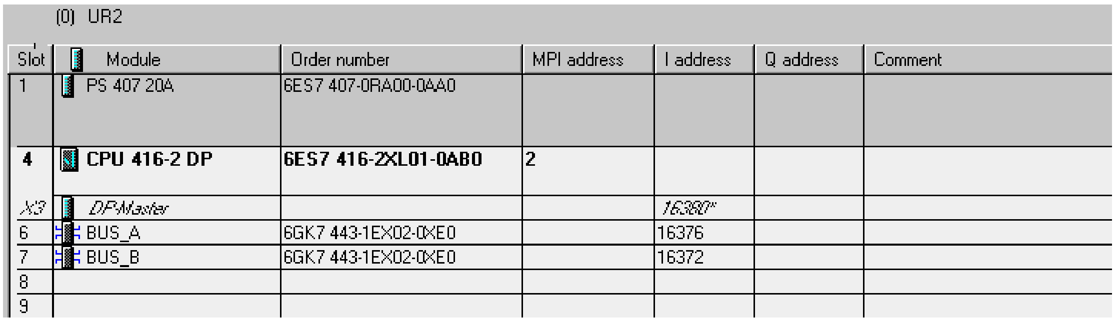
For PLC-PLC-Connections is the TSAP „**AG----AG**“ (Hexadezimalcode: 41 47 2D 2D 2D 2D 41 47).

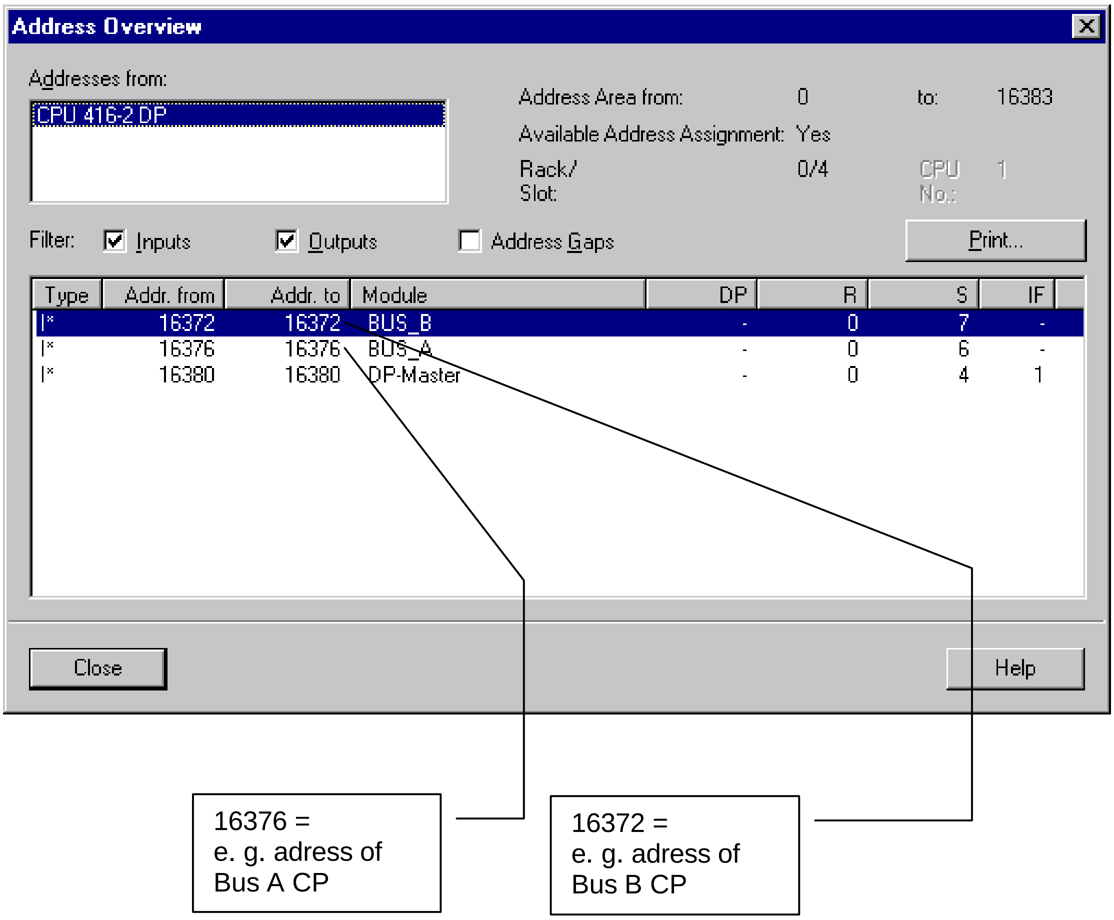
ACTIV / PASSIV

For PLC-PLC-Connections is defined: The communication partner with the smaller PLC no is **activ** for the connection setup.

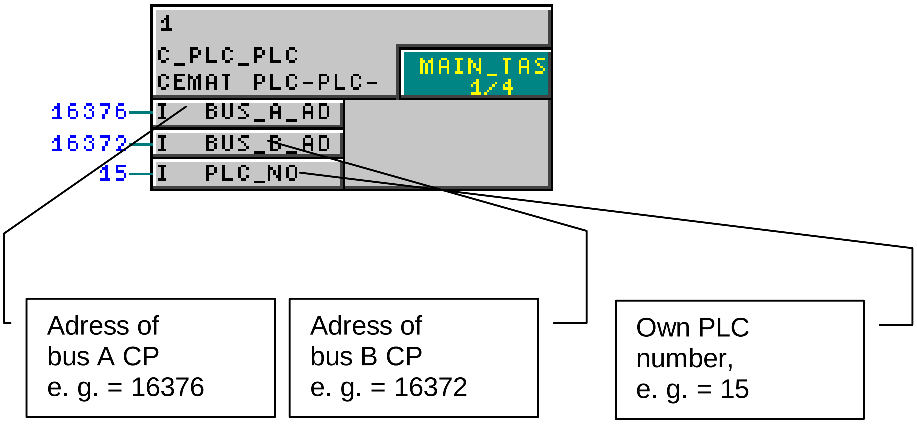
Project work for the system call of PLC PLC coupling

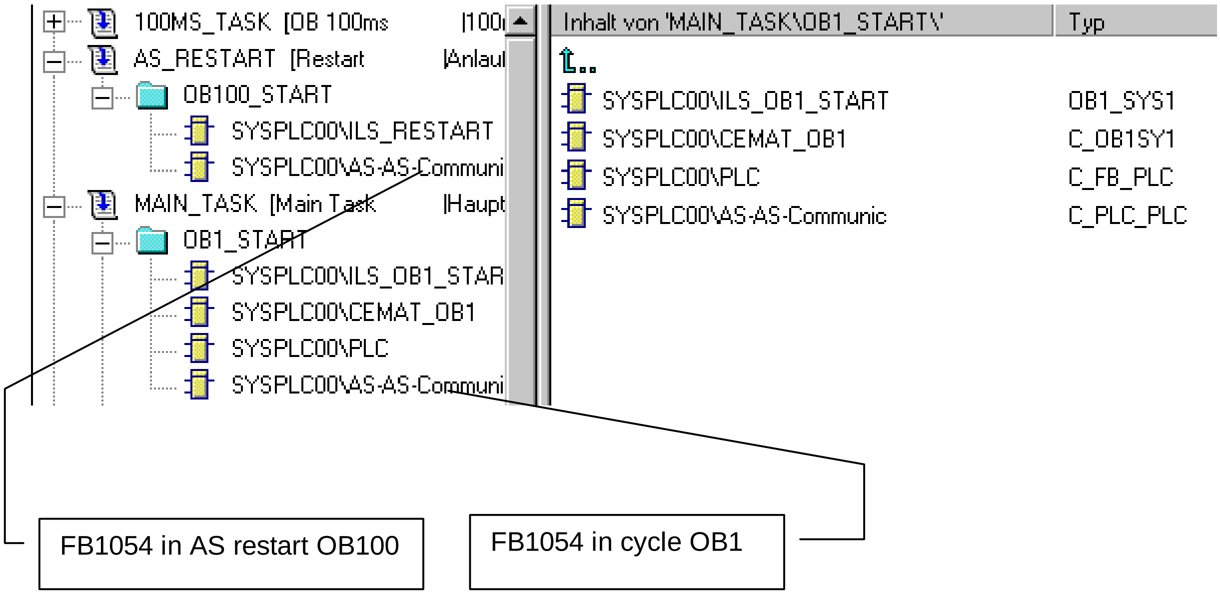
Open the hardware editor and check with „View -> Adress Overview“ the adresses of the CP443.





In the system chart SYSPLCxx the FB 1054 = C_PLC_PLC must be called on AS restart and in the cycle. The parameters for adress bus A, adress bus B and own PLC no. must be supplied. If bus B don't exist, please set the adress to „0". If the parameter PLC_NO = 0, then the PLC PLC coupling is switched off.





System Description

Diagnosis

The variables table AG_AG_KOM_Diagnose provides an aid for the diagnosis of the PLC-PLC coupling.

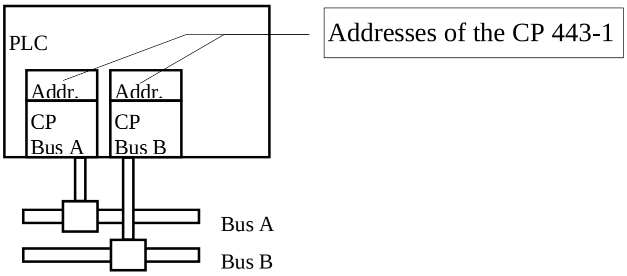
Remedies in Case of Faults

If the coupling does not work properly or not as expected one has to check the following:

- Is the bus interface CP443-1 inserted for bus A and bus B, respectively?
  Check: The bits 0.0 to 0.3 must be set to "0" in DB "00PLC_PLC_ENG"!
- Are the addresses of the CP443-1 correctly configured?
  The S7 hardware configuration and parameterization on FB C_PLC_PLC (FB1054) must be identical. The FB C_PLC_PLC (FB1054) must be called in the cycle (OB1) and during AS restart (OB 100)!
  Check: The bits 0.0 to 0.3 must be set to "0" in DB "00PLC_PLC_ENG"!
- Check the bus cable connection (Is the plug loose? Bus A or B mixed up?)
- Is the connection engineered and loaded to the PLC?
- Check the parameters on C_PLC_SEND (FB1052) and C_PLC_RECEIVE (FB1053).
  The output parameter ENG_ERR must be “0”. If the is a “1”, check the E_ERR_CO.
  (For error numbers refer to page 6)
  Check using the variables table "AG_AG_KOM_Projektierung":
- If there is no parameterization error, then check status of the connection!
  Check using the variables table ‘’AG_AG_KOM_Diagnose:
  (For error numbers refer to page 6) The status information must be 0!
- If no parameterization error exists and the corresponding connection is OK, then one must couple user telegrams.
  Check using the variables table ‘’AG_AG_KOM_Diagnose:
  Receive bit should be set to ‘1’ and/or initiation bit should be blinking.

  Check : Is the initiation bit set?

Structure of Double-Bus Interfacing of an Individual SMR-PLC



Function of the Double-Bus

The PLC-PLC Coupling is designed for double-bus operation. Of course, it also functions when a single-bus is used. Generally, "Bus A" is the master bus; "Bus B" is (provided it exists) only monitored with regards to function and is used exclusively in the event of the failure of "Bus A" for user telegrams. As soon as "Bus A" functions faultless again "Bus B" switches back to "Bus A".
This means there is no load distribution on "Bus A" and "Bus B".
The reversing logic is on the sending side and triggers the user telegram on the respective bus system. On the receiving side both bus systems have equal rights and receive the incoming telegrams. By checking the telegram in the receive buffer one determines whether it is a test telegram or a user telegram. The user telegram is copied to the user receive DB.

Structure of the Test Telegram

The test telegram for checking the connections is sent from the work-DB and consists of the length KF = 2 and the text KC = TEST.

Coupling from CEMAT S5 V X.X and other Control System or Third-Party PLCs

Refer to the SIMATIC S5 PLC-PLC Coupling description Chapter 8

OS Engineering

Content

General

The project settings which have to be carried out during the installation of a new PCS 7 project are no longer described here. These steps you can find in chapter 3 of this manual (03_PCS7-Project).

Chapter OS-Engineering contains all steps which are carried out during the engineering of the WinCC Project.

Basically this comprises the generation of the process pictures and archives.

Even before you start with the process pictures design you have to decide how the Process Objects shall be displayed and how you want to operate. The following aspects have to be considered:

- Size of the block icons
- Which additional information shall be displayed (Tagname, Mark, Quality, etc.)
- Which symbols shall be used?
- Do you have additional functions in your project which have to be enabled through parameterization at the block icons?
- Who can operate (User rights)?
- Positioning of the Faceplates
- Do you want to use any superordinated function and from where do you want to call it?
  - Display of simulated Objects in the AS
  - Buttons to open overview pictures (e. g. for INFO or MAINTENANCE)
  - Button to close all open windows
  - Buttons to open curve groups

We recommend creating a template picture for the project. The template picture C_@PCS7Typicals_CemV8_000.pdl, which has been copied during the Cemat installation, has a lot of examples. You have to decide yourself how the process objects shall be presented in your project.

For the archiving the time period and the type of archiving has to be considered.

Regarding the user archive you have to decide which information you want provide. The corresponding options in the Config files must be set.
The data for the user archive can be created in Excel and then imported (see engineering tools).

Generation of Process Pictures

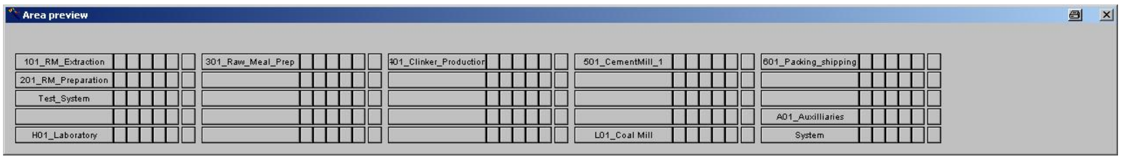
All process pictures for direct access from Overview Range (picture tree) have to be created in the process object view or plant view of the SIMATIC Manager.

Pictures which are not included in the picture tree (because they are called from other pictures) may be created in the component view of the SIMATIC Manager or directly in the GraCS Folder of WinCC.

Picture Navigation

In PCS 7 the Navigation between Process Pictures is defined via the OS Project Editor, based on the Area definition in the plant view. In the Runtime System the picture tree is shown in the Overview range (upper part of the Screen). The main picture of each Area (Overview) can directly be accessed via buttons in the Overview Area. If detail pictures exist, the arrow beside the Area button opens an additional dialog for the navigation to the levels below.

The number of Area buttons depends on the plant size and the number of Areas. Example for a picture tree with 5 by 5 AREA buttons:



Templates

During the Installation procedure, a Template-Picture has been copied into your project, which contains predefined symbols (block icons) for CEMAT Objects:

C_@PCS7Typicals_CemV8_000.pdl

The block icons in this picture are examples and have to be adapted according to the requirements of the project.

Note: Please consider the project code!
The default symbols in the upper part of the template picture were created for normal standard (key = 000). If you use a special standard version for your project (key > 000) you may have to adapt the symbols. Some examples for project versions you can find at the bottom of the picture. Please check the module states in configuration dialog (status, color) according to the object description, “module states” and correct the default symbols if required.

The picture **C_@PCS7Typicals_CemV8_000.pdl** contains the default symbols for automatic generation of block icons from SIMATIC Manager. The symbols of this template picture were enhanced by the new functions of PCS 7 V7.1 SP1. If you start with a new project with CEMAT V7.1 or higher, you should use the templates from this picture.

During the automatic generation of block icons PCS 7 V8.2 is looking for a template picture with the name @PCS7Typicals.pdl. Multiple template pictures are possible (@PCS7Typical**1**.pdl, @PCS7Typical**2**.pdl). The pictures are used in ascending order.

Before you can start with the engineering of your project, you have to create a template picture **@PCS7Typicals.pdl** according to the examples in **C_@PCS7Typicals_CemV8_000.pdl**. Your template picture should only contain the symbols which are really used. (If your template picture contains many “unused” block icons, the generation of the block icons takes unnecessarily long because always the complete picture is scanned!)

Note: A Cemat Template Picture **@Template.pdl** for manual Generation of Block Icons (with Property 'type' without index) is not delivered any more.

In the following description you can find the attributes, which are important for CEMAT.

Automatic generation of block icons

Using the ‘automatic generation of block icons’ the symbols in the process pictures are automatically created and linked to the corresponding variables. The template picture @PCS7Typicals.pdl (or @PCS7Typicals1.pdl, @PCS7Typicals2.pdl,.) must contain all the symbols used for automatic generation in the project.

The different block icons for the same block type distinguish in the Index, which is entered in the block icon properties under General, Type (e. g. @C_DRV_1D/100).
In the CEMAT template picture **C_@PCS7Typicals_CemV8_000.pdl** the index is written below each symbol in order to facilitate the searching.

In order to generate the block icons automatically, in the CFC in the Object properties of the block you have to select the option “Generate block icon” and to enter the corresponding index.

In PCS 7 V8.2 the function for the automatic generation of block icons is automatically carried out with the OS Compile. It can also be carried out in the Plant View or Process Object View of the SIMATIC Manager under Options -> Plant Hierarchy -> Create/Update Block Icons.

After the automatic generation the symbols are located in the upper left corner of the picture and must be moved to the correct position.

After that you can make further adaptations directly at the generated block icon, as e. g. for TooltipText, Formats (FontSize, FontBold), Window position (leftPos, topPos, defaultPos) when opening the faceplate.

In order <u>not</u> to overwrite the modifications in case of a repetition of the generation of block icons, the affected attributes must be listed in the configurations file @PCS7Typicals.cfg. The file @PCS7Typicals.cfg with Cemat specific adaptations was already copied into the OS-Project folder under WScripts.

Manual symbol generation

Beside the automatic generation of block icons you can also link the block icons manually (using Dynamic Wizard – Standard Dynamics – **Link a prototype to a structure or rename an existing link**).
In order to do this you can copy the symbol manually from the template picture @PCS7Typicals_CemV8_000.pdl into your process picture but you have to modify Attribute “type” (under Property System or General) by removing the @ sign.

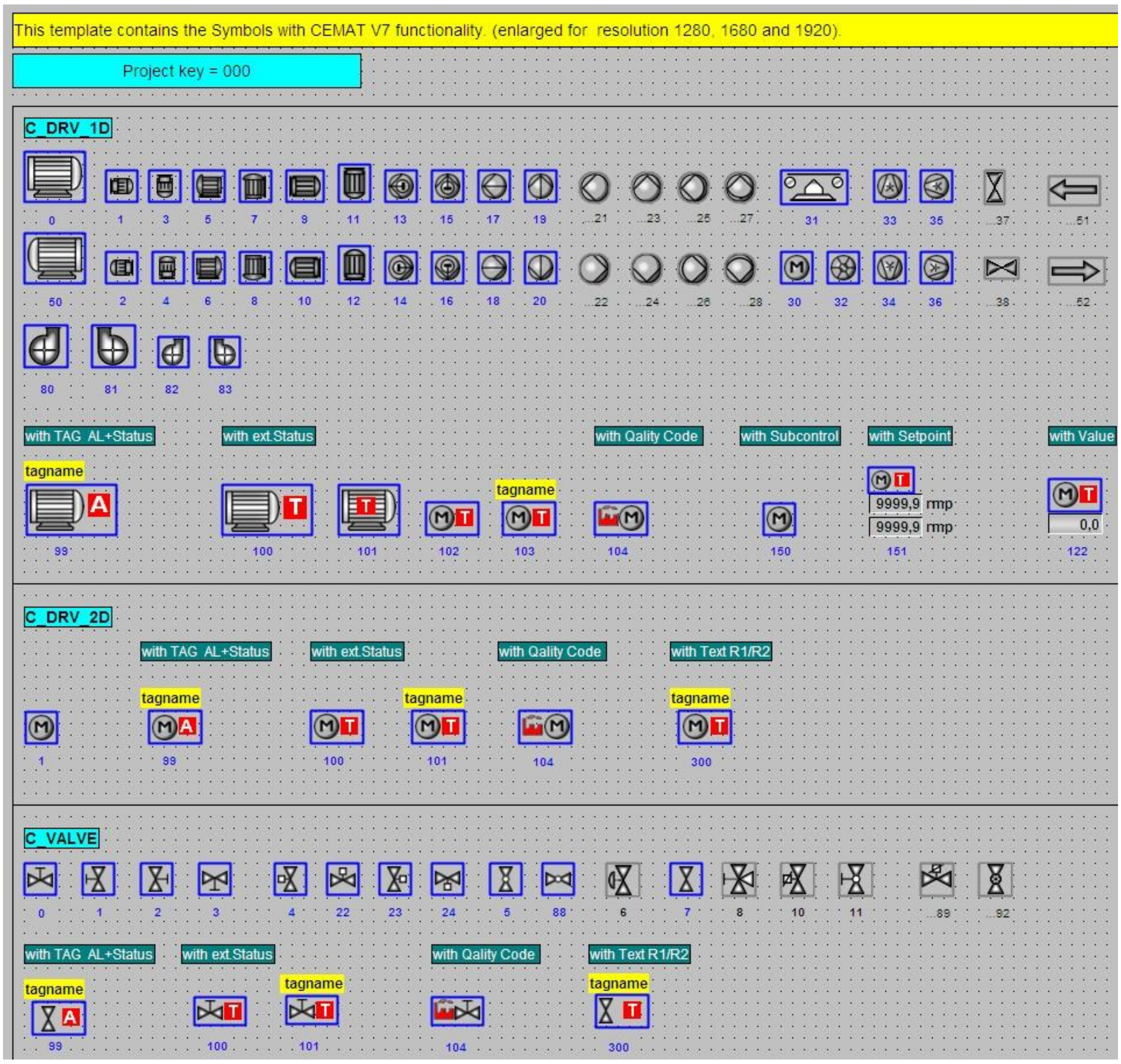
Note: If an @ sign exists, during the next automatic generation the block icon will be "actualized" and eventually deleted.

After that, as mentioned above you can adapt the symbol properties (e.g. ToolTipText).

C_@PCS7Typicals_CemV8_000.pdl

The template picture C_@PCS7Typicals_CemV8_000.pdl contains the block icons for automatic generation of CEMAT Objects.

Extract of the template picture:



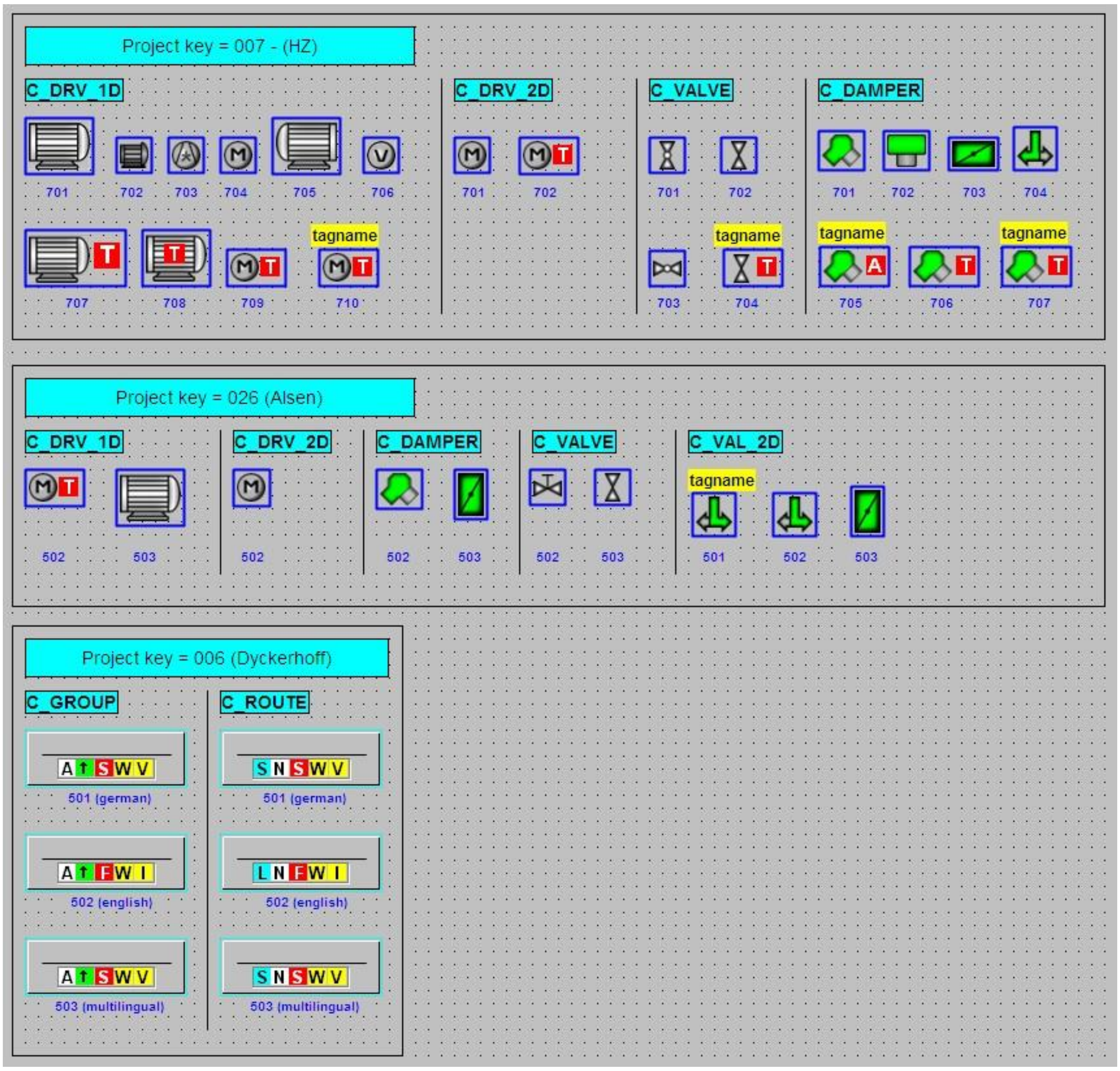
Caution: The Symbols from CEMAT V5 cannot be used in CEMAT V8.2 anymore!
The symbols from CEMAT V6 and Cemat V7.0 can only be used with restrictions. Functions as e. g. "Highlighting of the selected Object" have not been available in Cemat V6.
The block icons of C_GROUP, C_ROUTE, C_SELECT, C_MEASUR, and C_DAMPER cannot simply be used. These block icons have to be replaced or the old block icons must be adapted to the new functions.

Projectstandard related Styles (>=500)

For some of the project versions of CEMAT, modified symbols are required. These symbols you find at the bottom of the template pictures.

If you use a project version (Project key > 000) please check if special symbols are available! In this case, if you use the normal symbols, some functions are not guaranteed.

Example for Project Symbols:



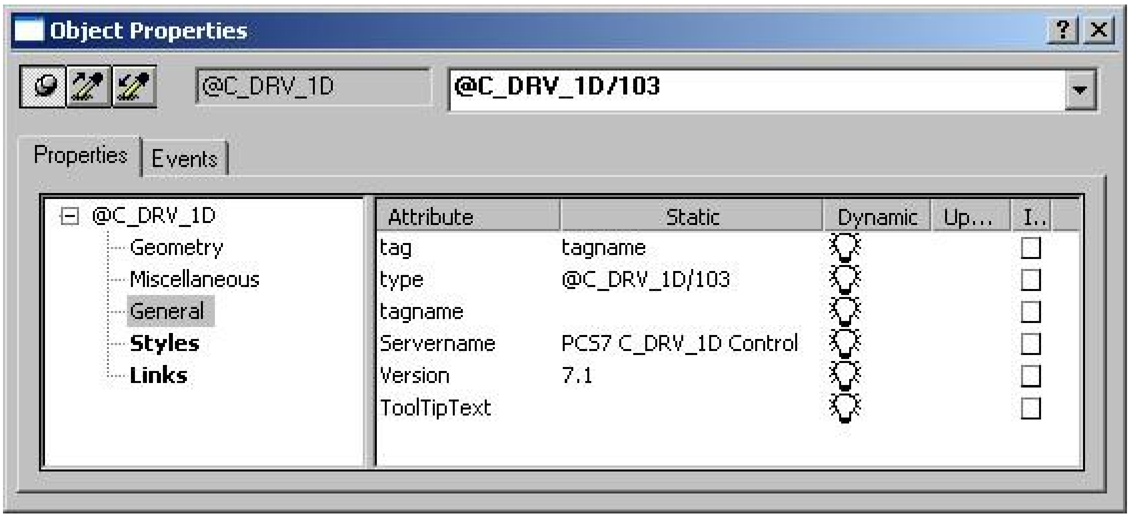
Properties of the Block Icons for CEMAT

All CEMAT objects are available as Customized object. They can be generated automatically using template picture @PCS7Typicals.pdl or linked manually.

The following description explains the most important properties of each block type:

- In Property **Miscellaneous** you find the attributes regarding Authorization and Password Level and regarding the Faceplate coordinates (at which position the faceplate should be displayed in the picture).
  For all objects which can create an alarm you also find here the color definition for highlighting the object in case the picture is opened via loop-in-alarm function.
- In Property **General** the main attributes for the object are defined, such as 'tag', 'type', 'tagname', 'Servername', 'Version' and the 'Tooltip-Text'.
  'tag' and 'tagname' will be replaced by the Object Tag during automatic generation of block Icons.
  The Tooltip-Text is not automatically generated and must be entered manually.

  The attribute 'type' has a special function regarding the automatic generation of block icons. The entry consists of @Blocktype and an index (e. g. @C_DRV_1D/103 for Index = 103).



In the template picture C_@PCS7Typicals_CemV8_000.pdl the index range is linked to a certain functionality, which means it is possible to estimate the function by index number:

from	0	- 98	simple status display
	99		status display with alarm symbol and "test mode" and "bad Quality" display
	100	- 199	Status display with „Testmode“ and „Bad Quality“ and partly with TAG indication
	200	- 299	Special representations
	300	- 399	like index 100++ but with additional properties for variable faceplate button text (C_DRV_2D, C_VALVE, C_DAMPER)
	>500		Project standards

For your own block icons you should use an index number outside of the default range (e. g. > 1000 or starting with a letter).

- In Property **Styles** the appearance of the block icon or the faceplate can be modified. The possible settings you will find in the description of the object types below:

  Special function StyleTag:
  Via attribute StyleTag you can switch faceplate buttons to visible or invisible. This option is used in the groups, routes and selections and in project version 026 also for the drive functions.
  In the faceplate call via block icon a number is transmitted to the faceplate.
  In the faceplate functions the transmitted StyleTag is evaluated. If the StyleTag > 0 the display is adapted according to the definition. At the same time the internal variable Tagname_ButtonStyle is created and stored on the computer.
  In case of an indirect faceplate call (e. g. via instance list or a linked object), in the faceplate the attribute StyleTag remains in the default setting (=0) and the number will be read from variable Tagname_ButtonStyle.

  Caution: If the StyleTag = 0 the internal variable is not stored. If you change the StyleTag of your block icon from > 0 to 0, you have to delete the variable manually. For the deletion the runtime can remain activated.
- In Property **Links** the symbol is connected to numerous variables of each object. The connection is carried out automatically during automatic generation of block icons.
- In Property **User Text** (for groups, routes and selections) a text can be defined which will be displayed in the symbol instead of the Tagname.

Name definitions

Picture Names

For the picture names there are no fixed rules. The maximum length is 24 characters.

Object names

The object names are created in the Engineering of the CFC. In the OS only selection of existing objects is possible.

C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE and C_SILOP

In the process pictures the blocks C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE and C_SILOP are represented by a drive symbol, which contains the status display (drive stopped, running, faulty and eventually the operation mode).

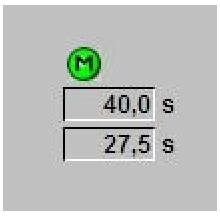
In this case, all further objects, belonging to the customized object are hidden behind the status display or invisible by default.

However there are block icons with additional indications and block icons with attributes which are evaluated in the faceplate itself.

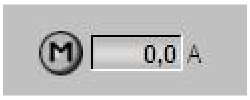
Example 1: with Tagname and display of Quality/Test mode/Simulation

Example 2: with Tagname and Signal status QC

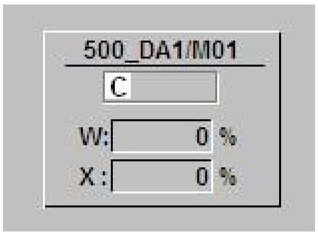
Example 3: Motor symbol with display of set point and actual value



Example 4: Motor symbol with related measuring value



Example 5: Damper positioner



Function overview for the different block icons:

C_DRV_1D:

Index 0-98	only status display
Index 99	Status display + Group display
Index 100-102	Status display + Quality/Test mode/Simulation
Index 103	Status display + Quality/Test mode/Simulation + Tagname
Index 104	Status display + Signal status QC
Index 122-123	Status display + measuring value
Index 150	Status display + Subcontrol function
Index 151	Status display + Quality/Test mode/Simulation + Set point/actual value

C_DRV_2D:

Index 1	only status display
Index 99	Status display + Group display
Index 100	Status display + Quality/Test mode/Simulation
Index 101	Status display + Quality/Test mode/Simulation + Tagname
Index 104	Status display + Signal status QC
Index 300	Status display + Button text

C_DAMPER:

Index 0-12	only status display
Index 99	Status display + Group display
Index 100	Status display + Quality/Test mode/Simulation
Index 101-102	Status display + Quality/Test mode/Simulation + Tagname
Index 104	Status display + Signal status QC
Index 200	Damper positioner
Index 300	Status display + Button text

C_VALVE:

Index 0-92	only status display
Index 99	Status display + Group display
Index 100	Status display + Quality/Test mode/Simulation
Index 101	Status display + Quality/Test mode/Simulation + Tagname
Index 104	Status display + Signal status QC
Index 300	Status display + Button text

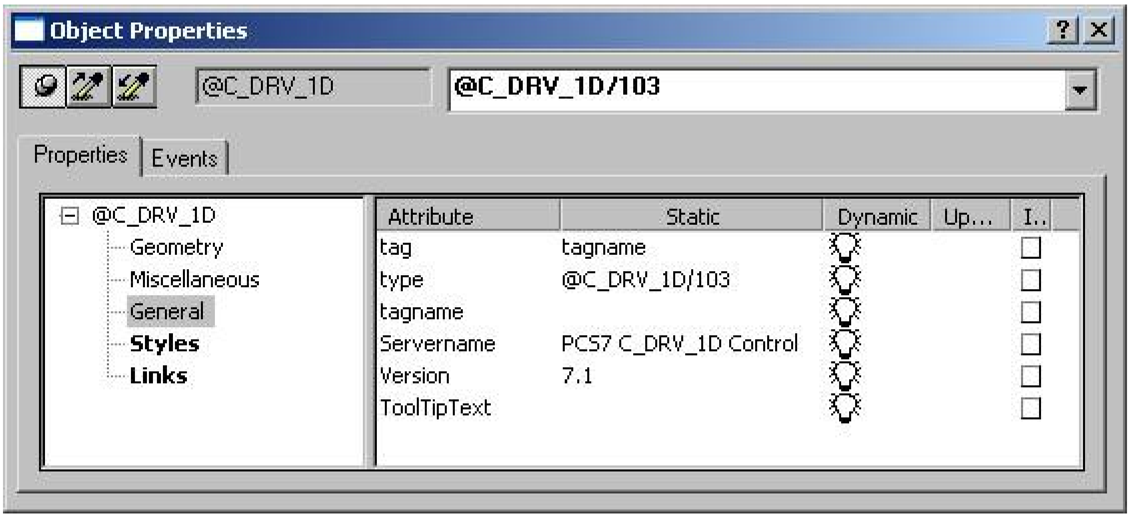
C_SILOP:

Index 1	only status display + Tagname
Index 2-3	only measure for silo level
Index 4	Status display + Quality/Test mode/Simulation

Caution: For project versions 007 Heidelberg Cement and 026 Alsen the block icons need special adaptations.
Examples you can find in the template picture C_@PCS7Typicals_CemV8_000.pdl
(Index 7xx for Project version 007 and Index 5xx for Project version 026).

Property General

Property **General** of object type C_DRV_1D:



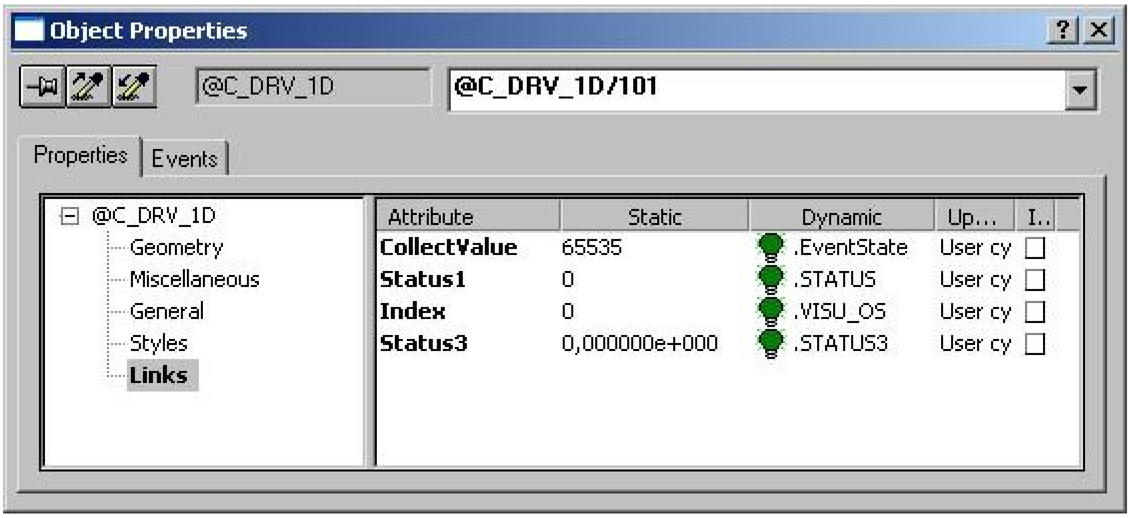
Caution: The screen shot of the object properties shows the property General in template picture C_@PCS7Typicals_CemV8_000.pdl.
Attribute 'type' is used for the automatic generation of block icons via OS compile and must therefore consist of an @-sign, the block type and an index.
For the manual link you should replace attribute 'type' by 'C_DRV_1D' (removing @-sign and index), in order not to overwrite the data in the next OS compile.

The Attribute **ToolTipText** is not entered automatically. You have to copy the text (normally the Tagname) manually.

Property Links

In Property **Links** all Attributes must be connected as follows to the variables of the corresponding object. (They will be connected automatically in case of automatic generation of Block Icons):

Property **Links** of object type C_DRV_1D:



Link Attributes of the drives:

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
Index	to	.VISU_OS	for the status display
Status1	to	.STATUS	for Bad Quality B, Test mode T, Simulation S
Status3	to	.STATUS3	Marking the Object (Function related. Objects)
QC	to	.RunSig1#ST	Display of the Signal status QC ,

Additional Link Attributes for indication of setpoint and actual value (only for C_DRV_1D, Index 151):

SP_IN	to	.SP_IN	for Set point display
SP_IN_UNIT	to	.SP_IN#shortcut	for the display of the set point unit
PV_IN	to	.PV_IN#value	for the display of the actual value
PV_IN_UNIT	to	.PV_IN#shortcut	for the display of the actual value unit

Additional Link Attributes for Subcontrol functions (only for C_DRV_1D, Index 150):

SubText	is the button text in the faceplate of the motor
SubTag	is the tagname of the subcontrol function
SubType	is the S7 structure name (block type)

Additional Link Attributes for measure display (only for C_DRV_1D, Index 122/123):

OutputValue	to	.PV#Value	for the display of the measure
BlockName	to	.PV#Jump	for the tagname of the measure
LeftHandText			for the tool-tip (tagname of the measure + comment)
StatusAlarm	to	.PV_Stat#Status	for the limit indications
UnitText	to	.PV_Stat#UNIT	for the unit of the measure (only Index 123)

Additional Link Attributes for damper positioner (only for C_DAMPER, Index 200):

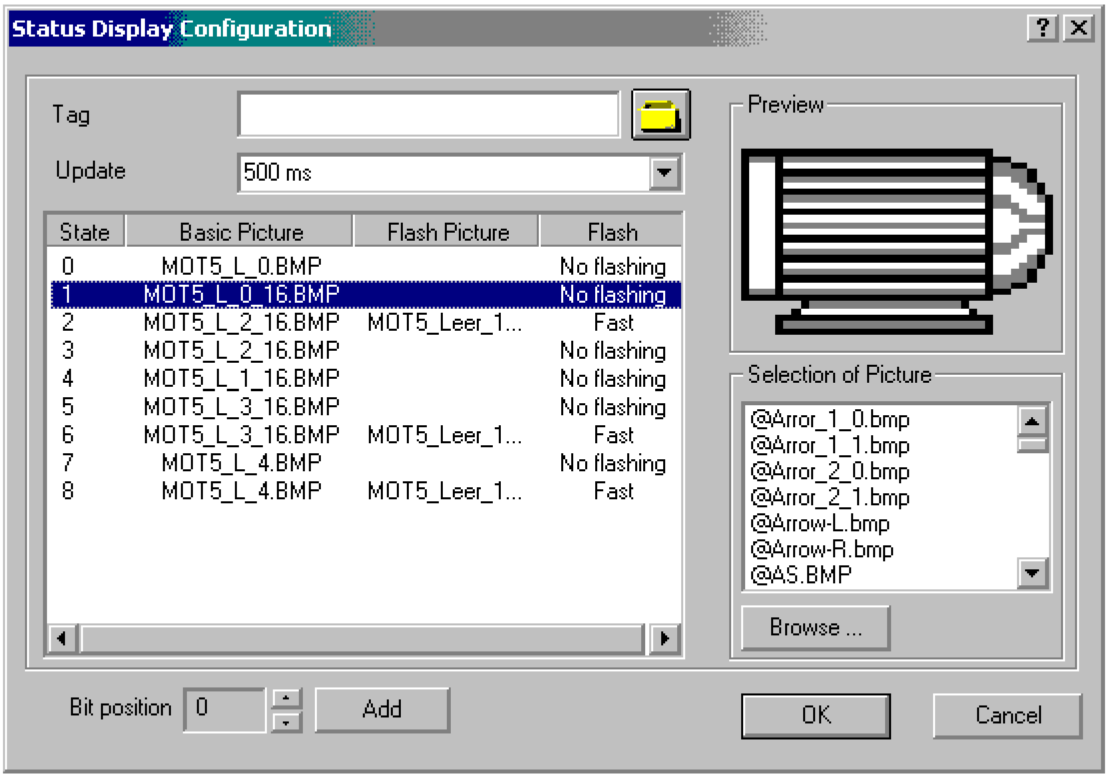
D_MODE	to	.STATUS	Damper mode
D_ExternalSetpoint	to	.STATUS2	External set point
D_Fault	to	.STATUS	Damper fault
D_FaultAnalog	to	.STATUS	Analog value fault
XValue	to	.X_POS_OS#Value	Actual value
XUnit		%	No connection because actual value is always in %
WValue	to	.W_OS	Set point
WUnit	to	.UNIT	Set point unit
WLimMax	to	.KWOG	for Set point upper limit
WLimMin	to	.KWUG	for Set point lower limit

Additional Link Attributes for Silopilot

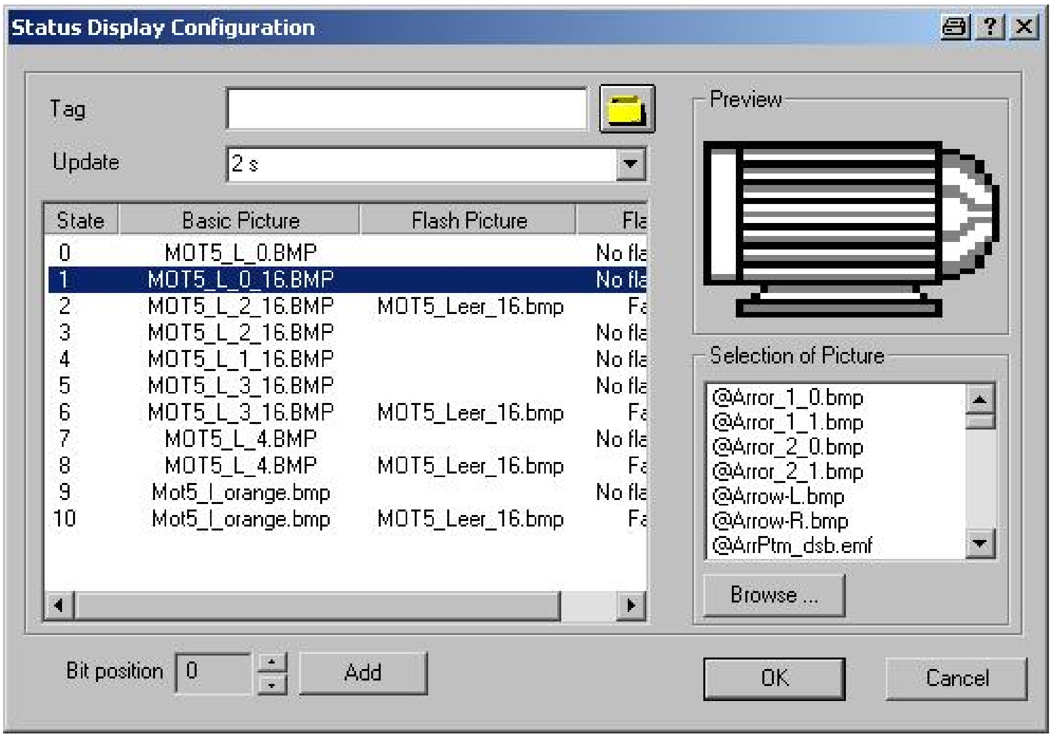
PV	to	.PV	for the display of the silo level
Unit	to	.UNIT	for the display of the unit

Status Display function is used to define the representation of the block icon the different states (running, off, faulty...). The possible states of each object are documented in the object description of the corresponding object type (Variable VISU_OS).

The following picture shows the states object type C_DRV_1D:



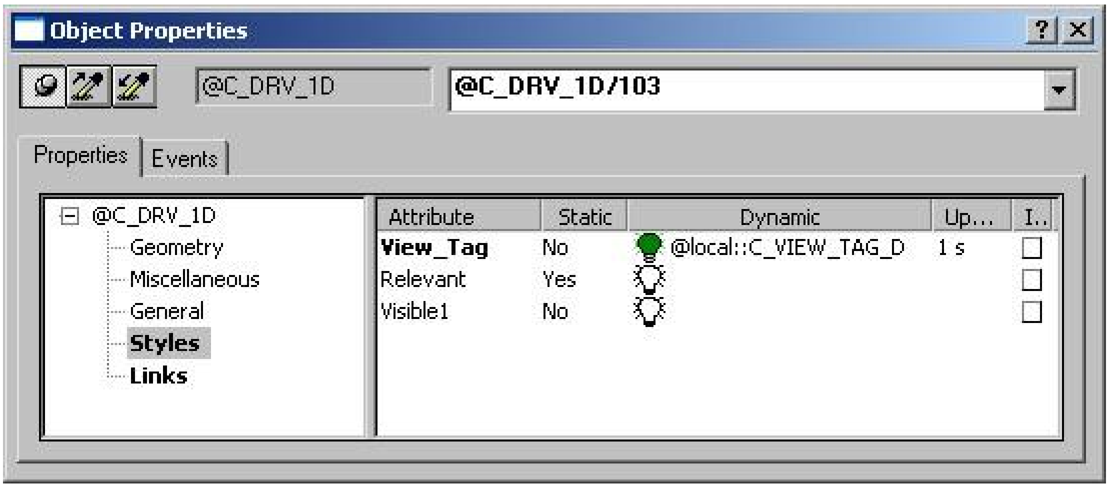
Achtung: The project version 007 Heidelberg Cement has additional status displays for non-interlocked single start mode (e. g. status 9 and 10 for C_DRV_1D). Examples you can find in the template picture C_@PCS7Typicals_CemV8_000.pdl.



Property Styles

In Property **Styles** the appearance of the symbol or the faceplate can be modified.

Property **Styles** of object type C_DRV_1D:



Style Attributes for the drives:

View_Tag	to	C_VIEW_TAG_D	Tagname will be switched to 'visible' via button "D"
Relevant		Yes	for Group display (no connection)
Visible1		No	(no connection)

Additional Style Attributes for the button text in the Facplate (only for C_DRV_2D, C_DAMPER and C_VALVE):

BText1	Button text for single start direction 1 (left button)
BText2	Button text for single start direction 2 (right button)

Zusätzliche Link-Attribute für Klappenpositionierer (nur bei C_DAMPER, Index 200):

ReturnPath	.U:CO_DKGREEN	internal (default setting must remain!)
StandardTrend	2	internal (default setting must remain!)
StyleTag	1,0000000	internal (default setting must remain!)
WBackColor	grey	internal (default setting must remain!)

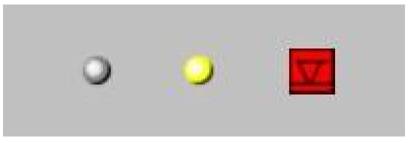
Caution: Project version 026 Alsen has an additional Style Attribute for C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE and C_VAL_2D, in order to switch the buttons for the operation mode change to local mode to invisible. Examples you can find in the template picture C_@PCS7Typicals_CemV8_000.pdl

StyleTag	0 = Buttons for local mode on/off are visible 1 = Buttons invisible

In order to make the StyleTag available also for indirect faceplate calls, if the StyleTag > 0, an internal variable Tagname_ButtonStyle is created on the computer, in order to save the StyleTag. (This happens automatically during the faceplate call via block icon).
If StyleTag = 0 the display follows the default settings.

C_ANNUNC und C_ANNUN8

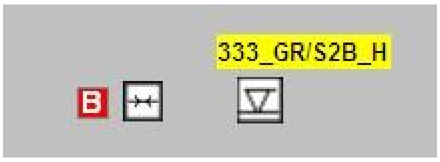
In the process pictures the blocks C_ANNUNC and C_ANNUN8 are represented by a status display only (Signal ok, Warning, Fault).



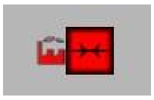
In this case, all further objects, belonging to the customized object are hidden behind the status display or invisible by default.

However there are block icons with additional indications, as e. g. tagname or signal status:

Example 1: with tagname and display of Quality/Test mode/Simulation



Example 2: with Signal status QC



Function overview for the different block icons:

C_ANNUNC:

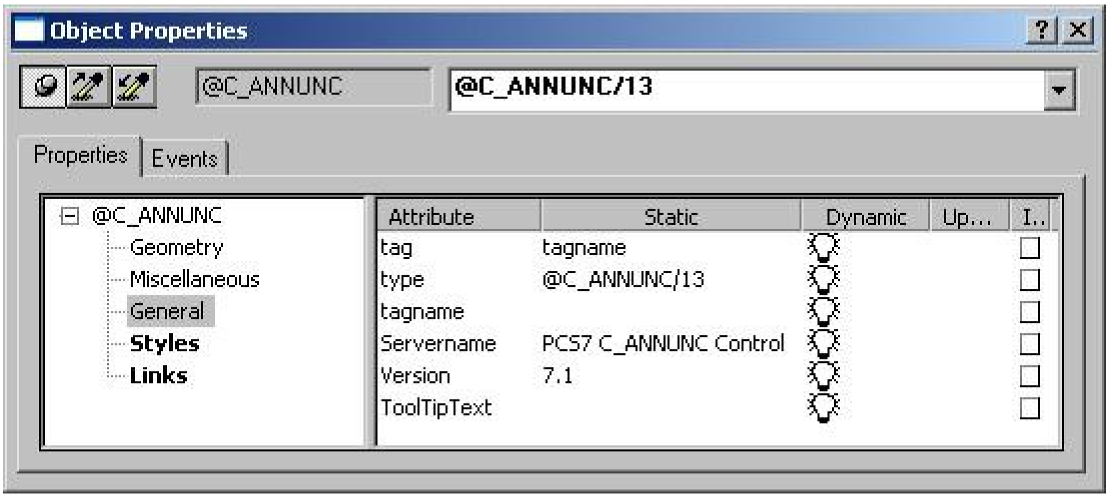
Index 0-13	Status display + Tagname
Index 14	only status display
Index 99	Status display + Group display + Tagname + QC
Index 104	Status display + Signal status QC

C_ANNUN8:

Index 1	Status display + Tagname
Index 2	Status display + Quality/Test mode/Simulation
Index 99	Status display + Group display + Tagname

Property General

Property **General** of object type C_ANNUNC:



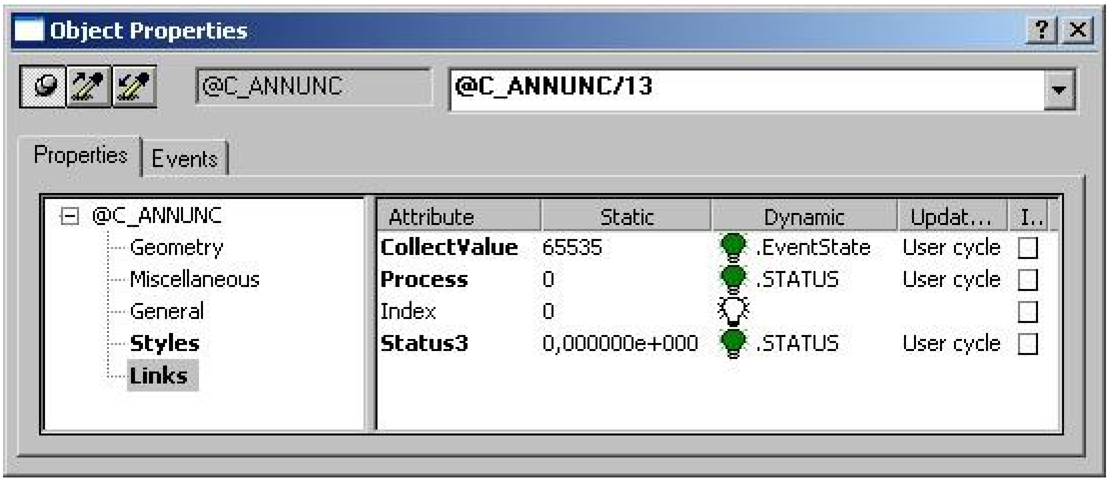
Caution: The screen shot of the object properties shows the property General in template picture C_@PCS7Typicals_CemV8_000.pdl.
Attribute 'type' is used for the automatic generation of block icons via OS compile and must therefore consist of an @-sign, the block type and an index.
For the manual link you should replace attribute 'type' e.g. by 'C_ANNUNC' (removing @-sign and index), in order not to overwrite the data in the next OS compile.

The Attribute **ToolTipText** is not entered automatically. You have to copy the text (normally the Tagname) manually.

Property Links

In Property **Links** all Attributes must be connected as follows to the variables of the corresponding object. (They will be connected automatically in case of automatic generation of Block Icons):

Property **Links** of object type C_ANNUNC:



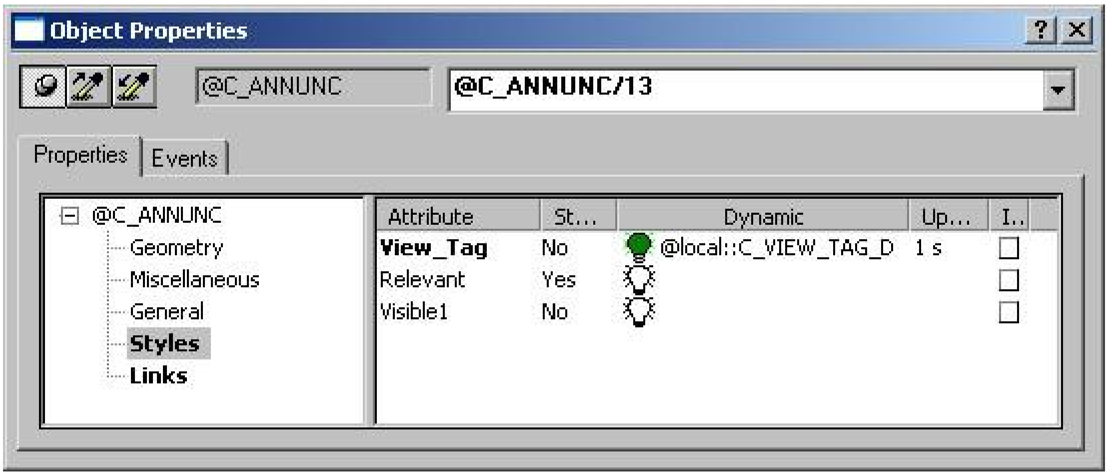
Link Attributes of the annunciation blocks:

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
Process	to	.STATUS	for Bad Quality B, Test mode T, Simulation S
Index		0	not used at the moment
Status3	to	.STATUS	Marking the Object (Function related. Objects)
QC	to	.RunSig1#ST	Display of the Signal status QC ,

Property Styles

In Property **Styles** the appearance of the symbol or the faceplate can be modified.

Property **Styles** of object type C_ANNUNC:



Style Attributes for the annunciation blocks:

View_Tag	to	C_VIEW_TAG_D	Tagname will be switched to 'visible' via button "D"
Relevant		Yes	for Group display (no connection)
Visible1		No	(no connection)

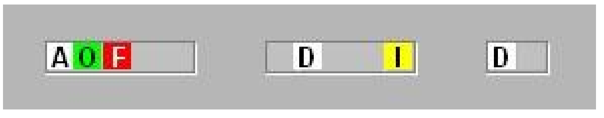
C_GROUP, C_ROUTE and C_SELECT

In the process picture the blocks C_GROUP, C_ROUTE, and C_SELECT are represented by a symbol in order to display the actual status (operation status, operation mode, summarizing fault, summarizing warning, interlocks and selections).

In this case, all further objects, belonging to the customized object are hidden behind the status display or invisible by default.

In the template picture you find various block icons in different sizes and attributes. Also the language has to be considered. Beside German and English block icons there are multilingual block icons which automatically adapt to the operating language.

Example 1: Single line representation without Tagname:



Function overview for the different block icons:

C_GROUP:

Index 1	big Symbol English
Index 2	small Symbol English
Index 101	big Symbol German
Index 102	small Symbol German
Index 201	big Symbol multilingual

C_ROUTE:

Index 1	big Symbol English
Index 2	small Symbol English
Index 101	big Symbol German
Index 102	small Symbol German
Index 201	big Symbol multilingual

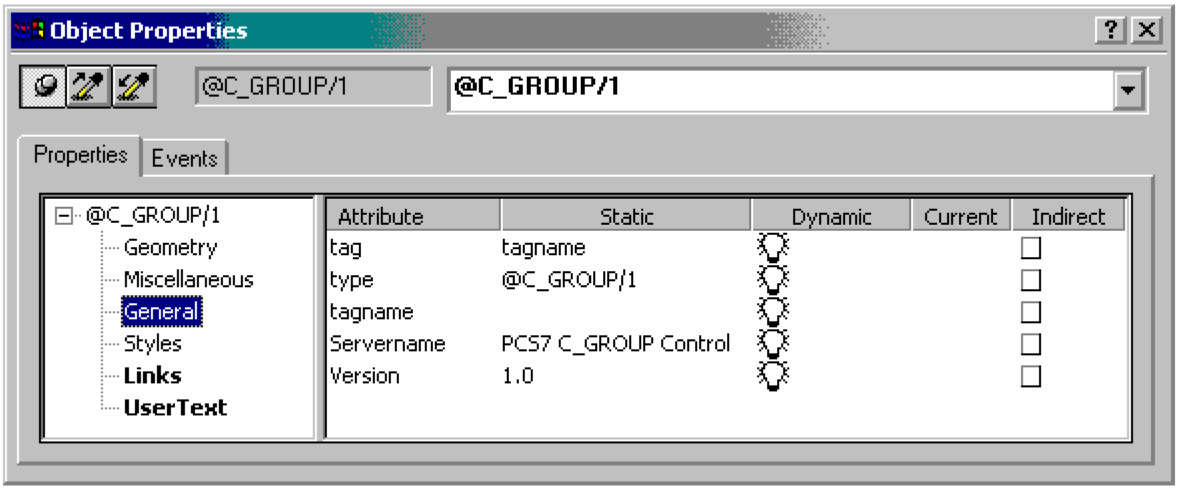
C_SELECT:

Index 1	big Symbol English
Index 2	small Symbol English with Tagname and mouse click left/right (left mouse-click for select/deselect; right mouse click opens faceplate)
Index 4	small Symbol English
Index 5	small Symbol with additional text (left mouse-click for select/deselect; right mouse click opens faceplate)
Index 6	small Symbol with additional text and interlock display (left mouse-click for select/deselect; right mouse click opens faceplate)
Index 101	big Symbol German
Index 102	small Symbol German with Tagname and mouse click left/right (left mouse-click for select/deselect; right mouse click opens faceplate)
Index 104	small Symbol German
Index 201	big Symbol multilingual

Caution: For Project version 006 Dyckerhoff especially adapted block icons exist for group and route (with modifications in the status display for manual interlocked and manual non-interlocked).
Examples you can find in the template picture C_@PCS7Typicals_CemV8_000.pdl (Index 5xx).

Property General

The following picture shows the **General** properties of object type C_GROUP:



Caution: The screen shot of the object properties shows the property General in template picture C_@PCS7Typicals_CemV8_000.pdl.
Attribute 'type' is used for the automatic generation of block icons via OS compile and must therefore consist of an @-sign, the block type and an index.
For the manual link you should replace attribute 'type' e.g. by 'C_GROUP' (removing @-sign and index), in order not to overwrite the data in the next OS compile.

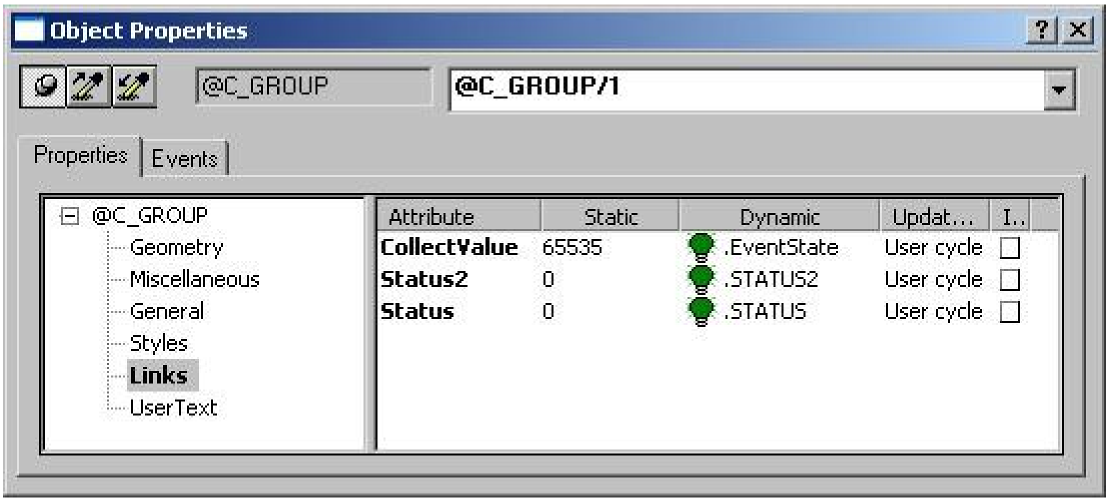
The Attribute **ToolTipText** is not entered automatically. You have to copy the text (normally the Tagname) manually.

In some block icons under Property 'General', you find the additional Attribute **TagOut**, which is used for a text output in the block icon.

Property Links

For each group in Property **Links** all Attributes must be connected as follows to the variables of the corresponding object. (They will be connected automatically in case of automatic generation of Block Icons):

Property **Links** of object type C_GROUP:



Link Attributes of the group:

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself).
Status2	to	.STATUS2	for the display of operation status and interlocking
Status	to	.STATUS	for the display of operation mode, fault and warning

Link Attributes or the route:

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
Status	to	.STATUS	for the display of locked, selection, fault, warning and interlocking

Link Attributes of the selection block:

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
Status	to	.STATUS	for the display of selection and interlocking

Additional Link Attributes for C_SELECT with Index 2 and 102:

S_Status	to	.STATUS	for the display of the selection
S_Interlock	to	.STATUS	for the display of the interlocking

Additional Link Attributes for C_SELECT with Text output (Index 4):

OutputValue	to	.STATUS	for the display of the selection
Operation1	to	.NON_INTL	in case of an interlocking the button text is grey.

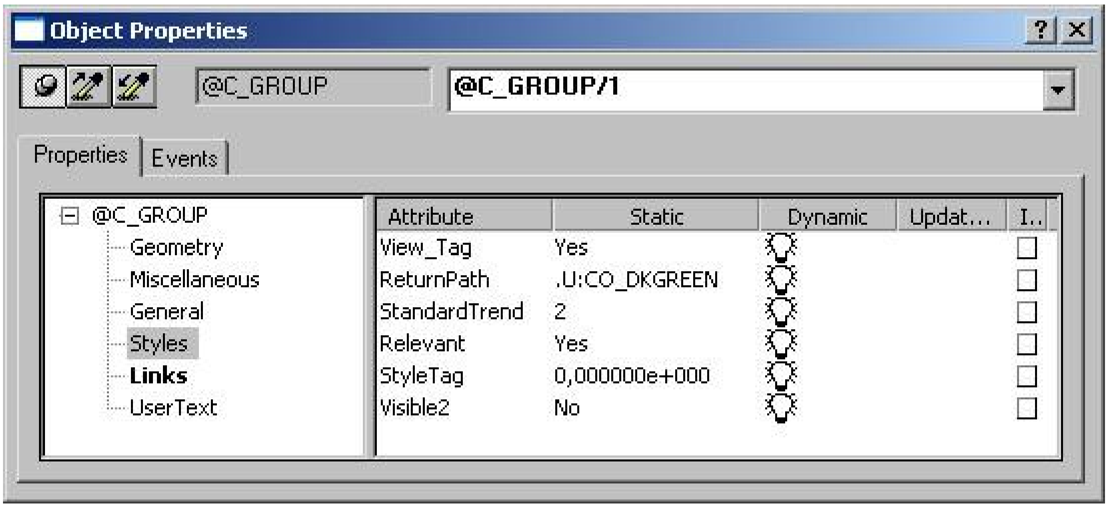
Additional Link Attributes for C_SELECT with Text output (Index 5):

Operation1	to	.NON_INTL	in case of an interlocking the button text is grey. **(**The display of the selection is carried out via Styles!)
ON		Yes	must remain

Property Style

In Property **Styles** the appearance of the symbol or the faceplate can be modified.

Property **Styles** of object type C_GROUP:



Styles attributes of the group:

View_Tag	to	C_VIEW_TAG_D	Tagname will be switched to 'visible' via button "D" this only makes sense in case of small block icons because the big block icons already contain the Tagname.
ReturnPath		.U:CO_DKGREEN	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
Relevant		Yes	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
StyleTag		0,0000000	modifies Faceplate Buttons:

0 – Start_Stop_Automatic_Local_Single (default)
1 – Buttons_Invisible
2 – Start_Stop_Automatic_Local_Single_QuickStop
3 – Start_Stop
4 – Start_Stop_Automatic_Local
5 – Start_Stop_Automatic_Single
6 – Start_Stop_Automatic_Local_QuickStop
7 – Start_Stop_Automatic_Single_QuickStop
8 – Start_Stop_Automatic_QuickStop
(only for Project Standard 006 Dyckerhoff)
For the display of the interrupt button you have to add +100 to the StyleTag number

In order to make the StyleTag available also for indirect faceplate calls, if the StyleTag > 0, an internal variable Tagname_ButtonStyle is created on the computer, in order to save the StyleTag. (This happens automatically during the faceplate call via block icon).
If StyleTag = 0 the display follows the default settings.

Caution: The project version 026 Alsen has different Style Attributes for C_GROUP, because there is no group-wise change of operation modes. There are just two buttons in order to enable all drives of the group for local mode or to switch the complete group back to automatic mode (for maintenance).

Style_Tag	0 – Start_Stop_Local on/Local off (default) 1 – Buttons_Invisible 3 – Start_Stop_Local on/Local off 8 – Start_Stop_ Local on/Local off_QuickStop 9 – Start_Stop_QuickStop 10 – Start_Stop For the display of the interrupt button you have to add +100 to the StyleTag number

In order to make the StyleTag available also for indirect faceplate calls, if the StyleTag > 0, an internal variable Tagname_ButtonStyle is created on the computer, in order to save the StyleTag. (This happens automatically during the faceplate call via block icon).
If StyleTag = 0 the display follows the default settings.

Style Attribute or the route:

View_Tag	to	C_VIEW_TAG_D	Tagname will be switched to 'visible' via button "D" this only makes sense in case of small block icons because the big block icons already contain the Tagname.
ReturnPath		.U:CO_DKGREEN	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
Relevant		Yes	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
StyleTag		0,0000000	modifies Faceplate Buttons: 0 – Selection_Deselection (default) 1 – Buttons invisible

Style Attributes of the selection block:

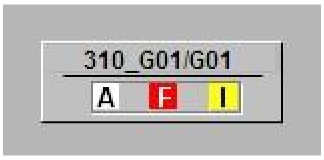
View_Tag	to	C_VIEW_TAG_D	Tagname will be switched to 'visible' via button "D" this only makes sense in case of small block icons because the big block icons already contain the Tagname.
ReturnPath		.U:CO_DKGREEN	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
Relevant		Yes	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
StyleTag		0,0000000	modifies Faceplate Buttons: 0 – Selection_Deselection (default) 1 – Buttons invisible
Visible2		No	internal

Additional Style Attributes for C_SELECT, Index 6:

ForeColor	must be connected manually to Tagname.STATUS
BackColor	must be connected manually to Tagname.STATUS

Property User Text

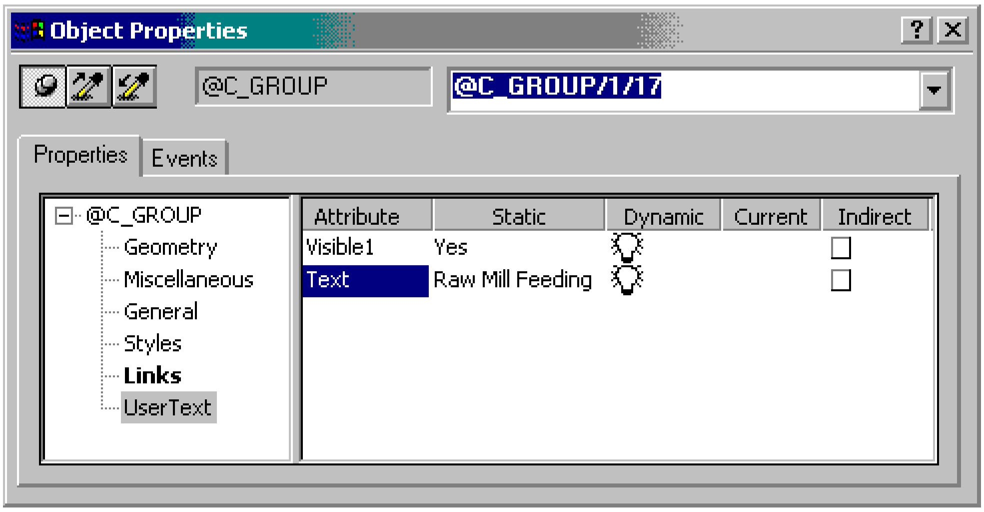
Normally the symbol shows the Tagname of the group, route or selection.



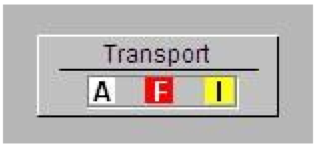
If a different description should be displayed in the object symbol, the property **User Text** can be used.

If **Visible1** is set to '**No**' the object tagname is displayed in Runtime mode.

In order to display a user defined text instead of the tagname the Attribute **Visible1** of Property **UserText** must be changed to '**Yes**' and the text must be entered into Attribute **Text**.

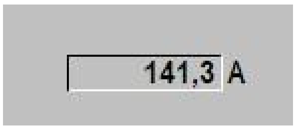


Display of a group with UserText Entry:



C_MEASUR

The block icon for C_MEASUR shows the actual measuring value, the dimension and eventual limit violations (color change).



In this case, all further objects, belonging to the customized object are hidden behind the status display or invisible by default.

However there are block icons with additional indications, as e. g. tagname or signal status and block icons with attributes which are evaluated in the faceplate itself.

Example 1: with tagname and display of Quality/Test mode/Simulation

Example 2: with signal status QC

Example 3: Line representation with Tagname and comment.

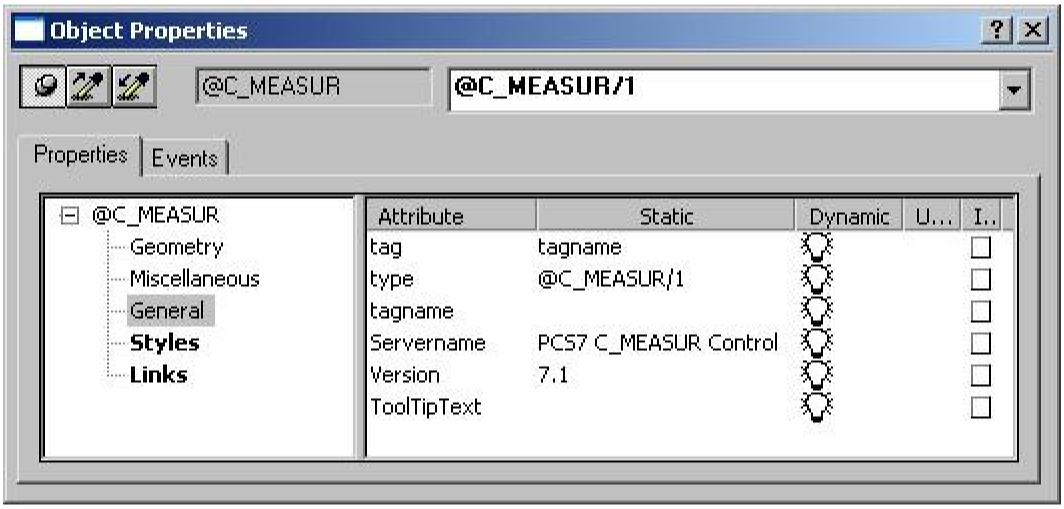
Function overview for the different block icons:

C_MEASUR:

Index 1	Measured value with Tagname.
Index 2	Measured value
Index 3	Line representation with Tagname and comment.
Index 4	Measured value with red frame in case of Simulation
Index 99	Measured value with big Group display, Tagname and status display for Bad Quality/Simulation
Index 100	Measured value with Tagname and status display for Bad Quality/Simulation
Index 104	Measured value with Signal status QC Colors for bar and Value can be adapted
Index 205	Measured value APL-like with big Group display
Index 206	Measured value APL-like with small Group display

Property General

Property **General** of object type C_MEASUR:



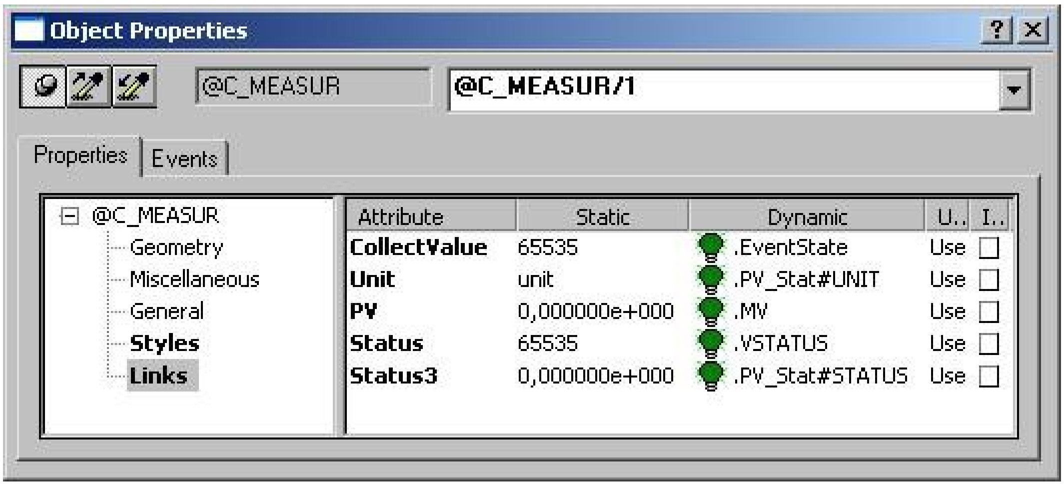
Caution: The screen shot of the object properties shows the property General in template picture C_@PCS7Typicals_CemV8_000.pdl.
Attribute 'type' is used for the automatic generation of block icons via OS compile and must therefore consist of an @-sign, the block type and an index.
For the manual link you should replace attribute 'type' by 'C_MEASUR' (removing @-sign and index), in order not to overwrite the data in the next OS compile.

The Attribute **ToolTipText** is not entered automatically. You have to copy the text (normally the Tagname) manually.

Property Links

In Property **Links** all Attributes must be connected as follows to the variables of the corresponding object. (They will be connected automatically in case of automatic generation of Block Icons):

Property **Links** of object type C_MEASUR:



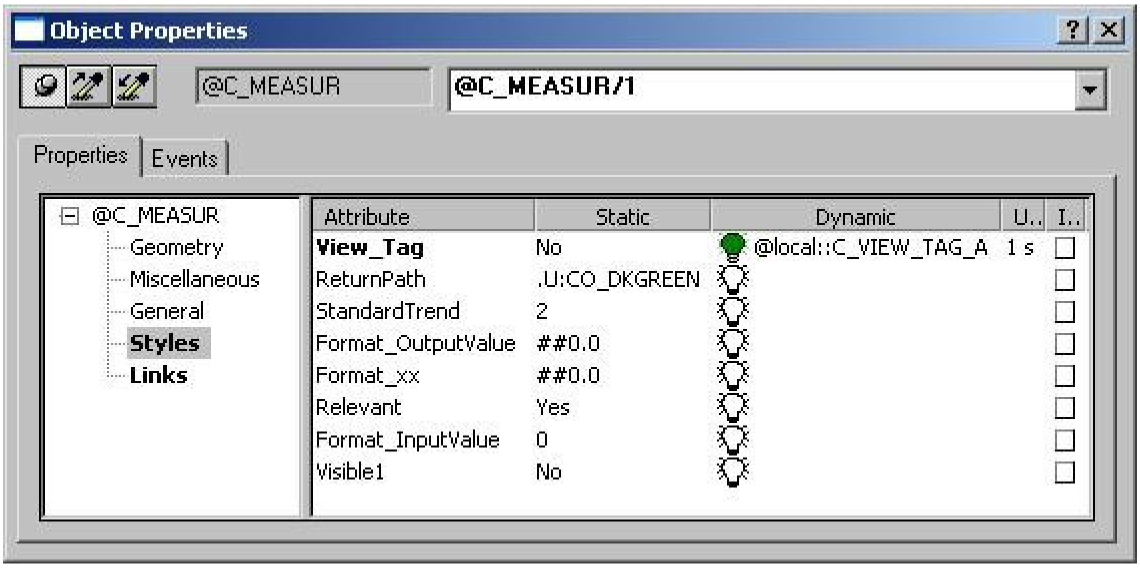
Link Attribute of the measured value block:

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself).
Unit	to	.PV_Stat#UNIT	for the display of the unit
PV	to	.MV	for the display of the process value
Status	to	.VSTATUS	for the colors
Status1	to	.PV_Stat#STATUS	for Bad Quality B, Test mode T, Simulation S
Status3	to	.PV_Stat#STATUS	Marking the Object (Function related. Objects)
QC	to	.PV_Out#ST	Display of the Signal status QC  , 

Property Styles

In Property **Styles** the appearance of the symbol or the faceplate can be modified.

Property **Styles** of object type C_MEASUR:



Style Attribute of the measured value block:

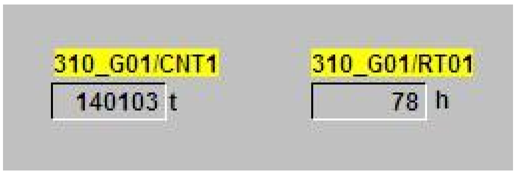
View_Tag	to	C_VIEW_TAG_A	Tagname will be switched to 'visible' via button "A"
ReturnPath		.U:CO_DKGREEN	Archive name (optional) → See 'Adding an archive tag to the process value archive'
StandardTrend		2	internal (default setting must remain!)
Format_OutputValue		##0.0	not relevant
Format_xx		##0.0	not relevant
Relevant		Yes	internal (default setting must remain!)
Format_InputValue		0	Output format for the measured value 0.0 one digit behind the comma 0.00 two digits behind the comma 0.0## one fix digit behind the comma and two optional digits (only if not 0). These settings are also used in the faceplate of the object for actual value and limits.
Visible1		No	internal

Additional Style Attributes for C_MEASUR, Index 104 (for color adaptations):

TrendColor1	dark green	for bar color
FillColor1	light green	for bar color
ValueColor1	dark green	for value color
ValueFillColor1	light green	for value color

C_COUNT and C_RUNNT

In the process pictures the blocks C_COUNT and C_RUNNT are represented by a block icon which shows the counter values.



In this case, all further objects, belonging to the customized object are hidden behind the status display or invisible by default.

The template picture contains only one block icon for each block type.

Function overview for the different block icons:

C_COUNT:

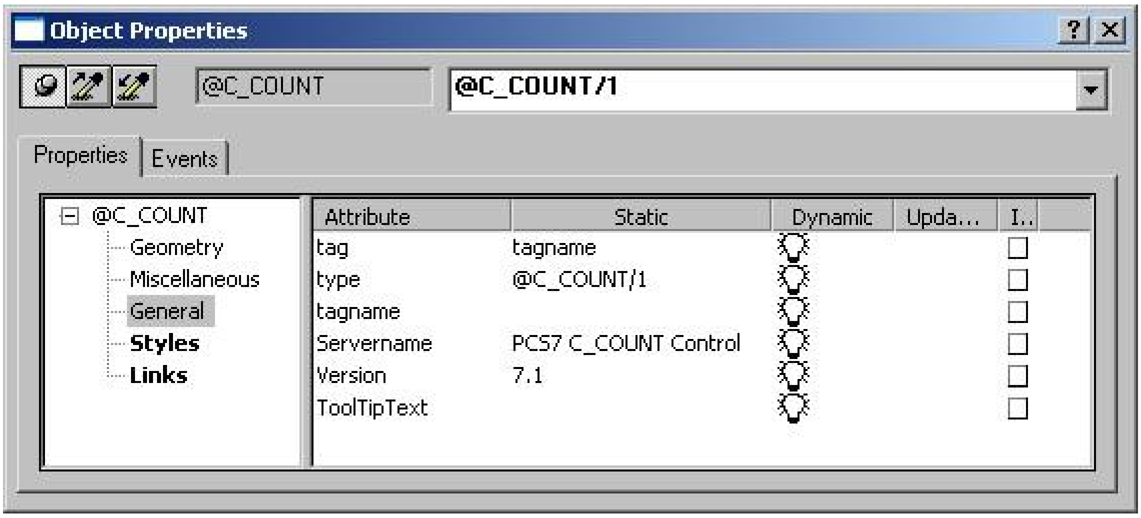
Index 1	Display of the counter value and the tagname

C_RUNNT:

Index 1	Display of the operating hours and the tagname.

Property General

Property **General** of object type C_COUNT:



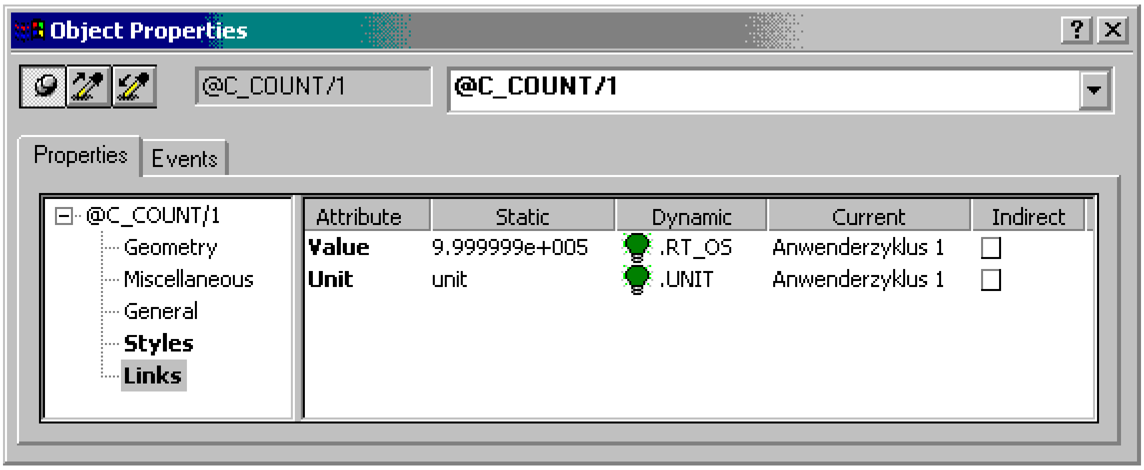
Caution: The screen shot of the object properties shows the property General in template picture C_@PCS7Typicals_CemV8_000.pdl.
Attribute 'type' is used for the automatic generation of block icons via OS compile and must therefore consist of an @-sign, the block type and an index.
For the manual link you should replace attribute 'type' e. g. by 'C_COUNT' (removing @-sign and index), in order not to overwrite the data in the next OS compile.

The Attribute **ToolTipText** is not entered automatically. You have to copy the text (normally the Tagname) manually.

Property Links

For each Counter in Property **Links** all Attributes must be connected as follows to the variables of the corresponding counter. (They will be connected automatically in case of automatic generation of Block Icons):

Property **Links** of object type C_COUNT:



Link-Attribute des C_COUNT:

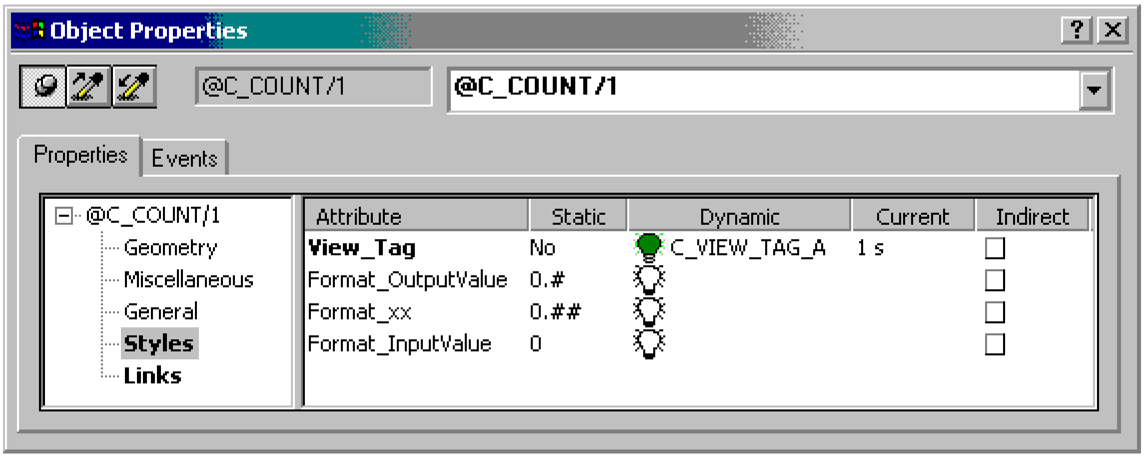
Value	to	.RT_OS	to show the counter value
Unit	to	.UNIT	to show the Unit

Link-Attribute des C_RUNNT:

Value	to	.RT_OS	to show the operating hours

Property Styles

Property **Styles** of object type C_COUNT:



Style Attributes of C_COUNT:

View_Tag to	C_VIEW_TAG_A	Tagname will be switched to 'visible' via button "A"
Format_OutputValue	0.#	not relevant
Format_xx	0.##	not relevant
Format_InputValue	0	Output format for the counter value: 0.0 one digit behind the comma 0.00 two digits behind the comma 0.0## one fix digit behind the comma and two optional digits (only if not 0). These settings are also used in the faceplate of the object.

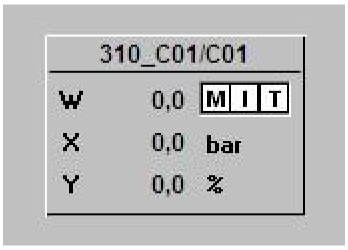
Style Attributes of C_RUNNT:

View_Tag to	C_VIEW_TAG_A	Tagname will be switched to 'visible' via button "A"
Format_OutputValue	0.#	not relevant
Format_xx	0.##	not relevant
Format_InputValue	0	Output format for the runtime (operating hours): 0.0 one digit behind the comma 0.00 two digits behind the comma 0.0## one fix digit behind the comma and two optional digits (only if not 0). These settings are also used in the faceplate of the object.
Text	h	time unit

CTRL_PID, CTRL_S and C_PID3

The controller functions CTRL_PID and CTRL_S are no CEMAT blocks and belong to the PCS 7 V7 Library. For further explanations see PCS 7 Manuals.
The controller function C_PID3 (Controller with three parameter tables) is an enhancement of block CTRL_PID. The properties of the block icon correspond to the properties of CTRL_PID.

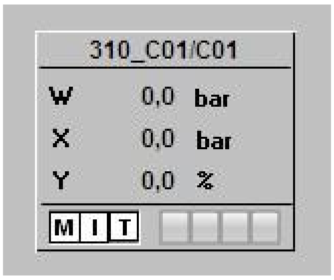
The faceplates for block CTRL_PID have been adapted to Cemat (same look and feel). In the block icon the most important controller values are displayed.

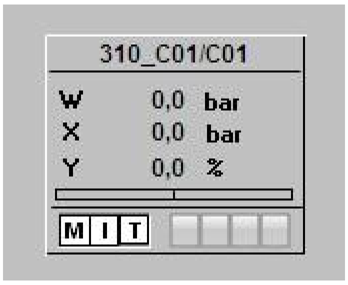


In this case, all further objects, belonging to the customized object are hidden behind the status display or invisible by default. However there are block icons with additional indications.

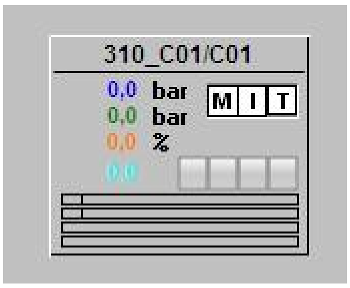
Example 1: Line representation with Tagname, Comment, Status and Set point

Example 2: with Group Display (Index 5) or with Group Display and bar for control deviation (Index 7)





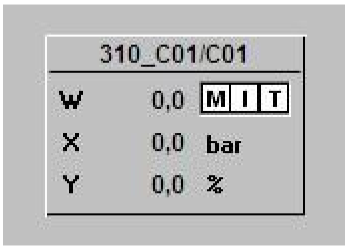
Example 3: with actual value, set point, output and output feedback, including bar indication (Index 6)



Beside this you can also use the standard block icons from the PCS 7 library. These block icons call the Standard faceplates of PCS 7:

Example 4: Block icons form the PCS 7 Standard library

The block icon of C_PID3 is identical to the block icon of CTRL_PID.



For block CTRL_S adapted Faceplates for Cemat do not exist. In case of the block icon, also the block icon of the PCS 7 standard library is used.

Function overview for the different block icons:

CTRL_PID:

Index 1 and 2	from PCS 7-Standard library (calls PCS 7 Standard Faceplates)
Index 3	with Setpoint, Actual value, Output and Status display
Index 4	Line representation with tagname, comment, status and set point
Index 5	with Set point, actual value, output, status and Group display
Index 6	shows the controller values, including bar indication and Group display
Index 7	with Group display and bar for controller deviation

CTRL_S:

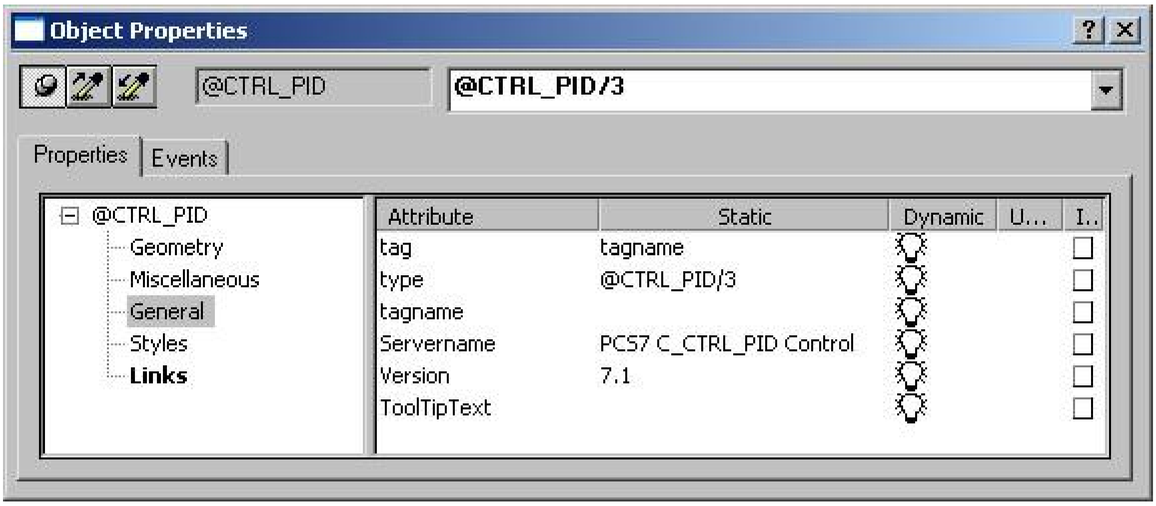
Index 1	from PCS 7-Standard library (calls PCS 7 Standard Faceplates)

C_PID3:

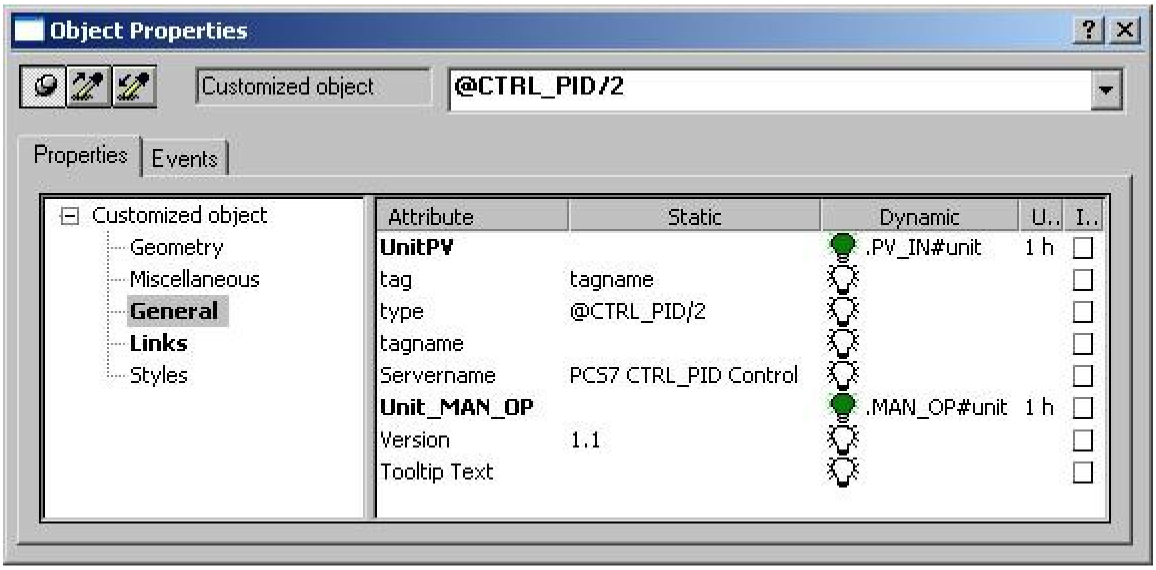
Index 1	with Set point, Actual value, Output and status display

Property General

Property **General** of object types CTRL_PID (with Cemat Faceplate call) and C_PID3



Property **General** of object type CTRL_PID (with call of PCS 7 Standard faceplate) or CTRL_S



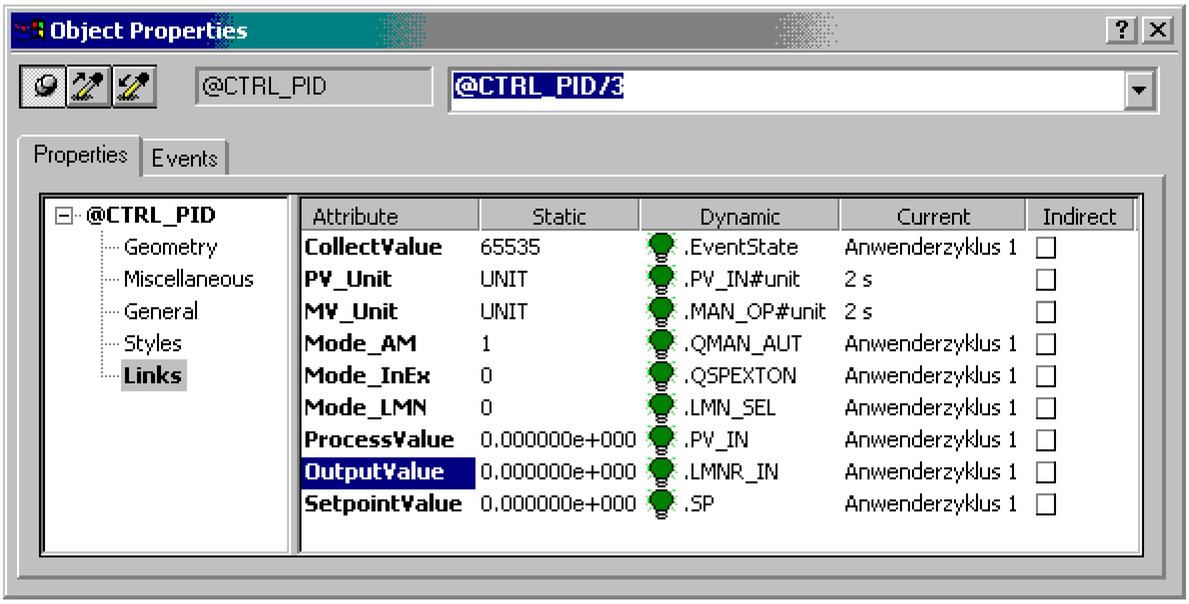
Caution: The screen shot of the object properties shows the property General in template picture C_@PCS7Typicals_CemV8_000.pdl.
Attribute 'type' is used for the automatic generation of block icons via OS compile and must therefore consist of an @-sign, the block type and an index.
For the manual link you should replace attribute 'type' e. g. by 'CTRL_PID' (removing @-sign and index), in order not to overwrite the data in the next OS compile.

The Attribute **ToolTipText** is not entered automatically. You have to copy the text (normally the Tagname) manually.

Property Links

For each Counter in Property **Links** all Attributes must be connected as follows to the variables of the corresponding counter. (They will be connected automatically in case of automatic generation of Block Icons):

Property **Links** for object type CTRL_PID:



Link Attributes of CTRL_PID:

CollectValue	to	.EventState	to show the group display
PV_Unit	to	.PV_IN#unit	to show the Unit for the Actual value
MV_Unit	to	.MAN_OP#unit	to show the Unit for the Controller Output
Mode_AM	to	.QMAN_OUT	to show the Mode Automatic/Manual
Mode_InEx	to	.QSPEXTON	to show the Mode Internal/External
Mode_LMN	to	.LMN_SEL	to show the Mode of the Output
ProcessValue	to	.PV_IN	to show the Actual Value
OutputValue	to	.LMNR_IN	to show the Output
SetpointValue	to	.SP	to show the Set point

Additional Attributes for the line representation (Index 4)

Comment	to	.#Comment	to show the block comment

Additional Link Attributes for bar indication (Index 6)

MV_IN	to	.MAN_OP
LLMR	to	.MO_PVLR
ULMR	to	.MO_PVHR
LLMR_MA	to	.MAN_LLM
ULMR_MA	to	.MAN_HLM

Additional Link Attributes for the display of the controller deviation (Index 7)

ER	to	.ER	to show the controller deviation
ER_AlarmHigh_%		50	Limit for alarm high
ER_AlarmLow_%		-50	Limit for alarm low

Link Attributes of CTRL_S:

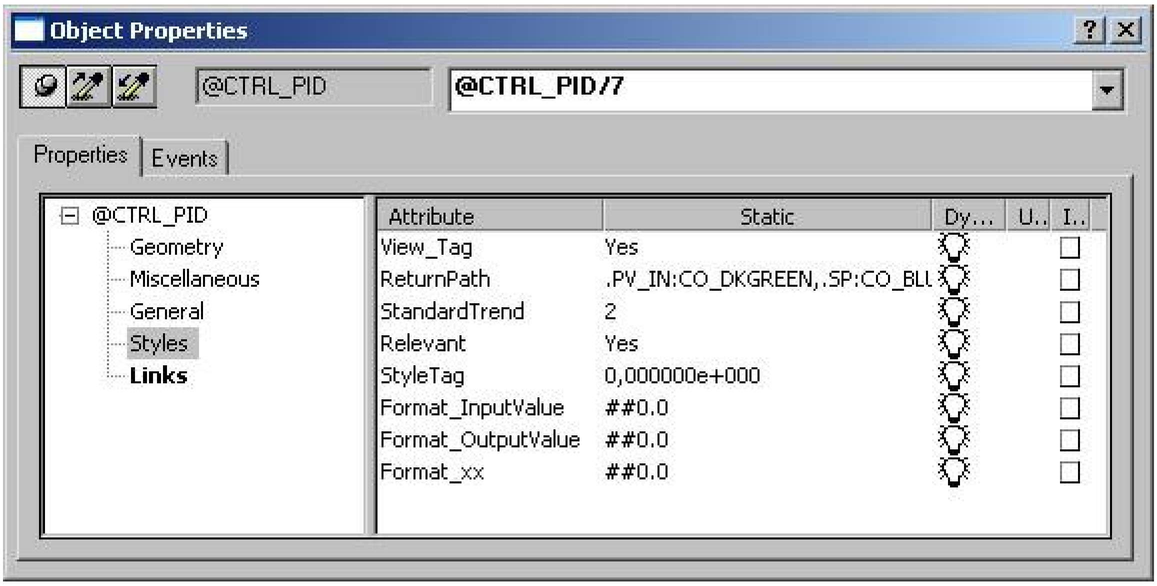
CollectValue	to	.EventState	to show the group display
SetpointValue	to	.SP	to show the Set point
ProcessValue	to	.PV_IN	to show the Actual Value
OutputValue	to	.LMNR_IN	to show the Output
QLMNR_ON	to	.QLMNR_ON	to show the Output
QLMNUP	to	.QLMNUP	to show the Output
QLMNDN	to	.QLMNDN	to show the Output
Mode_MAN_AUT	to	.QMAN_OUT	to show the Mode Automatic/Manual
Mode_INT_EXT	to	.QSPEXTON	to show the Mode Internal/External
LMN_SEL	to	.LMN_SEL	to show the Mode of the Output

Link Attributes of C_PID3:

CollectValue	to	.EventState	to show the group display
PV_Unit	to	.PV_IN#unit	to show the Unit for the Actual value
MV_Unit	to	.MAN_OP#unit	to show the Unit for the Controller Output
Mode_AM	to	.QMAN_OUT	to show the Mode Automatic/Manual
Mode_InEx	to	.QSPEXTON	to show the Mode Internal/External
Mode_LMN	to	.LMN_SEL	to show the Mode of the Output
ProcessValue	to	.PV_IN	to show the Actual Value
OutputValue	to	.LMNR_IN	to show the Output
SetpointValue	to	.SP	to show the Set point

Property Styles

Property **Styles** for object type CTRL_PID:



Style Attributes of CTRL_PID and C_PID3::

View_Tag	to	C_VIEW_TAG_A	Tagname will be switched to 'visible' via button "A"
ReturnPath		.U:CO_DKGREEN	internal (default setting must remain!)
StandardTrend		2	internal (default setting must remain!)
Relevant		Yes	internal (default setting must remain!)
Format_InputValue		##0.0	Output format: 0.0 one digit behind the comma 0.00 two digits behind the comma 0.0## one fix digit behind the comma and two optional digits (only if not 0). These settings are also used in the faceplate of the object.
Format_OutputValue		##0.0	see Format_InputValue
Format_xx		##0.0	see Format_InputValue
Visible2		No	intern

Style-Attribute of Standard CTRL_PID or CTRL_S:

View_Tag	Yes	visible
ReturnPath	.U:CO_DKGREEN	internal (default setting must remain!)
StandardTrend	2	internal (default setting must remain!)
Relevant	Yes	internal (default setting must remain!)
Format_InputValue	##0.0	Output format Set point
Format_OutputValue	##0.0	Output format Output value
Format_xx	##0.0	Output format Actual value
FontName	Arial	text description
FontSize	12	text description
FontNameNameOfTag	Arial	text description
FontSizeNameOfTag	12	text description

Window Position of the Faceplates

In Property **Miscellaneous** via the Attributes topPos, leftPos and defaultPos it can be defined at which screen position the faceplate appears when it is opened via click on the block icon.

For the Positioning the following possibilities exist:

a) The Faceplate appears at the Default Position

TopPos = irrelevant
leftPos = irrelevant
DefaultPos = yes

b) The Faceplate appears at the Curser-Position for

TopPos = 0
leftPos = 0
DefaultPos = no

c) The Faceplate appears at the set Position for

TopPos = Wert > 0
leftPos = Wert > 0
DefaultPos = no

Note: The Positioning of the Faceplates via Function PCS7_OpenGroupDisplay_V6_CEMAT (Cemat V6.0 Method) is not possible any more since Cemat V7.1.

Bitmaps

The bitmaps for the symbols are stored on all clients and servers in the standard directory (D:\CEMAT_CS\bitmap\).
Symbols used for the project must be copied into the <WinCC-Project directory>\GraCS, otherwise they cannot be used in the OS Engineering. The master for the bitmaps is the Server. The distribution can be done with a batch file.

For Drives and Valves all bitmaps must exist, which are theoretically foreseen for the Symbol, even if the status is never used in Runtime. WinCC will give an error message if any of the bitmaps does not exist.

In the Subfolder CEM_DRAFTS you find zip-files with further bitmaps and example pictures (Return from existing plants) which may be used as samples.

General Cemat Functions

The template picture C_Symbols.pdl contains general Cemat Functions. The buttons can be inserted in any process picture (e. g. in the system overview) in order to call superordinated functions.

The examples must be adapted and labeled accordingly.

Call of the Maintenance Overview

Via button "C_BtMaintenanceList" the Maintenance Overview will open:

Adaptations are not required.

Call of the Info Overview

Via button "C_BtInfo" a window will open which shows the last entry into the user archive. From there you can browse back or change to a spreadsheet format.

Adaptations are not required.

Query of the Object Browser

Via button "C_BtObjBrowser1" the operator will open the CEMAT Object Browser and will get a list of objects within an AS.

The function for left mouse click must be adapted (enter number of the AS):

long lPlcNo = 1; //The number of the PLC (e.g.: 1 ➔ SYSPLC01; 12 ➔ SYSPLC12;…)

Close all open windows

Button "C_BtCloseAllFaceplates" closes all the open faceplates at a time.

Adaptations are not required.

Open Curve Group on Full Screen

The button “C_BtOpenCurveGroup” opens the curve group on full screen (optional).

The function for left mouse click can be adapted (enter the name of the curve picture)
default = C_Curve_Group1.pdl

See chapter **Open curve group on full screen (optional)** below.

Open curve group on full screen (optional)

This option allows the display of pre-configured curve groups in full screen mode, which means, when opening the curve group the complete screen is used as working area. There is no alarm line, no overview range (picture tree) and not button line..

Caution: The full screen representation of curve groups is only possible for OS with only one screen. Multiscreen does not work.

On the Cemat installation CD under Cemat_AddOn\Full_Screen\WinCC\GraCS you find the required PDLs which you have to copy into your OS, if the function is required:

@screen_CEM_1280.pdl	Screen organization (1280x1024)
@screen_CEM_1680.pdl	Screen organization (1680x1050)
@screen_CEM_1920.pdl	Screen organization (1920x1200)
@screen_CEM_1920x1080.pdl	Screen organization (1920x1080)
C_Curve_Group1.pdl	Example 1 Curve groups default (1680x1050)
C_Curve_Group1_1280.pdl	Example 1 Curve groups (11280x1024)
C_Curve_Group1_1680.pdl	Example 1 Curve groups (1680x1050)
C_Curve_Group1_1920.pdl	Example 1 Curve groups (1920x1200)
C_Curve_Group1_1920x1080.Pdl	Example 1 Curve groups (1920x1080)
C_Curve_Group2.pdl	Example 2 Curve groups default (1680x1050)
C_Curve_Group2_1280.pdl	Example 2 Curve groups (11280x1024)
C_Curve_Group2_1680.pdl	Example 2 Curve groups (1680x1050)
C_Curve_Group2_1920.pdl	Example 2 Curve groups (1920x1200)
C_Curve_Group2_1920x1080.Pdl	Example 2 Curve groups (1920x1080)
C_Curve_Group3.pdl	Example 3 Curve groups default (1680x1050)
C_Curve_Group3_1280.pdl	Example 3 Curve groups (11280x1024)
C_Curve_Group3_1680.pdl	Example 3 Curve groups (1680x1050)
C_Curve_Group3_1920.pdl	Example 3 Curve groups (1920x1200)
C_Curve_Group3_1920x1080.Pdl	Example 3 Curve groups (1920x1080)

File @screen.pdl <u>must</u> be copied into the OS project in the required screen resolution.

The rest of the pictures contain examples for the display of curve groups, also in different screen resolutions, which you can use as template. Just copy it into a new file name and make the adaptations according to the process values you want to show.

For the call of the curve groups in full screen mode you have to use the button function from template picture C_Symbols.pdl.

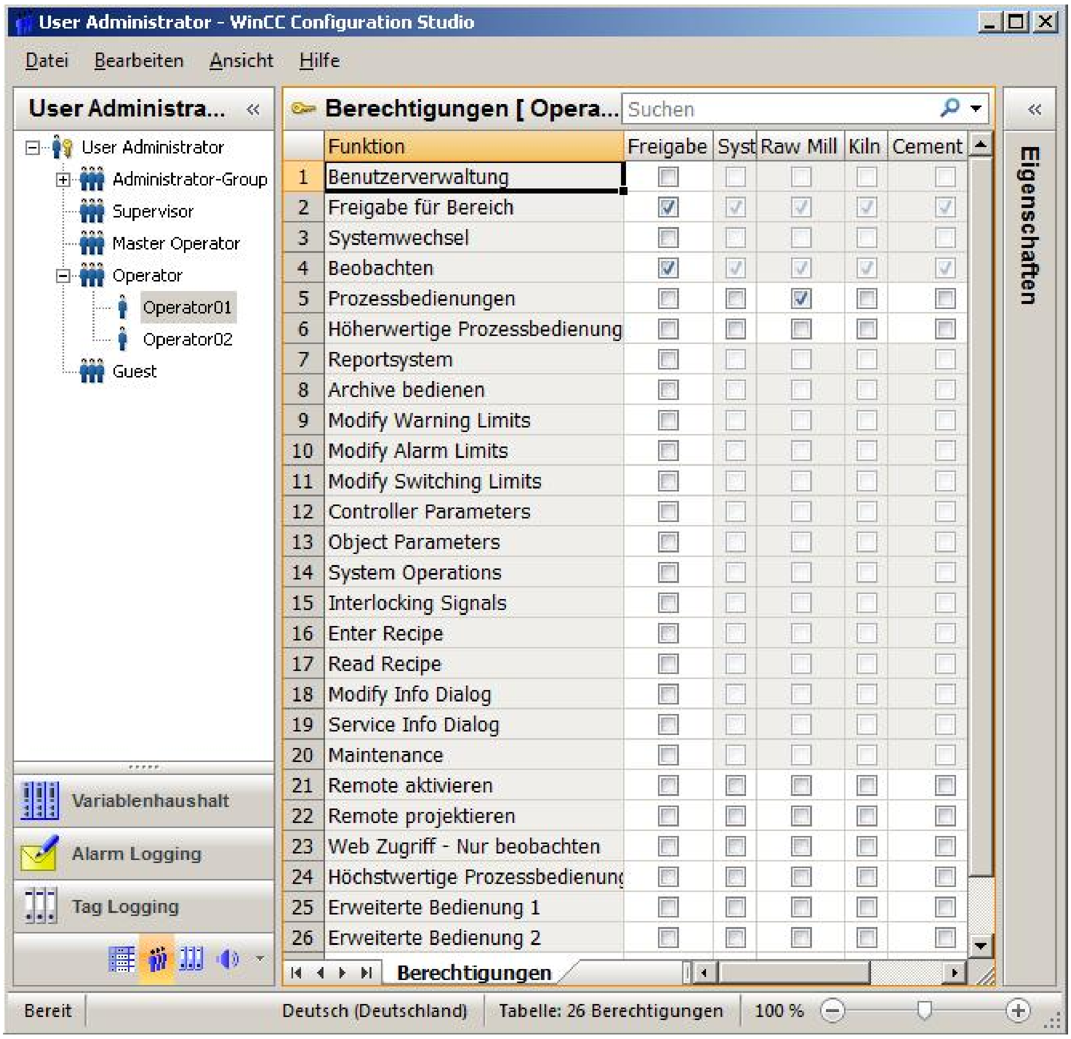
Copy this into your process picture and adapt the function for Mouse Action, by entering the name of the curve group:

```
#include "apdefap.h"
void OnClick(char* lpszPictureName, char* lpszObjectName, char* lpszPropertyName)
{
	TCHAR szPictureName[ ] = "C_Curve_Group1.pdl";	//Name of the Curve Group


	SetPropChar( "@screen", "CemCurve", "PictureName", szPictureName);

	SetPropWord("@screen", "CemCurve", "Top", 0);
	SetPropWord("@screen", "CemCurve", "Left", 0);
	SetVisible("@screen", "CemCurve", TRUE);
}
```

User Administration

Definition of user groups and users

In order to permit certain actions to certain persons depending on their responsibility, user groups and users must be defined in the User Administrator. It this has not already been done during the setup of the project it must be done during OS Engineering, in the WinCC Configuration Studio for User Administrator.



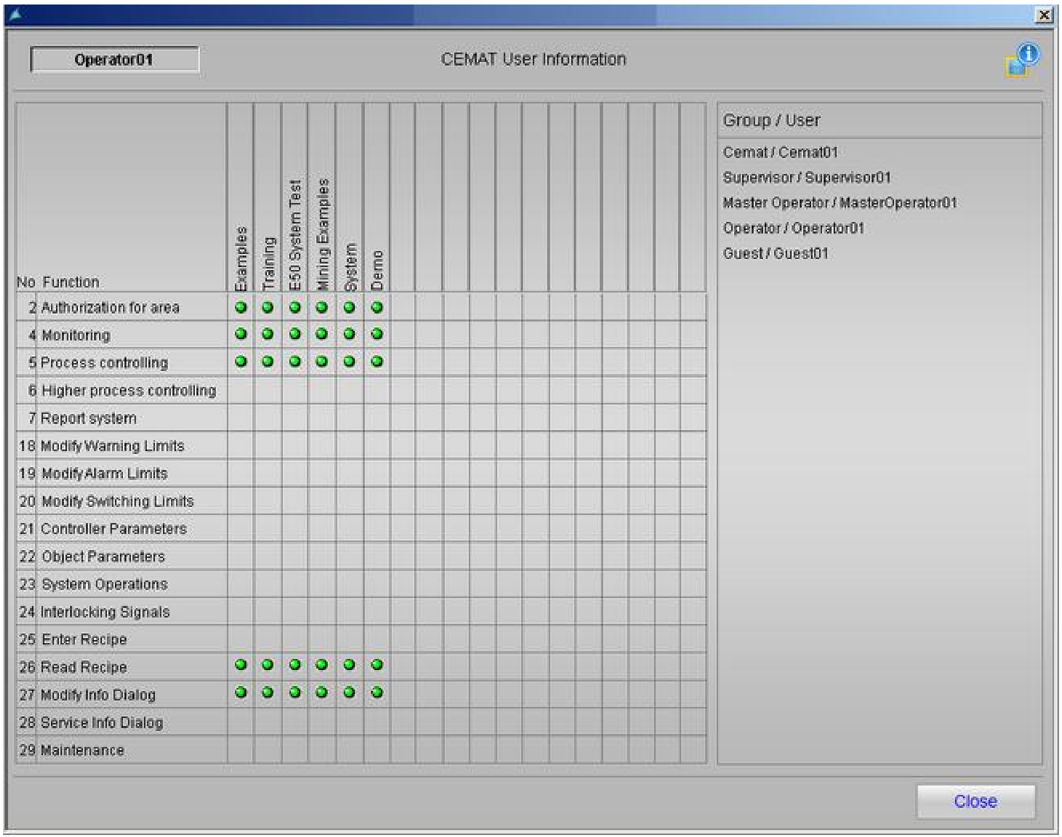
Each operation is linked to a specific authorization. For details see chapter PCS7_Project under "Operations and Authorization levels".
After the definition of user groups and users the individual authorizations can be enabled, either in general or per plant section (AREA).

After Login, each user can operate the system according to his authorization level.

The following authorization levels are used in Cemat:

002	Authorization for the Area	
004	Monitoring	call-up of the faceplates
005	Process operator control	manual operation changing the operation mode changes of setpoints
018	Modify warning limits	Measuring value
019	Modify alarm limits	Measuring value
020	Modify switching limits	Measuring value
021	Controller parameters	Parameter change for controller blocks
022	Object parameters	Change of process values
023	System operations	Reset Functions
024	Interlocking signals	Override and Simulation functions
025	Enter recipe data	
026	Read recipe data	
027	Info dialog value input	General Settings in the Info Dialog
028	Info dialog Service parameterization	Service Tab in the Info Dialog
029	Maintenance parameters	Change of Device information in the Maintenance Dialog

In the CEMAT User Information dialog you can see which authorizations are enabled for the actual user. To the right hand side you see a list of all User Groups and Users.


Operator01
CEMAT User Information
No Function
Examples
Training
E50 System Test
Mining Examples
System
Demo
2 Authorization for area
4 Monitoring
5 Process controlling
6 Higher process controlling
7 Report system
18 Modify Warning Limits
19 Modify Alarm Limits
20 Modify Switching Limits
21 Controller Parameters
22 Object Parameters
23 System Operations
24 Interlocking Signals
25 Enter Recipe
26 Read Recipe
27 Modify Info Dialog
28 Service Info Dialog
29 Maintenance
Group / User
Cemat / Cemat01
Supervisor / Supervisor01
Master Operator / MasterOperator01
Operator / Operator01
Guest / Guest01
Close

Proposal for the definition of User Groups:

User Right	Function	Administrator	Supervisor	Master Operator	Operator	Guest
1	User administrator	x				
2	Authorization for the Area	x	x	x	x	x
3	System change	x				
4	Monitoring	x	x	x	x	x
5	Process controlling	x	x	x	x	
6	Higher process control	x	x	x		
7	Report System	x	x			
8	Archive controlling	x				
18	Modify warning limits	x	x	x		
19	Modify failure limits	x	x	x		
20	Modify switching limits	x	x	x		
21	Controller parameters	x	x			
22	Object parameters	x	x			
23	System operations	x	x			
24	Interlocking signals	x	x	x		
25	Enter recipe data	x	x	x		
26	Read recipe data	x	x	x	x	x
27	Info dialog value input	x	x	x	x	
28	Info dialog Service parameterization	x				
29	Maintenance	x	x			

Instance specific Authorizations

The Authorizations are defined in the User Administrator of WinCC. Beside the authorizations of WinCC during the generation of the project additional authorizations have been defined for CEMAT (see chapter 3). There, you also find a list which explains which authorization is needed for which operation.

Defining different user groups and users you can enable or disable specific functions. This is possible per OS-Area.

In addition for some Operations it is possible to define instance specific rights. The settings have to be carried out at the block icon under Miscellaneous. The two attributes „Processcontrolling_backup“ and „HigherProcesscontrolling_backup“ are variables which are used in order to transmit free definable authorizations to the faceplate.

In the table in chapter 3 you can see for which operations instance specific rights are possible.

Extract of the table of authorizations from chapter 3 (example for C_DRV_1D):

Object type	Operation	Right/ default right	Instance specific settings possible via Attribute	Project Code
C_DRV_1D	Start/Stop Jump to Interlock Jump to rel. Group Jump to User Object Jump to Measure Obj. Jump to Ext. Setpoint Jump to SP Feedback	05: Process controlling	Processcontrolling_backup	all
	Process parameter	22: Object Parameters		all
	Reset Operation	23: System Operations		all
	Bypass speed monitor	24: Interlocking Signals	HigherProcesscontrolling_backup	all
	Maintenance functions	29: Maintenance		all
	auto/man. Interl./man. Non Interl.	05: Process controlling	Processcontrolling_backup	006
	non interl. Single mode Rel.	23: System Operations		007
	non interl. Single mode on/off	24: Interlocking Signals	HigherProcesscontrolling_backup	004/007

The first 5 operation functions in the list are possible for all project versions:

1. The Operations (**Start** and **Stop**, call of the **interlock faceplate**, call of the **related group**, call of a **user object**, call of the assigned **measure**, call of External **Setpoint** or **Setpoint feedback** source) are assigned to authorization 5 "Process controlling“. In this case it is not fix assignment, the authorization is transmitted via variable Processcontrolling_backup.
2. The Operations for the modification of the **Process parameters** is assigned fix to authorization 22 "Object parameters“.
3. The **Reset Operation** has a fix assignment to 23 "System Operations“.
4. The Operation **Bypass Speed Monitor** is assigned to authorization 24 "Interlocking Signals“. In this case it is not a fix assignment; the authorization is transmitted via variable HigherProcesscontrolling_backup.
5. The **Maintenance functions** are assigned to authorization 29 "Maintenance".

Some Operations are only available in Project Versions.

6. In Project Version "006" the Operation **auto/manual Interlocked/manual non Interlocked** is assigned to authorization 5 "Process controlling“. The assignment is carries out via variable Processcontrolling_backup.
7. In project version "007" the function **Release non-interlocked single-start-mode** is assigned fix to right 23 "System Operations“.
8. In the project versions "004" and "007" the function **non-interlocked single-start ON/OFF** is assigned to authorization 24 "Interlocking Signals". The assignment is carried out via variable HigherProcesscontrolling_backup.

Caution: Please keep in mind that the modifications of the attributes Processcontrolling_backup or HigherProcesscontrolling_backup, effect all operations which are assigned to Processcontrolling_backup or HigherProcesscontrolling_backup.
For the operations **Start** and **Stop** the same authorization is needed as for **auto/manual Interlocked/manual non Interlocked**.

Alarm logging

In the WinCC Alarm logging the message system is configured and the Message Archive is created. The adaptations in the Alarm logging which are needed for Cemat have already been carried out during the setup of the PCS 7 Project (see Engineering Manual, Chapter 03_PCS7_Project and refer to "Alarm logging").

If the Archive configuration has not been done yet you can do this now. Please refer to the PCS 7 Engineering Manuals.

Beside this, no further Engineering is required in the OS-Project.

Tag logging

In the WinCC Tag logging the Process Value Archive is configured. For each value which needs to be archived, an Archive tag must exist in the process value archive. Depending on the function of the value a suitable acquisition cycle and archiving cycle must be configured.

In the Archive Configuration you have to define the archive size, the time of the segment change and the Backup Location. For details refer to PCS 7 Engineering Manuals.

In Cemat mainly the measurement values need to be archived. The Archive tags will be created automatically if the block output *MV* is selected for "Archiving" or "Long-term archiving" before OS Compile. The Archive tags are created in a common Process Value Archive with name "SystemArchive".

The Archive tags can also be created manually in the Tag logging. This has been done in some existing installations, and it is a must in case of Archive splitting (Multiple Process Value Archived, e. g. one per AREA), because the OS Compile only supports one common archive.

⚠ **Caution:** In case of manual creation of Archive tags, make sure that the naming conventions are correct. Otherwise the C_MEASUR is not able to find the Archive Tag for curve display. See to Engineering Manual 14_Tips_Tricks and refer to "Process Value Archive Options".

Independent of automatic or manual creation of the Process Value Archive, in the WinCC Configuration Studio for Tag logging you have to configure the suitable Acquisition cycle and Archiving/Display cycle, because during automatic generation via OS Compilation always the minimum acquisition cycle is used.

- Quick archiving may be used for currents, pressures etc.
- Slow archiving may be used for temperatures etc.

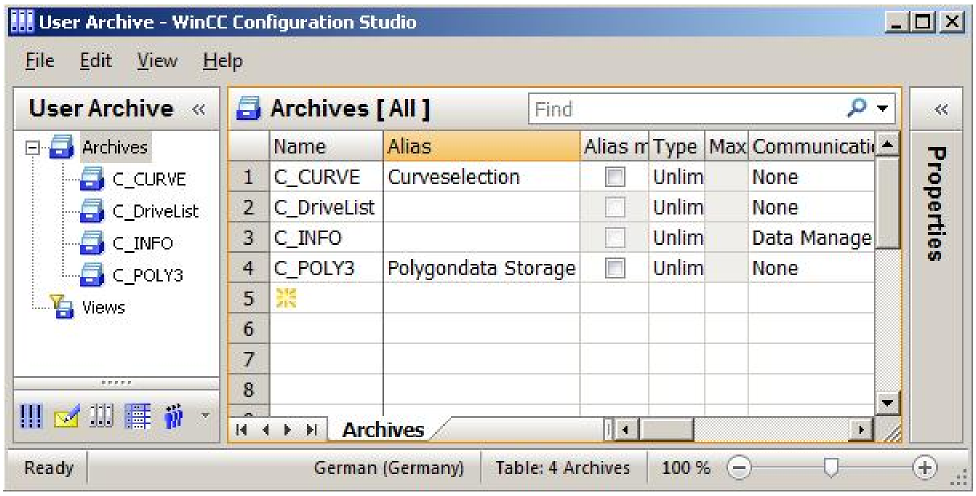
Example of typical configuration for cement plant:

Process value Type	Acquisition cycle in s	Archiving/Display cycle in s
Temperature	30	30
Level	10	30
Pressure	2	10
Flow	5	10

How to create a process value archive and the archive tags is described in the engineering manual of PCS 7 and not part of this description.

User Archives

During the setup of the PCS 7 Project, via import of Excel Workbook Import_UserArchives.xlsx four user archives have been created.



For the User Archives **C_INFO**, **C_CURVE** and **C_POLY3** at least one line of runtime data must exist, which has been created via import of the CSV files C_INFO.csv, C_CURVE.csv and C_POLY3.csv.

The User Archive **C_DriveList** (group instance list) can remain empty and does not need any further engineering.

The data for the User Archives **C_DriveList**, **C_CURVE** and **C_POLY3** will be provided online during operation of the Runtime system.

The User Archive **C_INFO** must contain the information data for each Cemat Object, in order to provide this in the Runtime System (Info Dialog and Maintenance Dialog of the Objects).

We recommend using the **Cemat Info-Tool** C_INFO.xlsm in order to create the Info-Data based on the PCS 7 Project data. The description for the Info-Tool you find in the Engineering Manual, chapter 09_Engineering_Tools and the Tool itself (Excel Macro) you will find under D:\Cemat_CS\TOOLS.

- The C_INFO.xlsm will create a csv file which contains all available information from the PCS 7 Project.
- The same tool can be used in order to add the information which is not part of the PCS 7 Project, using Excel. After adding the information you can create a csv file for the import with WinCC Configuration Studio for User Archives.

The following table shows the file structure of C_INFO data (Field names and Content). All data in **bold** field names are derived from the PCS 7 Project. The rest of the fields (italic) are optional and can be filled if required.

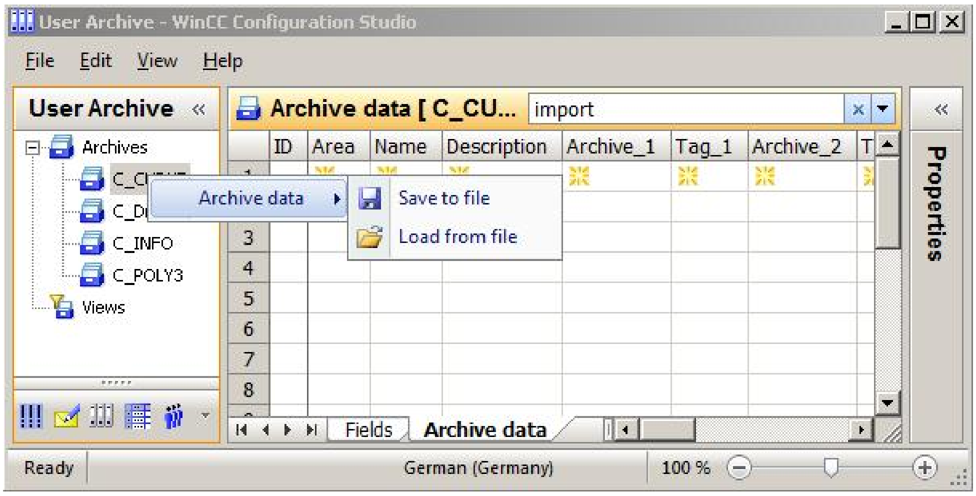
Field Name (Alias)	**Content**	**Used in**
Obj_ID	must be unique	General use
C_TAG	Tagname of the object - must be unique	General use
GL_Complex	Complex name	Info Dialog (General Location)
GL_Plant	Plant name	
GL_Plant_zone	Name of the Plant zone (AREA)	
GL_Equipment	Equipment name	
GL_LocationObject	Location of the Object	
GL_Location_LS	Not used	

Name	Remarks	Used in
G_LoopA	The entry in field G_LoopA serves as a reference to the process picture to be opened if the loop-in-Alarm Button is pressed. If this field remains empty, the system looks into the 'Component list Editor' of WinCC in order to open the 'Entry point picture'.	Info Dialog (Tab "Service)
G_Infotext	Info Text from Operator (will be entered online)	Info Dialog (Tab "Service) By entering the file name for manuals and video files, these documents can be opened from the Info-Dialog by additional buttons. →See Multi Media Interface
G_Sound_1	File name of a sound file (.wav)	
G_Sound_2	File name of the loop diagram	
G_DRAW_POS	File name of Operating instructions	
G_Video_Name	File name of a Video file (.avi)	
G_DOC_Name_1	File name of General functional specifications	
G_Spare	File name of Electrical schematics	
G_Spare_1	File name of MCC drawings	
I_Building	Building of the I/O Panel	Info Dialog (Tab "Input/Output")
I_Location	Place / Room of the I/O Panel	
I_Area	Area of the I/O Panel	
I_Cabinet	I/O Panel	
I_Rack_Q	Not used	
I_Slot_Q	Not used	
I_Rack_I	Not used	
I_Slot_I	Not used	
M_Building	MCC Building	Info Dialog (Tab "MCC Data")
M_Location	MCC Location	
M_Area	MCC Area	
M_Cabinet	MCC Cabinet	
M_Feeder	MCC Feeder	
M_Fuse	MCC Fuse Rate	
M_Bimetall	MCC Bimetal settings	
H_Building	Building of the AS Cabinet	Info Dialog (Tab "AS Hardware/ Software")
H_Location	Location of the AS Cabinet	
H_Area	Area of the AS Cabinet	
H_Cabinet	AS Cabinet Name	
H_Rack	AS Rack	
H_PLC_Name	AS Name	
H_Instanc_DB_N0	Instance DB of the block	

Name	Remarks	Used in
M_DT	Device type	Maintenance Dialog
M_MA	Manufacturer	
M_ON	Order Number	
M_Interval	Intervall	
M_US	User/Operator	
M_MES	Message	
M_INS	Instruction	
M_SER	Serial Number	
M_IND	Installation Date	
M_HWR	Hardware Revision	
M_SW	Software Revision	
M_LMT	Last Maint. Time	
M_DOC	Documentation file	
M_ET	End Time	
M_LMT_START	Last Main Time started	
M_INF1	Information 1	
M_INF2	Information 2	
M_INF3	Information 3	

Import of the Info Data:

With right mouse click on C_INFO → *Archive data* → *Load from file*, select file C_INFO.csv, and press Button *Import.*



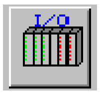
Important notes for the import:

1. ID and Obj_ID must be numbered consecutively.
2. Via Import function only **new** records can be imported, i. e. if you want to modify existing data records you have to export these records, and to delete it before you can import it again.

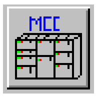
Multimedia Interface

From the Info Dialog object specific Documents can be opened. As these documents are not always available it can be configured for each object type, which means for each object type it is possible to decide which additional Buttons should be visible in the Info Dialog. The settings are carried out in directory D:\CEMAT_CS\CONFIG in the Configuration file of the particular object type.

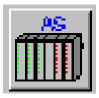
In the Folders "Input/Output", "MCC Data" and "AS Hardware and Software" a button will open respective drawings:



Folder 'Input/output' Assignment of the input/output cards (cabinet assignment)

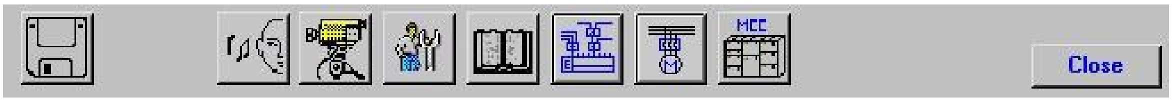


Folder 'MCC data' Assignment of the MCC cabinet (cabinet assignment)

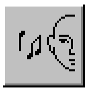


Folder 'AS hardware' Construction of the cabinet in which the AS is installed which this object belongs to (cabinet assignment).

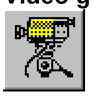
The additional Buttons at the bottom of the information dialog can be used to open a Sound file, a Video, a Operation procedure, a Function description, a Loop diagram, an Electrical schematic or a MCC Data.



Sound Sound replay (USER)

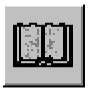


Video guide Video description for trouble-shooting (USER)

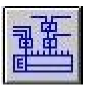


Operation procedures Descriptions how to operate (USER) as doc, pdf, xls

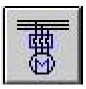
General function desc. Descriptions of general functions for this object (USER) as doc, pdf, xls



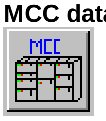
Loop diagrams Process engineering- and Software interlock as CAD drawing



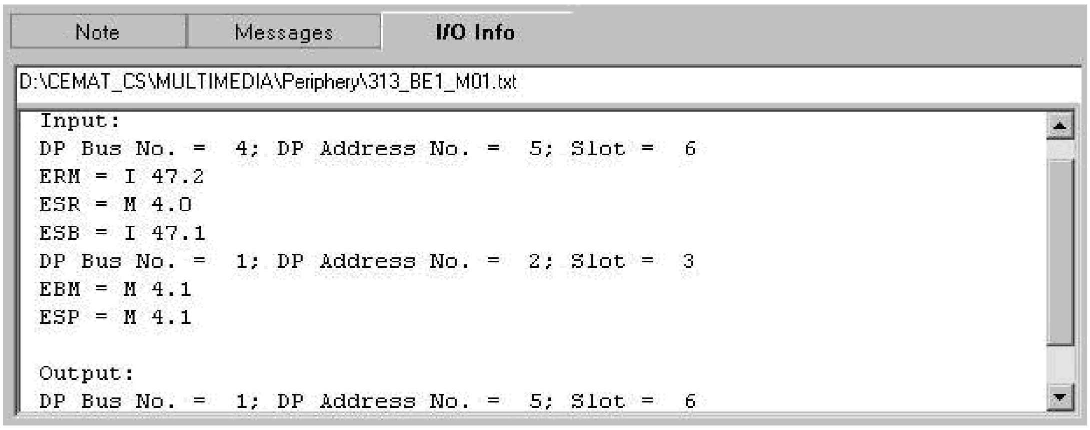
Electrical schematic Electrical schematic Drawings as CAD drawing



MCC data MCC cabinet drawings, settings as CAD drawing



In Folder I/O Info Text files with Input/ Output Information can be displayed.



The Text files for I/O Info can be created with an Excel Macro based on the data from the HW-Configuration and the Process object view. Refer to chapter 09_Engineering_Tools.

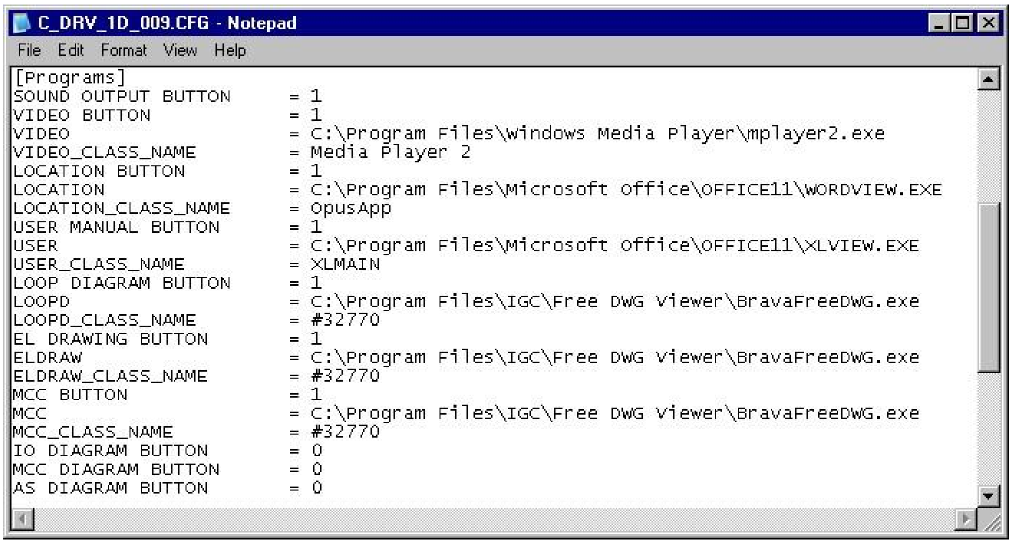
Enable the Multimedia Functions

In order to enable the user program neutral interfaces for multimedia applications, open in the directory D:\CEMAT_CS\CONFIG the module specific to file, e.g., C_DRV_1D_009.cfg and edit the area [Programs] as follows:

In the following example the Video Function and the Service Manual shall be enabled.

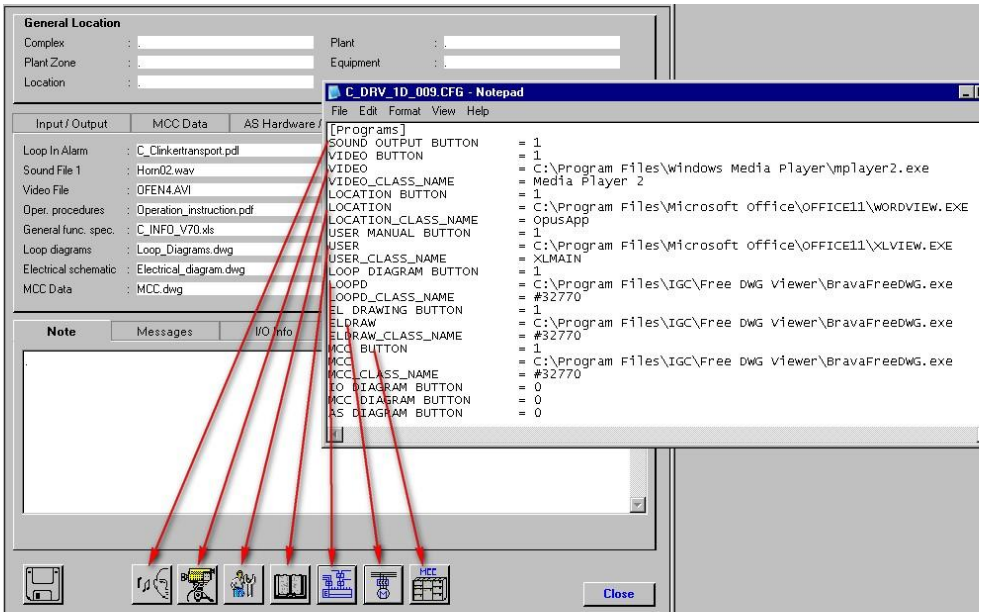
1. Switch buttons visible by place the parameter xxxxxBUTTON = 1

2. Write Path and Program name of the Viewer into "XXXXX"
   - write xxxx_Class_Name (see define Window-Class-Name)



```
[Programs]
SOUND OUTPUT BUTTON      = 1
VIDEO BUTTON             = 1
VIDEO                    = C:\Program Files\Windows Media Player\mplayer2.exe
VIDEO_CLASS_NAME         = Media Player 2
LOCATION BUTTON          = 1
LOCATION                 = C:\Program Files\Microsoft Office\OFFICE11\WORDVIEW.EXE
LOCATION_CLASS_NAME      = OpusApp
USER MANUAL BUTTON       = 1
USER                     = C:\Program Files\Microsoft Office\OFFICE11\XLVIEW.EXE
USER_CLASS_NAME          = XLMAIN
LOOP DIAGRAM BUTTON      = 1
LOOPD                    = C:\Program Files\IGC\Free DWG Viewer\BravaFreeDWG.exe
LOOPD_CLASS_NAME         = #32770
EL DRAWING BUTTON        = 1
ELDRAW                   = C:\Program Files\IGC\Free DWG Viewer\BravaFreeDWG.exe
ELDRAW_CLASS_NAME        = #32770
MCC BUTTON               = 1
MCC                      = C:\Program Files\IGC\Free DWG Viewer\BravaFreeDWG.exe
MCC_CLASS_NAME           = #32770
IO DIAGRAM BUTTON        = 0
MCC DIAGRAM BUTTON       = 0
AS DIAGRAM BUTTON        = 0
```

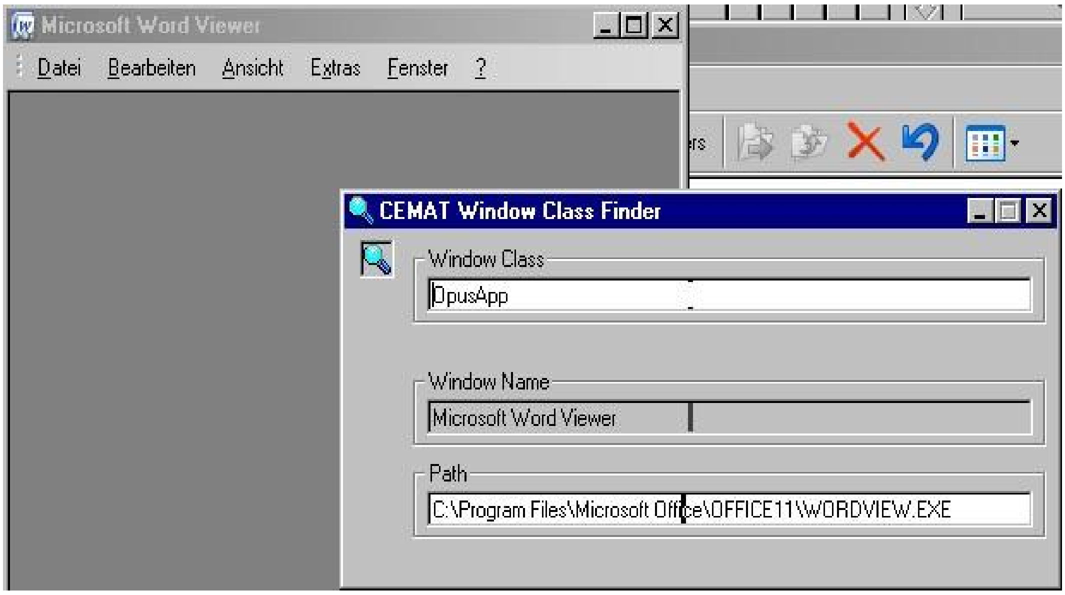
assignment of button to cfg Line



Define Window-Class-Name

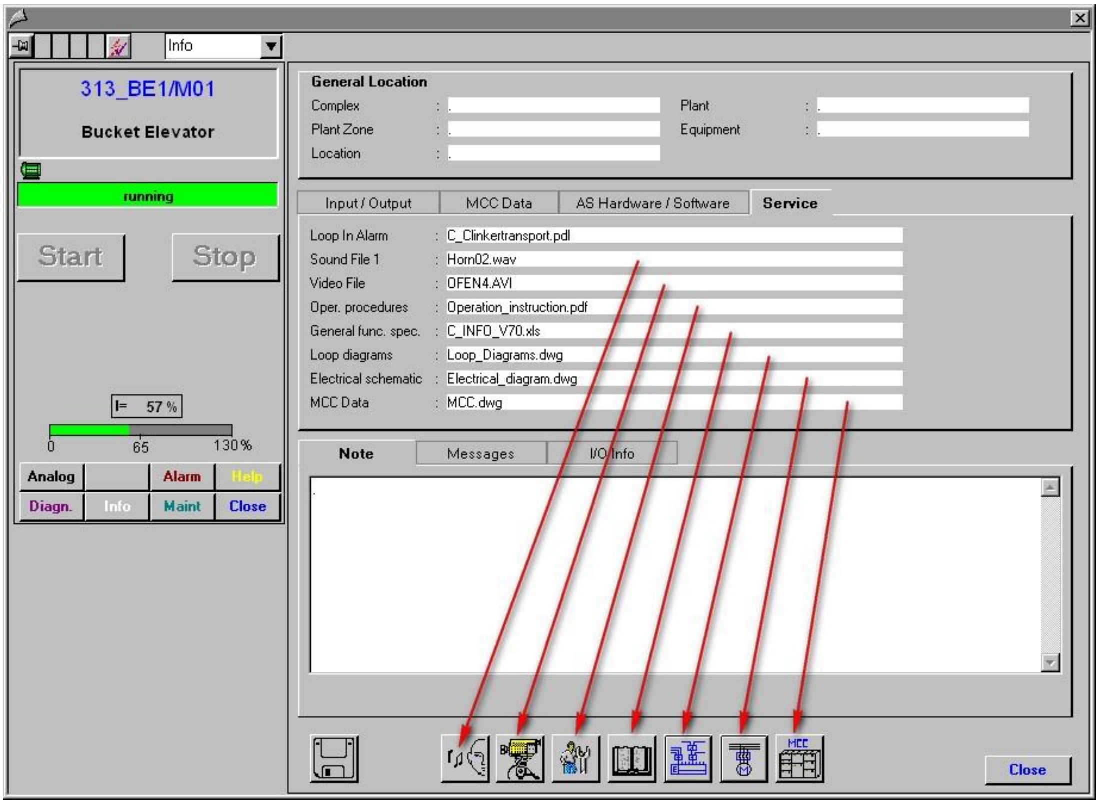
To view the external program „modal", it is necessary to know the Window-Class-Name. Please follow the steps:

1. Start the desired program (Word viewer)
2. Start then from „D:\CEMAT_CS\BIN" the program „CematClassFinder.exe"
3. Use the magnifier from „ClassFinder" and move them over the Frame of the external program .
   - it will be shown the „WindowClaasName" and the
   - Path (incl. programname)
4. Copy both into the corresponding lines of CFG-file.



Enter Filenames in the INFO Dialog

The names for a Video, a Operation procedure, a Function description, a Loop diagram, an Electrical schematic or an MCC Datasheet are entered for each object in the register 'service' without path.



Storage Location of the Multimedia Files

During the Cemat installation the multimedia directory is automatically created under D:\CEMAT_CS.

\Multimedia

\AS	CAD Pictures of AS objects
\ELECTRICAL_DIAGRAM	CAD pictures of electrical drawings (free name)
\IO	CAD Pictures input/output of the objects
\LOOP_DIAGRAM	Loop Diagrams CAD (free name)
\Manual	general function description (free name)
\Map	not used anymore
\MCC	CAD Pictures form the MCC
\OPERATING_PROCEDURES	Description of operating procedures (free name)
\Periphery	I/O Information from HW Config
\Sound	Sound files for info dialog
\Video	place of deposit of the video-files

The user files have to be located in the respective subfolders.

Naming conventions for the user files

The images for 'AS', 'IO', 'MCC' are always derived from the Tagname. In addition the following rule has to be considered:

- Special characters in the Tagname have to be substituted by "_"
- File endings must be .jpg or .tif (.bmp is not allowed).

The text files under 'Periphery' are always derived from the Tagname. In addition the following rule has to be considered:

- Special characters in the Tagname have to be substituted by "_"
- The file ending must be .txt

The file names for Video, Operation procedure, Function description, Loop diagram, Electrical schematic and MCC Datasheet are freely eligible.

- The corresponding file endings are .pdf, .doc, .xls, .wav, .avi, .dwg

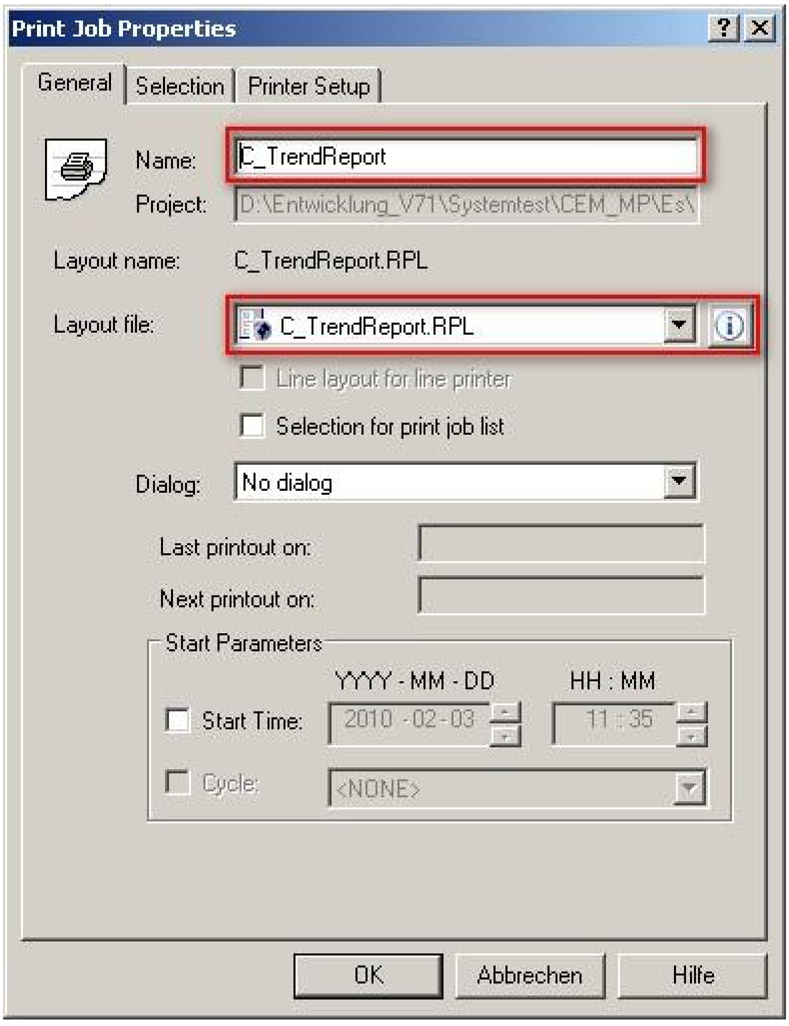
Note: The Names are case sensitive! Don’t use blank in the name.

Configuration of Print functions

For the printing of trend curves, with the Cemat delivery you received the Layout C_TrendReport.RPL, which has been copied into your OS Projects already during the setup of your PCS 7 Project.

After this the layout is available in the WinCC Project under Report Designer → Layouts → Language neutral.

For the configuration of the print functions you have to create a print job. Rename the print job into C_TrendReport and as Layout file select the Cemat Layout C_TrendReport.RPL.



Configuration of C_Config.cfg

Hardcopy to default printer

With Button “Print” the actual Faceplate can be printed at the standard printer.

If no printer is configured, the print jobs go to the print queue as for example „Microsoft Office Document Image Writer“. From there they are never used if no printer is connected.

In case a printer is configured for the station, the print function can be enabled under D:\CEMAT_CS\Config\C_CONFIG.cfg.

Proceed as follows:

- Open file C_Config.cfg (under D:\Cemat_CS\Config).

- There you find the lines:

```
  [CematSettings]
..;
  ;
... ;No Hard copy of the actual faceplate to the default printer…
...HardCopyToDefaultPrinter=0
```

- Change the last line into `HardCopyToDefaultPrinter=1`
  (Print button can be operated in Runtime)

Display mode of the object list

Via group object list function all elements which are linked to the group are displayed in a tree structure. By default the group only shows the objects which are directly linked to the group. In case of routes, it also shows the objects which are directly linked to the route.

In order to display the annunciations and measurements linked via O_LINK to the drive, the structure can be expanded via mouse-click on the '+' Button.

The default setting can be changed to direct expansion via setting in D:\CEMAT_CS\Config\C_CONFIG.cfg.

Proceed as follows:

- Open file C_Config.cfg (under D:\Cemat_CS\Config).

- There you find the lines:

```
[CematSettings]
..;
  ;
;Object List: Subobjects not expanded
ObjListExpanded=0
```

- Change the last line into `ObjListExpanded=1`
  (Expand Object list while opening)

Use of PDF Reader

In order to display the “PDF” Online Help files with an external PDF Reader you have to check or modify the settings in the “C_CONFIG.cfg” file. (The default is done with Adobe Reader V9.0).

The default setting can be changed to direct expansion via setting in D:\CEMAT_CS\Config\C_CONFIG.cfg.

Proceed as follows:
- Open file C_Config.cfg (under D:\Cemat_CS\Config).
- There you find the lines:

```
[CematSettings]
..;
  ;
;USE_PDF_READER=1 ==> help files can read/show with an installed pdf-
reader,
;find class name with "CEMAT Window Class Finder" - see "Define Window
Class Name" in 08_OS_Engineering_009.pdf
USE_PDF_READER=1
READER_PATH=C:\Program Files (x86)\Adobe\Reader 10.0\Reader\AcroRd32.exe
READER_CLASS_NAME=AcrobatSDIWindow
```

1. with “USE_PDF_READER” = 1 the external PDF Reader in the next line will be executed.
   = 0 it is not possible to start a PDF Reader

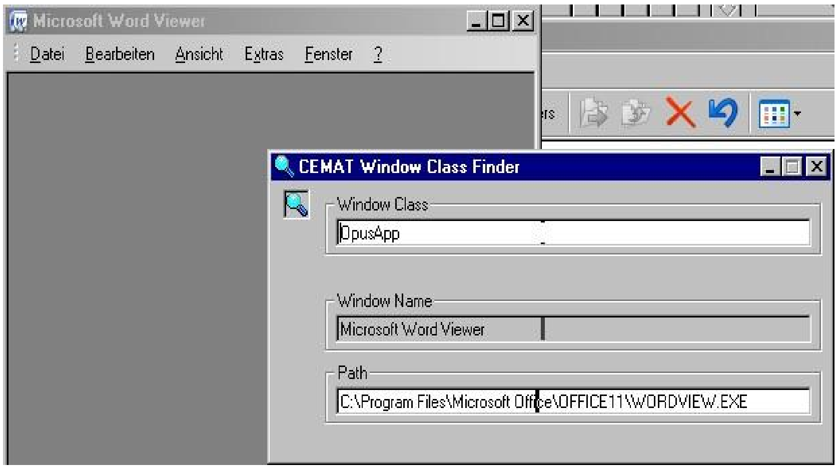
2. write the complete path and program name to “ READER_PATH”

3. define the Class-Name as it is descript below and write the Class-Name to “READER_CLASS_NAME”

Define Window-Class-Name

To view the external program „modal“, it is necessary to know the Window-Class-Name. Please follow the steps:

1. Start the desired program (Word viewer)
2. Start then from „D:\CEMAT_CS\BIN“ the program „CematClassFinder.exe“
3. Use the magnifier from „ClassFinder“ and move them over the Frame of the external program .
   - it will be shown the „WindowClaasName“ and the
   - Path (incl. programname)
4. Copy both into the corresponding lines of CFG-file.



CEMAT WEB

General

In addition to manual *PCS 7 - OS Web Option* this file contains information about the CEMAT WEB options.

Installation

CEMAT WEB can be used on the WEB Clients directly, without installing there any CEMAT software.

Note: Please consider the Settings for WEB Client:
For PCS 7 / CEMAT the WinCC classic design is not permitted.

Engineering

In order to enable opening the interlock faceplates on the CEMAT WEB Clients, you have to add the following tags in the tag management of **all os web servers**.

Open the Tag Management and add the internal variables for CEMAT_WEB:

Add new Group “CEMAT_WEB”

Within this group add the following variables:

C_< COMPUTERNAME >_WEBInfo1 - Text tag 8-bit character set - Computer-local
C_< COMPUTERNAME >_WEBInfo2 - Text tag 8-bit character set - Computer-local
C_< COMPUTERNAME >_WEBInfo3 - Text tag 8-bit character set - Computer-local

Replace < COMPUTERNAME > by the WEB station name, e. g. C_EF34930C_WEBInfo1. (COMPUTERNAME in capital letters)

Product Information

Publish from WinCC to process pictures

Library - function **CematUserAdmin\C_EnumUsersCallback.fct** may not be published.
Callback functions cannot be published.

Restrictions of the CEMAT OS WEB Client

The following functions are not supported by the CEMAT OS WEB Client:

Close, Help
The close keys and help keys in all Faceplates are locked.

Status
The group state call is not supported.

Objects
Group instance list is not supported.

Object Browser
The object list is not supported.

Alarm line info
The Info - button isn't available.

Alarm line LoopInAlarm
The LoopInAlarm - button isn't available.

CEMAT Reporting system
The CEMAT reporting system isn't available.

Maintenance
The operation in the maintenance picture is disabled.

Print Faceplate
The function *Print Faceplate* is disabled.

Quick View
The function *Quick View* of measured values is disabled.

The following functionality is limited:

The individual faceplate positioning is switched off.

Info
The operation in the info picture is limited.

Engineering Tools

Content

User Archive C_INFO

With CEMAT V8.2 an Excel Macro is available, which can be used to create the Info Data from the PCS 7 Engineering Data (Hardware Configuration, Process Object view). The Excel Macro generates a file C_INFO.csv, which can be imported into the User Archive.

The procedure how to generate the Info-Data is described in the following chapter.

Generation of the user archive C_INFO

This chapter describes the generation of the user archives C_INFO from the data of the PCS 7 Project. For this purpose necessary excel file C_INFO.xlsm are available in the directory D:\CEMAT_CS\TOOLS.

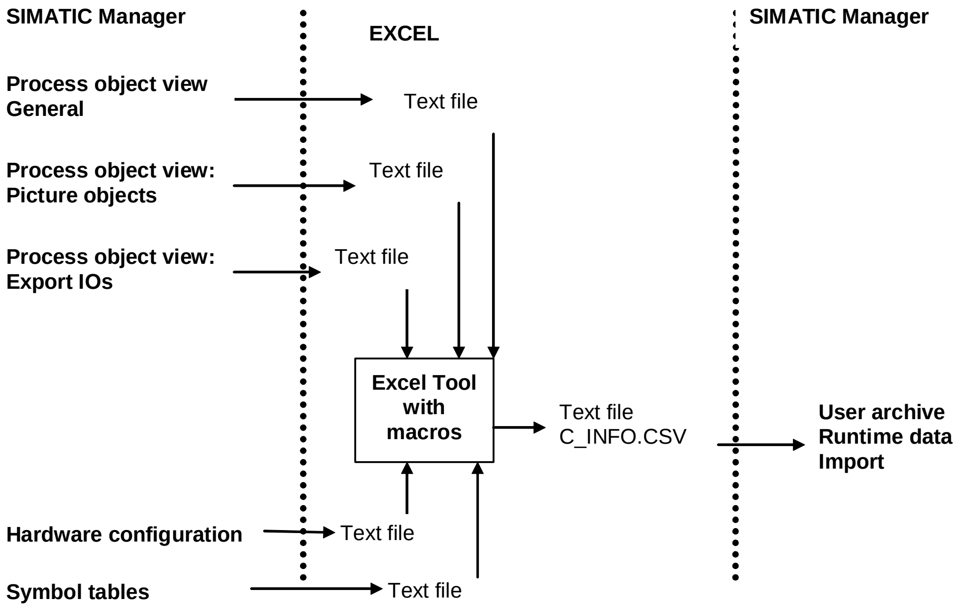
For the generation of the User Archive C_INFO from the process object view and the hardware configuration of the PLCs the following requirements must be met:

- The hardware configuration of each PLC must be exported as CFG files.
- The symbol tables of each PLC must be exported as SDF files.
- The content of the process objects views „General“ and „Picture objects“ must be copied into text files.
- All I/Os of the process objects view must be exported into text files.

The following columns are filled during this procedure:

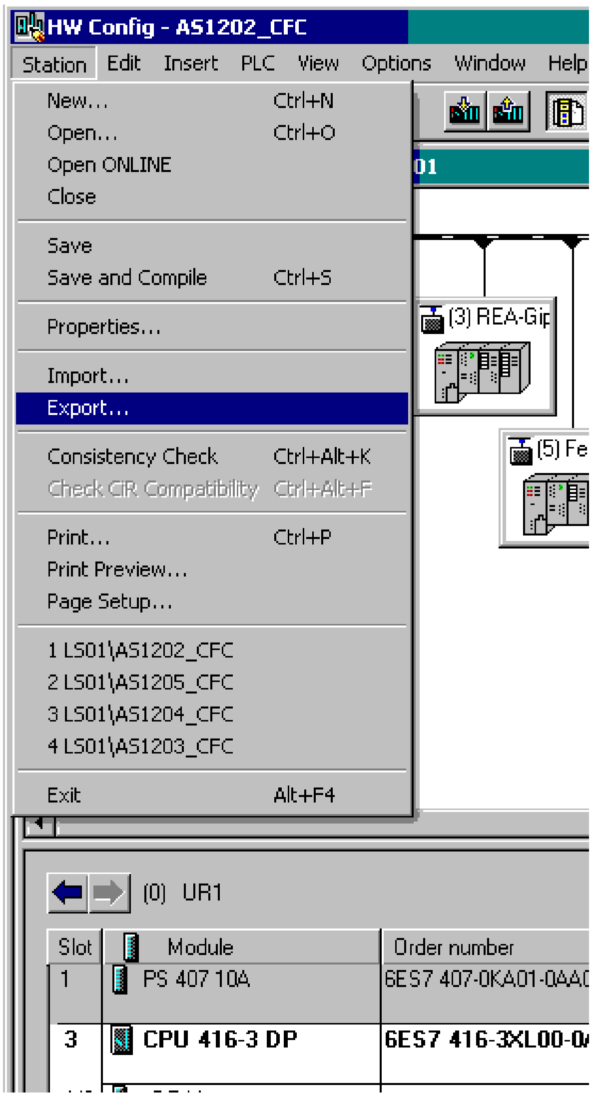
- ID -> will be numbered continuously.
- C_TAG = TAG
- GL_Complex = Complex = Manual input = identically for all objects.
- GL_Plant = Project = Manual input = identically for all objects.
- GL_Plant_zone = Plant zone = 1. Hierarchy folder.
- G_LoopA = Process mimic = from process object view „picture objects“.
- H_PLC_Name = PLC = from process object view “General” or hardware configuration.

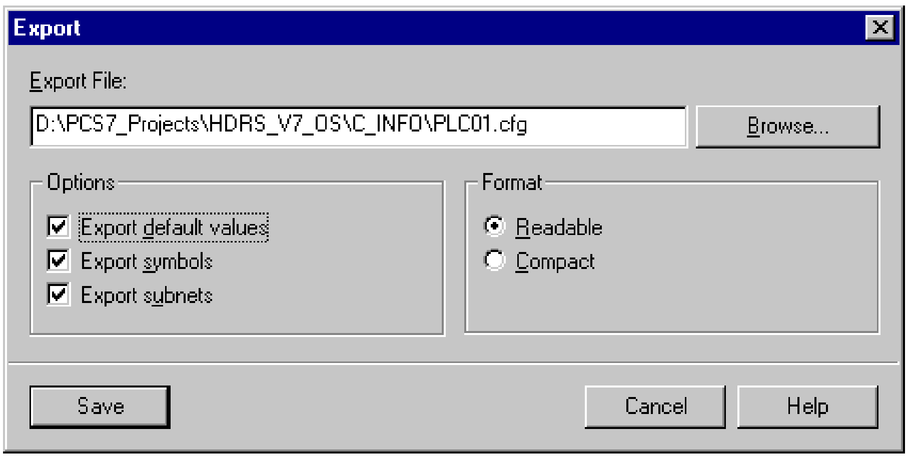
Overview



Export of the hardware configuration of the PLCs

1. Open the hardware configuration of the 1. PLC.
2. Select menu „Station -> Export“.
3. Enter export file, e. g. 1202_cfc.cfg and „Save“.

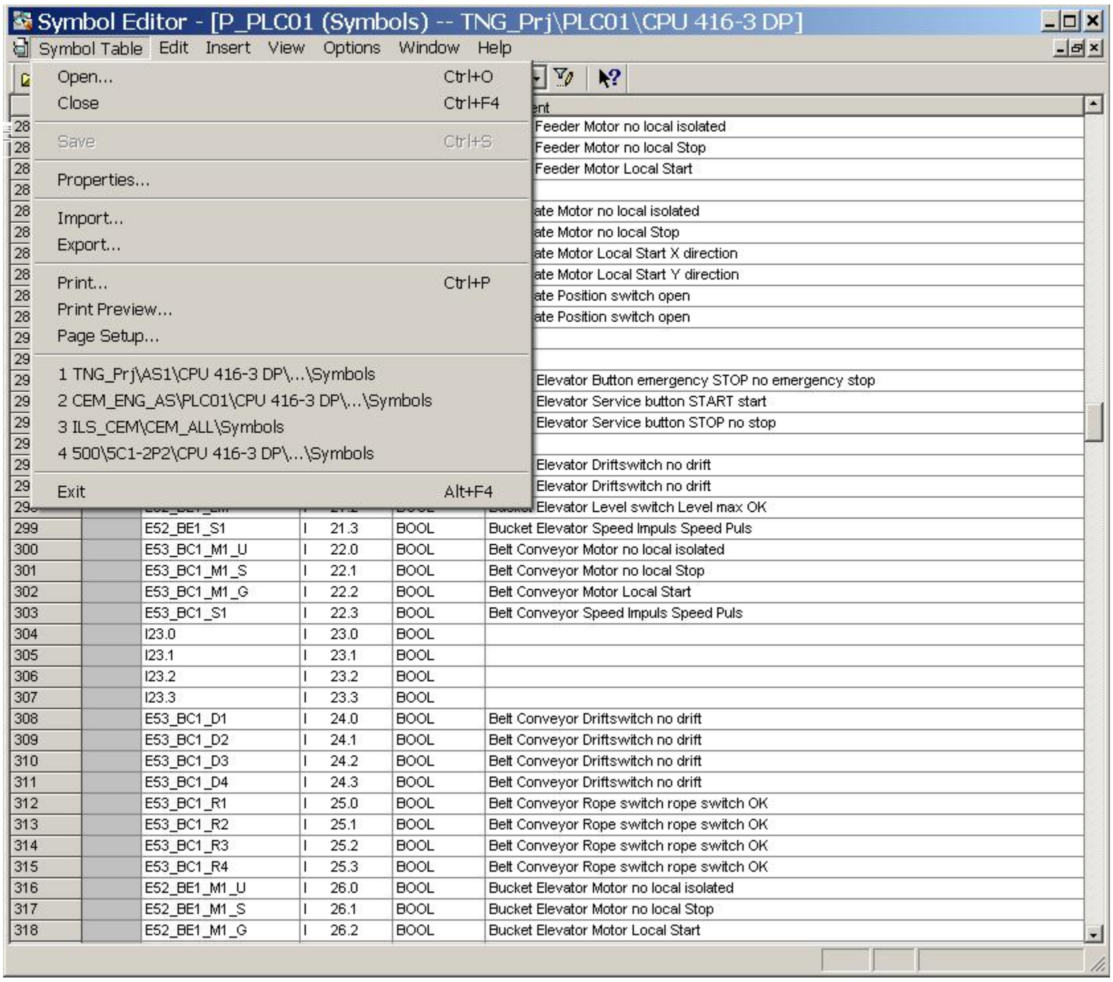




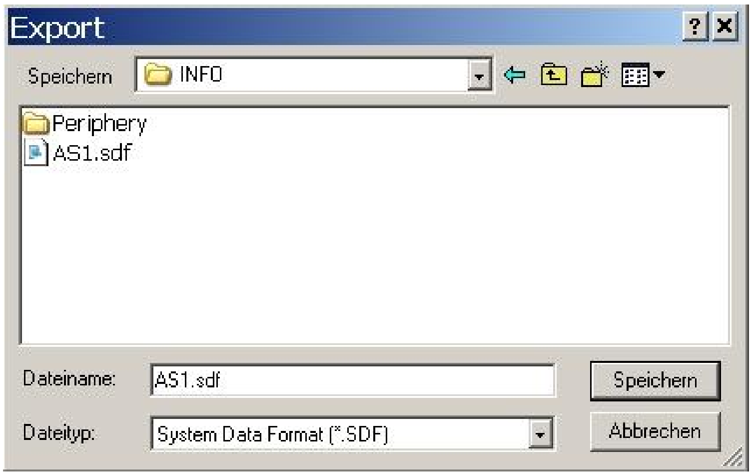
Repeat Step 1 to 3 for each PLC.

Export of the symbols file

Open the symbols table and select Symbol Table → Export…



Export the symbols in System Data Format (*.SDF)

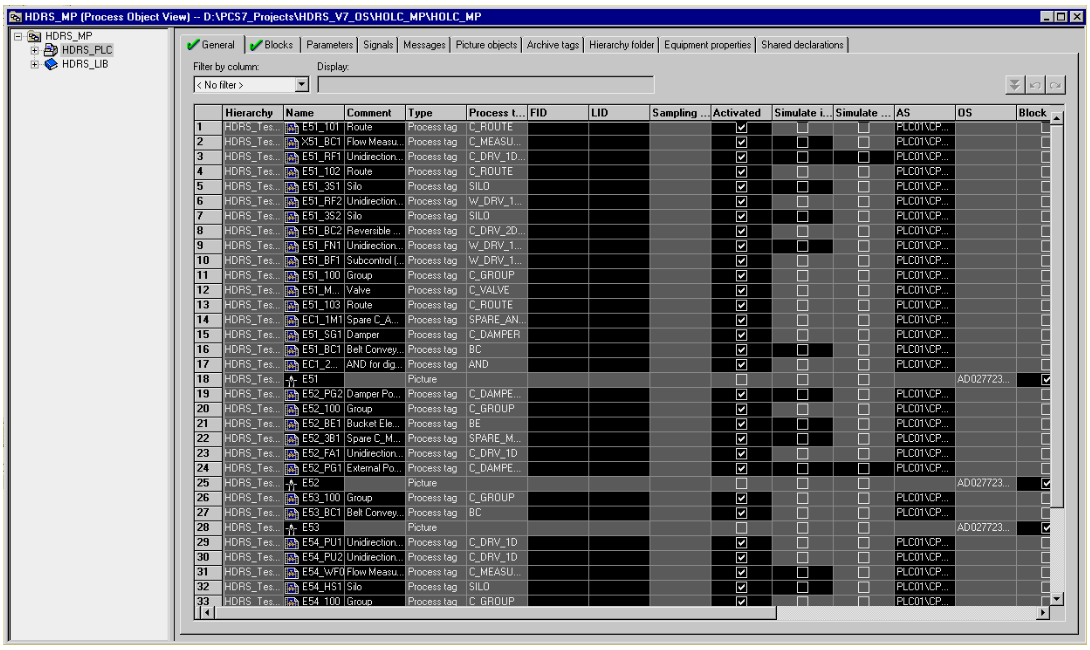


Export from the process object view

Process object view „General“

Copy the process object view „General“ as a text file, e.g. as GENERAL PO.txt.

1. Select process object view.
2. Select the topmost hierarchy folder
   (from the project with the PLCs, not the multi project folder).
3. Refresh the view (F5).
4. Select in all lines all fields, except the line numbers.
5. Copy with CTRL-C all lines into the clipboard.
6. Start the text editor (notepad) and insert all lines with CTRL-V.
7. Save the text file, e.g. as GENERAL PO.txt.

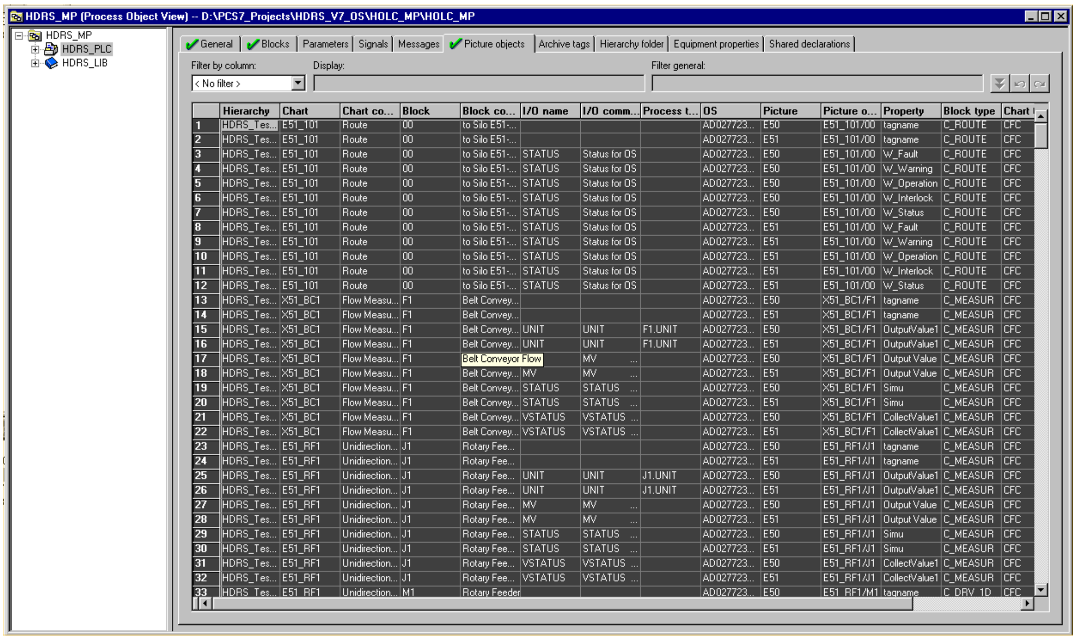


If the AS devices are stored in different single projects of a multi project, then you have to select each single project and copy the “General” view of each single project into files, e. g. GENERAL PO.txt. The specific file names must be entered in the sheet “Presettings” of the Excel tool.

Process object view „Picture objects“

Copy the process object view „picture objects“ as a text file, e.g. as PO picture.txt.

1. Select process object view “Picture objects”.
2. Select the topmost hierarchy folder
   (from the project with the PLCs, not the multi project folder).
3. Refresh the view (F5).
4. Select all lines.
5. Copy with CTRL-C all lines into the clipboard.
6. Start the text editor (notepad) and insert all lines with CTRL-V.
7. Save the text file, e.g. as PO picture.txt.

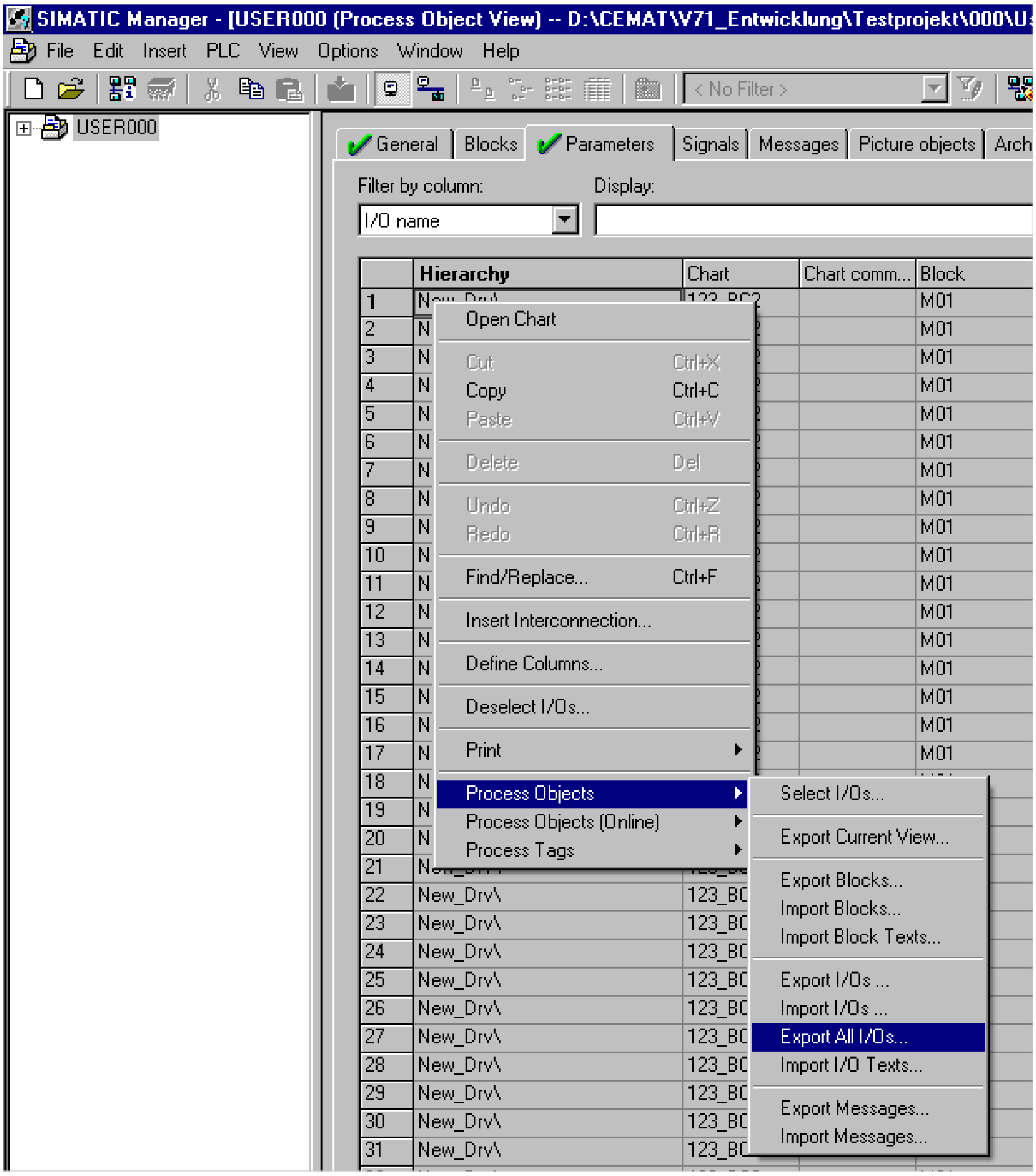


If the AS devices are stored in different single projects of a multi project, then you have to select each single project and copy the process object view “Picture objects” of each single project into files, e. g. PO_picture.txt. The specific file names must be entered in the sheet “Presettings” of the Excel tool.

Process objects view „Export I/Os“

Export the block I/Os into a text file, e.g. into LS01_PS.csv.

1. Select process object view “Parameters.
2. Select the topmost hierarchy folder
   (from the project with the PLCs, not the multi project folder).
3. Refresh the view (F5).
4. Select with the right mouse button the menu “Process objects -> Export All I/Os”.
5. Save the export file, e.g. as LS01_PS.csv.



If the AS devices are stored in different single projects of a multi project, then you have to select each single project and the process object view “Parameter” of each single project must be stored in a separate file. The specific file names must be entered in the sheet “Presettings” of the Excel tool.

Import into Excel

Open the file C_INFO.xlsm and do in the table “Presettings” the following presettings:

PO general = name of the text file with the process object view „General“.

PO export files with parameter = name of the text files with the I/O export from the object process view “Parameter”.

Names of the files with hardware config = export files with the hardware configuration of each PLC.

Files with symbol list = export files with the symbol tables of each PLC.

ID start number = start number for the data records

List separator = list separator for the CSV files „;“ for German MS Windows or „,“ for English MS Windows (refer to regional settings in control panel).

Language = D or E.

Complex = same for all objects.

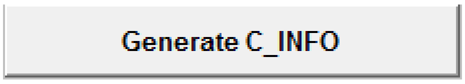
Project = same for all objects

PO picture objects = name of the text file with the process object view „picture objects“.

	1st AS	**2nd AS**	**3rd AS**	**4th AS**
Process Object General	Crusher_GE.txt	Handling_GE.txt	Rawmill_GE.txt	Blending_GE.txt
Process Object All IOs	CRUSHER_PS.txt	HANDLING_PS.txt	RAWMILL_PS.txt	BLENDING_PS.txt
AS Hardware Configuration	crusher.cfg	handling.cfg	rawmill.cfg	blending.cfg
AS Symbol List	Crusher.sdf	Handling.sdf	Rawmill.sdf	Blending.sdf
Process Objects Picture Objects	Crusher_PIC.txt	Handling_PIC.txt	Rawmill_PIC.txt	Blending_PIC.txt
Project wide settings:				
Archive - Start No	1			
List Separator	;			
Language	E			
Complex	Cemat plant			
Project	Cemat			

Copy all export files into the same directory like the Excel file C_INFO.xlsm.

If all necessary files are present, you can start the Excel macro „CINFO“ with the button “Generate C_INFO”.



Wait until "Ready" appears again in the status bar of Excel. 

For a complete line the macro could run very long., maybe up to one hour runtime. "Excel 2003" is limited to 65535 rows. If there are many objects in your project, it could be that the Excel macro stops by a fault because of this limit. In this case we recommend to use "Excel 2010". Store the XLS file as XLSM file. **Don't** run the file in the "compatibility mode". In the "compatibility mode" there is still the limit of 65535 rows!

Check:

- The file „C_INFO.CSV“ should be created by the macro. Attention: An existing file C_INFO.CSV will be overwritten without a query. Import this file as runtime data with the user archive editor from WinCC. A detailed description of the Runtime data import you will find in chapter 08_OS_Engineering.
- For every tag there should be created a text file in the subdirectory “Periphery”. These text files contain the I/O information of the tag. Also these files will be overwritten without a query.
  Copy these files to D:\CEMAT_CS\MULTIMEDIA\Periphery\. In this case, the I/O information will be shown in the object faceplate “Info” -> I/O Info”.

Edit and / or convert the User Archive C_INFO

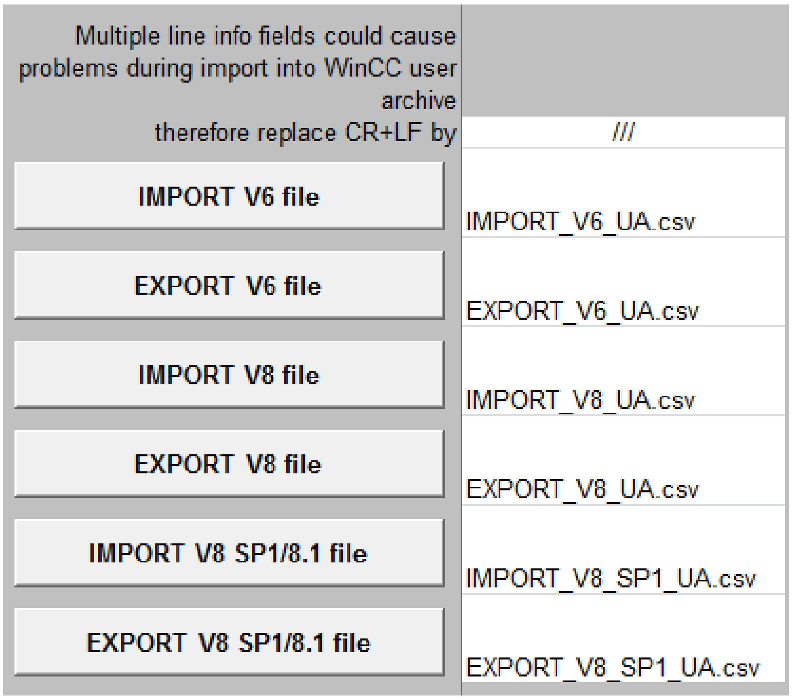
The following Macros of C_INFO.xlsm can be used in order to convert the user Archive data of C_INFO from an older Cemat Version into the structure of the latest version
or
in order to edit/complete the user archive data with Excel. With this method a proper file structure is guaranteed.

Working method:

1. Export the Runtime data into a CSV file
2. Import the CSV file into C_INFO.xlsm. After the import the data will be in sheet “UA”.
3. Edit/complete the User archive data in Excel.
4. Export the data into a CSV file.
5. Delete the existing data sets in C_INFO; otherwise the datasets cannot be imported.
6. Import the modified Runtime data into with User Archive Editor.

Under “Presettings” you have to enter the file name of the CSV file beside the corresponding start button.

The files must be copied into the same folder as C_INFO.xlsm.



<u>For importing the Runtime data into Excel tool the following Macros exist:</u>

IMPORT V6 file: Import the runtime data of C_INFO in V6 (or V7) Structure
IMPORT V8 file: Import the runtime data of C_INFO in V8 Structure
IMPORT V8 SP1 file: Import the runtime data of C_INFO in V8 SP1 Structure

Enter the file name of the CSV file in the corresponding line under “Presettings” and press the IMPORT button. Wait until “Ready” appears again in the status bar of Excel.

The user archive data of "C_INFO" can now be edited in the sheet "UA".

<u>For exporting the Runtime data the following Macros exist:</u>

EXPORT V6 file: Export the C_INFO data in V6 (or V7) Structure after editing
EXPORT V8 file: Export the C_INFO data in V8 Structure after editing
EXPORT V8 SP1 file: Export the C_INFO data in V8 SP1 Structure after editing

Enter the file name of the CSV file in the corresponding line under “Presettings” and press the EXPORT button. Wait until “Ready” appears again in the status bar of Excel.

Existing datasets in the WinCC User Archive cannot be overwritten. Before importing the data again you have to delete the existing datasets.

Cemat CFC Engineering Tool

Function description

Requirement

The PCS 7 license "SIMATIC PCS 7 Import-Export Assistant V8.2" must be installed on the Engineering Station. The user has to know, how to create "Process Tag Types", how IEA files are build up and how to import IEA files to create CFC charts.

The new Cemat Engineering tool works only after installation of PCS 7 V 8.2.

"Excel 2003" is limited to 65535 rows. If there are many objects in your project, it could be that the Excel macro stops by a fault because of this limit. In this case we recommend using "Excel 2010". Store the XLS file as XLSM file. **Don't** run the file in the "compatibility mode". In the "compatibility mode" there is still the limit of 65535 rows!

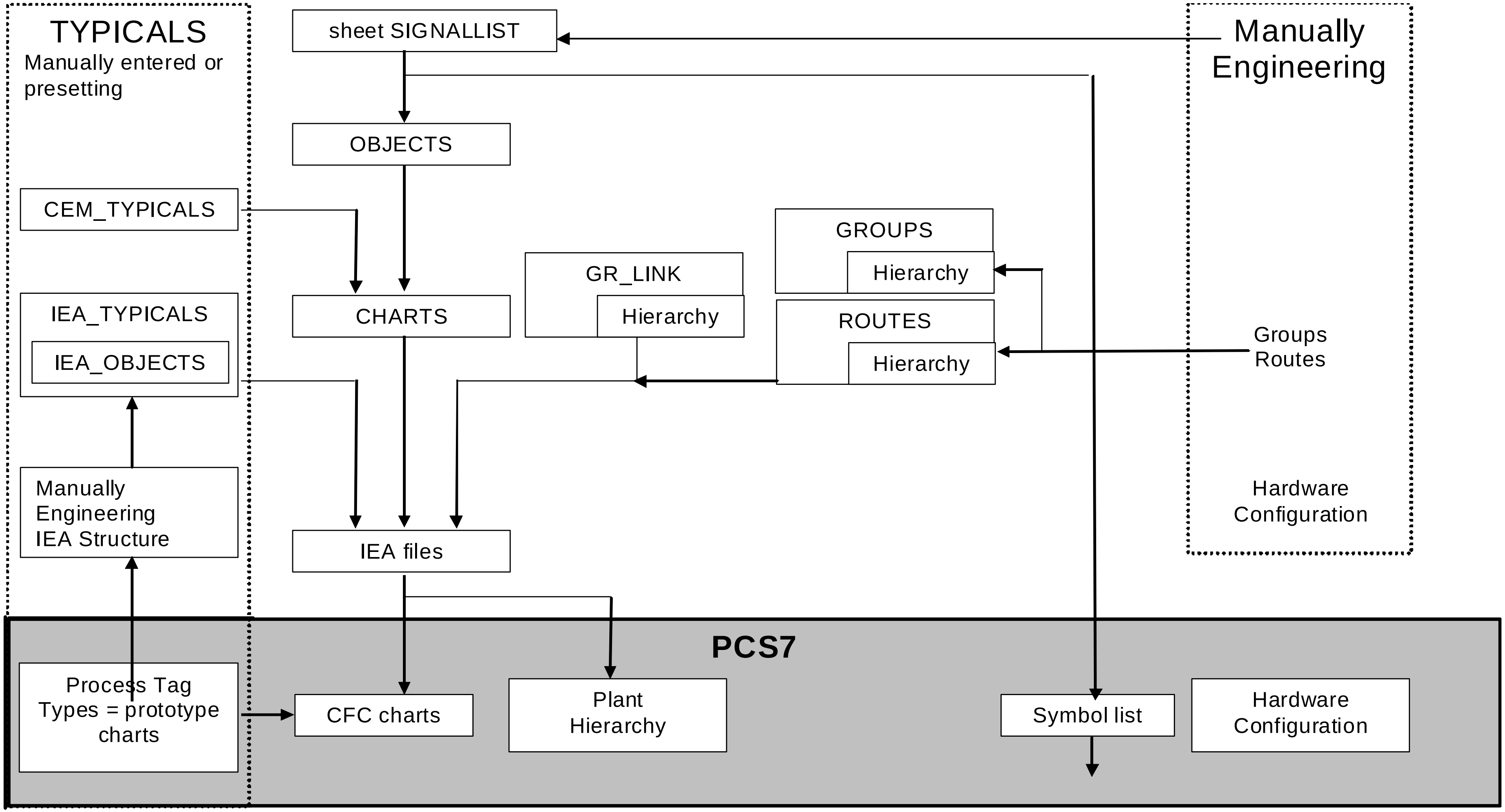
Functions

The user has to fill the sheet "SIGNALLIST" with the signals of a PLC. The user can either use the existing Process Tag Types (Typicals, prototype charts) or better, he creates with PCS 7 Process Tag Types, which fit better to the requirements of his plant.

- The Cemat Engineering Tool creates from the signal list (sheet "SIGNALLIST") a symbol table for Simatic program (SDF file). This SDF file can be imported into the Symbol editor.
- The Cemat Engineering Tool creates from the sheet "SIGNALLIST" an object list (block list).
- The Cemat Engineering Tool creates from the object list a CFC chart list.
  Sheet CEM_TYPICALS contains a description of predefined Process Tag Types (prototype charts). The user can add the description of own designed Process Tag Types to this sheet.
  The Cemat Engineering Tool first determines by chart name, which objects (blocks) shall be placed in the same chart and it tries to find the most suitable Process Tag Type (the Process tag type which has as many as possible of the required objects).
  Of course, not for every chart, which has to be created, a perfect Process Tag Type exists. For some charts the Engineering Tool chooses Process Tag Types which contain too many objects. In this case the objects which are not needed must be deleted manually afterwards.
  In some cases more than one Process Tag types is needed to cover the function, which means more than one chart will be created. The objects of these charts must later on be manually moved together into one chart.

The operating principle of the Cemat Engineering Tool you find on the following page.

Operating principle



Short summary of the engineering sequence

1. Create a PCS 7 project according to the Cemat guidelines. See Cemat Engineering manual 03_PCS7_Project_009.pdf.
2. Select the Cemat Project standard, e. g. "000 Cemat".
3. Copy the delivered Process Tag Types into the Master Data Library of your PCS 7 multi project. Take care, that you use the correct Cemat project standard! The block types in your project, the Master Library and the delivered libraries must be identical. If necessary update in all S7 programs the block types with the latest version. You can only copy the Process Tag Types and generate charts, if you use the identical block types in all block and chart containers!
4. If necessary, create your own Process Tag Types (prototype charts).
5. Create your signal list. See the structure of the signal list on page 25. One line for each hardware signal.
6. Macro: Create Symbol list (SDF file), (Button "Create symbol list").
7. Macro: Create object list (block list), (Button "Create Cemat object list").
   An Excel macro creates from the signal list an object list. Each block call (motor, measured value, annunciation, etc.) in the CFC corresponds to one line in the Excel sheet OBJECTS.
8. Macro: Create CFC chart list (Button "Create Cemat chart list").
   In the object list it is defined, which objects have to be placed together on one chart. The macro checks which process tag types are available and it tries to find the most suitable process tag type for the required chart.

   If the macro doesn't find an ideal Process Tag Types, the function is spit into several process tag types; i. e. more than one chart will be created. The content of the additional charts must later be copied manually into the main chart.

   - Example: A chart E51_BC1 should be created. The tool creates e. g. the charts E51_BC1, E51_BC1_2, E51_BC1_3 and E51_BC1_4. The block calls from the charts E51_BC1_2, E51_BC1_3 and E51_BC1_4 have to be moved manually into the chart E51_BC1. Afterwards the empty charts E51_BC1_2, E51_BC1_3 and E51_BC1_4 can be deleted.

   It may also happen that the macro chooses a Process Tag Type, which contains more objects than needed for the function. The unnecessary block calls have to be deleted later on manually.

   - Example: A belt conveyor needs two C_ANNUNC blocks as rope switches. The most suitable a Process Tag Type has four C_ANNUNC blocks for rope switch annunciations. The two unnecessary C_ANNUNC block call have to be deleted manually afterwards.

   The CFC chart list is the sheet CHARTS. Each CFC chart corresponds to one line in the Excel sheet CHARTS.
9. In the sheet OBJECTS, only the block types are listed which are connected to hardware signals. The lists of groups, routes and selections have to be created manually. In the sheet GROUPS you have to list all group modules, in the sheet ROUTES you have to list all route modules and in the sheet SELECT you have to list all selection modules.

10. Each chart has to be assigned at least to one group and / or one route. These assignments have to be done in the sheet GR_LINK. With a macro the chart list from CHARTS can be taken over into the sheet GR_LINK. For charts, which are split first into several charts, e. g. E51_BC1, E51_BC1_2, E51_BC1_3 and E51_BC1_4, only the main chart has to be entered in GR_LINK, e. g. E51_BC1. In the columns "Group 1", "Route 1", "Group 2" and "Route 2" you have to enter manually the assigned groups and routes. In the column "Hierarchy" you have to enter the assigned hierarchy folder.
11. Macro: "Check chart list" (Button "Check chart list").
    This macro checks, if each chart from CHARTS is listed in GR_LINK and if the charts are assigned at least to one group.
    The macro colorizes the sheet CHARTS in order to show the user whether the existing process tag types cover all the functions, or if it is better to design some more suitable process tag types.
    Each chart gets alternating another color. Charts, which are split into several charts, e. g. E51_BC1, E51_BC1_2, E51_BC1_3 and E51_BC1_4, get the same color.
    Objects, which have to be deleted afterwards, like e. g. not needed rope switches, will be yellow.
    Create new Process Tag Types if needed. According to the new Process Tag Types extend the sheets CEM_TYPICALS, IEA_TYPICALS and IEA_OBJECTS. The new entries in the sheets CEM_TYPICALS, IEA_TYPICALS and IEA_OBJECTS can be checked with the macro "Check typicals and IEA structure".
    Are your new typicals OK, create the sheet CHARTS again new -> see point 8.
12. Macro: "Create IEA files" (Button "Create IEA files").
    First of all, the macro creates for all entries in sheet IEA_TYPICALS an IEA file, which only consists of 4 header lines. After this, for all entries in the sheets CHARTS, GROUPS, ROUTES and SELECT one line is attached in the corresponding IEA file.
    E. g. the Process Tag Type MOT_2PI_2PIA is chosen for the chart E51_BC1. After the macro has created file MOT_2PI_2PIA00.IEA with the 4 header lines, one more line with the data for chart E51_BC1 will be attached to MOT_2PI_2PIA00.IEA.
13. Please check: Are the newest Cemat function blocks in the AS project and in the Master Data Library? Are the function blocks identical in the AS project and in the Master Data Library? Are the block types updated in the chart folders (Options -> Block Types…)? You can only generated charts in the next step, if you have in all block and chart containers the identical block types!
14. Create CFC charts (Import of the IEA files with PCS 7)
    Copy the IEA files, which you create in point 12, into the "Global" directory of the Master Data Library. Each Process Tag Type has to be assigned with his corresponding IEA file (See IEA PCS 7 manual). If all Process Tag Type are assigned with an IEA file, the import of the Process Tag Types can be started. During the import, the CFC charts will be created.
15. Completing the CFCs:
    For not used Process Tag Types the Cemat Engineering tool creates a dummy chart. Theses dummy charts are created in the hierarchy folder "Delete". Delete this hierarchy folder "Delete" including the dummy charts.

    Not all objects of generated charts are needed. For some charts, Process Tag Types are used which contain unnecessary block calls. These blocks have been renamed to "DELETE_XX", e. g. to "DELETE_D1", "DELETE_M1" or "DELETE_L1", etc. All these objects / blocks, whose name starts with the name "DELETE_ " have to be deleted. Use the Process Object View to find all objects which name start with "DELETE". Unfortunately they can not be deleted from the process object view!
    Proceed as follows: Open the Process Object View -> Select Tab "Blocks" and sort by block name. Select all blocks, which names start with "DELETE_" and choose with the right mouse button the menu "Open chart". Delete now manually in the opened charts all objects, which names start with "DELETE_".

Some blocks could not be renamed, because a block with the same name already exists in the typical. The existing block will later be renamed or deleted. The blocks, which could not be renamed to its final designation, get a temporary name "RENAME_XX".
E. g. an annunciation block with the name L1 should be created, but in the used typical chart exists already a measured value block with the name "L1". The chart will be created with the following block name: The annunciation block will get the name "RENAME_L1" and the not needed measured value block will get the name "DELETE_L1". So after deleting the not used block "DELETE_L1" the Annunciation block "RENAME_L1" can be manually renamed into "L1". Use the Process Object View to find all objects which name start with "RENAME_".

For some charts for which no suitable Process Tag Type was detected. In this case, for some charts additional process tag types are created, i. e. instead of one chart several charts will be generated. The content of these additional charts have to be copied manually later on into the main chart.
E. g. for the chart E51_BC1 the charts E51_BC1 and three additional charts E51_BC1_2, E51_BC1_3 and E51_BC1_4 are created. All objects from the extra charts E51_BC1_2, E51_BC1_3 and E51_BC1_4 have to be moved manually to the chart E51_BC1. Afterwards delete the empty extra charts (e. g. chart E51_BC1_2, E51_BC1_3 and E51_BC1_4).

16. During the IEA import, some textual interconnections are generated. With the menu "Options -> Close textual interconnections" all still existing textual interconnections will be closed.
17. After the closing of textual interconnection no more textual interconnections should remain. Please check this with the menu "Options -> Delete textual interconnections". The window with the textual interconnections should be empty. If not, this indicates a problem. Please search for the reason!
18. Runtime editor. The objects from the additional charts, e. g. E51_BC1_2, E51_BC1_3 and E51_BC1_4, are located in the runtime groups E51_BC1_2, E51_BC1_3 and E51_BC1_4. Unfortunately the objects remain in these runtime groups, even if they are moved into the main destination chart. Please move the objects manually into the runtime group of the main chart (e. g. E51_BC1). Delete afterwards the empty runtime groups of the additional charts.
19. Now the bulk engineering is over and the "real engineering" work in the CFC starts. You can add now the start interlocks, operating interlocks, etc.

Scope of delivery

The Cemat Engineering Tool is an Excel file, delivered together with 9 libraries (Process Tag Types) and 9 PSC7 example projects. The libraries and the master data libraries of the example projects contain the same Process Tag Types which are defined in the Excel tool (sheet CEM_TYPICALS). The Excel tool and the libraries are not on the Cemat DVD. The Tool and the libraries will only be delivered on demand.
Please copy the Process Tag Types from the delivered library (choose the correct project standard) into the Master Data Library of your multi project.

Assignment project standard – libraries and example projects

Project standard	Project-standard-code	ZIP file with library	ZIP file with example project
Cemat	000	CEM_ENG_LIB.ZIP	CEM_ENG_MP.ZIP
Holcim	004	HOL_ENG_LIB.ZIP	HOL_ENG_MP.ZIP
Dyckerhoff	006	DYC_ENG_LIB.ZIP	DYC_ENG_MP.ZIP
HZ	007	HZ_ENG_LIB.ZIP	HZ_ENG_MP.ZIP
Vigier	023	VIG_ENG_LIB.ZIP	VIG_ENG_MP.ZIP
Busher	024	BUS_ENG_LIB.ZIP	BUS_ENG_MP.ZIP
Caima	025	CAI_ENG_LIB.ZIP	CAI_ENG_MP.ZIP
Alsen	026	ALS_ENG_LIB.ZIP	ALS_ENG_MP.ZIP
Lafarge	027	LAF_ENG_LIB.ZIP	LAF_ENG_MP.ZIP

Attention: Create your PCS 7 project according the Cemat rules and don't use in any case one of the example project as your project. Copy only the Process Tag Types from the delivered library (choose the correct project standard) into the Master Data Library of your multi project. The example projects are incomplete. E. g. there is no system chart SYSPLC00 and therefore the PLC program is not working. Furthermore the example projects or the Process Tag Type libraries could be obsolescent regarding the blocks. It is not guaranteed that the blocks and the symbol list are up-to-date. The projects are only examples and show the working principle of the Excel tool and the Process Tag Types.

Cemat Engineering Tool

Description of the Excel Worksheets used in the Cemat Engineering Tool:

PRESETTINGS	Presettings (Cemat Project Standard, file names, PCS 7 Project names).
SIGNALLIST	Signal list. The Signal list has to be filled by the user and must contain the hardware signals of all Cemat objects.
OBJECTS	Cemat Object list. Will be created by the Engineering Tool from the signal list.
CHARTS	CFC chart list. Will be created by the Engineering Tool from the object list.
GROUPS	Group list. The group list has to be filled by the user and must contain all groups to be generated.
ROUTES	Route list. The route list has to be filled by the user and must contain all routes to be generated.
SELECT	Selection list. The selection list has to be filled by the user and must contain all all selection modules to be generated.
GR_LINK	Group / route assignment for all charts from the chart list CHARTS. The group / route assignment list has to be filled by the user.
IEA_Struc	IEA header structure for used keyword in the sheets IEA_OBJECTS xxx.
LIST_DATA	Lists of allowed signal functions, blocks, keywords, etc.
BLOCKFUNCTIONS	Lists of allowed block functions
HEADER	Header lines for the sheets "OBJECTS", "CHARTS", "GROUPS", "ROUTES", "SELECT", "GR_LINK".
IEA keyword	Allowed keywords for user function blocks (non Cemat).
IEA_Block_Names	Block names of user function blocks (non Cemat).
SIGNALTYPES 000 SIGNALTYPES 004 SIGNALTYPES 006 SIGNALTYPES 007 SIGNALTYPES 023 SIGNALTYPES 024 SIGNALTYPES 025 SIGNALTYPES 026 SIGNALTYPES 027	Allowed signal types and the relevant signal function for each block. For each Cemat project standard a corresponding signal type list exists.
CEM_TYPICALS CEM_TYPICALS_004 CEM_TYPICALS_026 CEM_TYPICALS_027	List of the existing Process Tag Types (prototype charts). The most suitable process tag type for a function is evaluated based on this sheet. For each Process Tag Type, the assigned objects (block calls) are listed together with the block name and the block function. The project standards "Holcim 004", "Alsen 026" and "Lafarge 027" use their own Process Tag Type list. The content of the sheet CEM_TYPICALS must fit to the Process Tag Types of the Master Data Library (compare definition of CEM_TYPICALS and example project of delivery state).

IEA_TYPICALS IEA_TYPICALS_004 IEA_TYPICALS_026 IEA_TYPICALS_027	Structure of the IEA file of the respective Process Tag Types. In the sheet IEA_TYPICALS, only the sequence of the blocks in the IEA file is defined. The always repeating IEA detail structures for blocks have to be defined in the sheet IEA_OBJECTS. The project standards "Holcim 004", "Alsen 026" and "Lafarge 027" use their own IEA structure list. The content of the sheet IEA_TYPICALS must fit to the Process Tag Types of the Master Data Library (compare definition of IEA_TYPICALS and example project of delivery state).
IEA_OBJECTS 000 IEA_OBJECTS 004 IEA_OBJECTS 006 IEA_OBJECTS 007 IEA_OBJECTS 023 IEA_OBJECTS 024 IEA_OBJECTS 025 IEA_OBJECTS 026 IEA_OBJECTS 027	IEA detail structures for blocks. In Process Tag Types blocks are often used several times. For these blocks always the same parameters are chosen as variable places. In this case the IEA file contains the repeated columns, e. g. for each C_ANNUNC block. Therefore the block structure is defined in a separate file and the IEA_TYPICALS contain only a reference to this. For each Cemat project standard exists an own IEA_OBJECTS sheet. The content of the sheet IEA_OBJECTS must fit to the Process Tag Types of the Master Data Library (compare definition of IEA_OBJECTS and example project of delivery state).
LOG	If an error occurs during generating of the sheets OBJECTS or CHARTS, the macro will list these errors in the sheet LOGS.

Predefined Process Tag Types (Prototype charts, Typicals)

Die following Process Tag Types are necessary for the system and it is not allowed to delete them:

AND8	AND gate with 8 inputs. All digital inputs, which can not be connected to a Cemat standard block, will be connected to the block AND8. If there is only 1 digital input, the system will use the Process tag Type BIT1.
BIT1	AND gate with 1 input. All digital inputs, which can not be connected to a Cemat standard block, will be connected to the block BIT1. If there is more than 1 digital input, the system will use the Process tag Type AND8.
OUT	AND gate with 1 variable output. All digital outputs, which can not be connected to a Cemat standard block, will be connected to the block OUT.

Die following Process Tag Types are examples and can be copied, changed or extended. According to requirements of the plant, the Process Tag Types have to be changed before creating the IEA files and the CFC charts.

All project standards except Lafarge:

DRV_1D	Unidirectional drive without additional signals
DRV_1D_SC	Unidirectional drive with Simocode, but without additional signals
DRV_2D	Bi-directional drive without additional signals.
DRV_2D_SC	Bi-directional drive with Simocode, but without additional signals
MOT_CURR	Drive with a flow measurement.
MOT_2PI_2PIA MOT_2PI_3PIA (Holcim)	Drive with 2 protection interlock annunciations and with 2 automatic mode protection interlock annunciations.
MOT_4PI_4PIA MOT_4PI_5PIA (Holcim)	Drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations.
MOT_SC_4PI_4PIA MOT_SC_4PI_5PIA (Holcim)	Simocode drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations.
MOT_2PI_2PIA_CURR MOT_2PI_3PIA_CURR (Holcim)	Drive with 2 protection interlock annunciations and with 2 automatic mode protection interlock annunciations and with motor current measurement.
MOT_4PI_4PIA_CURR MOT_4PI_4PIA_CURR (Holcim)	Drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations and with motor current measurement.
MOT_8PI_8PIA_CURR MOT_8PI_9PIA_CURR (Holcim)	Drive with 8 protection interlock annunciations and with 8 automatic mode protection interlock annunciations and with motor current measurement
MOT_MAIN	Main drive and auxiliary drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations and with motor current measurement and 1 common annunciation.

MOTR_CURR	Bi-directional drive with motor current measurement.
MOTR_2PI_2PIA MOTR_2PI_3PIA (Holcim)	Bi-directional drive with 2 protection interlock annunciations and with 2 automatic mode protection interlock annunciations.
MOTR_4PI_4PIA MOTR_4PI_5PIA (Holcim)	Bi-directional drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations.
MOTR_SC_4PI_4PIA MOTR_SC_4PI_5PIA (Holcim)	Simocode Bi-directional drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations.
MOTR_2PI_2PIA_CURR MOTR_2PI_3PIA_CURR (Holcim)	Bi-directional drive with 2 protection interlock annunciations and with 2 automatic mode protection interlock annunciations and with motor current measurement.
MOTR_4PI_4PIA_CURR MOTR_4PI_5PIA_CURR (Holcim)	Bi-directional drive with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations and with motor current measurement.
MOT_SPEED	Drive with speed measurement and set point.
MOT_SINA_GS	Drive with SINA_GS (Variable speed drive).
ANNUNC	Annunciation block.
ANNUNC2	2 Annunciation blocks.
ANNUNC6	6 Annunciation blocks.
DAMPER	Damper without additional signals.
DAMPER_SC	Simocode Damper without additional signals.
DAMPER_P	Damper with positioning function.
VALVE	Valve without additional signals.
VALVE_SC	Simocode Valve without additional signals.
VALVE_2D (only Alsen)	Bi-directional Valve without additional signals.
VALVE_2D_SC (only Alsen)	Simocode Bi-directional Simocode Valve without additional signals.
MEASUR	Measurement
MEASUR2	2 Measurements
MEASUR6	6 Measurements
ANNUNC2_MEASUR2	2 Annunciation blocks and 2 Measurements.
GROUP	Group module
ROUTE	Route module
SELECT	Selection module
AO	Channel output module with CH_AO
PROFB (only Holcim)	Process feedback block
SILOPILOT (not for Holcim)	Silo pilot
ANNUN8	ANNUN8 with 7 annunciations

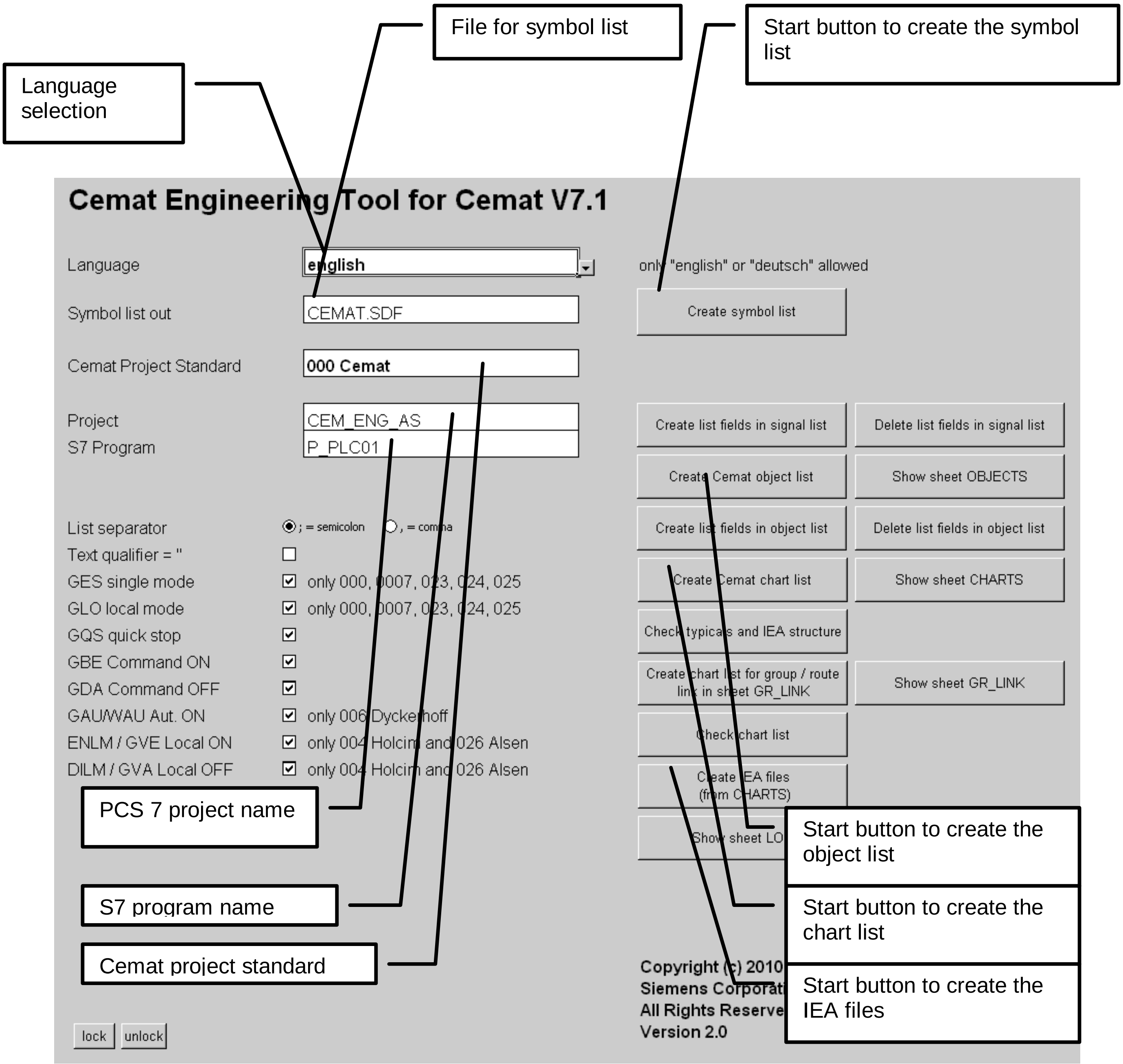
USER_BLOCK	Example for a user defined block

Lafarge:

BPB	Simple drive block
M2B	Drive without additional signals (Unidirectional or Bi-directional
M2B_CURR	Drive with motor current measurement.
M2B_2HS_2PE2	Drive with 2 emergency shut down interlock annunciations and with 2 automatic mode protection interlock annunciations.
M2B_2HS_2PE1_2PE2	Drive with 2 emergency shut down interlock annunciations and with 2 protection interlock annunciations and with 2 automatic mode protection interlocks annunciations.
M2B_2HS_2PE2_CURR	Drive with 2 emergency shut down interlock annunciations and with 2 automatic mode protection interlock annunciations and with motor current measurement.
M2B_2HS_2PE1_2PE2_CURR	Drive with 2 emergency shut down interlock annunciations and with 2 protection interlock annunciations and with 2 automatic mode protection interlock annunciations and with motor current measurement.
M2B_4HS_4PE1_4PE2_CURR	Drive with 4 emergency shut down interlock annunciations and with 4 protection interlock annunciations and with 4 automatic mode protection interlock annunciations and with motor current measurement.
DIB	Annunciation block (Digital input block)
DIB2	2 Annunciation blocks (Digital input blocks).
DIB6	6 Annunciation blocks (Digital input blocks).
DABMAB	Damper without additional signals.
AAB	Analog positioning function.
DAB	Valve without additional signals.
AIB	Measurement (Analog input block)
AIB2	2 Measurements (Analog input blocks)
AIB6	6 Measurements (Analog input blocks)
DIB2_AIB2	2 Annunciation blocks (Digital input blocks) and 2 Measurements (Analog input blocks).
SSB	Group module
SELECT	Selection module
AO	Channel output module with CH_AO
SPEEDM	Speed monitor block
SILOPILOT	Silo pilot
USER_BLOCK	Example for a user defined block

Presettings

In the sheet PRESETTINGS you have to do all predefinitions (file names, project names, etc.). There you also find the buttons for starting the macros.



Caution: The Cemat Engineering Tool and the export files must be located in the same directory.

Recommendation: The sheets "SIGNALTYPES xxx", "CEM_TYPICALS xxx", "IEA_TYPICALS xxx" and "IEA_OBJECTS xxx" exists for each Cemat project standard. Delete the unnecessary sheets, before you start with the engineering for a concrete plant. This reduces the number of sheets and gives you a much better overview.
If you work e. g. with the Cemat standard 000, then it is enough if you keep only the sheets "SIGNALTYPES 000", "CEM_TYPICALS", "IEA_TYPICALS" and "IEA_OBJECTS 000".

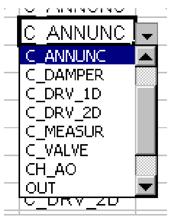
Structure of the signal list

Field	Comment	Allowed values
Signal name	Signal description	Text
IOType	IO Type	DI, DO, AI, AO
ConvType	Signal conversion type	NO (DI, normally open) NC (DI normally closed) Puls (DI pulse) 0-20mA, 4-20mA, 0-10V, PT100 (AI / AO analogue signal) DB (DI, DO, AI, AO virtual signal from a sub system)
SignalType	Signal type	2w (two wiring) or 4w (four wiring)
SignalLocation	Signal location	Field, MCC
IOAddress	Absolute address	E. g. I 4.0
SignalText	Signal text	Text
ChartName	CFC chart name	Text, follow the PCS 7 rules
ObjectName	Block name	Text, follow the PCS 7 rules
BlockComment	Block comment	Text, follow the PCS 7 rules
EventText	Event text	Text, follow the PCS 7 rules
SignalFunction	Signal function	e. g. feedback, allowed function see in the following pages
ObjectType	PCS 7 block type	e. g. C_DRV_1D
RangeMin	Measuring range minimum	Real value
RangeMax	Measuring range maximum	Real value, has to be bigger than RangeMin
Unit	Measuring unit	Text
AlarmType	Alarm type Alarm or Warning	WA = warning, all other values are an alarm

In the field "IOType", "ConvType", "SignalType", "SignalLocation", "SignalFunction" and "ObjectType" only specific entries are permitted.

With the button "Create list fields in signal list" 

the cells in the Excel sheet will be changed into Excel list fields. With the list fields you can enter only permitted values.

Example list field for the block type: 

With the button "Delete list fields in signal list" 

the list fields will be changed back to normal Excel cells, i. e. now all values could be entered.

Signal functions

Function name	Function description
feedback	Feedback ON relay, e. g. ERM, RME, R, B1
feedback1	Feedback ON direction 1 relay, e. g. ERM1, RME1, RX, B1
feedback2	Feedback ON direction 2 relay, e. g. ERM2, RME2, RY, B2
limitpos1	Limit position 1, e. g. KWE1, ESL, VE1, Y, BP1
limitpos2	Limit position 2, e. g. KWE2, ESR, VE2, X, BP2
available	Electrical ready, available, e. g. ESB, K, AV
overload	Bimetal, mechanical fault, e. g. EBM, MSB, T, SMUE
local	Local switch, e. g. EVO, U
locstart	Local start, e. g. ESR, TSE, G, L1
locstart1	Local start direction 1, e. g. ESR1, TSE1, GY, L1
locstart2	Local start direction 2, e. g. ESR2, TSE2, GX, L2
locstop	Local Stop, e. g. ESP, S, LS
torque1	Torque switch direction 1, e. g. KDR1, TY, PE1
torque2	Torque switch direction 2, e. g. KDR2, TX, PE2
speedmon	Speed monitor, e. g. EDRW, SW_SPEED, IN_SIG, PULS_SSM, BP, PULS
ON	Command ON, e. g. EBE, SEE, D, C1
ON1	Command ON direction 1, e. g. EBE1, SEE1, DY, C1
ON2	Command ON direction 2, e. g. EBE2, SEE2, DX, C2
alarm	Alarm signal MST0 for C_ANNUNC or DIG for C_DIB
operation	Alarm signal MST0 for C_ANNUNC or DIG for C_DIB (Operation interlock drive = EBVG and interlock block)
protection	Alarm signal MST0 for C_ANNUNC or DIG for C_DIB (Protection interlock drive = ESVG, PINT1, PE1 and interlock block)
shutdown	Alarm signal DIG for C_DIB ("Emergency shut down" drive HS and interlock block) (only Lafarge)
autoprotection	Alarm signal MST0 for C_ANNUNC or DIG for C_DIB (Protection interlock drive in automatic mode = ESVA, PINT2, PE2 and interlock block)
fault1	Fault 1 for C_ANNUN8 = FLS1
fault2	Fault 2 for C_ANNUN8 = FLS2
fault3	Fault 3 for C_ANNUN8 = FLS3
fault4	Fault 4 for C_ANNUN8 = FLS4
fault5	Fault 5 for C_ANNUN8 = FLS5
fault6	Fault 6 for C_ANNUN8 = FLS6
fault7	Fault 7 for C_ANNUN8 = FLS7
analogvalue	Analogue input MV_CARD for C_MEASUR, AI for C_AIB, VALUE for CH_AI or VALUE for SINA_GS, etc.
actposition	Position value AMV for C_AAB (only Lafarge)
AO	Analogue output VALUE for CH_AO
BIT	Common digital input signal, will be connected to an AND gate
OUT	Common digital output signal, will be connected to the output of an AND gate
beltbreak	Belt break silo pilot
limitpos	Upper limit position silo pilot
puls	Input pulses silo pilot
Simocode_I	Simocode start address input area
Simocode_Q	Simocode start address output area

Creating Process Tag Types (Typicals) in PCS 7

If you want to use the delivered Process Tag Types and don't create your own Process Tag Types, you can jump directly to the chapter "Create the symbol list".

Typical types in PCS 7

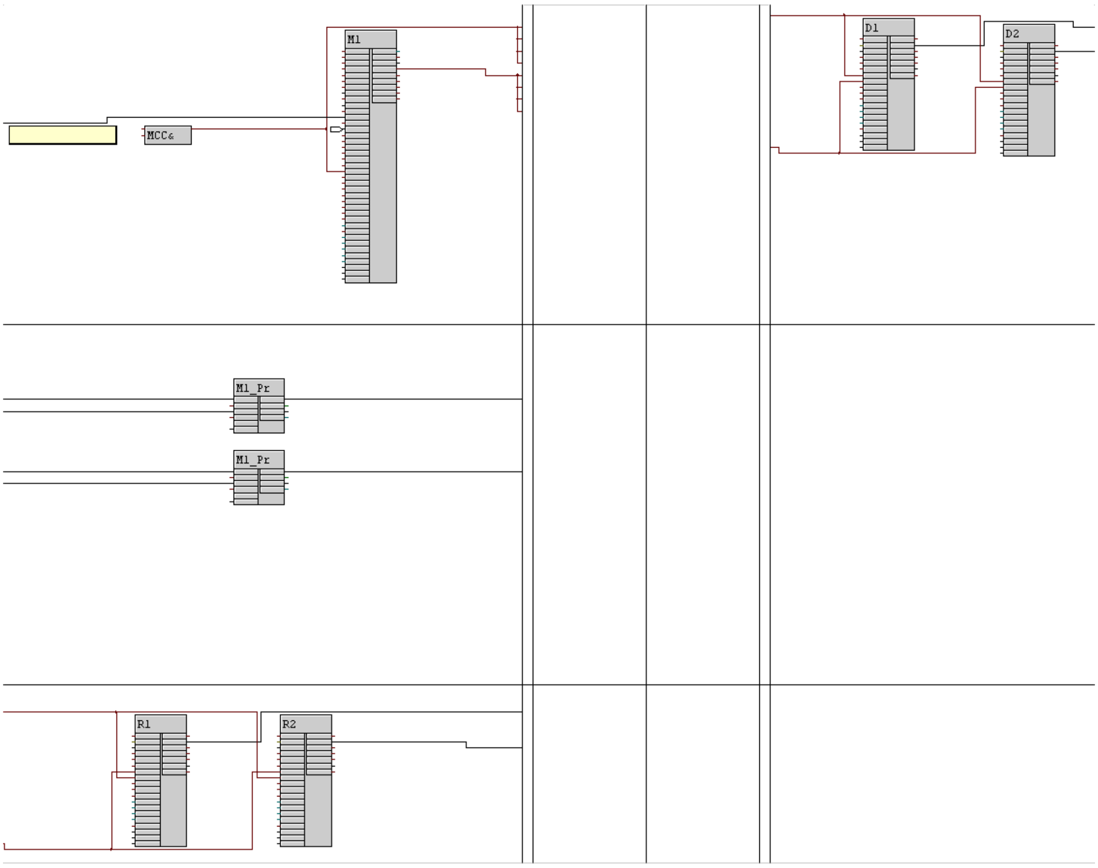
In PCS 7 you can define two types of typicals:
1. Models: A PCS 7 Model could consist of several charts and several hierarchy folders.
2. Process Tag Types: A Process Tag Types consist of only one chart.

The Cemat Engineering Tool uses only Process Tag Types!

Create a Process Tag Types in PCS 7

Process Tag Types can only be created in a PCS 7 multiproject with a Master Data Library (see PCS 7 manual). In a PCS 7 project or in a normal PCS 7 library you can't create Process Tag Types. The Master Data Library contains the Process Tag Types for the multiproject. The Process Tag Types are templates. Based on these templates CFC charts will be generated.

Example: Drive with 2 protection interlock annunciations and with 2 automatic mode protection interlock annunciations = CFC chart MOT_2PI_2PIA.



The example Process Tag Type MOT_2PI_2PIA consists of the following objects:
M1 = main drive = Cemat bock C_DRV_1D
M1_ProtG = interlock block for protection = PCS 7 block Intlk02
M1_ProtA = interlock block for automatic mode protection = PCS 7 block Intlk02
D1 = drift switch = Cemat block C_ANNUNC
D2 = drift switch = Cemat block C_ANNUNC
R1 = rope switch = Cemat block C_ANNUNC
R2 = rope switch = Cemat block C_ANNUNC

Based on this Process Tag Type MOT_2PI_2PIA, multiple copies of this chart be generated, e. g. E51_BC1, E51_BC2, etc.
The block names M1, D1, R1, etc. are variable and can be changed during the chart generation. E. g. instead of a "D2", a "D3" has to be generated. In this case, the block "D2" will be renamed during chart generating into "D3".

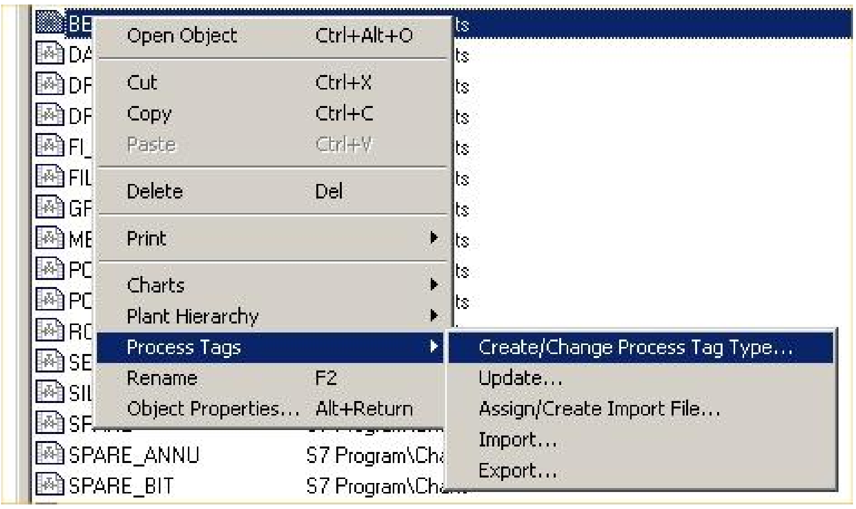
Editing a Process Tag Types

In your Master Data Library, you first of all need a hierarchy folder, where the Process Tag Types of your project are stored. You should use a folder name like "Process Tag Types". In this folder you first create a normal CFC chart. This chart must contain all blocks for the new process tag type, including all static interconnections, as for example the links from the C_ANNUNC blocks to the interlock blocks, from the interlock blocks to the drives, from the motor current measurements to the drives, etc. Only the variable connections will not be created.
Variable interconnections will be created during the CFC chart generation. (E. g. the links between drive blocks to the group). In the next step you have to select all spots (parameters), which are variable and should be changed for every instance of the chart.

Example:

- Block parameter, which will be connected to symbols from the symbol list.
- Block parameter, which will be connected to another block (select only the input parameter!)
- Block parameter, which values will be adapted, e. g. range min and range max of a measured value, etc.

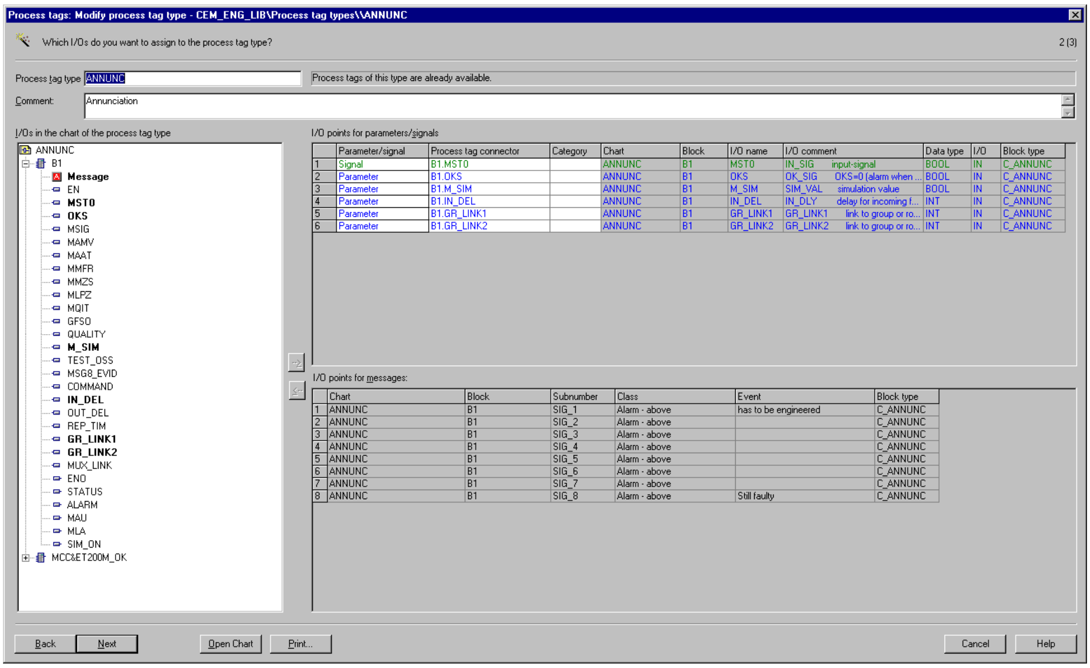
To create a Process Tag Type, you have to select the CFC chart in the Master Data Library (Plant View) and with the right mouse button you call the menu: Process Tags -> Create/Change Process Tag Type…



Confirm the following window with "Next". In the next window you see on the left side a list with all blocks and all parameters of the blocks of the chart. In this list you have to select these parameters, which are variable and which interconnection or value shall be adapted to the IEA file.
Select the parameter with a double click or select the parameter and click on the arrow button.

Example: In the following screen shot, you see the Process Tag Type ANNUNC. For C_ANNUNC are the following parameters are selected:
MST0 – on this parameter, the symbol from the symbol list should be connected.
OKS – this parameter gets a 0 or a1, depending on "normally closed" or "normally open".
M_SIM – this parameter gets a 0 or a1, depending on "normally closed" or "normally open".
IN_DEL – this parameter gets the default value 0, but to the "Event text" will be written to the property "Identifier".
GR_LINK1 – this parameter gets the textual interconnection to the first group / route.
GR_LINK2 – this parameter gets the textual interconnection to the second group / route.

If you like to change also the alarm texts during generating of the charts, you have to select also the parameter "message". In this way all variable places of the 8 messages of the ALARM8 block can be modified via IEA file.
In Cemat this will be used to enter the "event text" and the "TAG comment"



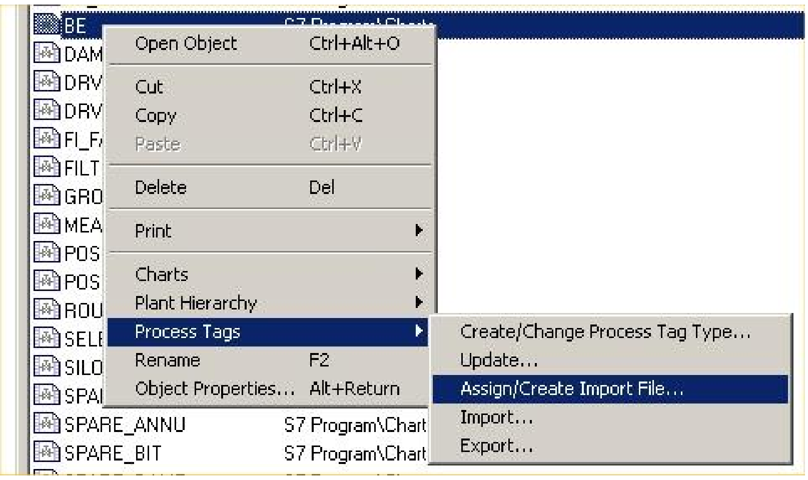
Create a template IEA file for a Process Tag Type

After all variable parameters have been selected the IEA file has to be created. The structure of an IEA file is similar to a CSV file. As list separator the list separator from the regional settings (Control Panel) is used, else the semicolon ";" or the comma",". The corresponding list separator could be set in the sheet PRESETTINGS. The data field between the list separators is always limited by the text limiter """. An IEA file has always 4 header lines. The first header line is a comment line. In the next 3 header lines, there is a description of the variable places of a Process Tag Type (see chapter before "Create a Process Tag Types in PCS 7").
The Engineering Tool will later add one line for each chart, which has to be generated. The corresponding columns contain the variable data (symbol names, values, message texts, textual interconnections, etc.).
In the Cemat Engineering Tool you have to define, which Process Tag Types exists and how the IEA file structure for the Process Tag Types is built.

You can create a template IEA file with the Import/Export Assistant of PCS 7. Based on this template you can parameterize the Cemat Engineering Tool.

Create a template IEA file:

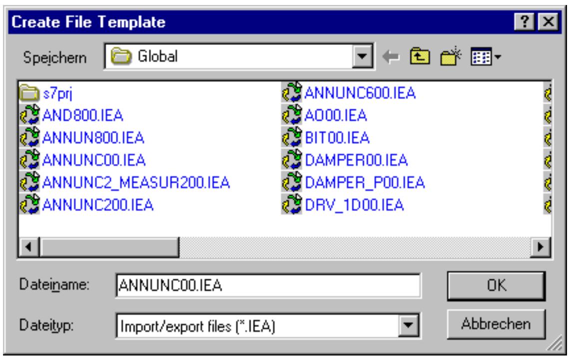
Select the CFC chart in the Master Data Library (Plant View) and with the right mouse button call the menu: Process Tags → Assign/Create Import File…



Confirm the following window with "Next".

In the next window you select the button "Create File Template…" .

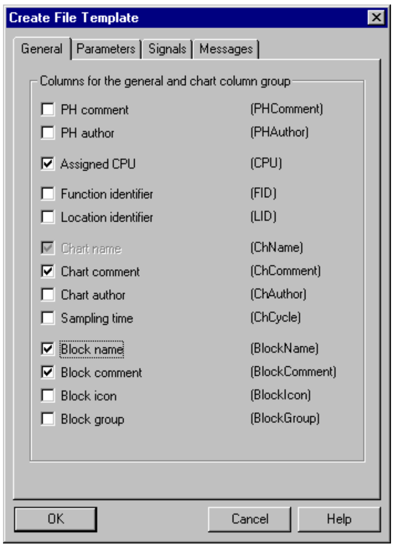
In the next box you have to enter the file name and the file location. We recommend accepting the default settings: File name = "Name of the Process Tag Types & 00.IEA". Location = directory of the Master Data Library, Subdirectory "Global".



After the confirmation of the file name with "OK" a new window opens. Please carry out the following settings, in order to get a template file with the same file structure, which is created by the Cemat Engineering Tool:

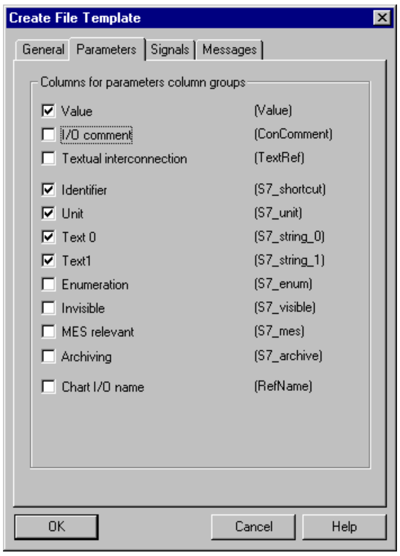
Select in tab General:

- Assigned CPU
- Chart comment
- Block name
- Block comment



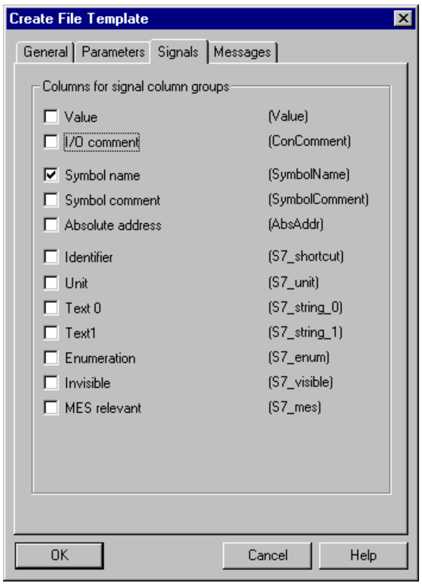
Select in tab Parameters:

- Value
- Identifier
- Unit
- Text 0
- Text 1



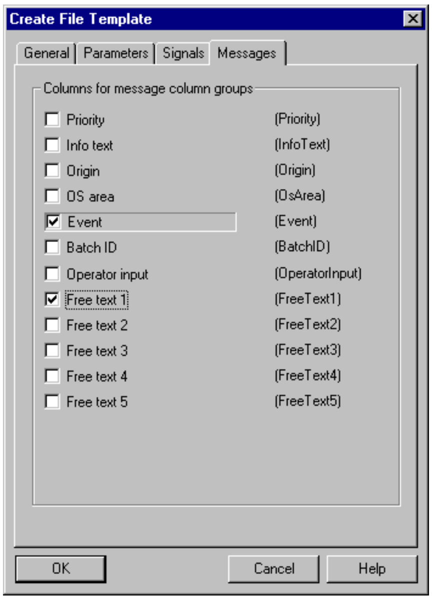
Select in tab Signals:

- Symbol name



Select in tab Messages:

- Event
- Free text 1



Confirm all tabs with "OK". In the specified directory you will find the new template IEA file. The IEA file can be displayed and edited with the PCS 7 IEA File Editor.

⚠ **Attention:** According to this generated template IEA file you have to design the entries for this Process Tag Type. The parameterization is done in sheets CEM_TYPICALS, IEA_TYPICALS and IEA_OBJECTS of the Cemat Engineering Tool. See also example of the delivered Process Tag Types.

Definition of PCS 7 Process Tag Types in the Cemat Engineering Tool

After creating new Process Tag Types in PCS 7, you have to define them also in the Cemat Engineering Tool (to be considered in the chart recognition and the generation of the IEA files). For the definition you have to modify 3 sheets:

1. CEM_TYPICALS
   In this sheet, all existing objects in the Process Tag Type are listed. For each block, you have to define the block name, the block type and the function.

2. IEA_TYPICALS and IEA_OBJECTS
   In these two sheets, the structure of the corresponding IEA files is defined. If the Process Tag Type has more than one block of the same type (several C_ANNUNC blocks), the IEA structure definition of each block type is repeating.
   In order to avoid repetitions, the IEA_TYPICALS describes only the block sequence within the IEA file and it has a reference to a detailed description in IEA_OBJECTS.
   In IEA_OBJECTS the detailed IEA file structure for an individual block is described

Details see in the next chapters.

Sheet CEM_TYPICALS

For the project standards "Holcim 004", "Alsen 026" and "Lafarge 027" please use the sheets CEM_TYPICALS_004, CEM_TYPICALS_026 or CEM_TYPICALS_027. In this sheet all existing objects in the Process Tag Type are listed. For each block, you have to define the block name, the block type and the function.

found func block	found block	No. Of del. Obj.	No. Of Obj.	Name	Obj1	Obj2	Obj3	Obj4	Obj5
				MOT_2PI_2PIA	M1	D1	D2	R1	R2
				MOT_2PI_2PIA	C_DRV_1D	C_ANNUNC	C_ANNUNC	C_ANNUNC	C_ANNUNC
				MOT_2PI_2PIA	main	autoprotection	autoprotection	protection	protection
				MOT_2PI_2PIA					

In upper sheet there is the example for the Process Tag Type MOT_2PI_2PIA. For each Process Tag Type, 4 lines are reserved in the sheet CEM_TYPICALS. The columns “A” to “D” are empty and will be used later on from the Excel macros. In the column “E” you have to enter the name of the Process Tag Type in all 4 lines (e. g. "MOT_2PI_2PIA“).
In the next columns you have to enter the objects of the Process Tag Type. In the first line you enter the name of the object, e. g. “M1“. In the second line you have to enter the block type, e. g. "C_DRV_1D". In the third line you have to enter the object function, e. g. "main" = main drive.
The example Process Tag Type "MOT_2PI_2PIA" (defined in the upper sheet) consists of the following objects:

Object name	Block type	Function	Description
M1	C_DRV_1D	Main	Main drive
D1	C_ANNUNC	autoprotection	Drift switch
D2	C_ANNUNC	autoprotection	Drift switch
R1	C_ANNUNC	protection	Rope switch
R2	C_ANNUNC	protection	Rope switch

Permitted entries in CEM_TYPICALS

Follow the PCS 7 rules and the Cemat rules for block names.

All permitted types are listed in sheet LIST_DAT, column "M" (Lafarge column "N"). The user can add new block types in the corresponding columns.

AND	Common digital input to an AND gate
C_ANNUN8	C_ANNUN8
C_ANNUNC	C_ANNUNC
C_DAMPER	C_DAMPER
C_DRV_1D	C_DRV_1D
C_DRV_2D	C_DRV_2D
C_MEASUR	C_MEASUR
C_VALVE	C_VALVE
CH_AO	PCS 7 driver block CH_AO
OUT	Common digital output, which will be connected to the output of an AND gate
C_GROUP	C_GROUP
C_ROUTE	C_ROUTE
C_SELECT	C_SELECT
C_SILOP	C_SILOP
SINA_GS	Block SINA_GS from the library SINAMICS for speed controlled drives
C_VAL_2D	C_VAL_2D
C_PROFB	C_PROFB
IEA_USER	IEA_USER = example for an user function block

Permitted functions are listed in sheet LIST_DATA, column "F".

main	Main drive
main_SC	Main drive with Simocode
aux	Auxiliary drive
aux_SC	Auxiliary drive with Simocode
alarm	Common alarm
operation	Operation message
autoprotectio n	Protection interlock alarm in automatic mode
protection	Protection interlock alarm
shutdown	"Emergency shut down" interlock alarm (only Lafarge)
analogvalue	Analogue input
drive_curr	Power or Current measurement for drive
drive_speed	Speed measurement for drive
speedmon	Speed monitor (only Holcim and Lafarge)
silo	Silopilot
annun8	7 common messages
AO	Analogue output VALUE for CH_AO
BIT	Common digital input signal, will be connected to an AND gate
OUT	Common digital output signal, will be connected to the output of an AND gate

For each new Process Tag Type you have to add 4 lines in the sheet CEM_TYPICALS. Please refer to the entries for already existing Process Tag Types and use them as examples.

Sheet IEA_TYPICALS

For the project standards "Holcim 004", "Alsen 026" and "Lafarge 027", please use the sheets IEA_TYPICALS_004, IEA_TYPICALS_026 or IEA_TYPICALS_027. Sheets IEA_TYPICALS and IEA_OBJECTS contain a description of the IEA files for the Process TAG Types. Based on these structures the Cemat Engineering Tool generates the IEA files for the existing Process Tag Types.
Normally for the same block types (C_DRV_1D, C_ANNUNC, etc.) you will select always the same parameter /signals as variable places in a Process Tag Types (e. g. the connectors MST0, OKS, M_SIM, IN_DEL, GR_LINK1, GR_LINK2 for a C_ANNUNC). More objects of the same block type in a Process Tag Types result in a repetition of the structures within the IEA file. E. g. for each C_ANNUNC you have always the same structure as part of the complete structure within the IEA file.
The structure of the message texts (event texts / block comments) is also identical for each block type. If you have more objects of the same block type in a Process Tag Types, you also have repeating structures for the message texts within the IEA file.
All these repeating sections for objects / blocks within an IEA file are defined in the sheet IEA_OBJECTS. The common structure of an IEA file and the sequence of the objects within the IEA file is defined in the sheet IEA_TYPICALS. The sequence of the blocks within an IEA file is determined by the alphabetic order of the block names.

For each Process Tag Type 2 lines are reserved in the sheet IEA_TYPICALS. In these two a detailed description of the IEA file must be configured. The description must fit exactly to the PCS 7 definition of the Process Tag Types.

- The first four elements of an IEA file are standard definitions (always 1. Project, 2. Hierarchy, 3. PLC name and 4. CFC chart name) and not listed in the sheet IEA_TYPICALS.
- In column “A” of both lines you have to enter the name of the Process Tag, e. g. "MOT_2PI_2PIA“.
- In column “B” of both lines you have to enter the file name of the IEA file, which is assigned to the Process Tag Types, e. g. "MOT_2PI_2PIA00.IEA“.
- The column “C” is empty and will be used later on from the Excel macros.

Starting from column "D" the objects of the Process Tag Types are listed. In the first line you have to enter the object name (block name), e. g. “D1“.
In the second line you have to enter the name of the IEA part structure from the sheet IEA_OBJECTS, e. g. "ANNUNC_B".
For interlock blocks you to enter the block names of all involved objects, separated by "/". (drive name /Interlock name/Object1/Object2/etc., e. g. "M1/ESVA1/D1/D2").
The sequence of the objects is the alphabetic order and the objects with messages are listed twice in the IEA file: First all block parameter and block signals and then all object again in alphabetic order with their message IEA structures for the ALARM8.
See the existing examples and compare the structure and sequence in IEA_TYPICALS and IEA_OBJECTS with an IEA template file, which you can create with PCS 7.

Attention: Follow the rules for interlock names. For the Cemat interlock blocks C_INTER5 und C_INTERL there are strict rules for the interlock block names, e. g. M1_ESVA1 (refer to manuals). For the new PCS 7 interlock blocks Intlk02 to Intlk16 there are no rules, except it is not allowed to give them names which are conform to the rules for names for C_INTER5 and C_INTERL.

Example with Cemat Interlock blocks C_INTER5: The IEA structure description for the Process Tag Type "MOT_2PI_2PIA" consists of the definition of the variable parameters and signals for the drive M1 (C_DRV_1D), the additional annunciation blocks D1/D2/R1/R2 (C_ANNUNC) and the interlock blocks M1_ESVG1/M1_ESVA1 (C_INTER5).

Columns A to G

A	B	C	D	E	F	G
Typical Name	IEA file	used				
MOT_2PI_2PIA	MOT_2PI_2PIA00.IEA		D1	D2	M1	M1/ESVA1/D1/D2
MOT_2PI_2PIA	MOT_2PI_2PIA00.IEA		ANNUNC_B	ANNUNC_B	DRV1D_COM	INTER5

Columns H to O

H	I	J	K	L	M	N	O
M1/ESVG1/R1/R2	R1	R2	D1	D2	M1	R1	R2
INTER5	ANNUNC_B	ANNUNC_B	ANNUNC_A	ANNUNC_A	DRV1D_A	ANNUNC_A	ANNUNC_A

Example with PCS 7 interlock blocks Intl02: The IEA structure description for the Process Tag Type "MOT_2PI_2PIA" consist of the definition of the variable parameters and signals for the drive M1 (C_DRV_1D), the additional annunciation blocks D1/D2/R1/R2 (C_ANNUNC) and the interlock blocks M1_ProtG/M1_ProtA (Intl02).

Columns A to G

A	B	C	D	E	F	G
Typical Name	IEA file	used				
MOT_2PI_2PIA	MOT_2PI_2PIA00.IEA		D1	D2	M1	M1/ProtA/D1/D2
MOT_2PI_2PIA	MOT_2PI_2PIA00.IEA		ANNUNC_B	ANNUNC_B	DRV1D_COM	S_INTER2

Columns H to O

H	I	J	K	L	M	N	O
M1/ProtG/R1/R2	R1	R2	D1	D2	M1	R1	R2
S_INTER2	ANNUNC_B	ANNUNC_B	ANNUNC_A	ANNUNC_A	DRV1D_A	ANNUNC_A	ANNUNC_A

For each new Process Tag Type you have to add 2 lines in the sheet IEA_TYPICALS. Please use already existing Process Tag Types as a template for your new Process Tag Types.

Sheet IEA_OBJECTS

For each Cemat project standard exist an own sheet IEA_OBJECTS. For "Cemat 000" please use the sheet IEA_OBJECTS_000. For all other Cemat project standards use the sheets with the according code number. In the sheet IEA_OBJECTS the IEA structures of an individual block / object are defined. These structures are only a part/segment of the structure of an IEA file and can be used several times within the structure of one IEA file.
Each of these segments gets a unique name (e. g. "DRV_1D_COM" = drive with group commands). By this name the sheet IEA_TYPICALS refers to this structure segment.
E. g., if in a Process Tag Type the C_ANNUNC block is used 8 times, then the IEA file contains for each of this 8 C_ANNUNC the same structure. This applies for the parameter structures as well for the message structures.
With a single definition of the block types in sheet IEA_OBJECTS and a reference to IEA_OBJECTS in sheet IEA_TYPICALS the entries in sheet IEA_TYPICALS are much more simple and short.

For each new IEA structure section you have to add 2 lines in the sheet IEA_OBJECTS. In these two lines you have to enter a detailed description of the IEA structure segment. Please use the already existing definitions in IEA_OBJECTS as a template for new IEA segment.

Example: Drive with interconnections to the group commands:

DRV1D_COM	C_DRV_1D								
ERM	|	Blockname	|	Comment	ESB	EBM	EVO	ESP	ESR

LOCAL	SINGLE	EBFE	EBFA	QSTP	GR_LINK1	GR_LINK2	EBE

1. Line 1. Column = IEA structure segment name, e. g. DRV1D_COM = with interconnections to the group commands (Start / Stop / Single mode / etc.).

1. Line 2. Column = Block type, e. g. "C_DRV_1D" or
"Alarm8" for an IEA structure segment with the message definition of an object.

1. Line 3. Column = Block type, if column 2 = "Alarm8", e. g. "C_DRV_1D"

1. Line x. Column = Block type for further blocks in this IEA structure segment, e. g. a "CH_AI" for a measured value block or AND / OR gates for groups and routes ("G_RUNNING", "G_STOPPED", etc.)

2. Line: elements (keywords) of the IEA definition.
The following elements / keywords are permitted.

Signals for Objects, which are listed in the sheets "SIGNALTYPES xxx". For each Cemat project standard exist an own sheet "SIGNALTYPES xxx". "xxx" is the project standard code, e. g. 000" for the Cemat Normal Standard. In these sheets is the assignment between parameter <-> function defined. E. g. for parameter "ERM", block type "C_DRV_1D" is the function "feedback" defined. All parameter from the column "E" (sheet SIGNALTYPES xxx) could used in the sheet IEA_OBJECTS. The macro will enter the signal (symbol name) into the IEA file.

Otherwise refer to the following table with allowed IEA_OBJECTS elements / keywords

<table>
<tr><th>Keyword</th><th>Entry into the IEA file</th></tr>
<tr><td>BLOCKNAME</td><td>Block name, e. g. "M1"</td></tr>
<tr><td>BLOCKNAME_AI</td><td>Block name + appendix "_AI", e. g. "P1_AI"</td></tr>
<tr><td>BLOCKNAME_DI</td><td>Block name + appendix "_DI", e. g. "D1_DI"</td></tr>
<tr><td>BLOCKNAME_SC</td><td>Block name + appendix "_SC", e. g. "M1_SC"</td></tr>
<tr><td>BLOCKNAME_INTERL</td><td>Block name of the Interlock block, e. g. "M1_ESVG1</td></tr>
<tr><td>COMMENT</td><td>Block comment</td></tr>
<tr><td>COMMENT_ INTERL</td><td>Block comment for Interlock blocks</td></tr>
<tr><td>EVENT</td><td>Event text</td></tr>
<tr><td>EVENT_SIG1, EVENT_SIG2, EVENT_SIG3, EVENT_SIG4, EVENT_SIG5, EVENT_SIG6, EVENT_SIG7,</td><td>Event text of the according signals of a C_ANNUN8</td></tr>
<tr><td>EVENT1, EVENT2, EVENT3, EVENT4, EVENT5, EVENT6, EVENT7, EVENT8, EVENT9, EVENT10, EVENT11, EVENT12, EVENT13, EVENT14, EVENT15, EVENT16</td><td>Event text of the according signals of a C_INTER5 or C_INTERL or Intlk02 or Intlk04 or Intlk08 or Intlk16</td></tr>
<tr><td>INTERL_1, INTERL_2, INTERL_3, INTERL_4, INTERL_5, INTERL_6, INTERL_7, INTERL_8, INTERL_9, INTERL_10</td><td>Empty field. The property "Text_1" will be filled with the event text of the signal. Because of the empty field, the existing interconnection will remain.</td></tr>
<tr><td>|</td><td>|, no text limiter """ and no list separator ";"</td></tr>
<tr><td>ONLY |</td><td>Empty field with "|" (inclusive text limiter """ and list separator ";"</td></tr>
<tr><td>EMPTY</td><td>Empty field (inclusive text limiter """ and list separator ";"</td></tr>
<tr><td>EBFE, EBFE1, EBFEX, EBFEY, VBFE, VBFE1, KEB1</td><td>Textual interconnection to group (GBE) or to route (WBE)</td></tr>
<tr><td>SST1</td><td>Textual interconnection to group (ST), Lafarge</td></tr>
<tr><td>WEBW</td><td>Textual interconnection to group (GBE)</td></tr>
<tr><td>EBFA, VBFA, KEB2</td><td>Textual interconnection to group (GDA) or to route (WBA)</td></tr>
<tr><td>WABW, WGWA (Holcim)</td><td>Textual interconnection to group (GDA)</td></tr>
<tr><td>LOCAL</td><td>Textual interconnection to group (GLO)</td></tr>
<tr><td>SINGLE</td><td>Textual interconnection to group (GES)</td></tr>
<tr><td>ENLM</td><td>Textual interconnection to group (ENLM), Holcim</td></tr>
<tr><td>DILM</td><td>Textual interconnection to group (DILM), Holcim</td></tr>
<tr><td>EAUT, KAUT, VAUT</td><td>Textual interconnection to group (GAU) or to route (WAU), Dyckerhoff</td></tr>
<tr><td>SWK</td><td>Textual interconnection to group (WK), Lafarge</td></tr>
<tr><td>RST</td><td>Textual interconnection to group (RS), Lafarge</td></tr>
</table>

SRT	Textual interconnection to group (RT), Lafarge
QSTP	Textual interconnection to group (QSTP)
GSTP	Textual interconnection to group (GR_STP), Holcim
G_LINK	Textual interconnection to group (G_LINK)
GR_LINK1	Textual interconnection to group (G_LINK) or to route (R_LINK)
GR_LINK2	Textual interconnection to group (G_LINK) or to route (R_LINK)
SCB, LPHL	Scale begin for C_MEASUR / C_AIB
SCE, HPHL	Scale end for C_MEASUR / C_AIB
UNIT	Unit for C_MEASUR / C_AIB
VAL_HH, HH	HH Limit for C_MEASUR / C_AIB
VAL_H, H	H Limit for C_MEASUR / C_AIB
VAL_L, L	L Limit for C_MEASUR / C_AIB
VAL_LL, LL	LL Limit for C_MEASUR / C_AIB
VAL_SHH	SHH Limit for C_MEASUR
VAL_SH, HPRO	SH Limit for C_MEASUR / C_AIB
VAL_SL, LPRO	SL Limit for C_MEASUR / C_AIB
VAL_SLL	SLL Limit for C_MEASUR
SCB_OUT	Textual interconnection from driver block CH_AI to C_MEASUR (SCB_OUT), if not PT100
SCE_OUT	Textual interconnection from driver block CH_AI to C_MEASUR (SCE_OUT), if not PT100
REL_MVC	1, if current / power measurement for drive exists, otherwise 0
OKS, SIM_VAL, M_SIM, NS	0, if signal is normally open = 0 1, if signal is normally closed = 1
IN_DEL	Default value 0. The property "Identifier" will be filled with the event text of the signal
WMOD	0, if alarm type = alarm 1, if alarm type = WA = warning, Holcim
TYP	0, if alarm type = alarm 1, if alarm type = WA = warning, Lafarge
ON_DLY	Default value 10. The property "Identifier" will be filled with the event text of the signal, only Holcim and block type C_PROFB
IN_SIG	Speed monitor signal, no pulse signal, only Holcim and block type C_PROFB
PULS_SSM	Speed monitor signal, pulse signal, only Holcim and block type C_PROFB
EN_SSM	0, if speed monitor signal is no pulse signal 1, if speed monitor signal is a pulse signal only Holcim and block type C_PROFB
REL_SSM	0, if no speed monitor signal exists

	1, if a speed monitor signal exists, Lafarge
I_ADDR	Simocode start address input area for C_SIMOS
O_ADDR	Simocode start address output area for C_SIMOS
BEGINTEXT	' = text limiter for S_TEXT, C_INTER5
ENDTEXT	' = text limiter for S_TEXT, C_INTER5
;	; = separator character within S_TEXT, C_INTER5
SIG_TXT1, SIG_TXT2, SIG_TXT3, SIG_TXT4, SIG_TXT5, SIG_TXT6, SIG_TXT7	Signal description as annunciation text for C_ANNUN8
SSB_BF1, SSB_BF2, SSB_BF3, SSB_BF4, SSDB, G_DRIVES_HE, G_DRIVES_HV, G_LOCAL, G_R_RUNNING, G_R_STOPPED, G_RUNNING, G_STOPPED, R_DRIVES_HE, R_DRIVES_HV, R_RUNNING, R_STOPPED	Block names for according AND / OR gates of the groups and routes
COMMENT_BF1, COMMENT_BF2, COMMENT_BF3, COMMENT_BF4, COMMENT_HE, COMMENT_HV, COMMENT_LOCAL, COMMENT_ROUTES_RUNNING, COMMENT_ROUTES_STOPPED, COMMENT_RUNNING, COMMENT_STOPPED, COMMENT_ SEQUENCE,	Block comments for according AND / OR gates of the groups and routes
GHA, GLA	Empty fields, group signals, not used at the moment
SAV	Default value = 0, SSB block Lafarge
T1	Default value = 0, SSDB block Lafarge
GREZ1, GREZ2, …, GREZ50	Textual interconnection to drive (EVS, EVS1, EVS2, EVSY, EVSX, KVS2, KVSX, VVS2, VVSX)
GRAZ1, GRAZ2, …, GRAZ50	Textual interconnection to drive (EVS, EVS1, EVS2, EVSY, EVSX, KVS1, KVSY, VVS1, VVSY)
R_GREZ1 and R_GREZ2	Textual interconnection to route (WRE)
R_GRAZ1 and R_GRAZ2	Textual interconnection to route (WRA)
FB_L1, FB_L2, …, FB_L50	Textual interconnection to drive (LOCAL) only Holcim
HV1, HV2, …, HV50	Textual interconnection to drive (EHV, KHV, VHV) only Dyckerhoff
HE1, HE2, …, HE50	Textual interconnection to drive (EHE, KHE, VHE) only Dyckerhoff
R_HV1 and R_HV2	Textual interconnection to route (WHV) only Dyckerhoff
R_HE1 and R_HE2	Textual interconnection to route (WHE) only Dyckerhoff
BF1_1, BF1_2, …, BF1_50	Textual interconnection to drive (ST1) only Lafarge

BF2_1, BF2_2, …, BF2_50	Textual interconnection to drive (ST1) only Lafarge
BF3_1, BF3_2, …, BF3_100	Textual interconnection to object (FW) only Lafarge
BF4_1, BF4_2, …, BF4_100	Textual interconnection to object (FT) only Lafarge
AEVG	1 = Default value Select block
DESELECTED	'DESELECTED' = Default value Select block
SELECTED	'SELECTED' = Default value Select block

Check the consistency of the IEA structure description in the sheets CEM_TYPICALS, IEA_TYPICALS and IEA_OBJECTS.

With the button "Check typicals and IEA structure" 

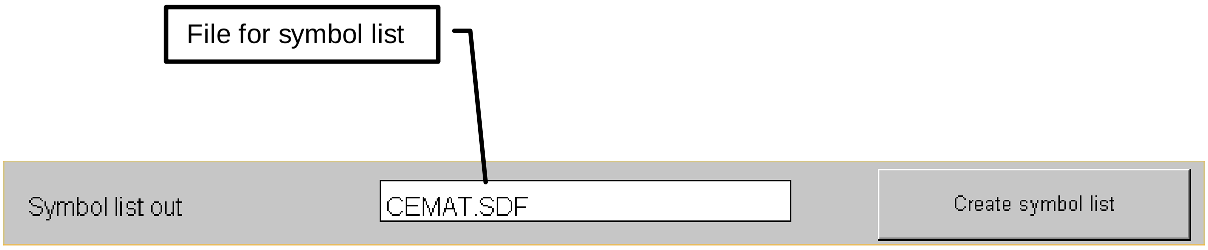
you can check the consistency of the sheets CEM_TYPICALS, IEA_TYPICALS and IEA_OBJECTS. If the macro finds a wrong entry, you will get an error message within a message box.

Create the symbol list

With the button "Create symbol list", a macro creates a symbol list (SDF file), based on the sheet SIGNALLIST. Errors will be listed in the sheet LOGS. Depending on the language settings "deutsch" or "English", the symbol list will be created with German or English mnemonics, e. g. E 3.4 or I 3.4.

Lines with errors will be not written into the SDF file!

The SDF file can be imported with the Symbol Editor of the Simatic Manager.



SIMATIC Manager

↑

Symbol Editor
Import function

↑

CEMAT.SDF

Create the object list (block list)

With the button "Create Cemat object list" 

, a macro creates an object list (sheet OBJECTS), based on the sheet SIGNALLIST.

The object is defined by chart name and object name (block name). The signal list SIGNALLIST list has to be sorted in a way that all signals for the same object are in sequence; all object signals for the same chart also have to be in sequence.

For each signal you have to define an object type (e. g. block type = C_DRV_1D) and a function (e. g. "feedback"). In the sheets "SIGNALTYPES xxx" (xxx = project standard code) the allowed combinations of block type and function are listed.

For each new object name/chart name in the signal list, the Engineering Tool creates a new line in table OBJECTS. If the macro finds the combination block type and function in the sheet "SIGNALTYPES xxx", then the signal name is entered in the matrix of the sheet OBJECTS.
If the macro doesn't find the combination block type and function in the sheet "SIGNALTYPES xxx", then a new object will be created according to the IO type (DI, DO, AI, AO). See following table:

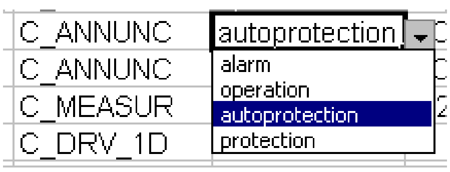
IO type signal list	**Object type block type**	**Description**
AI	C_MEASUR	Analogue signal connected to a C_MEASUR block
AO	CH_AO	Analogue signal connected to a CH_AO block
DI	AND	Digital signal connected to the input of an AND gate
DO	OUT	Digital signal connected to the output of an AND gate

For the field "function", depending on the block type, only certain entries are allowed.

With the button "Create list fields in object list" 

the cells in column "function" can be changed into Excel list fields.
This has the advantage that via list fields only permitted values can be entered.

Example list field for the function of a C_ANNUNC block:



With the button "Delete list fields in object list" 

the list fields will be changed back to normal Excel cells, i. e. now all values can be entered.

Example object list:

Block functions

A	B	C	D	E	F	G	H	I	J	K
Chart	**Object**	**Block text**	**Event text**	**Cemat_Object**	**function**	**ConvType**	**AlarmTyp**	**SCB**	**SCE**	**Unit**
E51_BC1	M1	Belt Conveyor Below E52-3B1		C_DRV_1D	main					
E51_3S1	T1	Silo E51-3S1 Temperature		C_MEASUR	analogvalue	PT100		0	200	°C

L	M	N	O	P	Q	R	S	T	U
VAL_HH	**VAL_H**	**VAL_L**	**VAL_LL**	**VAL_SHH**	**VAL_SH**	**VAL_SL**	**VAL_SLL**	**feedback**	**feedback1**
								E51_BC1_M1_ERM	
200	200	0	0	200	200	0	0		

V	W	X	Y	Z	AA	AB
feedback2	**limitpos1**	**limitpos2**	**intlimitpos1**	**intlimitpos2**	**available**	**overload**
					E51_BC1_M1_ESB	E51_BC1_M1_EBM

AC	AD	AE	AF	AG	AH	AI	AJ
Local	**locstart**	**locstart1**	**locstart2**	**locstop**	**torque1**	**torque2**	**speedmon**
E51_BC1_M1_EVO	E51_BC1_M1_ESR			E51_BC1_M1_ESP			E51_BC1_SI

AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU
ON	**ON1**	**ON2**	**alarm**	**operation**	**protection**	**shutdown**	**autoprotection**	**fault1**	**fault2**	**fault3**
E51_BC1_M1_EBE										

AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF	BG
fault4	**fault5**	**fault6**	**fault7**	**analogvalue**	**actposition**	**AO**	**BIT**	**OUT**	**beltbreak**	**limitpos**	**puls**
				E51_3S1_T1							

Signal analogue

Signals drive

Die header line will be copied from the sheet HEADER, line 2. Please don't change this line!

The macro automatically determines a function for each object:

The first drive of a chart will get the function "main" = main drive, all other drives on the chart will get the function "aux" = auxiliary drive.
For analogue values, keywords are defined in sheet LIST_DATA, column "I" to "L". If the macro finds a keyword in the "Block text" and a main drive exists, then the block function will be changed from "analogvalue" into "drive_curr" = current / power measurement main drive or "drive_speed" = speed measurement main drive. Please check the function of the objects. If the macro determines a wrong function, please change it manually to the correct function.

Later, during chart detection, a macro is searching for the most suitable Process Tag Type (Typical) for each chart. The Process Tag Type searching considers not only the block type, but also the block function.

Object functions

Block	Function name	Description
C_DRV_1D, C_DRV_2D, C_BPB, C_M2B, C_DAMPER, C_DABMAB, C_AAB, C_VALVE, C_VAL_2D, C_DAB	main	Main drive, if the is a current or power or speed measurement, then the measurement will be connected to the main drive
C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE, C_VAL_2D	main_SC	Main drive with Simocode
C_DRV_1D, C_DRV_2D, C_BPB, C_M2B, C_DAMPER, C_DABMAB, C_AAB, C_VALVE, C_VAL_2D, C_DAB	aux	Auxiliary drive
C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE, C_VAL_2D	aux_SC	Auxiliary drive with Simocode
C_ANNUNC, C_DIB	alarm	Alarm message
C_ANNUNC, C_DIB	operation	Alarm message, which is connected via interlock block to the operation interlock of the drive = EBVG
C_ANNUNC, C_DIB	protection	Alarm message, which is connected via interlock block to the protection interlock of the drive = ESVG, PINT1, PE1
C_DIB	shutdown	Alarm message, which is connected via interlock block to the emergency shut down interlock of the drive = HS (only Lafarge)
C_ANNUNC, C_DIB	autoprotection	Alarm message, which is connected via interlock block to the protection interlock automatic mode of the drive = ESVA, PINT2, PE2
C_MEASUR, C_AIB	analogvalue	Analogue value input
C_MEASUR, C_AIB	drive_curr	Current or power measurement for the main drive
C_MEASUR, C_AIB	drive_speed	Speed measurement for the main drive
C_ANNUN8	annun8	7 fault messages
C_SILOP	silo	Silo pilot
C_PROFB	speedmon	Speed monitor (only Holcim)
C_SPEEDM	speedmon	Speed monitor (only Lafarge)
CH_AO	AO	Analogue output VALUE for CH_AO
AND	BIT	Common digital input signal, will be connected to an AND gate
OUT	OUT	Common digital output signal, will be connected to the output of an AND gate
SINA_GS	drive_speed	Speed measurement for the main drive

Create CFC chart list

From the object list (sheet OBJECTS) a chart list is created (sheet CHARTS). The most suitable Process Tag Type will be selected from the list of Typicals (sheet CEM_TYPICALS). The Typical (Process Tag Type) should contain all objects which belong to the chart. If no suitable Process Tag Type is found, a Process Tag Type will be chosen which fit best as possible for the charts, which should be created.

Priority for the searching:

1. Current or speed measurement exists and also in the Process Tag Type
2. More identical block types and block functions.
3. More identical block types.
4. Less not needed objects, which have to be deleted afterwards

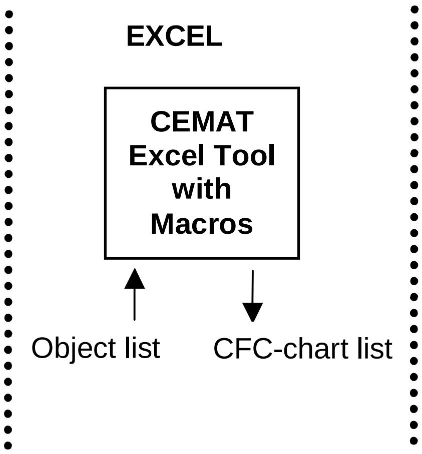
For the rest of the objects, which are not covered by the first Process Tag Type, again a Process Tag Type is searched. The chart gets the original chart name with the extension "_2", e. g. E51_BC1_2. If still objects are remaining, again a typical search is started. This happens until all needed objects are placed in a chart. All these extension charts have the same name like the first chart and the extensions "_2", "_3", "_4", etc. Later on, during editing the CFCs, all this objects in these extension charts have to be moved into the first chart. After these moves, the extension charts are empty and have to be deleted.

With the button "Create Cemat chart list" 

the chart list will be created. The button is in the sheet PRESETTINGS.

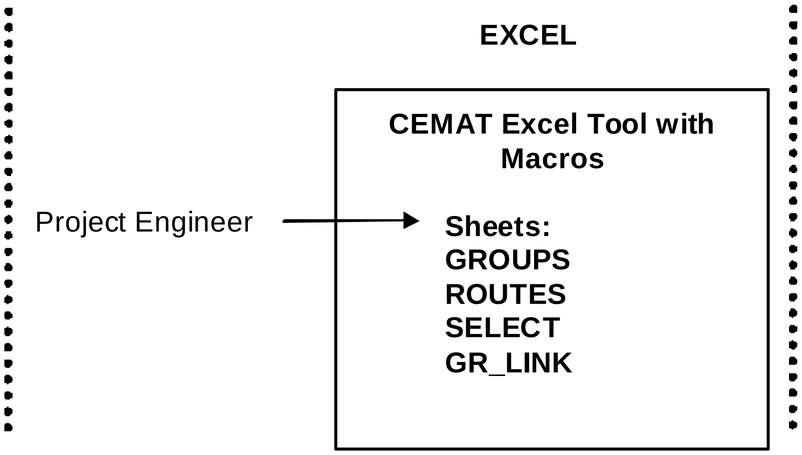
Preconditions for the creating of the CFC chart list are:

- All existing Process Tag Types have to be listed in CEM_TYPICALS.
- The IEA file structures of all existing Process Tag Types have to be defined in the sheets IEA_TYPICALS and IEA_OBJECTS.



Assignment of the charts to groups, routes and hierarchy

In the sheets GROUPS, ROUTES and SELECT the chart name of the groups, routes and select modules have to be entered (Lafarge only groups = SSB blocks and select modules). The object name for groups is always "G“. The object name for routes is always "R“. For Lafarge the group name is always SSB. In the sheet GR_LINK for each chart the associated groups (max. 2) and eventually the associated routes (max. 2) must be entered. The link to further routes and groups must be carried out manually after the generation of the charts. In column "Hierarchy“ of sheet GROUPS, each group must be assigned to the hierarchy folder. After the chart generation the group chart will be found in this hierarchy folder.



Structure sheet GROUPS

Groups	Hierarchy	BlockText	GLA	GHA
E51_100	Test_System\E51	Block Text Group 1		
E52_100	Test_System\E52	Block Text Group 2		
E53_100	Test_System\E53	Block Text Group 3		

The additional columns "Route 1", "Route 2", "Drives", "Drive 1", "Drive 2", etc. will be used by the macros.

Structure sheet ROUTES

Routes	Hierarchy	BlockText
E51_101	Test_System\E51	Block Text Route 1
E51_102	Test_System\E51	Block Text Route 2
E51_103	Test_System\E51	Block Text Route 3

The additional columns "Drive 1", "Drive 2", etc. will be used by the macros.

Structure sheet GR_LINK

CHART	Hierarchy	Group 1	Route 1	Group 2	Route 2	Chart Comment
E51_3S1	Test_System\E51	E51_100	E51_101		E51_102	E51 Concrete - Silo 1
E51_3S2	Test_System\E51	E51_100		E52_100		E51 Concrete - Silo 2

The first column "CHART" could be filled automatically by starting a macro with the button "Create chart list……" 

. All charts from the sheet CHARTS will be copied, except the extension charts like E51_BC1_2, i. e. there is only one line for chart E51_BC1 in GR_LINK, even when extension charts like E51_BC1_2 or E51_BC1_3 exist. If the sheet GR_LINK is already filled, you can choose to keep it and to add only new charts from sheet CHARTS to add to sheet GR_LINK. Extension charts must not listed in GR_LINK. The columns "Hierarchy", "Group 1", "Route 1", "Group 2", "Route 2"and "Chart Comment" has to be filled manually. Enter for each chart the hierarchy folder, the chart - group / route assignment and the chart comment. In the column "Group 1" there has to be an entry!
Additionally you can enter a second group in "Group2", but then no entry in "Route 1" or Route 2" is allowed. Or you enter additionally to "Group 1" 2 routes in "Route 1" and "Route 2". All other group or route assignments have to be edit later on in the generated charts, especially, when you have to use a C_MUX block.

Structure sheet SELECT

Select	Hierarchy	Chart	Gen. Chart	BlockText
S1	HDRS_Test\E51	E51_FN1	E51_FN1_3	E51 Fan Motor Select 1
S2	HDRS_Test\E51	E51_FN1	E51_FN1_4	E51 Fan Motor Select 2
S1	HDRS_Test\E51	E51_RF1	E51_RF1_3	E51 Rotary Feeder 1 Aux drive
S1	HDRS_Test\E52	E52_BE1	E52_BE1_4	E52 Bucket Elevator 1 Aux drive
S1	HDRS_Test\E53	E53_100	E53_100_2	Group E53 Fan Select

The column "Gen. Chart" will be filled by the macro. The macro checks, is the chart name already used by a chart from sheet CHARTS or from a group or from a route. If the chart name is already used, the select module will be generated in an extension chart. The chart name in "Gen. Chart" will be used for the chart generation.

Check Charts regarding the existing Process Tag Types

With the button Check chart list you can check the consistency of the sheets GROUPS, ROUTES and GR_ LINK. The macro checks also how good the used Process Tag Types fit to charts, which has to be created. The list will get colored. The color of lines is changing for each chart (turquoise or azure); i. e. a chart and his extension charts will be colored with the same color. In this way it is easy to see, if a similar combination of objects is used several times in your project and leads always to the same extension charts. In this case it is recommended to create a new Process Tag Type to cover all needed objects in one Process Tag Type. All fields for not needed objects will be colored yellow. If a Process Tag Type is several times used and always you have to delete the same objects, so you see always the same pattern of yellow fields.

Summary: If Process Tag Types are used several times and for the charts you have to do always the same edit jobs (Copy objects together into one chart or deleting objects), then it is recommended to create new Process Tag Types, which cover better the charts, which have to be created. Enter for test the new Process Tag Types only in CEM_TYPICALS and create the sheet CHARTS new and check with "Check chart list" new. If you have less extension charts and less objects, which you have to delete afterwards, then it is better to create the new Process Tag Types really in PCS 7. Of course you have to increase then the sheets IEA_TYPICALS and IEA_OBJECTS with the structures of the new Process Tag Types. Check your entries with the button "Check typicals and IEA structure".

Example of a colored chart list:

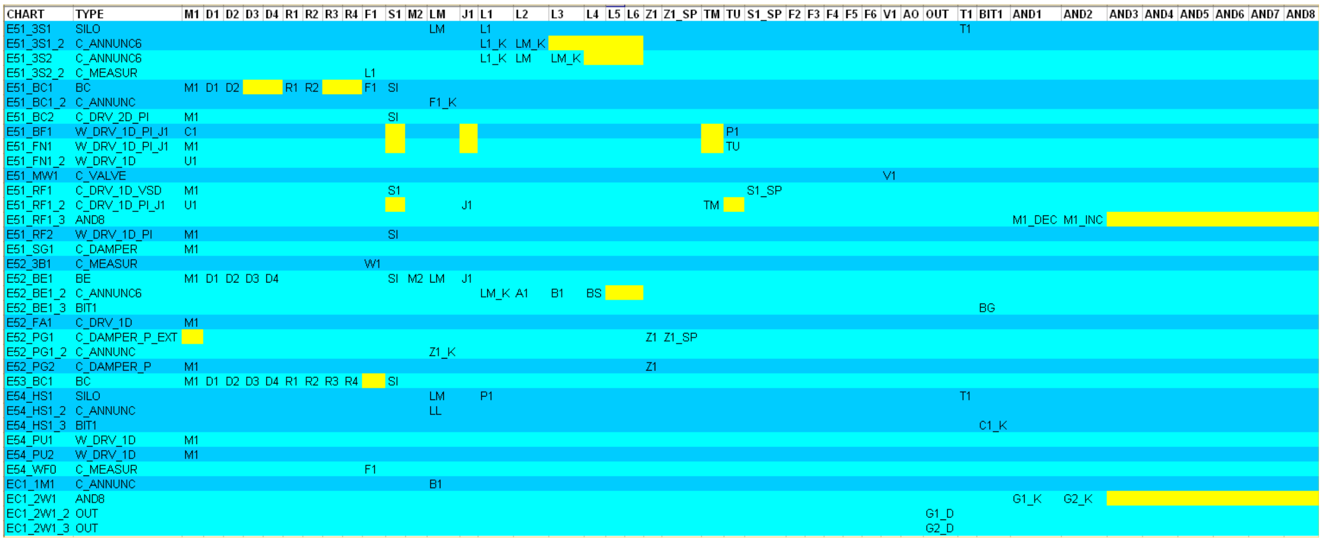


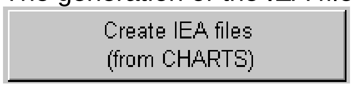
CHART	TYPE	M1	D1	D2	D3	D4	R1	R2	R3	R4	F1	S1	M2	LM	J1	L1	L2	L3
E51_3S1	SILO													LM		L1		
E51_3S1_2	C_ANNUNC6															L1_K	LM_K	
E51_3S2	C_ANNUNC6															L1_K	LM	LM_K
E51_3S2_2	C_MEASUR										L1							
E51_BC1	BC	M1	D1	D2			R1	R2			F1	SI						
E51_BC1_2	C_ANNUNC													F1_K				
E51_BC2	C_DRV_2D_PI	M1										SI						
E51_BF1	W_DRV_1D_PI_J1	C1																
E51_FN1	W_DRV_1D_PI_J1	M1																
E51_FN1_2	W_DRV_1D	U1																
E51_MW1	C_VALVE																	
E51_RF1	C_DRV_1D_VSD	M1										S1						
E51_RF1_2	C_DRV_1D_PI_J1	U1													J1			
E51_RF1_3	AND8																	
E51_RF2	W_DRV_1D_PI	M1										SI						
E51_SG1	C_DAMPER	M1																
E52_3B1	C_MEASUR										W1							
E52_BE1	BE	M1	D1	D2	D3	D4						SI	M2	LM	J1			
E52_BE1_2	C_ANNUNC6															LM_K	A1	B1
E52_BE1_3	BIT1																	
E52_FA1	C_DRV_1D	M1																
E52_PG1	C_DAMPER_P_EXT																	
E52_PG1_2	C_ANNUNC													Z1_K				
E52_PG2	C_DAMPER_P	M1																
E53_BC1	BC	M1	D1	D2	D3	D4	R1	R2	R3	R4		SI						
E54_HS1	SILO													LM		P1		
E54_HS1_2	C_ANNUNC													LL				
E54_HS1_3	BIT1																	
E54_PU1	W_DRV_1D	M1																
E54_PU2	W_DRV_1D	M1																
E54_WF0	C_MEASUR										F1							
EC1_1M1	C_ANNUNC													B1				
EC1_2W1	AND8																	
EC1_2W1_2	OUT																	
EC1_2W1_3	OUT																	

CHART	L4	L5	L6	Z1	Z1_SP	TM	TU	S1_SP	F2	F3	F4	F5	F6	V1	AO	OUT	T1	BIT1	AND1	AND2	AND3	AND4	AND5	AND6	AND7	AND8
E51_3S1																	T1									
E51_3S1_2																										
E51_3S2																										
E51_3S2_2																										
E51_BC1																										
E51_BC1_2																										
E51_BC2																										
E51_BF1							P1																			
E51_FN1							TU																			
E51_FN1_2																										
E51_MW1														V1												
E51_RF1								S1_SP																		
E51_RF1_2						TM																				
E51_RF1_3																			M1_DEC	M1_INC						
E51_RF2																										
E51_SG1																										
E52_3B1																										
E52_BE1																										
E52_BE1_2	BS																									
E52_BE1_3																		BG								
E52_FA1																										
E52_PG1				Z1	Z1_SP																					
E52_PG1_2																										
E52_PG2				Z1																						
E53_BC1																										
E54_HS1																	T1									
E54_HS1_2																										
E54_HS1_3																		C1_K								
E54_PU1																										
E54_PU2																										
E54_WF0																										
EC1_1M1																										
EC1_2W1																			G1_K	G2_K						
EC1_2W1_2																G1_D										
EC1_2W1_3																G2_D										

Generation of the IEA-Files

For each Process Tag Type the structure of the IEA files must be described in the sheets IEA_TYPICALS and IEA_OBJECTS (see Chapters “Sheet IEA_TYPICALS” and "Sheet IEA_OBJECTS").

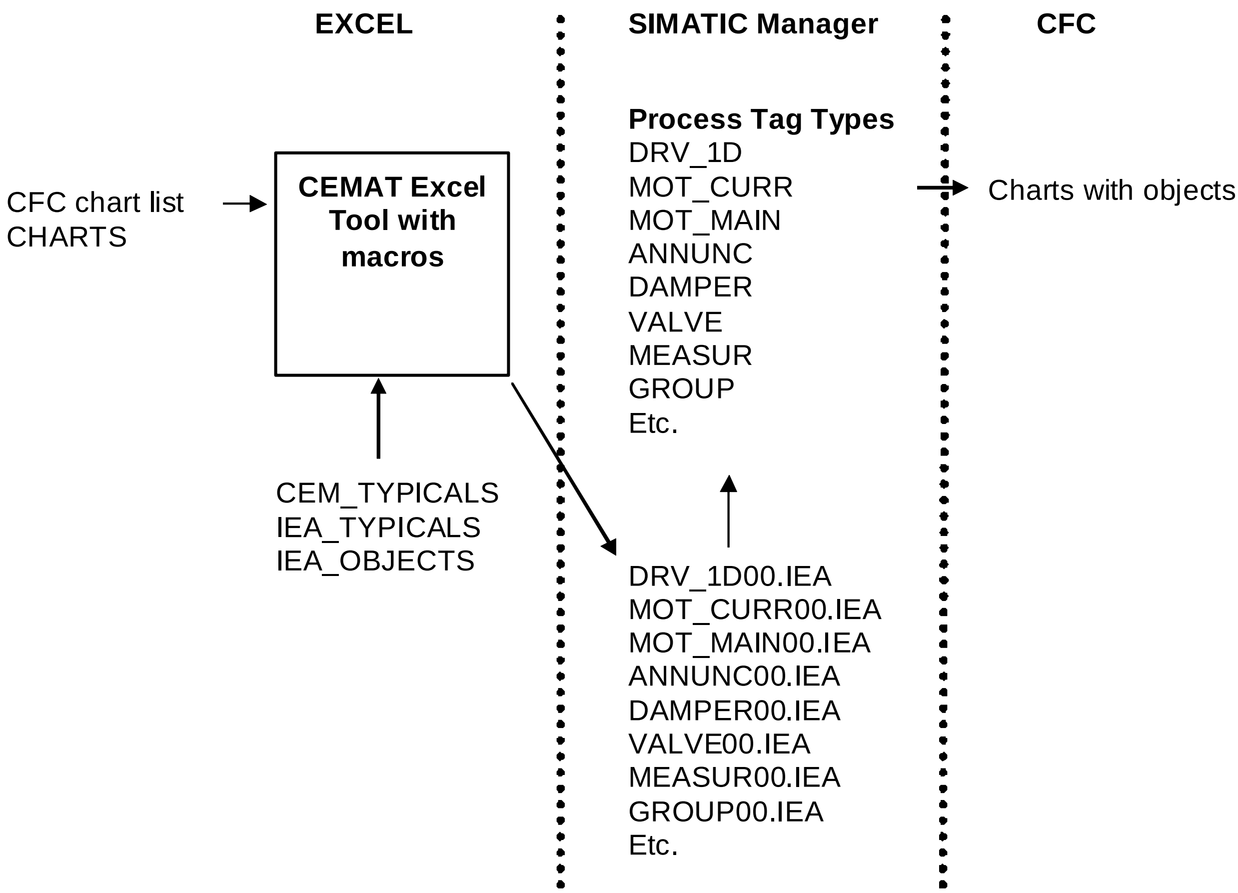
The generation of the IEA files will be started with the button "Create IEA files".



The button "Create IEA files" is in the sheet PRESETTINGS.

Based on the definitions in IEA_TYPICALS and IEA_OBJECTS will be first an IEA file created, which contain only the 4 header lines of an IEA file. For each chart, listed in the sheets CHARTS, GROUPS, ROUTES and SELECT, will be one line attached to the corresponding IEA file. For not used Process Tag Types will be one line added, which contain the data for a dummy chart. This avoids an errors message, if the assigned IEA file is missing or don't contain any data. The not needed dummy charts could be easily deleted afterwards.

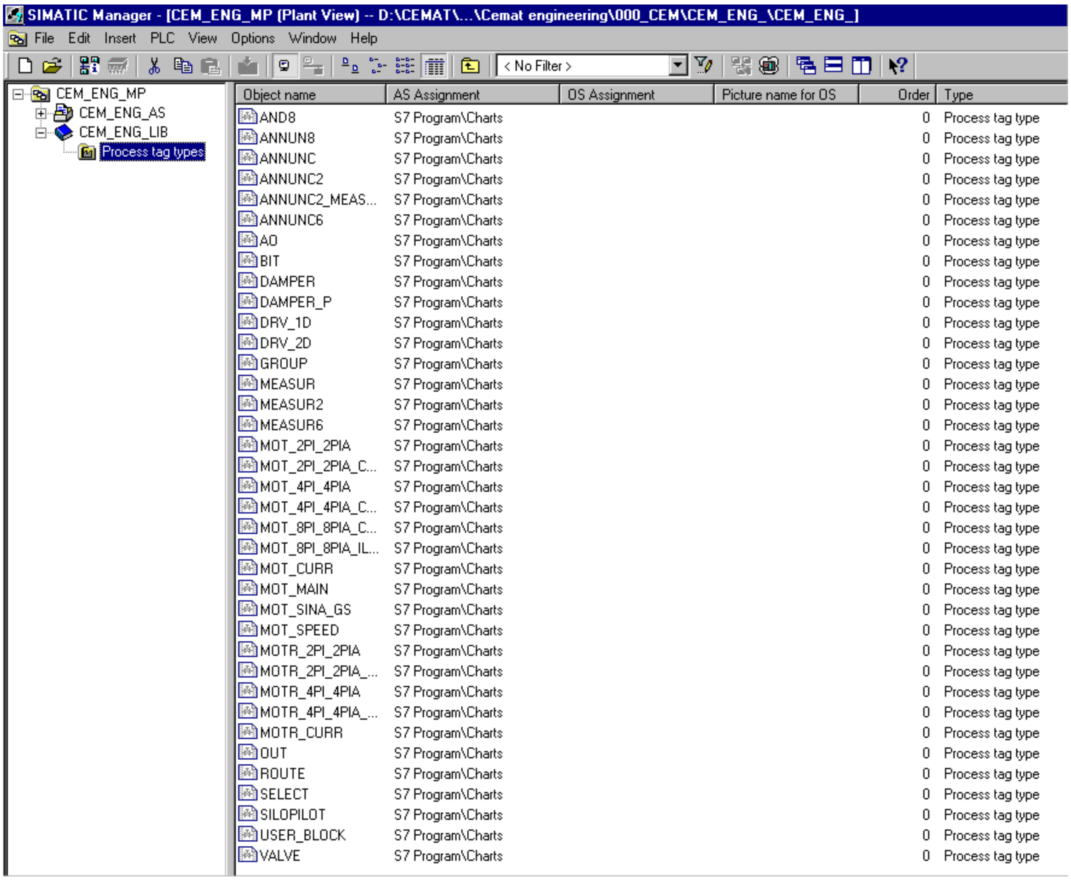
In a later step, through import of these IEA files in the Simatic Manager, the CFC-Charts will be generated. The import of the IEA files has to be carried out in the Simatic Manager. For this import you need the IEA Simatic license. There is no trial license!



CFC Generation with the IEA Editor (PCS 7)

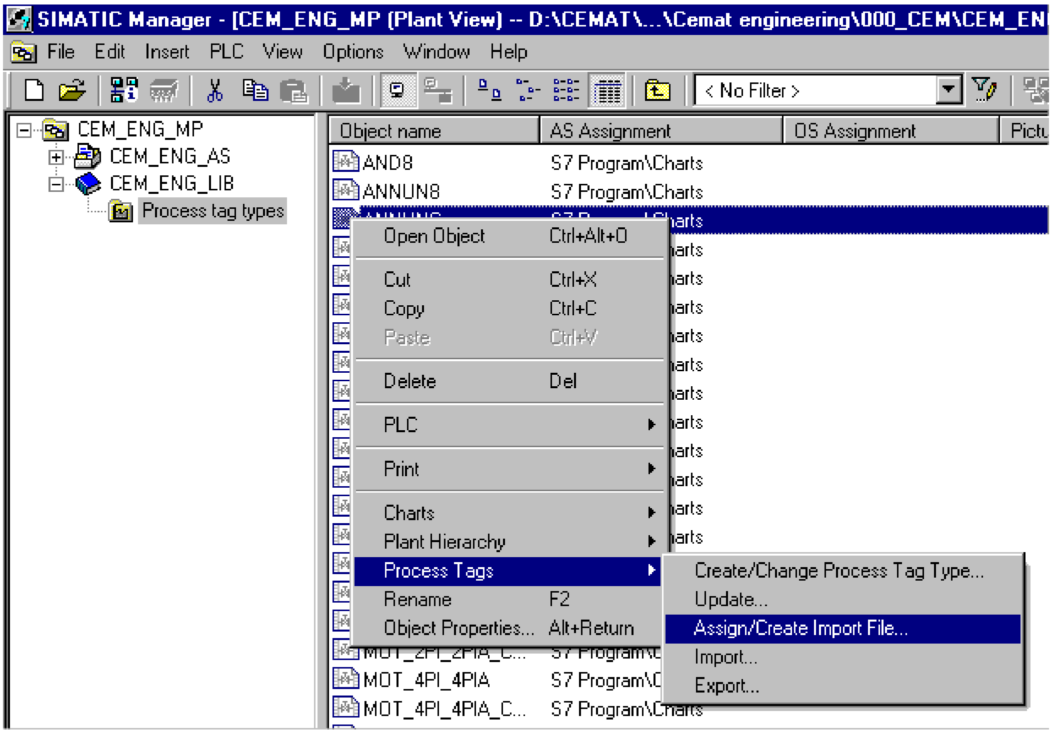
Using already existing Process Tag Types

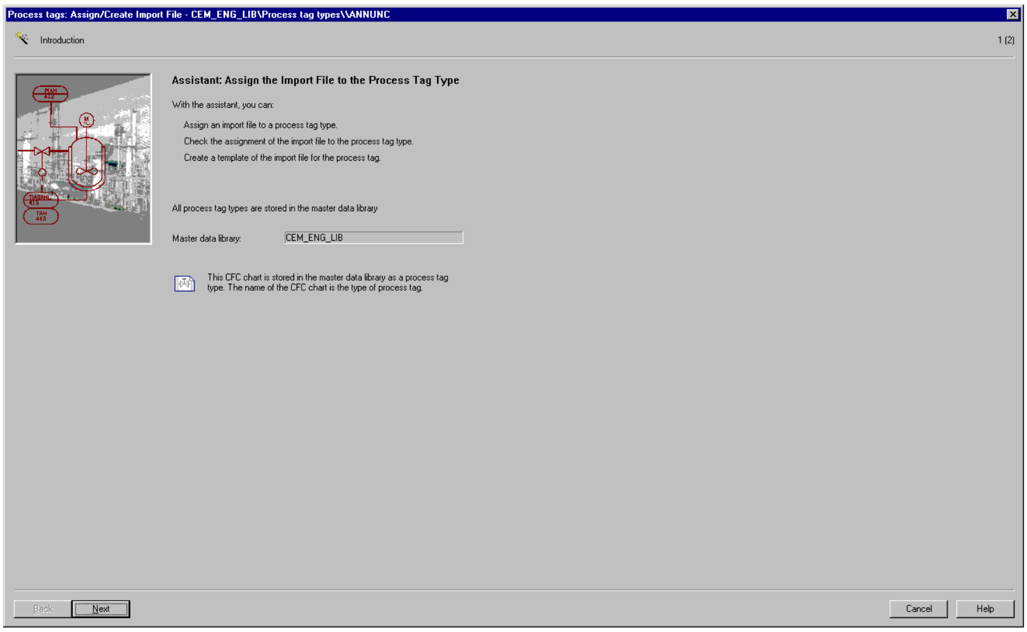
Process Tag Types could only be used, if they are located in the Master Data Library of a Multiproject in a hierarchy folder. The Master Data Library has a blue icon.  See PCS 7 manual "Master Data Library"! Copy all delivered examples (Process Tag Types) into the Master Data Library of your Multiproject. Use during coping the "Plant View" and copy all Process Tag Types into one Hierarchy folder, e. g. "Process Tag Types". In this hierarchy folder you could define also your own Process Tag Types.
Before you copy the delivered Process Tag Types into your Master Data Library, update all libraries and AS programs with the latest version of the block types. Don't forget to update also the chart folders with "Options -> Blocktypes….". If the block types in source and destination are different, you can't copy the Process Tag Types.



Assigning the IEA files to the Process Tag Types

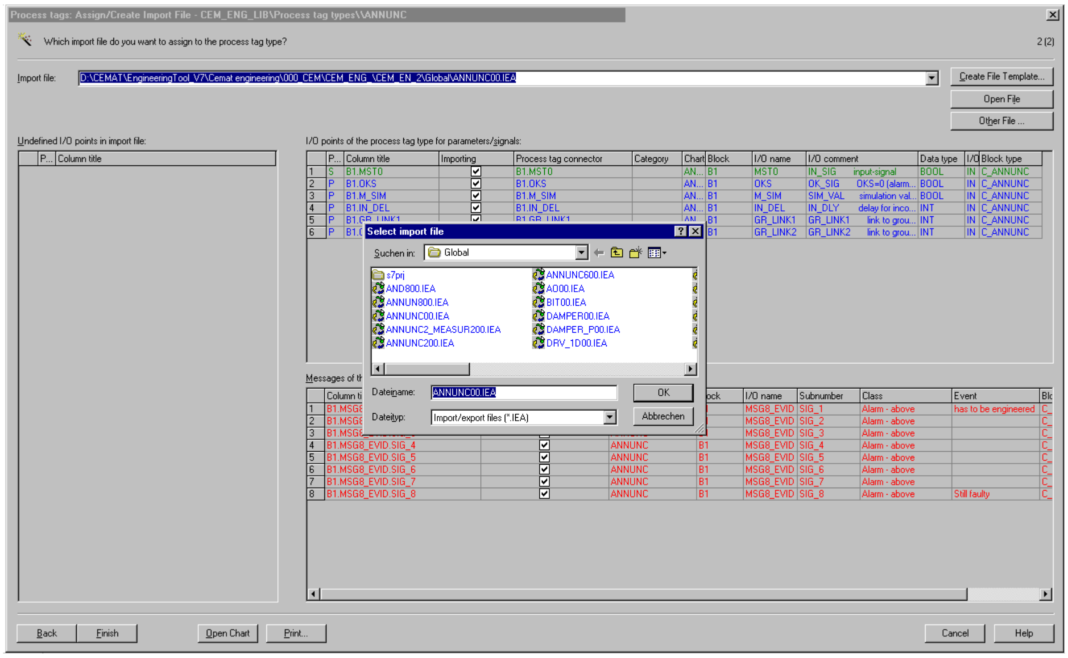
The Process Tag Types must be assigned to the created IEA-Files:
In the plant view of the SIMATIC manager you have to assign each Process Tag Type to his according IEA file. In the plant view of the SIMATIC manager select the Process Tag Type chart with right mouse button and chose "Process Tags -> Assign/Create Import File....“.





Acknowledge the introduction window with "Next".

With the button "Other File…" you can choose the IEA file.



For the Process Tag Type MOT_2PI_2PIA you have to choose the IEA file MOT_2PI_2PIA00.IEA, for the Process Tag Type MEASUR you have to choose the IEA file MEASUR00.IEA, etc.

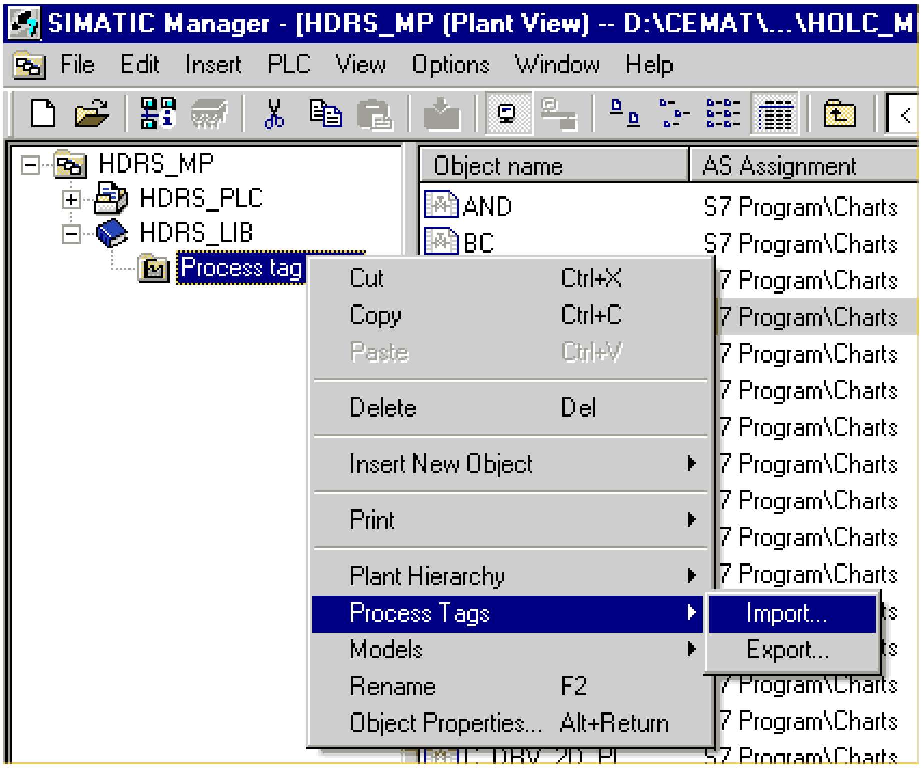
With the button "Finish" the IEA file assignment will be finished.

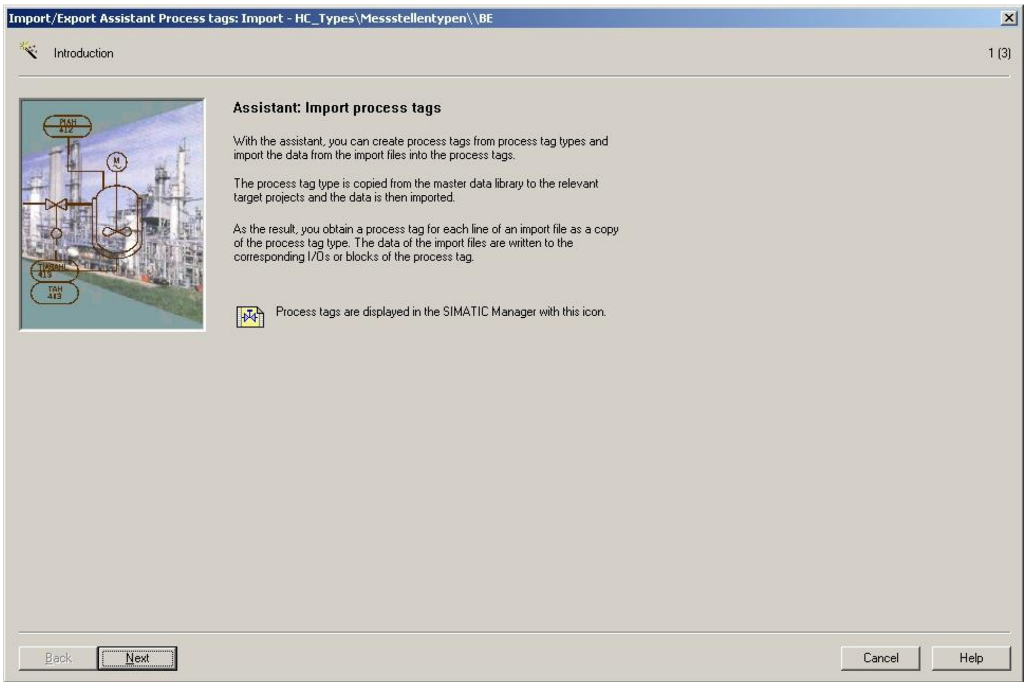
The default directory for IEA files is the subdirectory "Global" of the storage location of the Master Data Library. The assignment is much easier, if you copy all IEA files, created from the Cemat Engineering Tool, into the subdirectory "Global" of the storage location of the Master Data Library. This proceeding has another advantage. If you like to create the charts for several PLCs with IEA files, you have to assign only ones the IEA files. For further PLCs overwrite the IEA files in "Global" with the next IEA files of the next PLC.

Generating the CFC charts

To generate the CFC charts with the IEA, please select the hierarchy folder in the Master Data Library with the Process Tag Types in the "Plant View" of the Simatic Manager. Select with the right mouse button the menu "Process Tags -> Import...“.

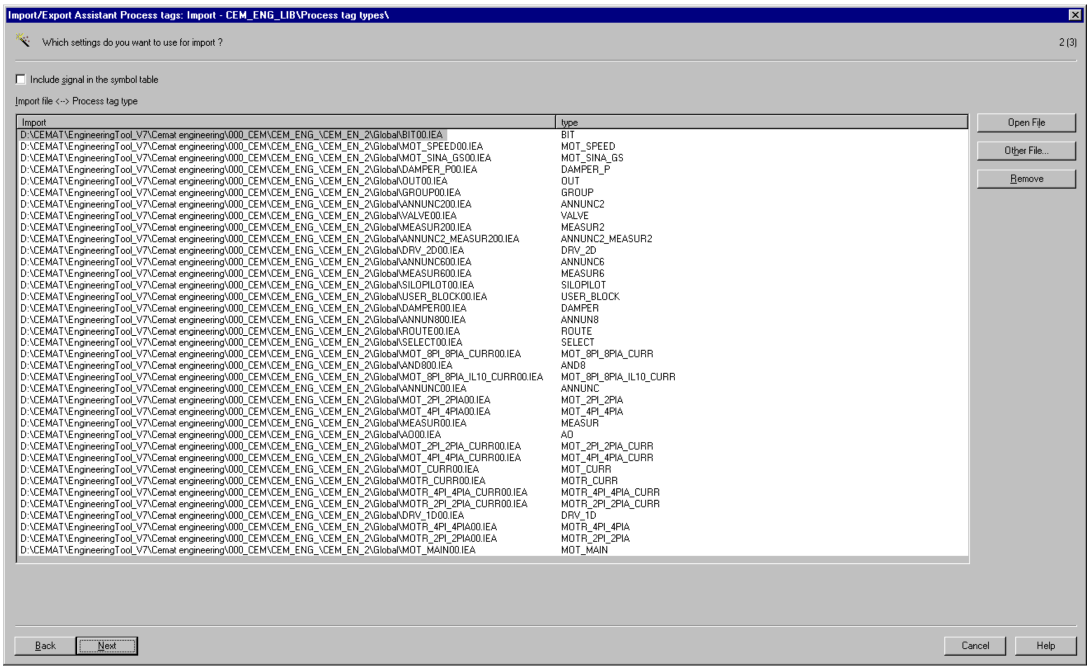
If you use blocks from other libraries, e. g. SINA_GS from the function block library for SINAMICS, please copy the block to the Master Data Library and to the block folder of the PLC. Update also the block in the chart folder of the Master data Library and in the chart folder of the PLC. Otherwise you will have problems during chart generation.





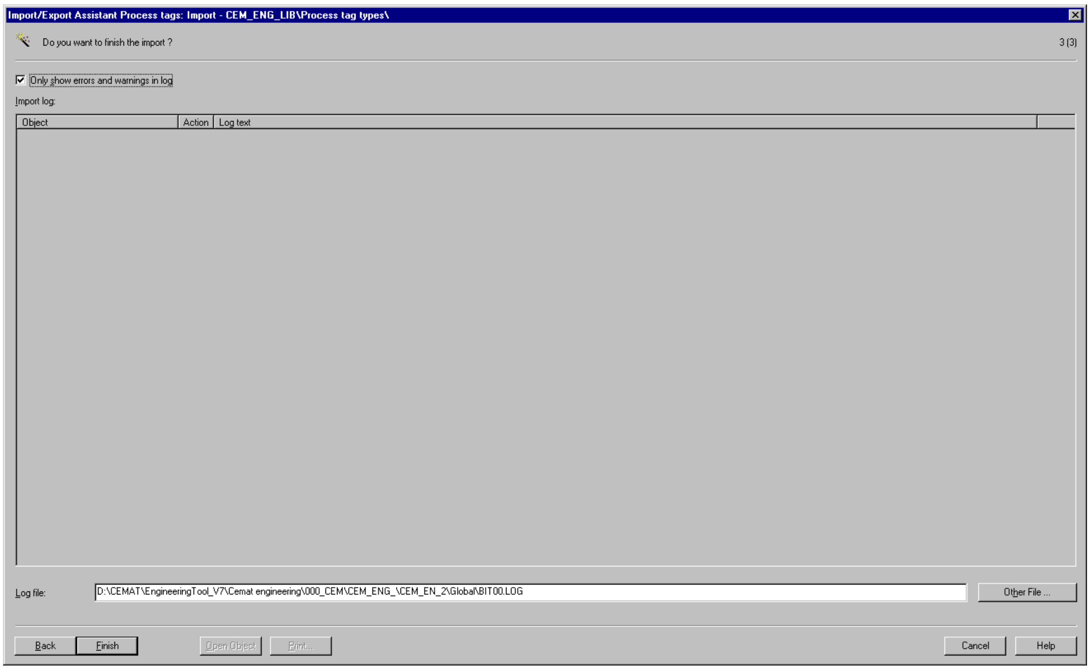
Confirm the introduction screen with "Next“.

In the following screen all Process Tag Types are listed together with the associated IEA-Files:

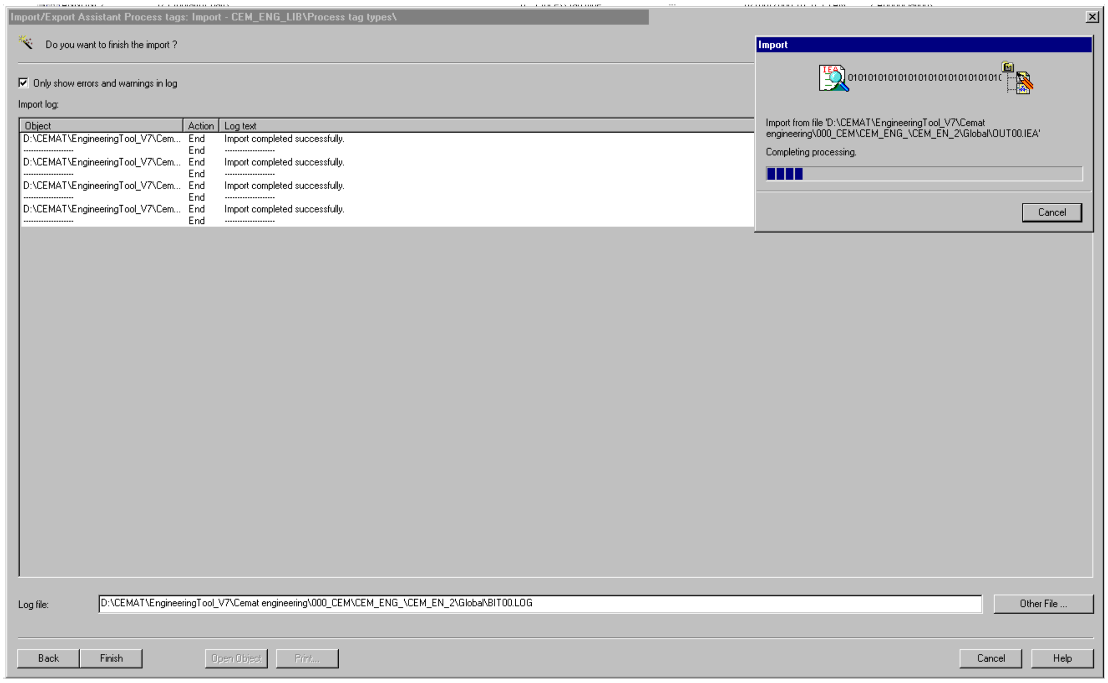


For the Process Tag Types, which are not needed for the creation of the charts, has the Cemat Engineering Tool although IEA files generated. In these IEA files are only data for a dummy chart. In this way each Process Tag Type should be assigned with an IEA file. This is necessary, because the IEA file import of PCS 7 stops completely, if one of the used Process Tag Types is assigned with a not valid IEA file (minimum one user line). The by this way created IEA files could be deleted later on.

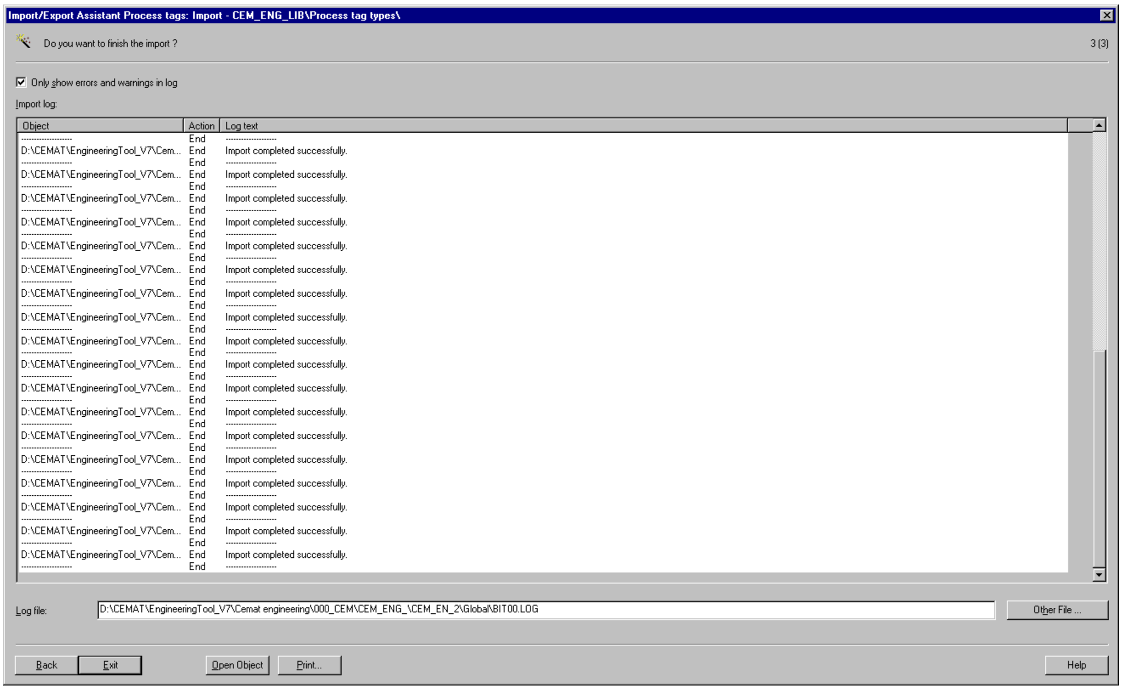
Confirm the Process Tag Type list with "Next“. With the button "Finish" the import of the IEA files will be started. We recommend to select the option "Only show errors and warnings in log". If you do so, then it is easier to see, is there a problem during import. The location of the log file is also shown in this window.



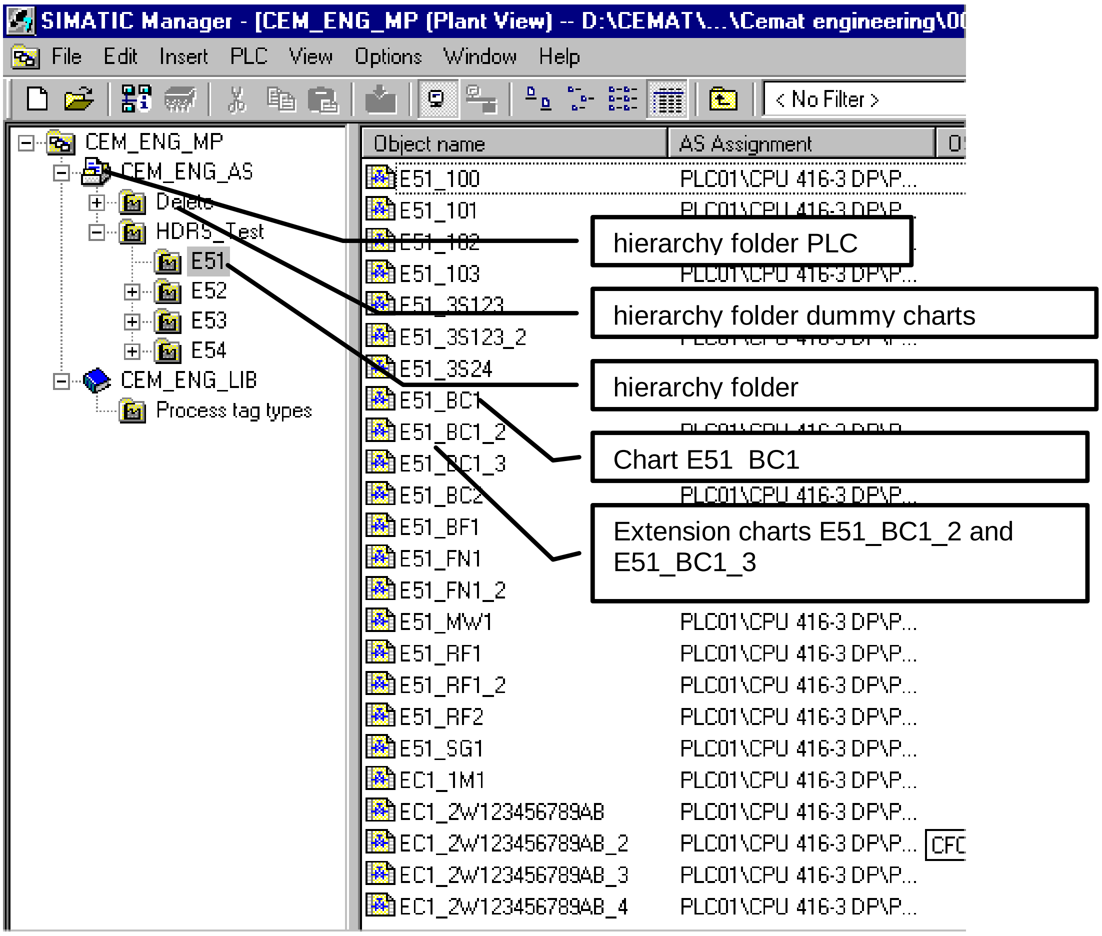
The import can take more than an hour:



After the import is completed, you have to acknowledge with the “Exit“ button:



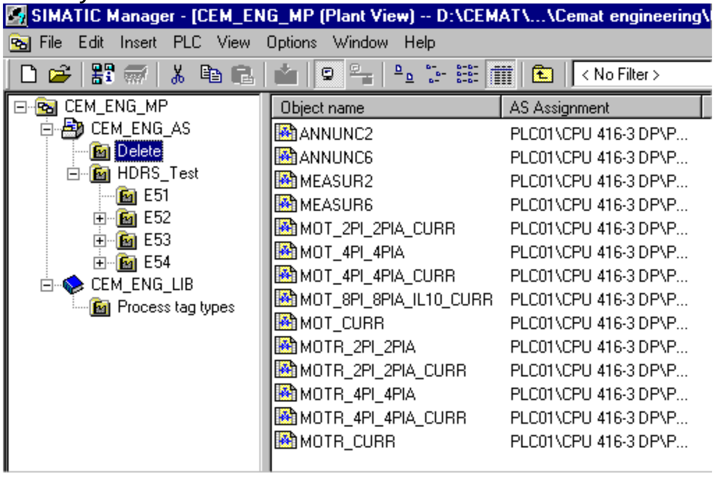
After the import the Plant Hierarchy and the CFC charts have been generated:



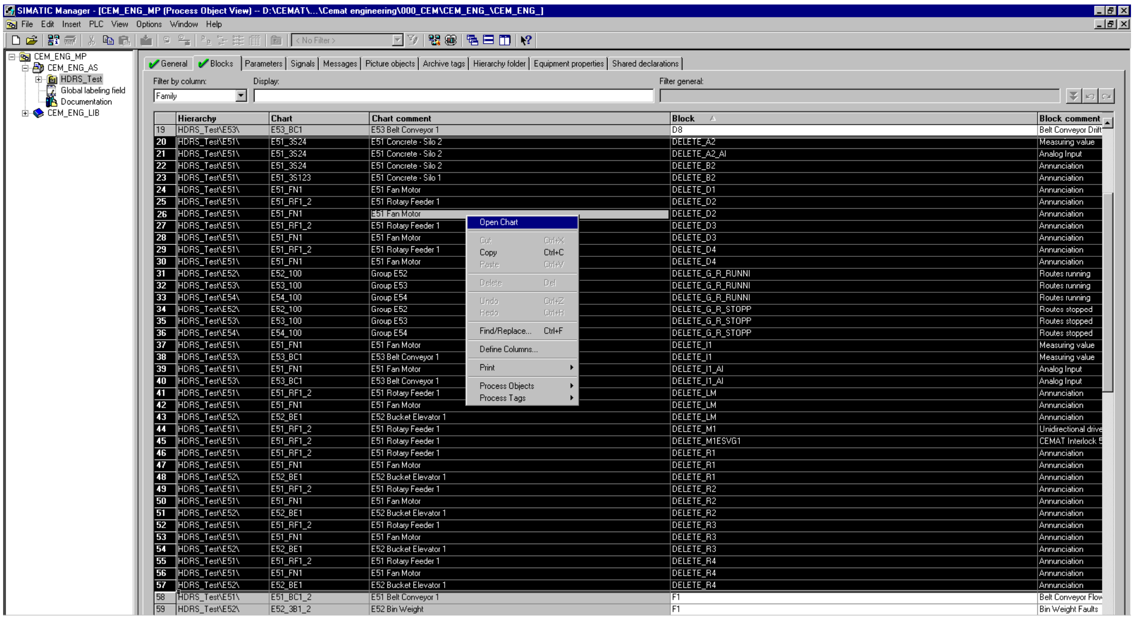
Manually post CFC Engineering

1. Deleting not needed objects (blocks and charts)

A hierarchy folder with the name "Delete" has been generated. In this folder "Delete" are the dummy charts of the not used Process Tag Types. Delete the folder "Delete" inclusive of the dummy charts.



In some charts the IEA has not needed objects (blocks) generated. This happens, if not all objects of the Process Tag Types are needed for the wanted chart. The Engineering Tool renamed these blocks into "DELETE_XX", e. g. in "DELETE_D1", "DELETE_M1" or "DELETE_L1". E. g. the Process Tag Type MOT_4PI_4PIA has 4 drift switch annunciations (D1 to D4). If for a concrete chart this Process Tag Type will be used and the concrete chart has only 2 drift switch annunciations (e. g. D1 and D2), then the not needed drift switch annunciations (D3 and D4) will be renamed into "DELETE_D3" and "DELETE_D4". These blocks have to be deleted manually. With the "Process Object View" of PCS 7 it is easy to find these objects, which name start with "DELETE_".
Open the Process Object View. Select the highest hierarchy folder of your PLC. Select Tab "Blocks". Sort by "Blocks" = block name. Select all blocks, whose names start with "DELETE_" or with "DEL_". Select with the right mouse button the menu "Open chart". Search in all opened charts the blocks, whose names start with "DELETE_" or with "DEL_" and delete them manually. Unfortunately the blocks could not be deleted directly in one step from the Process Object View.



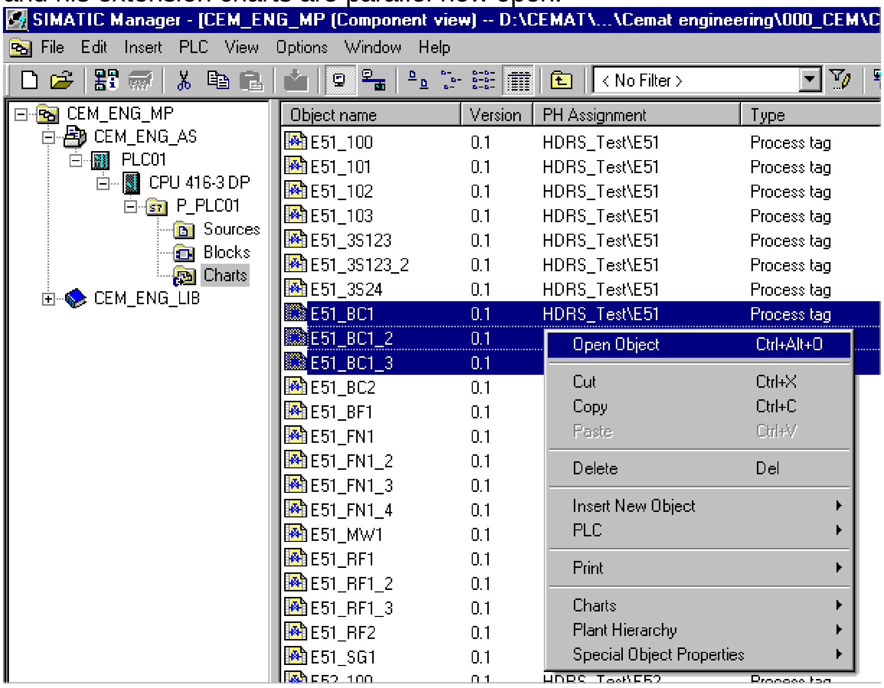
2. Rename objects

Some of the blocks could not get during IEA import the final destination name. This happens, if a block with this name already in the Process Tag Type exists. This is so, even if the already existing block will be later deleted or renamed. All blocks, which can't renamed into the final destination name get the temporary name "RENAME_XX". E. g. an annunciation block C_ANNUNC should be created with the name "LM". Another not needed block in the Process Tag Type has the name "LM". This already existing block "LM" will later on be renamed and deleted. Our annunciation block will get during IEA import the temporary name "RENAME_LM". After chart generation with IEA there exists an annunciation block with the name "RENAME_L1" and the not needed block has now the name "DELETE_LM". The not needed block "DELETE_LM" is already deleted during the first editing step. The block "RENAME_L1" could be now manually renamed into "LM". With the "Process Object View" of PCS 7 it is easy to find these objects, which name start with "RENAME_".
Open the Process Object View. Select the highest hierarchy folder of your PLC. Select Tab "Blocks". Sort by "Blocks" = block name. Select all blocks, whose names start with "RENAME_". Select with the right mouse button the menu "Open chart". Search in all opened charts the blocks, whose names start with "RENAME_" and rename them manually. The blocks could also easily be renamed directly in the Process Object View.

118	HDRS_Test\E51\	E51_101	to Silo E51-3S1	R_RUNNING	Drives running
119	HDRS_Test\E51\	E51_103	to E53-BC1	R_STOPPED	Drives stopped
120	HDRS_Test\E51\	E51_102	to Silo E51-3S2	R_STOPPED	Drives stopped
121	HDRS_Test\E51\	E51_101	to Silo E51-3S1	R_STOPPED	Drives stopped
122	HDRS_Test\E51\	E51_3S24	E51 Concrete - Silo 2	RENAME_LM	Concrete - Silo Lev
123	HDRS_Test\E51\	E51_RF2	E51 Rotary Feeder 2	S1	Rotary Feeder
124	HDRS_Test\E51\	E51_RF1	E51 Rotary Feeder 1	S1	Rotary Feeder Spe
125	HDRS_Test\E53\	E52_BE1_4	E52 Bucket Elevator 1 Aux drive	S1	E52 Bucket Elevat
126	HDRS_Test\E53\	E53_100_2	Group E53 Fan Select	S1	Group E53 Fan Se
127	HDRS_Test\E53\	E51_FN1_3	E51 Fan Motor Select 1	S1	E51 Fan Motor Sel

3. Move objects together from extension charts into the destination chart

Mostly the existing Process Tag Types doesn't fit perfect for the charts, which have to be created. For the rest of the objects, the Cemat Engineering Tool searches for another Process Tag Type = extension chart. This will happen as long as all objects of a chart, which has to be created, are placed in a Process Tag Type. Only the first chart of the first found Process Tag Type get the original chart name, e. g. E51_BC1. Each of the extension charts gets the chart name and a "_" and an extension number. Example: For the chart E51_BC1 we need 2 more extension charts to cover all objects. The three generated charts have the names: E51_BC1, E51_BC1_2, and E51_BC1_3. In the next step you have to move manually all objects of the extension charts (e. g. E51_BC1_2 and E51_BC1_3) into the original chart (e. g. E51_BC1). After the manual moving the extension charts (e. g. E51_BC1_2 and E51_BC1_3) are empty and could be deleted.
Open the Component View and select the chart folder of your PLC. Select the chart and his extension charts and select with the right mouse button the menu "Open Object". The chart and his extension charts are parallel now open.

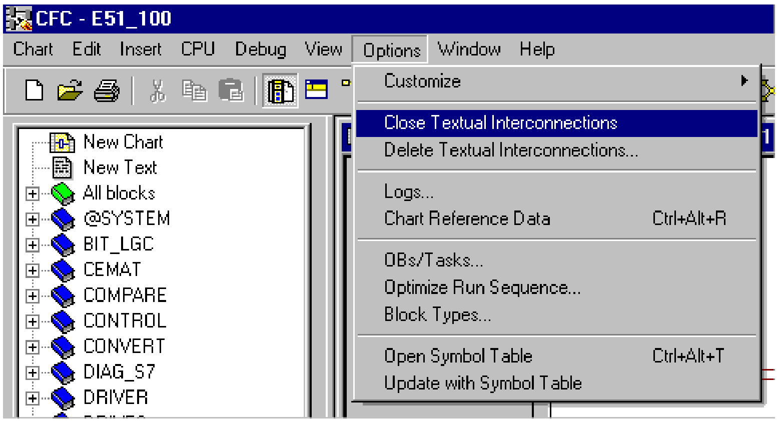


Arrange all windows of the opened charts and select in all charts the view "Overview". Now you can move all blocks from the extension chart into the original chart. The empty extension charts could be deleted afterwards.

⚠ **Attention:** Check that the extension charts are really empty, before you delete them. If you forget some objects there and you delete the extension charts, the generated objects are lost!

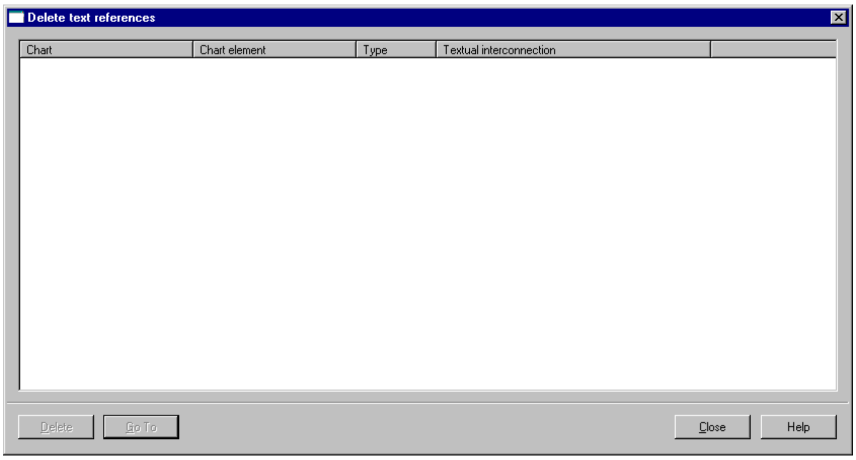
4. Close Textual interconnections

If all blocks are moved to their destination chart and all extension charts are deleted, then you can close the Textual interconnections. During the import of the IEA files, some objects will be create before the group blocks or vice versa, or the blocks are located first in the extension charts. Because of this, the textual interconnections between groups, routes and drives could not be closed and remain as Textual Interconnection. After the moving of objects into their original charts, all Textual interconnection can be closed now.
Choose the menu "Options -> Close Textual Interconnections".



5. Delete Textual Interconnections

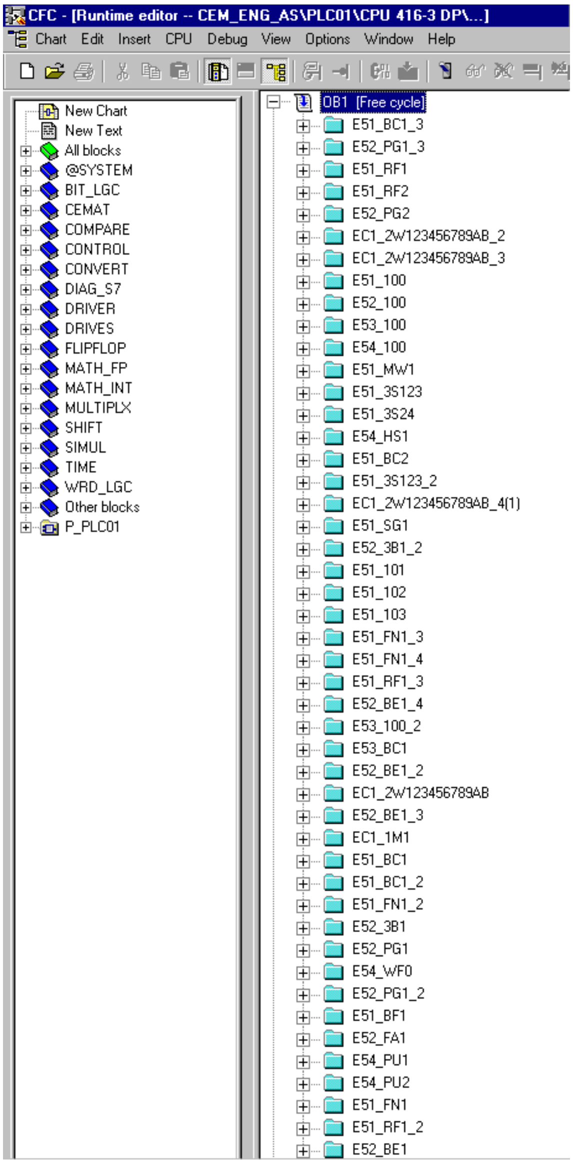
After closing the Textual Interconnections, you can check with the menu "Options -> Delete Textual Interconnections", do there still Textual Interconnections exist or not. Normally there should not remain a "Textual Interconnection". In this case the following window is empty, like in the following screen shot.



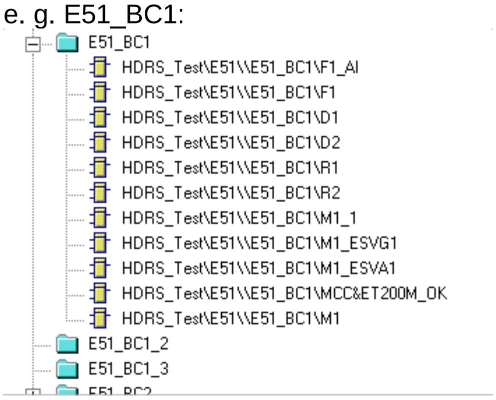
If there is still a Textual Interconnections, then this points to an edit fault. Maybe you forgot to move an object from an extension chart and you delete this object together with the extension chart. Or you during deleting the "DELETE_" object you select also real object and you delete the real object together with a not needed object. Please check the reason for the still existing Textual Interconnection.

6. Edit the runtime groups with the Runtime Editor

The objects of the extension charts e. g. E51_BC1_2, E51_BC1_3 and E51_BC1_4, are placed into the runtime groups E51_BC1_2, E51_BC1_3 and E51_BC1_4. Unfortunately the objects remain in these runtime groups, although the blocks are moved already into their final destination chart.

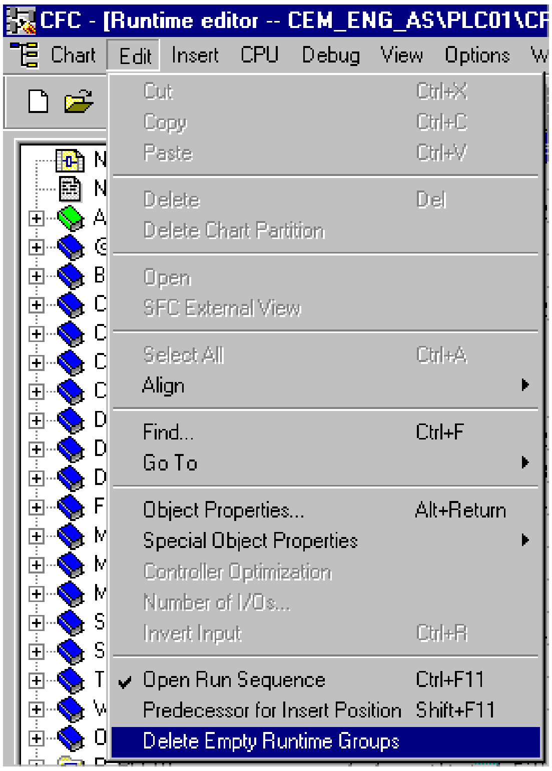


It's better to move the objects (blocks) into the runtime group for the destination main chart, e. g. E51_BC1:

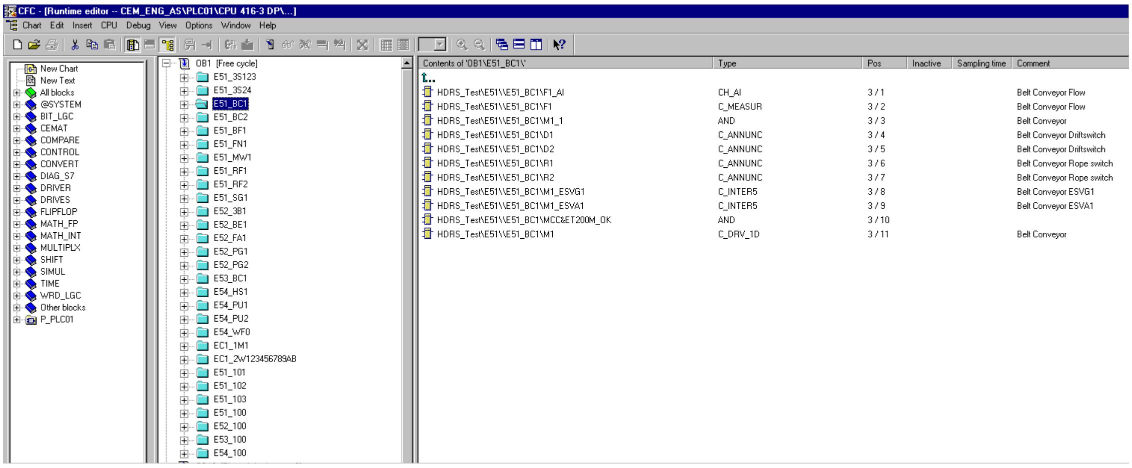


The runtime groups E51_BC1_2 and E51_BC1_3 are empty after the movement.

Delete the empty runtime groups with the menu "Edit -> Delete Empty Runtime Groups".



Result: There are only the runtime groups of the original destination charts.



Appendix A: Structure of the sheets SIGNALTYPES xxx

"xxx" = Project standard code.

A	B	C	D	E	F	G
Keyword header	english	deutsch	Block	parameter	ConvType	IOType
feedback	Feedback ON	Rückmeldung EIN	C_DRV_1D	ERM	NO,NC,DB	DI

Keyword header = Signal function = Column from sheet OBJECTS or signal function from sheet SIGNALLIST
english = description in English
deutsch = description in German
block = Block type PCS 7
parameter = Block parameter = entry in IEA_OBJECTS
ConvType = Signal conversion type from sheet SIGNALLIST
IOType = IO Type from sheet SIGNALLIST

Appendix B: Structure of the sheet IEA_Struc

A	B	C	D	E
ERM	C_DRV_1D	ERM	S|	SymbolName
LOCAL	C_DRV_1D	ELOC	P|	TextRef
VAL_HH	C_MEASUR	VAL_HH	P|	Value
IN_DEL	C_ANNUNC	IN_DEL	P|	Value|S7_shortcut

Column A = keyword from IEA_OBJECTS
Column B = Block type
Column C = Parameter for header in IEA file (header line 2)
Column D = Indicator, weather signal or parameter or message in IEA file (header line 3)
Column E = Indicator, weather value or Textual interconnection or symbol, etc. in IEA file (header line 4)

Appendix C: Structure of the sheet LIST_DATA

A	B	C	D	E	F	G	H
IOType	ConvType	SignalType	SignalLocation	Language	Functions	blocks signal list	blocks lafarge signal list
AI	4-20mA	2w	Field	english	feedback	AND	AND
AO	0-20mA	4w	MCC	deutsch	feedback1	C_ANNUN8	C_AAB
DI	0-10V				feedback2	C_ANNUNC	C_AIB
DO	DB				limitpos1	C_DAMPER	C_BPB
	NO				limitpos2	C_DRV_1D	C_DAB
	NC				available	C_DRV_2D	C_DABMAB
	PT100				overload	C_MEASUR	C_DIB
	Puls				local	C_VALVE	C_M2B
					locstart	CH_AO	C_SPEEDM
					locstart1	OUT	CH_AO
					locstart2	C_SILOP	OUT
					locstop	SINA_GS	C_SILOP
					torque1	C_PROFB	SINA_GS
					torque2		
					speedmon		
					ON		
					ON1		
					ON2		
					alarm		
					operation		
					protection		
					shutdown		
					autoprotection		
					fault1		
					fault2		
					fault3		
					fault4		
					fault5		
					fault6		
					fault7		
					analogvalue		
					actposition		
					AO		
					BIT		
					OUT		
					beltbreak		
					limitpos		
					puls		
					Simocode_I		
					Simocode_Q		

Allowed values for the sheet SIGNALLIST. These data will be copied into the sheet SIGNALLIST to create Excel list fields for input.

In the columns I to L are the keywords for the recognition of current, power and speed measurement. During the object detection the macro searches for the keywords in the signal description of analog values. If the macro finds one of the keywords, then the object function will be changed from a normal analogue value into a current or speed measurement.

I	J	K	L
keyword current english	keyword speed english	keyword Strom deutsch	keyword Geschwindigkeit deutsch
curr	speed	strom	geschw
power		leist	

In the columns M and N (Lafarge) are the allowed block types for the typicals (Process Tag Types).

M	N
blocks typical	blocks lafarge typical
AND	AND
C_ANNUN8	C_AAB
C_ANNUNC	C_AIB
C_DAMPER	C_BPB
C_DRV_1D	C_DAB
C_DRV_2D	C_DABMAB
C_MEASUR	C_DIB
C_VALVE	C_M2B
CH_AO	C_SPEEDM
OUT	CH_AO
C_GROUP	OUT
C_ROUTE	C_SSB
C_SELECT	C_SSDB
C_SILOP	C_SELECT
SINA_GS	C_SILOP
C_VAL_2D	SINA_GS
C_PROFB	IEA_USER
IEA_USER	

In the columns O / P (Lafarge) and Q (2. line) are the allowed keywords IEA_TYPICALS. (All entries, except object names and IEA section names from IEA_OBJECTS).

O	P	Q
keywords IEA typical	keywords IEA typical Lafarge	keywords 2 IEA typical
AND_8	AND_8	

In the columns R (C_ROUTE), S (C_GROUP), U (Lafarge group C_SSB) and T (objects, except groups, routes and select modules) are the allowed keywords for the sheets IEA_OBJECTS.

R	S	T	U
IEA keywords ROUTE	IEA keywords GROUP	IEA keywords Objects	IEA keywords SSB
BLOCKNAME	BLOCKNAME	|	BLOCKNAME
COMMENT	COMMENT	AEVG	COMMENT
|	|	Blockname	|
ONLY |	ONLY |	Blockname_AI	ONLY |
EMPTY	EMPTY	Comment	EMPTY
WEBW	GLA	DESELECTED	SAV
WABW	GHA	EBFA	SSB_BF1
WGWA	G_R_RUNNING	EBFE	SSB_BF2
G_LINK	G_R_STOPPED	EBFE1	SSB_BF3
R_RUNNING	G_RUNNING	EBFEY	SSB_BF4
R_STOPPED	G_STOPPED	EBFEX	Comment_BF1
COMMENT_RUNNING	COMMENT_ROUTES_RUNNING	EMPTY	Comment_BF2
COMMENT_STOPPE D	COMMENT_ROUTES_STOPPE D	Event	Comment_BF3
R_DRIVES_HE	COMMENT_RUNNING	GR_LINK1	Comment_BF4
R_DRIVES_HV	COMMENT_STOPPED	GR_LINK2	Comment_Sequence
COMMENT_HE	R_GRAZ1	IN_DEL	BF1_1
COMMENT_HV	R_GRAZ2	LOCAL	BF2_1
	R_GREZ1	M_SIM	BF3_1
	R_GREZ2	WMOD	BF4_1
	R_HE1	SIM_VAL	SSDB
	R_HE2	OKS	T1
	R_HV1	only |	
	R_HV2	PULS_IN	
	G_DRIVES_HE	QSTP	
	G_DRIVES_HV	GSTP	
	COMMENT_HE	REL_SSM	
	COMMENT_HV	REL_MVC	
	G_LOCAL	SCB	
	COMMENT_LOCAL	SCB_OUT	
		....	
		....	
		....	
		SIG_TXT8	
		SIG_TXT9	
		SIG_TXT10	
		Blockname_USE R	
		U_SIGNAL	
		U_TYP	

In the columns V, W (Holcim) and X (Lafarge) are the allowed keywords for interlock blocks in the sheets IEA_OBJECTS.

V	W	X
IEA keywords Interlock	IEA keywords Holcim Interlock	IEA keywords Lafarge Interlock
ESVG	PINT1	HS
ESVA	PINT2	PE1
EEVG	Prot1	PE2
EBVG	Prot2	
EEVG1		
EEVG2		
EBVG1		
EBVG2		
KEV1		
KBV1		
KSV1		
KEV2		
KBV2		
KSV2		
VEVG		
VBVG		
VSVG		
ProtG		
ProtA		

Appendix D: Structure of the sheet BLOCKFUNCTIONS

A	B	C	D	E	F	G	H
C_DRV_1D	C_BPB	C_DRV_2D	C_M2B	C_DAMPER	C_DABMAB	C_AAB	C_VALVE
main	main	main	Main	Main	main	main	main
aux	aux	aux	Aux	Aux	aux	aux	aux
main_SC		main_SC		main_SC			main_SC
aux_SC		aux_SC		aux_SC			aux_SC

I	J	K	L	M	N	O	P
C_VAL_2D	C_DAB	C_ANNUNC	C_DIB	C_MEASUR	C_AIB	C_ANNUN8	C_SILOP
main	main	alarm	Alarm	analogvalue	analogvalue	annun8	silo
aux	aux	operation	operation	drive_curr	drive_curr		
main_SC		autoprotection	autoprotection	drive_speed	drive_speed		
aux_SC		protection	protection				
			shutdown				

Q	R	S	T	U	V	W	X
C_SPEEDM	AND	CH_AO	OUT	SINA_GS	C_PROFB	C_SIMOS	IEA_USER
speedmon	BIT	AO	OUT	drive_speed	speedmon	simocode	BIT

Allowed object functions (block functions) for the sheet OBJECTS. These data will be copied into the sheet OBJECTS to create Excel list fields for the object function input.

Appendix E: Structure of the sheet HEADER

	A	B	C	D	E	F
1	Header Sheet "OBJECTS"					
2	Chart	Object	BlockText	...	...	...
3						
4						
5	Header Sheet "CHARTS"					
6	CHART	ORG_CHART	TYPE	...	...	...
7						
8						
9	Header Sheet "GROUPS"					
10	CHART	Hierarchy	Group 1	...	...	...
11						
12						
13	Header Sheet "ROUTES"					
14	CHART	Hierarchy	Group 1	...	...	...
15						
16						
17	Header Sheet "SELECT"					
18	CHART	Hierarchy	Group 1	...	...	...
19						
20						
21	Header Sheet "GR_LINK"					
22	CHART	Hierarchy	Group 1	...	...	...

Header lines:
Line 2 = Header line sheet OBJECTS
Line 6 = Header line sheet CHARTS
Line 10 = Header line sheet GROUPS
Line 14 = Header line sheet ROUTES
Line 18 = Header line sheet SELECT
Line 22 = Header line sheet GR_LINK

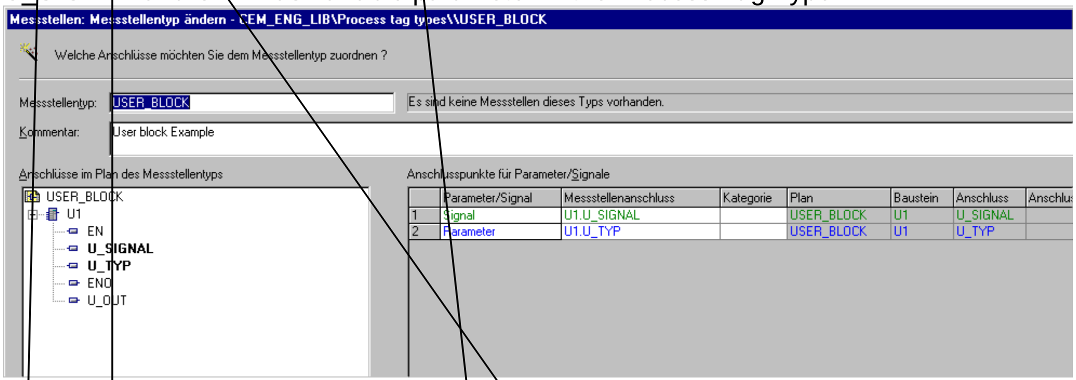
Appendix F: How to integrate a user function block in a Process Tag Type

1. Create your user function block

In our example we use the block IEA_USER = FB1801. The block has two input parameter:
U_SIGNAL – a signal from the signal list SIGNALLIST should be connected
U_TYP – depending on an entry in the signal list SIGNALLIST, a value should be changed.

2. Create Process Tag Type

Place the user block IEA_USER in a CFC and define the CFC as Process Tag Type. Select U_SIGNAL and UTYP as variable parameter in the Process Tag Type:



3. Add the signal type in the Cemat Engineering Tool

Add in the sheet "SIGNALTYPES xxx" the following line:

"xxx" = Project Standard Code.

Keyword	header	English	deutsch	block	parameter	ConvType	IOType
BIT		User Signal	User Signal	IEA_USER	U_SIGNAL	NO,NC,DB	DI

4. Define IEA header lines

Add in the sheet "IEA_Struc" the following line:

U_SIGNAL	IEA_USER	U_SIGNAL	S|	SymbolName
U_TYP	IEA_USER	U_TYP	P|	Value

5. Define name extensions

If you like to add to the object name an extension, e. g. like "_AI" to each CH_AI for each measured value, then you have to add in sheet "IEA_Block_Names" the following line:

Block IEA	**object IEA**	**object extension**
IEA_USER	object	_EXT

In this example, the name of the user block IEA_USER will be always extended by "_EXT".

6. Add in sheet CEM_TYPICALS xxx the Process Tag Type data

"xxx" = Project standard code for Holcim and Lafarge.

Enter in the sheet the Process Tag Type name, the object name and the block type:

found func block	found block	No. Of del. Obj.	No. Of Obj.	Name	Obj1
				USER_BLOCK	U1
				USER_BLOCK	IEA_USER
				USER_BLOCK	
				USER_BLOCK	

7. Add in sheet IEA_TYPICALS xxx the Process Tag Type data

"xxx" = Project standard code for Holcim and Lafarge.

Enter in the sheet the Process Tag Type name, the IEA file name, the object name and the IEA section structure name in IEA_OBJECTS:

A	B	C	D
Typical Name	IEA file	used	
USER_BLOCK	USER_BLOCK00.IEA		U1
USER_BLOCK	USER_BLOCK00.IEA		USER1_B

8. Add in sheet IEA_OBJECTS xxx the IEA section structure

"xxx" = Project standard code.

Enter in the sheet the IEA section structure for user block:

A	B	C	D	E	F
USER1_B	IEA_USER				
U_SIGNAL	|	Blockname_USER	|	Comment	U_TYP

9. Add in sheet "IEA keyword" the possibilities for a value parameter

List all parameters, which value should be modified, depending on entries from the signal list. Enter the keyword from IEA_OBJECTS and all possible entries in the signal list and the reaction value for the parameter.
Example:
The parameter U_TYP should be changed depending from values from the signal list. The values from the signal list shall be taken from column 7 = "ConvType". If the "ConvType = "NC", then the U_TYP shall get the value "1". If the "ConvType" = "NO", then the U_TYP shall get the value "0". For all other "ConvType" values, U_TYP shall get the default value "0".

keyword	**column object list (1. column = 0)**	**default**	**possibility 1**	**value 1**	**possibility 2**	**value 2**
U_TYP	7	0	NC	1	NO	0

Check Tool

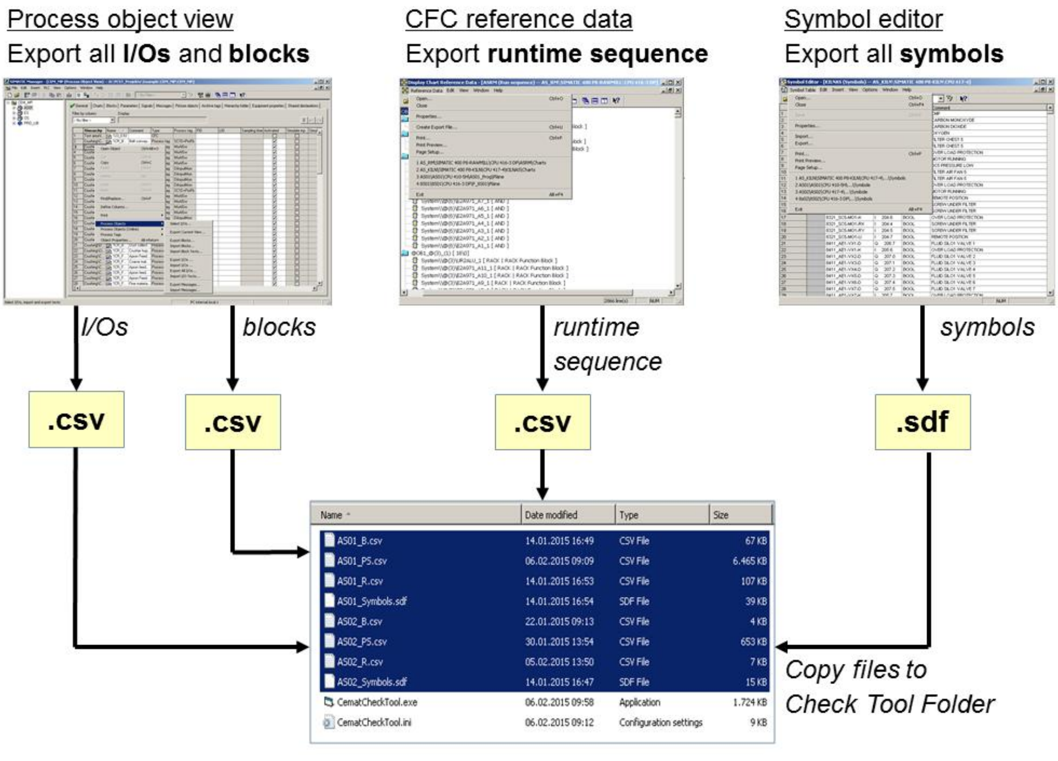
The CEMAT Check Tool V8.2 checks on programming errors resulting from e.g. missing or wrong connections, incorrect installation of CEMAT (system) blocks or inconsistent parameterization.

The CEMAT Check Tool V8.2 can be used for Cemat Project Standard 000, 001, 004, 006, 007, 023, 024, 025, 026 and 028. If you need to check other Project Standards please call the competence center for an updated version.

⚠ **Note:** The CEMAT Check Tool V8.2 can check multiple user projects at the same time, but each single project must contain only one AS. Multiple AS per user project are not supported.

Overview: Extraction of required data from CEMAT project

The information which is needed has to be exported from the PCS 7 project as follows:



If multiple projects shall be checked at the same time, the following steps have to be repeated for each user project.

The file names are standardized and spelling must be exact (case sensitive!)

For Chart Reference data use name	*MyProject*_**R.csv**
For Export I/Os use name	*MyProject*_**PS.csv**
For Export all blocks use name	*MyProject*_**B.csv**
For the Symbol Table use name	*MyProject*_**Symbols.sdf**

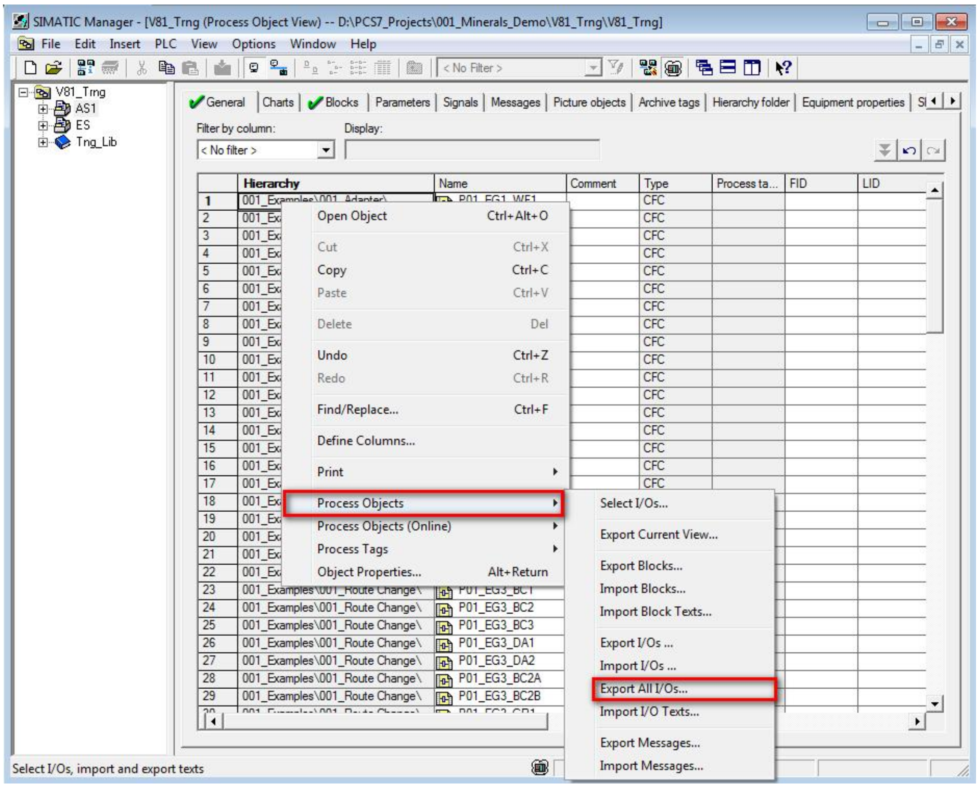
MyProject hast to be replaced by the name of the AS.

Export data from the Process Object View

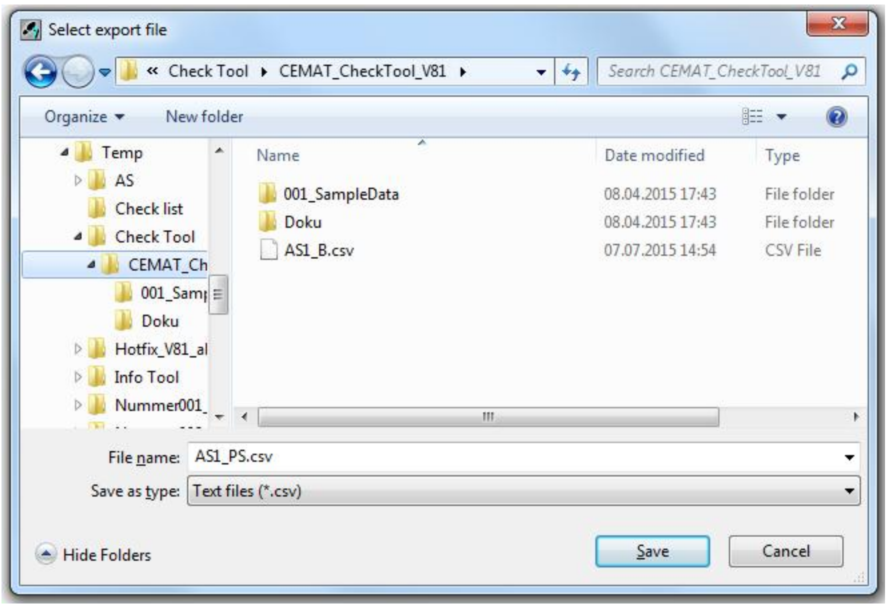
It is necessary to create two export files from the process object view. To do so, please follow the steps described below.

Export all I/Os

1. Select the user project
2. Right click on any object and select “Export All I/Os”

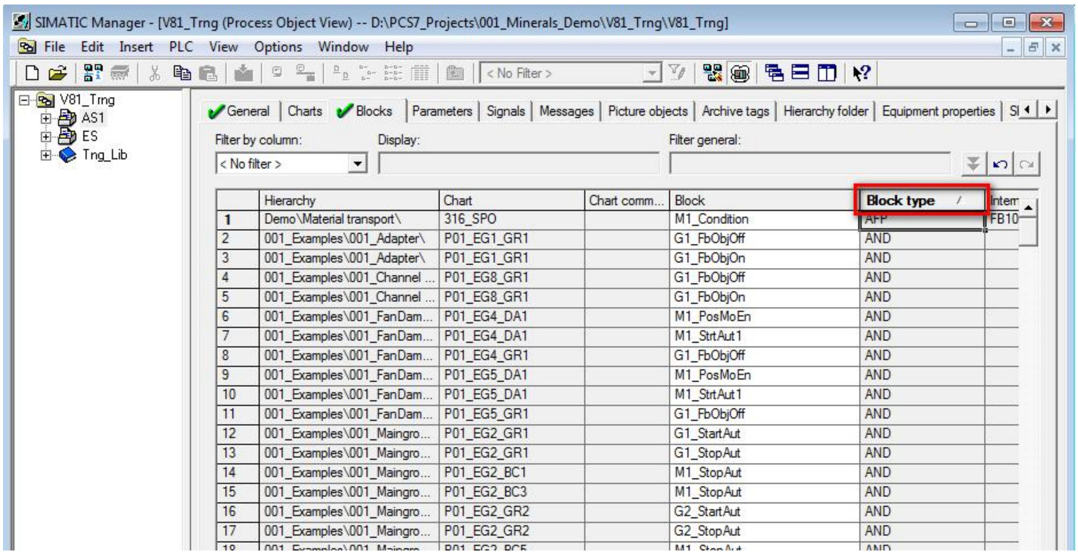


3. Press “Save” – Don’t change the file name!
   The file always must be named like *MyProject*_PS.csv!

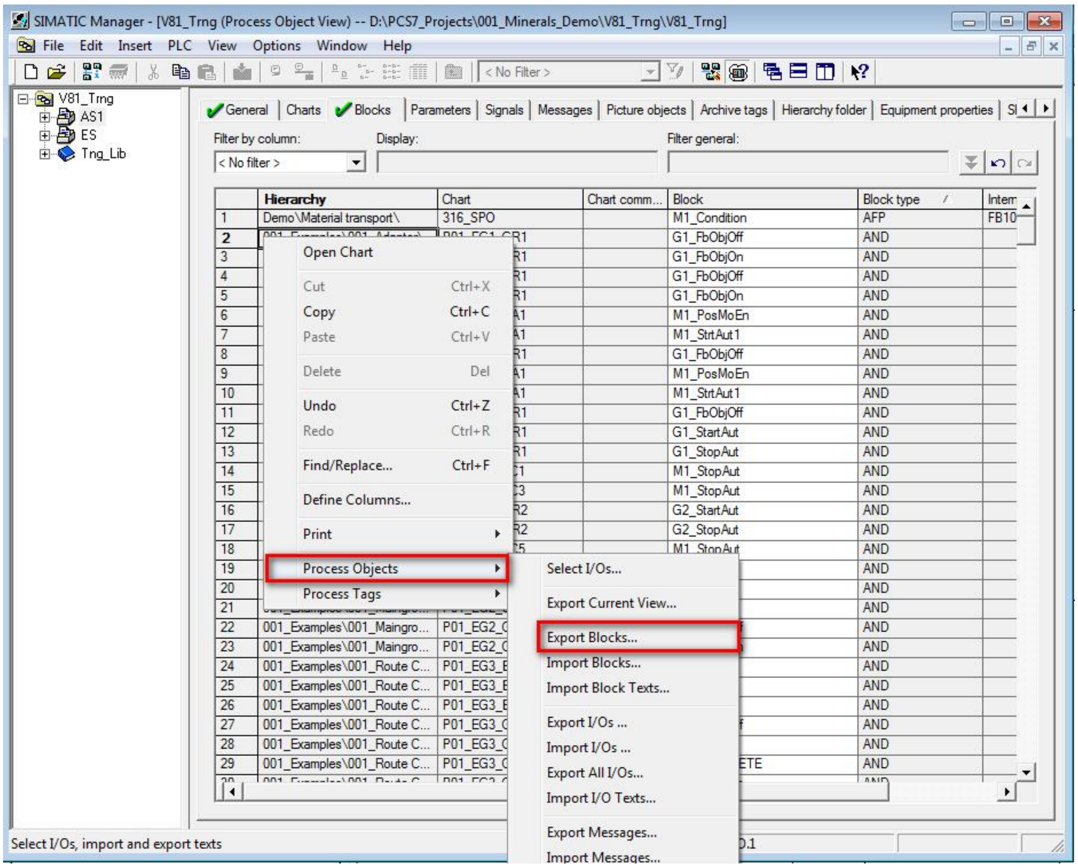


Export all blocks

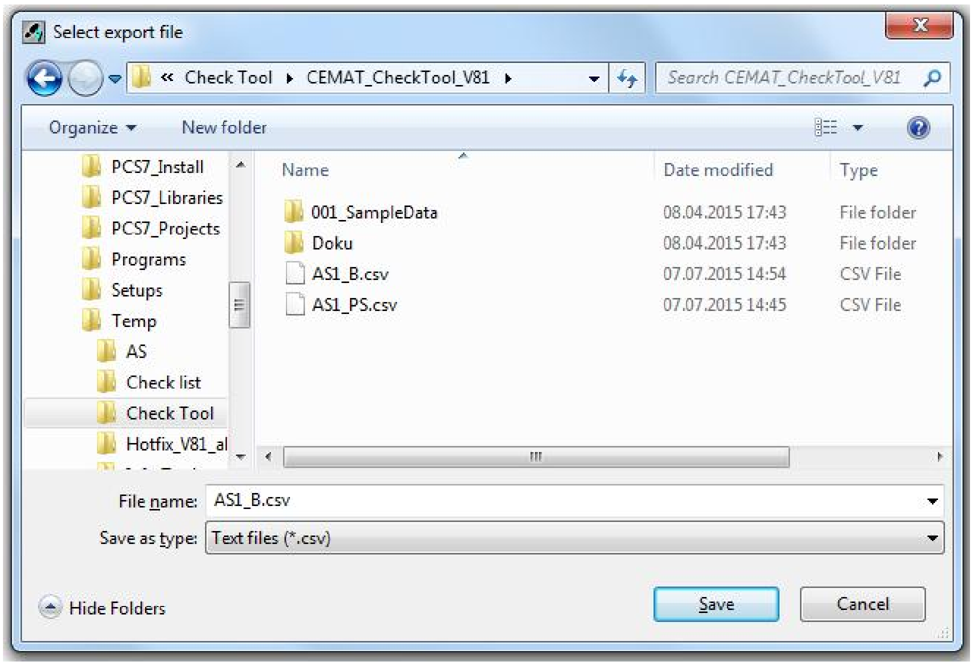
1. Select the user project
2. Change to the tab “Blocks”
3. **Sort by column “Block type” in alphabetical ascending order**



4. Right click on any object and select “Export Blocks”



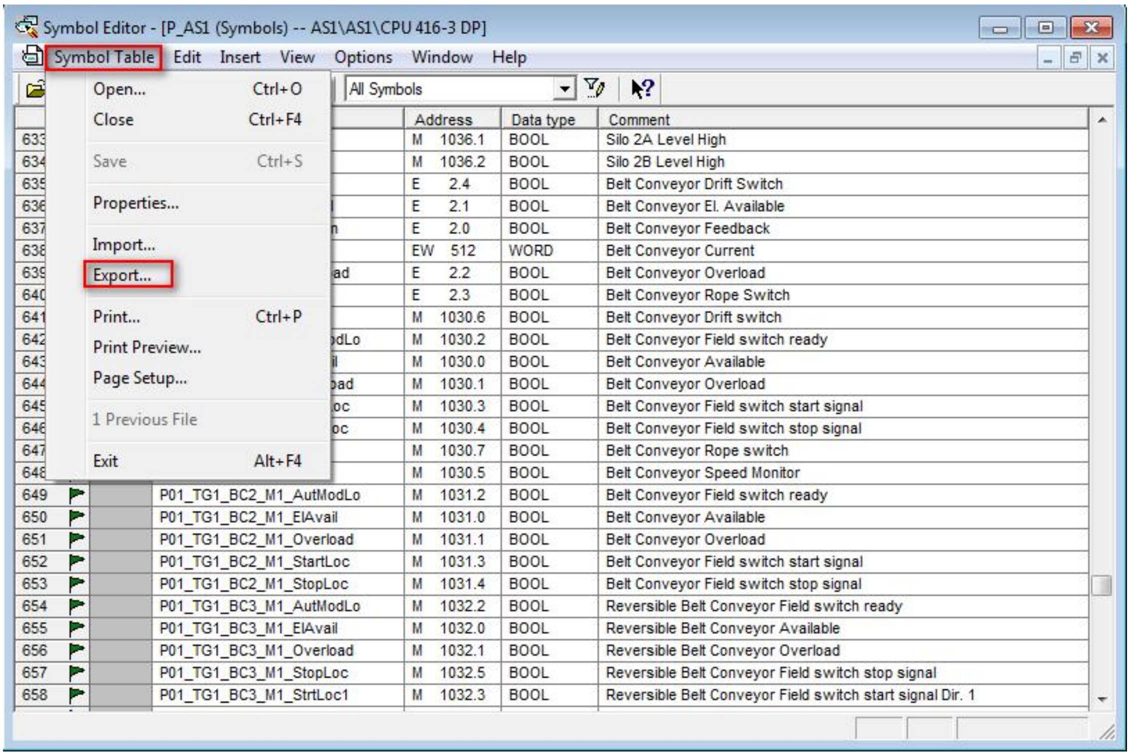
5. Press “Save” – Don’t change the file name!
   The file always must be named like *MyProject*_B.csv!



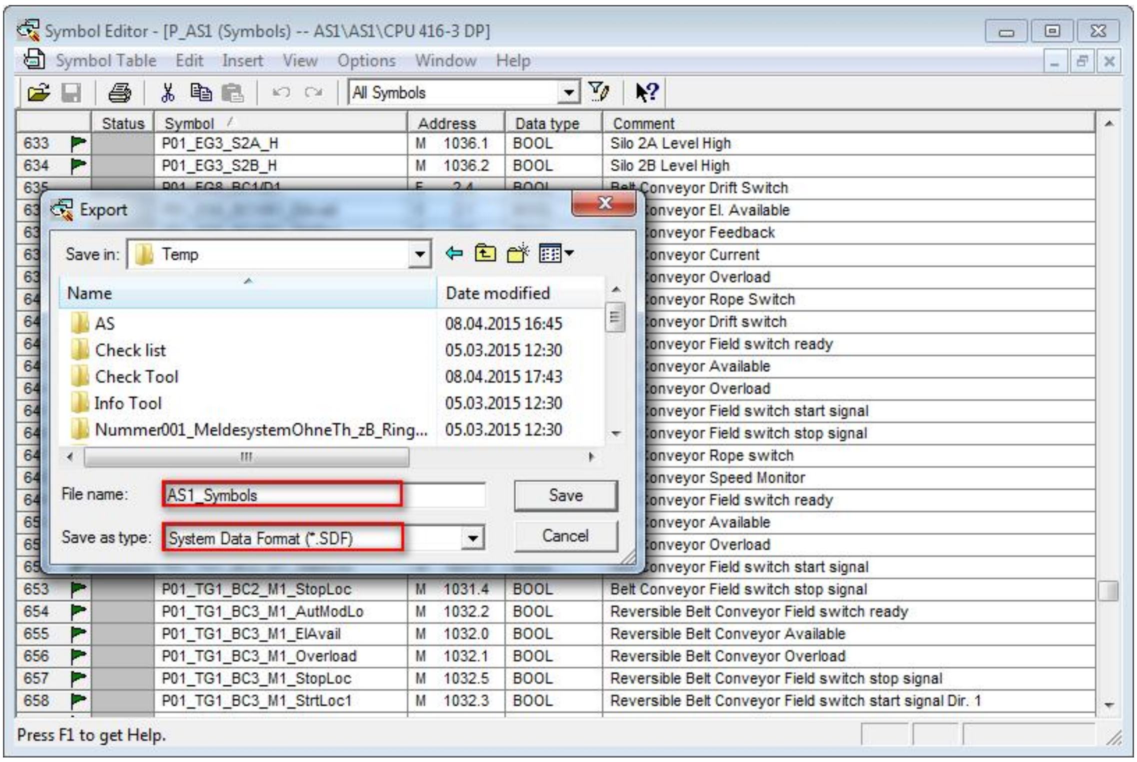
Export symbol table

It is necessary to export the symbol table. To do so, please follow the steps described below.

1. Open the symbols table and select Symbol Table → Export…



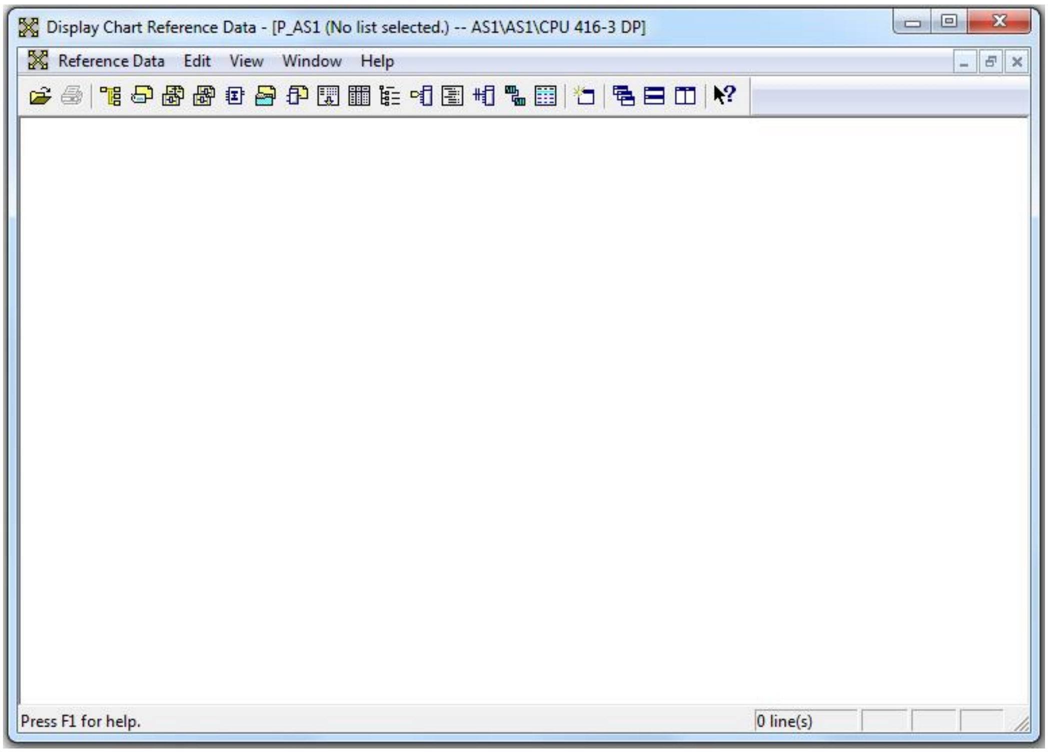
2. Change the export file name to *MyProject*_Symbols.sdf. Please ensure that the file type is "sdf":



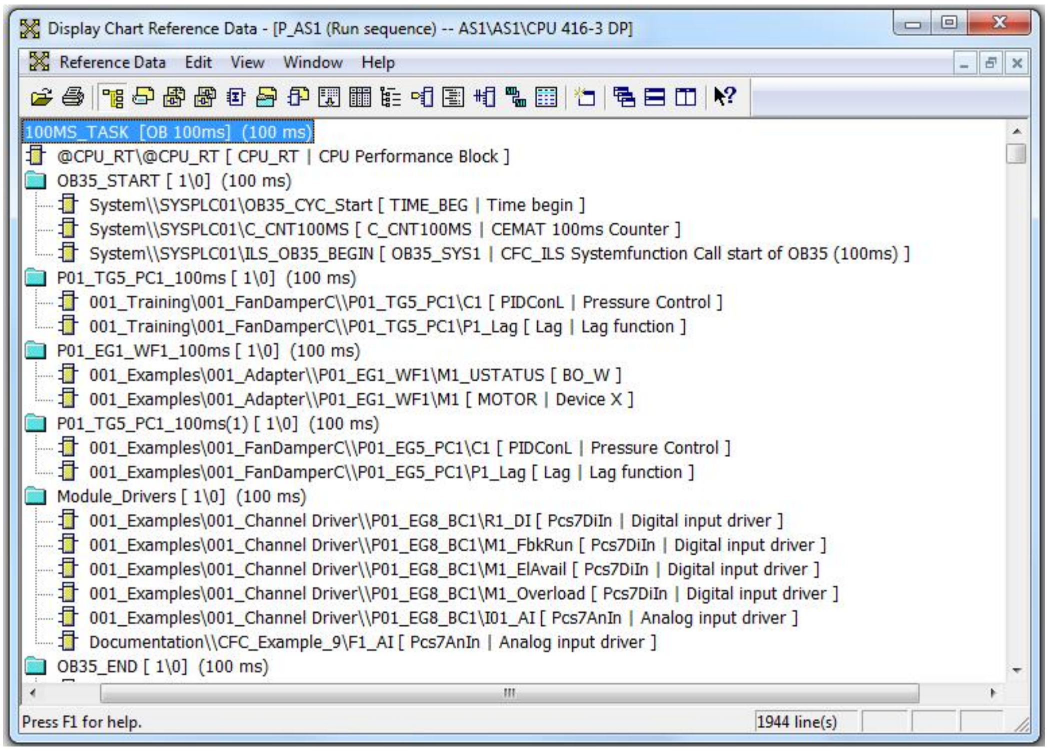
Export runtime sequence

It is necessary to export the runtime sequence from the CFC reference data. To do so, please follow the steps described below.

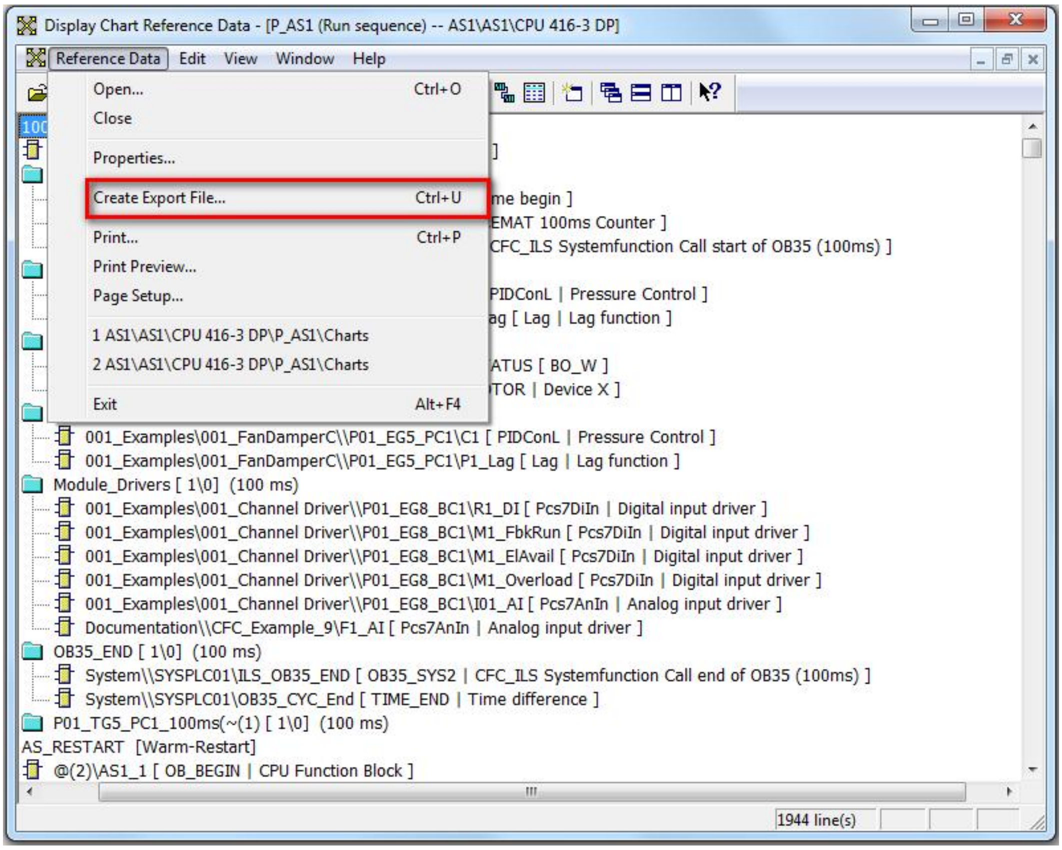
1. Open any CFC from your project and go to the Chart Reference Data



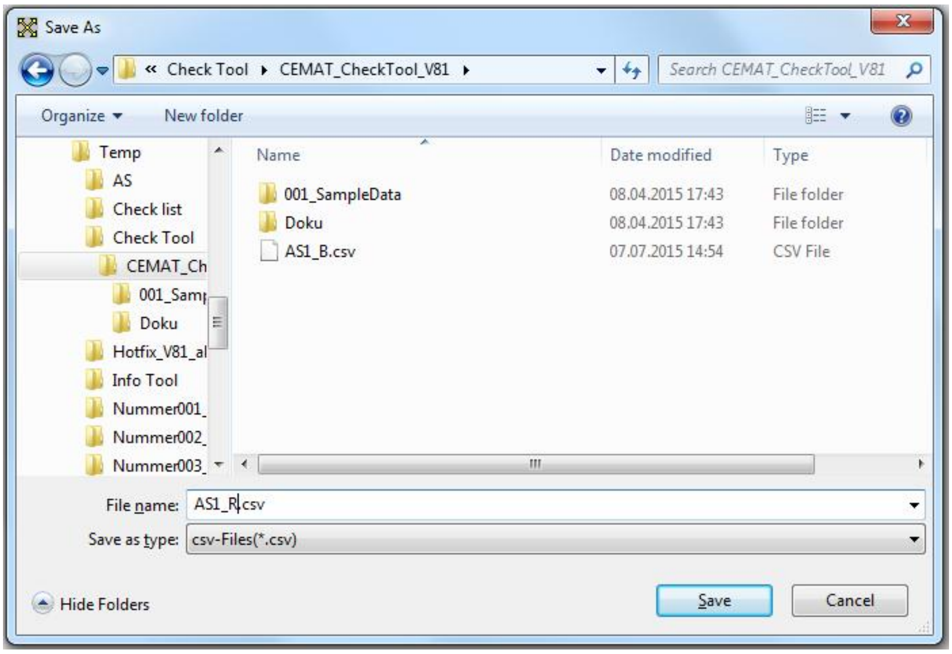
2. Select “Run Sequence”



3. Go to “Reference Date → Create Export File…”

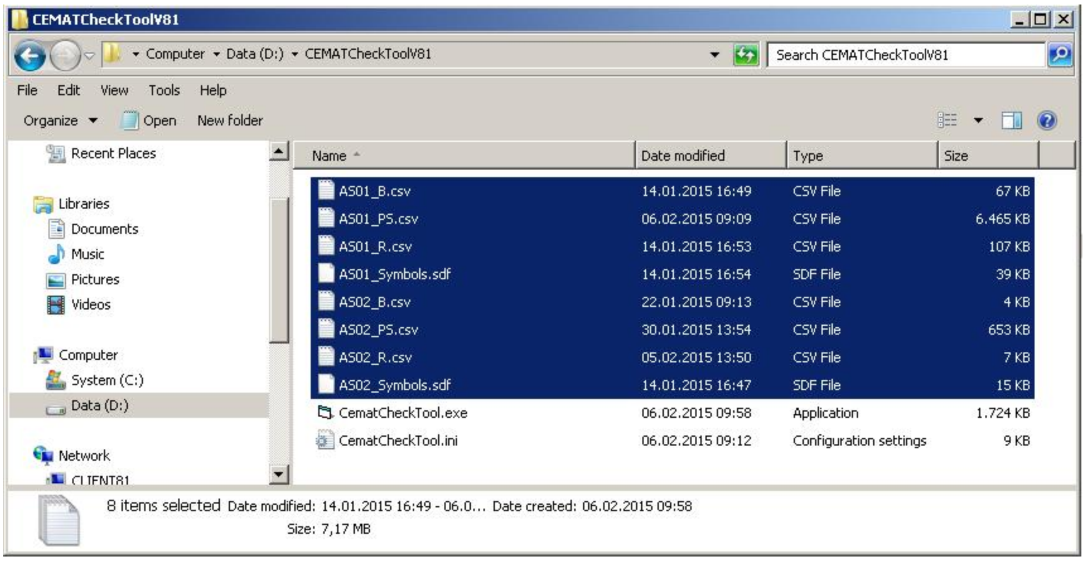


4. Change the export file name to *MyProject*_R.csv



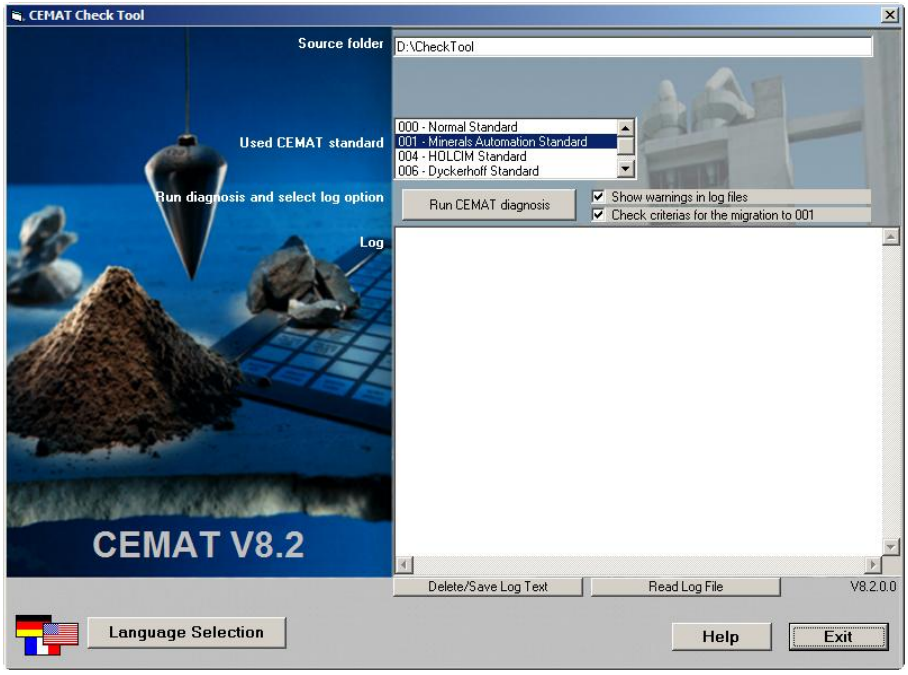
Copy files to check tool folder

Copy all exported files (for all projects which shall be checked) into the folder of the CEMAT Check Tool V8.2.



Running the check tool and evaluation of the results

Open the CEMAT Check Tool V8.2 by starting the program “CematCheckTool.exe”.

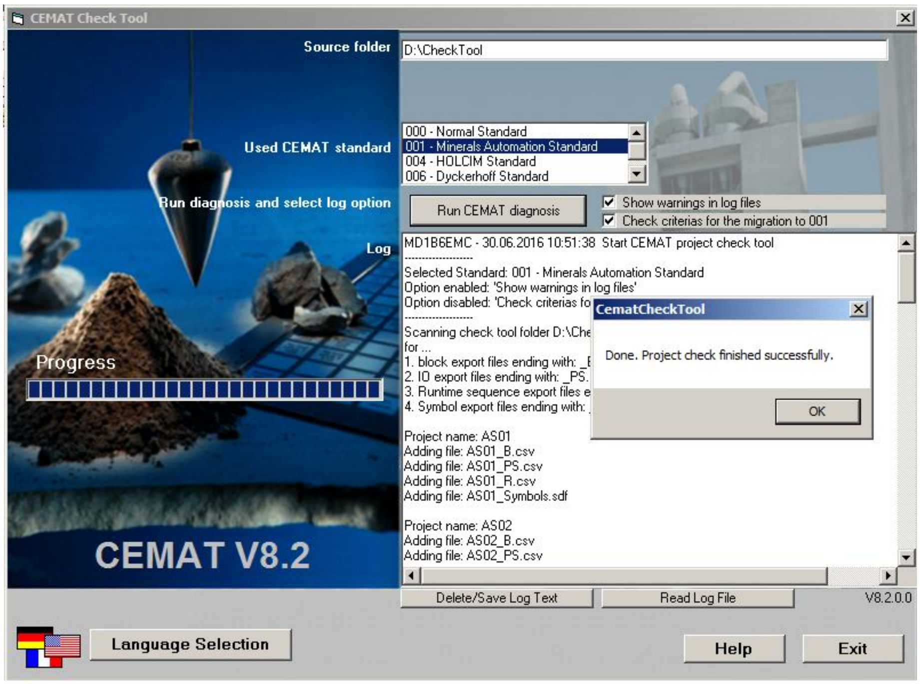


Select the CEMAT standard you have used in your PCS 7 project.

Note: The CEMAT standard must be the same for all projects which you want to check at the same time. Currently the Project Standards 000, 001, 004, 006, 007, 023, 024, 025, 026 and 028 can be checked.

Execute the CEMAT Check Tool V8.2 by clicking the “Run CEMAT diagnosis” button.

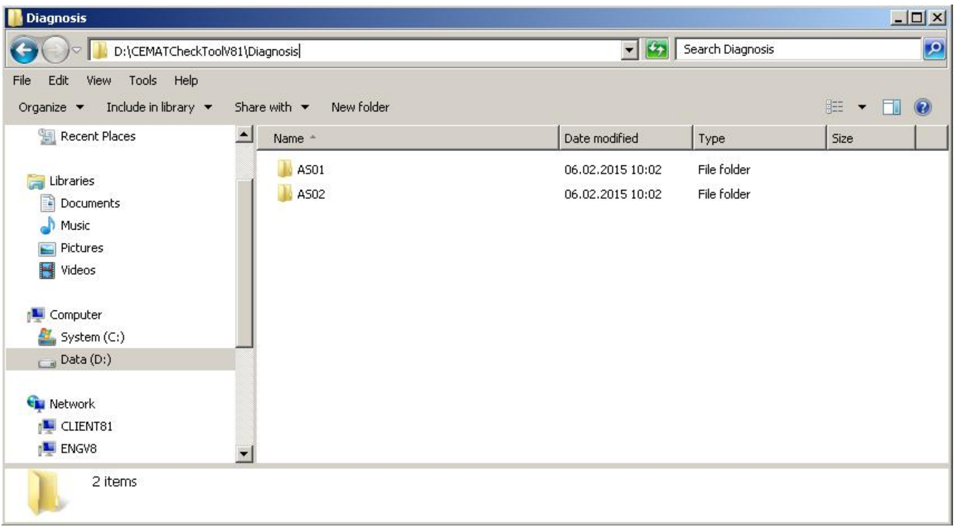
Wait until the message appears that the CEMAT Check Tool V8.2 has finished its check.



If any export file could not be found (e.g. because file name or file type is not as expected), it will create an error message in the log file.

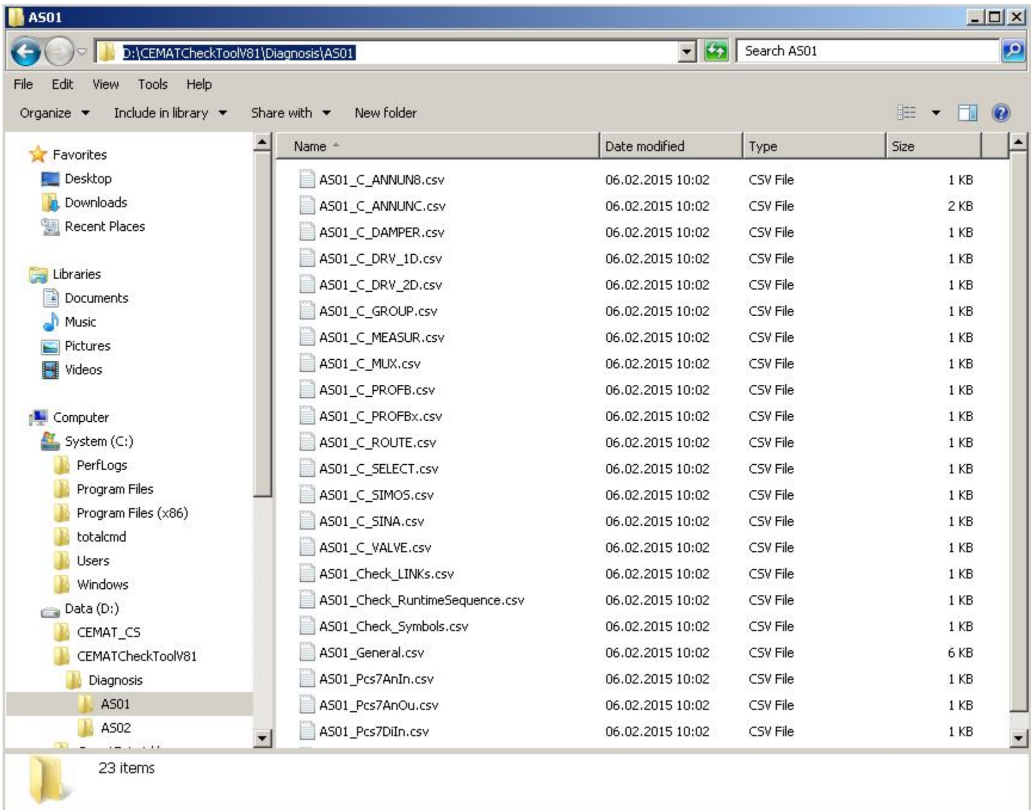
The log file (in the log window) only gives a short summary about what has been checked and which errors have been found. The log file will be saved as “CematCheckProtocol.txt” after closing the CEMAT Check Tool V8.2

The detailed diagnosis entries have been created now in the folder “…\Diagnosis_xxDATExx_xxTIMExx\”.



There is one folder for each project which has been checked. Inside each folder there are now several files for block types and general check criteria. Please note that information from general files and block type files can be redundant.

However, it is always necessary to go through all files for getting all found errors and warnings.



Log file **xxx_General.csv** contains general information such as Number of installed block types and general faults/warnings within the AS.

Log file **xxx_Check_LINKs.csv** contains missing links between Objects and groups. Each CEMAT Object must be linked to a CEMAT Group block (either directly or indirectly).

Log file **xxx_Check_RuntimeSequence.csv** contains incorrect installation positions of the CEMAT blocks.

Log file **xxx_Check_Symbols.csv** contains engineering errors in the Symbols file such as usage of memory range which is reserved for CEMAT Standard blocks and therefore excluded for the CFC.

Log file **xxx_Migration_relevant.csv** contains actions which have to be carried out and corrections which have to be made if the project shall be migrated to Minerals Automation Standard (001).

In addition you find one log file for each Object Type, containing the following information:

Classification:	Fault or Warning
Number:	Fault or Warning code
Description:	Short description of the fault or warning
CFC:	Chart name of the object
Block name:	Block name of the object
Comment:	

For more detailed information see chapter below or study the CEMAT Object descriptions for the corresponding interface.

Check functions

The check functions differ between the different CEMAT Project standards. The following list contains the check functions for **Project Standard 000 – Normal Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
18	ERM not connected
19	Function block not enabled, parameter EN = 0.
21	ERM1 not connected
22	ERM2 not connected
23	KWE1, VE1 not connected
24	KWE2, VE2 not connected
25	MST0, PV, FLS1 – FLS7 not connected
26	START1, START2 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.

66	CEMAT function blocks installed in the runtime sequence in OB35 after the last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD
72	SW_SPEED, SSM not connected if REL_SSM = 1
74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
34	EMFR, VMFR, MMFR, KMFR, UMFR = 0, Messages are permanently suppressed
37	EBFE, VBFE, WEBW, GEBG not connected
38	EBFE1 or EBFE2 (KBFE1 or KBFE2) is not connected
39	EBFA, VBFA, WABW, GABG not connected, in case EBFE, VBFE, WEBW, GEBG is connected
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 004 – Holcim Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
18	R not connected
19	Function block not enabled, parameter EN = 0.
21	RY not connected
22	RX not connected
23	ZY, Y not connected
24	ZX, X not connected
25	MST0, IN_SIG, FLS1 – FLS7 not connected
26	START1, START2 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
40	ProFb, PRFB not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the

	last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD
72	PULS_SSM not connected if EN_SSM = 1
74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
34	EMFR, VMFR, MMFR, KMFR, UMFR = 0, Messages are permanently suppressed
37	EBFE, VBFE, WEBW, GEBG not connected
38	EBFEY or EBFEX (KEBY or KEBY) is not connected
39	EBFA, VBFA, WABW, GABG not connected, in case EBFE, VBFE, WEBW, GEBG is connected
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 006 – Dyckerhoff Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
18	RME not connected
19	Function block not enabled, parameter EN = 0.
21	RME1 not connected
22	RME2 not connected
23	ESL not connected
24	ESR not connected
25	MST0, PV, FLS1 – FLS7 not connected
26	START1, START2 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the

	last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD
72	SW_SPEED, SSM not connected if REL_SSM = 1
74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
34	EMFR, VMFR, MMFR, KMFR, UMFR = 0, Messages are permanently suppressed
37	EBFE, VBFE, WEBW, GEBG not connected
38	EBFE1 or EBFE2 (KBFE1 or KBFE2) is not connected
39	EBFA, VBFA, WABW, GABG not connected, in case EBFE, VBFE, WEBW, GEBG is connected
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 007 – Heidelberg Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
18	ERM not connected
19	Function block not enabled, parameter EN = 0.
21	ERM1 not connected
22	ERM2 not connected
23	KWE1, VE1 not connected
24	KWE2, VE2 not connected
25	MST0, PV, FLS1 – FLS7 not connected
26	START1, START2 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the

	last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD
72	SW_SPEED, SSM not connected if REL_SSM = 1
74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
34	EMFR, VMFR, MMFR, KMFR, UMFR = 0, Messages are permanently suppressed
37	EBFE, VBFE, WEBW, GEBG not connected
38	EBFE1 or EBFE2 (KBFE1 or KBFE2) is not connected
39	EBFA, VBFA, WABW, GABG not connected, in case EBFE, VBFE, WEBW, GEBG is connected
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 023 – Vigier Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
19	Function block not enabled, parameter EN = 0.
25	MST0, PV, FLS1 – FLS7 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD

74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 024 – Bushehr Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
19	Function block not enabled, parameter EN = 0.
25	MST0, PV, FLS1 – FLS7 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on

	MV_PHYS, PV / MV_CARD
74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 025 – CAIMA Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
19	Function block not enabled, parameter EN = 0.
25	MST0, PV, FLS1 – FLS7 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD

74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 026 – Alsen Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
19	Function block not enabled, parameter EN = 0.
25	MST0, PV, FLS1 – FLS7 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD

74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

The following list contains the check functions for **Project Standard 028 – Rossi Standard**:

Faults

Number	Description
1	Usage of bit memory with an address below M200.0. This range is reserved for CEMAT system functions and it must not be used by any user application.
2	CEMAT system chart name is used more than once (if check is performed for multiple programs/CPUs).
3	CEMAT system chart name does not match the CEMAT naming convention.
4	CEMAT system chart is not within the process hierarchy.
8	CEMAT system function block called more than once within one S7 program.
11	GR_LINK and O_LINK are both connected
12	MUX_LINK and O_LINK are both connected
13	GR_LINK not connected to a group (C_GROUP) or route (C_ROUTE) block
14	G_LINK not connected to a group (C_GROUP)
15	MUX_LINK not connected to a multiplexer (C_MUX) block
16	MUX_IN not connected to a multiplexer (C_MUX) block
17	O_LINK not connected to C_SEND_G (only C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE)
19	Function block not enabled, parameter EN = 0.
25	MST0, PV, FLS1 – FLS7 not connected
35	MV_PHYS / PV is not connected
36	MV_CARD is not connected
41	I_ADDR or O_ADDR of C_SIMOS or C_SINA is not configured (=0)
58	GREZ, WREZ is not connected
59	GRAZ, WRAZ is not connected
60	Not all CEMAT system function blocks have been found. Note: Depending on your CEMAT version (<8.1) this error has to be verified. Check the SYSPLC00 chart of the installed CEMAT library if the as “missing” defined blocks are already part of your CEMAT version.
61	CEMAT function blocks installed in the runtime sequence in OB1 prior to the first CEMAT system function blocks.
62	CEMAT function blocks installed in the runtime sequence in OB35 prior to the first CEMAT system function blocks.
63	Runtime sequence of CEMAT system function blocks is wrong in OB1.
64	Runtime sequence of CEMAT system function blocks is wrong in OB35.
65	CEMAT function blocks installed in the runtime sequence in OB1 after the last CEMAT system function block.
66	CEMAT function blocks installed in the runtime sequence in OB35 after the last CEMAT system function block.
68	All CEMAT blocks are not called in the same cycle (=OB). This does NOT include blocks from the CEMAT system chart. Please check your runtime sequence.
69	PV_In not connected on channel driver block
70	PV_Out not connected on channel driver block
71	Configuration of parameter TYP does not correspond to connection on MV_PHYS, PV / MV_CARD

74	Basic Type of block C_SIMOS, C_SIMO_A, C_SIM_AD is not set to 1 or 2
75	Cycle time for communication of block C_SEND_G or C_RECV_G is set 0
76	Watchdog time of block C_SEND_G or C_RECV_G is set <=1
77	No S7 connection ID configured for block C_SEND_G or C_RECV_G

Warnings:

Number	Description
5	PLC number (parameter PLC_NO of block C_FB_PLC) does not match the PLC number of the CEMAT system chart name.
9	No CEMAT link (GR_LINK, MUX_LINK or O_LINK) connected
10	Link output G_LINK (C_GROUP) or R_LINK (C_ROUTE) not connected
20	Function block dynamically enabled. Parameter EN is interconnected.
73	Signal is not used in program, check if signal is spare

Migration relevant:

Number	Description
79	Dublicate block name found in one CFC (check upper/lower case)
80	Block name has a leading blank which has to be removed
81	Square bracket in block name which has to be removed
82	Apostrophe in block name which has to be removed
83	Degree sign in block name which has to be removed
84	Asterisk in block name which has to be removed
85	Accent in in block name which has to be removed
86	CEMAT system chart SYSPLC00 from MinAS library (001) has to be implemented
87	No CFC internal connection on a block parameter which is a STRUCT in MinAS (001)
88	Parameter doesn’t exist in MinAS (001), workaround required

Check list

Content

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\10_Check_list_009.doc

General

This document contains check lists for

- Plant configuration
- Project Settings
- AS Engineering
- OS Engineering
- Function test

If you follow these lists you can avoid general mistakes in each phase of the engineering.

We recommend to work through the check list before the FAT, in order to avoid deviciencies in advance.

Project Name:	
Location:	
Customer:	

PCS 7 Version:	
Cemat Version:	
Project Standard:	

Engineering (Company):	
Location:	
responsible Engineer:	

inspected by:	
Date of inspection:	
Date of Report:	

Hardware definition

PC Stations:

Station Type	Computer Name *)	WORK-GROUP	Infobus TCP/IP Address	Plant Bus MAC Address	MS Login	Password	WinCC Project Name	WinCC Login	Password
Engineering Station 1	*ENG01*								
Engineering Station 2	*ENG02*								
Server1	*SERVER1A*								
Server1 (Standby)	*SERVER1B*								
Server2	*SERVER2A*								
Server2 (Standby)	*SERVER2B*								
Process Historian	*PH01*								
Process Historian (Standby)	*PH02*								
Web Server	*WEBSRV01*								
Open PCS 7	*OPCSRV01*								

*) Only Letters and Numbers are allowed for Computer names (no special characters)

Station Type	Computer Name	WORK-GROUP	Infobus TCP/IP Address	Plant Bus MAC Address	MS Login	Password	WinCC Project Name	WinCC Login	Password
OS Client 1	*OSC01*			---					
OS Client 2	*OSC02*			---					
OS Client 3	*OSC03*			---					
OS Client 4	*OSC04*			---					
OS Client 5	*OSC05*			---					
OS Client 6	*OSC06*			---					
OS Client 7	*OSC07*			---					
OS Client 8	*OSC08*			---					
OS Client 9	*OSC09*			---					
OS Client 10	*OSC10*			---					
OS Client 11	*OSC11*			---					
OS Client 12	*OSC12*			---					
OS Client 13	*OSC13*			---					
OS Client 14	*OSC14*			---					
OS Client 15	*OSC15*			---					
OS Client 16	*OSC16*			---					

AS Stations:

Station Name	**Function**	**Plant Section (AREA)**	**AS Number (for SYSPLCxx)**	**CPU Type**	**CP143 MAC Address**	**S7 Program Name**
			01			
			02			
			03			
			04			
			05			
			06			
			07			
			08			
			09			
			10			
			11			
			12			
			13			
			14			
			15			
			16			
			17			
			18			
			19			
			20			
			21			
			22			

Project Settings (according to Engineering Manual Chapter 3)

Designations:

Item	**Remark**	**Status**
Name of the Multiproject	Should discribe the complete Plant	
Name of the Project(s)	The project name for the OS (Servers) must not have an Underscore (because of Server Prefix)	
Station Name of the AS	has to be unique in the complete MP e. g. AS01	
AS Program Folder	has to be unique in the complete MP e. g. P_PLC01	
Station Name OS PC	Following the rules of the PCS 7 Readme and the WinCC Help the Computer name should not contain any special Character. Unterscore is also not allowed for Servers!!! **Caution:** Once WinCC (and SQL Server) has been started, the modification of the station name requires a new installation of the Operating System, PCS 7 ……..	
Station Name of the OS (as used in the PCS 7 Project and in Station Configuration Editor)	Should match with the real Station name.	
OS Project name	has to be unique in the complete MP	
Name of the System Chart	Must be e. g. SYSPLC01 has to be unique in the complete Control System	

Cemat Installation:

Item	Remark	Status
Cemat Version	The actual version is Cemat V8.2 (+ Hotfix?)	
Screen Resolution	Possible settings: 1680x1050, 1280x1024, 1600x1200, 1920 x1080 or 1920x1200	
Windows Appearance	must be Classic view (not XP)	
Character Size	must be small (normal) characters	
Firewall	On the Servers the CEMATRS.exe must be listet under "Exceptions".	

Hardware Configuration:

Item	Remark	Status
CPU Setting	Settings according to Installation Manual?	
CP Setting	Settings according to Installation Manual?	

Hardware Configuration:

Item	Remark	Status
Network connections	Settings according to Installation Manual?	

Blocks:

Item	Remark	Status
Cemat Library	The actual Version is Cemat V8.2 Hotfix required?	

CFC Settings:

Item	Remark	Status
Settings for Chart compilation	Settings according to Installation Manual? Installed block per OB had to be changed to >=300	

OS Configuration (WinCC Project):

Item	Remark	Status
OS Project Editor	Screen Resolution = 1920x1080? → 1680x1050, 1280x1024, 1600x1200 or 1920x1200 require exchange of some Cemat standard pictures Settings for Message display and Runtime Window correct? How many area selection buttons? Areas sorted?	
Computer Settings	Additional Application 'CEMATRS.exe' added.	
Tag Management, internal Tags	Variables for group ‘Cemat’ existing for Single Stations and Servers? Redundancy Tags existing? (Even if it is only a Single Station the Redundancy Tags are needed for Cemat.)	
Alarm logging	Definition of the Text blocks? Color and Acknowledgement settings? Configuration of the message archive (default: 1 week)	
Tag logging	Archive variable must have the same name as the process variable. Acquisition time < Archiving/Display time?	
User Archive	User Archives C_INFO, C_CURVE, DriveList, and POLY3 available? Data for C_INFO, C_CURVE and POLY3 available? Has the Excel Macro C_INFO.xlsm been used? Hardware Information for Inputs and Outputs available?	
Horn Configuration	Sounds file defined for each used message class?	
Time Synchronisaton	Time Synchronisation is crucial. Normally: Server = Master; Clients and AS = Slaves Make sure that there is only 1 (redundant) time master.	

AS Engineering

AS Engineering:

Item	Remark	Status
SYSPLC01	Naming correct? Is the System chart assigned to a Hierarchy folder in the plant view? Are the blocks of the system chart inserted at the right position in runtime group? Check with the tool "CEMAT_CheckTool.xlsm"	
Message configuration	Message texts complete for all blocks? The message text for block comment can be edited in the Process object view. Copy column "Block comment" to "Free Text 1"	
Group link	All objects (C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE, C_ANNUNC, C_ANNUN8, C_MEASUR and C_ADAPT must be linked to the group or route via GR_LINK1 or GR_LINK2 (or using C_MUX) Check with the tool "CEMAT_CheckTool.xlsm"	
Single-Start Release	If the Single-start-mode is enabled group-wise the output GES of the Group must be connected to all Signals xEIZ of the Drives	
Local Realease	If the local mode is enabled group-wise the output GLO of the Group must be connected to all Signals xLOC of the Drives	
Start command	No continuous Signal used for EBFE, KEB1, KEB2, VBFE. Only group/route commands (no EVS)	
Stop command	From group or route. Make sure that the feeding drive has been stopped before	
Quick Stop	Quick Stop is used? Connection of output GQS of the group to the QSTP of the drives? Butten in the group faceplate visible?	
Start-up-warnig	Connect the GLA from the group and the HORN output of all drives belonging to this group to an OR-Gate. Connect the GHA from the group and the HORN output of all drives belonging to this group to an OR-Gate.	

Item	Remark	Status
Group Feedback	Connection of GREZ and GRAZ complete? In case of routes the inverted WRA has to be used for the GRAZ.	
Annunciation blocks for Drive fault annunciations	For each signal which is used as protection interlock for a motor an annunciation module must exist. Alarm Activation is essential (MAAT)!!!	
Annunciation blocks for group status call	Annunication blocks existing for the indication of interlocking conditions of groups and routes (in the status call).	
Annunciation blocks for Interlocking Signals	In the annunciation blocks for interlocking signal (as e. g. Silo Levels) the interface MMZS must be set to 1-Signal in order not to show the summarizing indication in the group display.	
Annunciation blocks, Message configuration	For each annunciation block the 'Event text' (max. 16 characters) and the 'Free text 1' (max. 40 characters) must be configured.	
Annunciation blocks, Event text for status call function	The text under IN_DEL Property "Identifier" will generate an internal variable in the Tag management of WinCC. This text (max. 16 characters) will be used for the status call function. Must be equal to Event Text in the Message System.	
Programming order Runtime sequence	Check (especially after updates) if OB_START and OB_END is at the correct position. Cemat blocks must be called in OB1. Controllers, Timers and other PCS 7 blocks using SAMPLE_T must be called in a time interrupt OB. Check with the tool "CEMAT_CheckTool.xlsm"	
Stop delay for the drives	"Previous drive stopped" included in EBFA? This makes the configuration of the stop delay easier. The operator only has to configure the delay between one and the next drive.	
Interlock blocks	In general the Interlock blocks can be used with any interface if the Cemat blocks. However the most useful application is together with Interlocking conditions (e. g. EEVG, EBVG, ESVG and ESVA).	

Item	Remark	Status
Structure Interlock blocks	Interlock blocks Intlk02, Intlk04, Intlk08 or Intlk16 used for display of interlocking conditions (IntStart, IntOper, IntProtG, IntProtA) Simulation only enabled if necessary? Block C_SIMU_L connected for list of active simulations?	
Treiberbausteine	The driver blocks are not necessary in Cemat (except for CH_AO) and are only an option. It has to be considered that the driver wizard creates a lot of blocks (expecially when using ET200S) which cost additional resources (PBK).	
CPU_RT	In order to disable Load Shedding, connect "0" to parameter MAX_RTRG	
PID Controllers	The library PID_Expl contains some examples for PID functions.	
Annunciation Release	In case of loss of control power of e.g. Profibus fault we do not want to give alarms for the effected drives. Therefore the Annunciation Release Interface EMFR, KMFR, VMFR, MMFR, UMFR must be connected with a signal "Control Power ok". 0-Signal at xMFR will freeze the alarm generation for the Cemat blocks.	
Cycle time	Check the actual Cycle time	

After the AS Engineering is completed and before you start your FAT or Comissioning, we highly recommend using the Cemat Check Tool in order to check the general file structure and most important links. You will find the Excel file CEMAT_CheckTool.xlsm under D:\CEMAT_CS\TOOLS.
The description for the Check tool you find in the Engineering Manual, chapter 09_Engineering_Tools.

OS Engineering

OS Engineering:

Item	Remark	Status
Simocode block icon	Has a block icon for the SIMOCODE Adapter been generated? (If the block icon is missing, the group display in the overview range will not be updated.) The block icon can be hidden behind the motor symbol as it is only needed for the group display function.	
User rights	User Groups and Users defined?	
LoopInAlarm	The Entry of the Picture name in the User Archive C_INFO is not absolutely necessary. As a default, Cemat uses the Picture from Component List Editor. (The entry in C_INFO can still be used; e. g. if the object is in more than on picture and a specific picture should be opened).	

Function Test

General Functions:

Item	Remark	Status
Help Functions	Help files can be opened via Cemat Faceplates?	
Diagnosis Pictures	Can, for Modules with Interlock function, the blue dots be seen in the diagnosis window? Can the interlock faceplate be opened?	
Info Dialog	Object Information available? I/O information has been generated with Excel Macro? Can the operator comments be saved for the Objects? Additional button functions used, such as Sound, Manual, Video, Map? Can the additional files be opened?	
Annunciation Release	In case of a power supply failure, only one message must come!	

Message System:

Item	Remark	Status
Alarm line	Are the alarms in the alarm line displayed correctly (for the areas for which the operator has the permission)?	
Alarm line, Loop in Alarm function	Alarm picture is opened using loop-in-alarm button in message line?	
Alarm line, Info function	The faceplate of the object in the alarm line will open?	
Alarm line, Acknowledge	With acknowledge button in the alarm, for the AS which produced the alarm all faults are acknowledged?	
Cemat Alarm list	Selections possible? Selections can be saved?	
Object Alarmwindow	Alarms and Operation annunciation are shown correctly in the Alarm window of the Cemat Objects.	

Group (Route) Functions:

Item	Remark	Status
Status Call	Status call function shows all existing faults within the group/route? The fault text is shown correctly, also for annunciation blocks and non-Cemat blocks?	
Instance List	The group instance list shows all objects in the group (or route)? The status of the object is shown correctly (also for non-cemat blocks)? With double-click on the object the faceplate will open? Sorting correct? → Through a respective call-up order in the runtime-editor the objects automatically appear in a correct sequence and must not be sorted manually).	
Assigned group/route to the Object	Instance list saved? → The new function "show groups/routes for the selected object" only works if the Instance list has been saved before.	
Open assigned Gruppe/Weg	Via button "G" in the route faceplate, the group which is connected via GR_LINK1 will be opened.	
Show related Objects	Via button "O" in the route or group faceplate, all related objects in the process picture are marked.	
User Object	Via button "A" the faceplate of a user object can be opened.	

Object Functions:

Item	Remark	Status
Open assigned group/route	For drives, measures and annunciation modules: via button "G" the faceplate of the group (or route), which is connected via GR_LINK1 will be opened.	
User Object	For drives, measures and annunciation modules: Via button "A" the faceplate of a user object can be opened.	
Display of a measure (current or power) in %	Measure for motor current (or power) can be displayed in the drive faceplate (C_DRV_1D und C_DRV_2D). Percentage value correct? (High limit = 100%)	
Display of a measure	Display of an additional measure in the drive faceplate (for C_DRV_1D, C_DRV_2D and C_DAMPER). Can the Measure faceplate be opened directly from the drive faceplate? Display of Archive possible? (See naming conventions for Measuring value archives and Archive variables in Engineering Manual Chapter 3).	
SIMOCODE	The faceplate for Simocode Adapter can be called from drive faceplate (C_DRV_1D, C_DRV_2D, C_DAMPER or C_VALVE)? Does the group display also work for SIMOCODE faults (block icon for SIMOCODE adapter block exist)?	
SUBCONTROL	The Subcontrol faceplate can be called from drive faceplate of C_DRV_1D? Button description ok?	
Setpoint input	Setpoint input and display of the actual value directly from faceplate of C_DRV_1D.	

Project administration

Content

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\11_Project_Administration_009.doc

Distributed Engineering

A PCS 7 project can be edited by more than one user. Since PCS 7 V7 it is even possible that more than one user can make modifications in the same S7-Program (z. B. in order to make modifications in different charts). However, some steps like the compilation of the AS or the OS can not be carried out simultaniously by more than one user. In this case the system blocks the function.
Nevertheless, working with more than one user in the same AS requires a certain discipline.

Since PCS 7 V8.1 WinCC Projects can be opened more than one time but with restricted functionality. This provides the option that e.g. more than one person can edit different process pictures within same WinCC Project.
Again, working with more than one user in the same OS requires a certain discipline.

If the situation demands that a project is created at different times or at different locations, you can break down a master project (Multiproject) into subsections (partial projects). You can, for example, assign a station or a program to each person involved. The procedure is analogous for distributing work on several operator stations.

See PCS 7 - Configuration Manual Engineering Station, Chapter 5, Basic Concepts of Engineering.

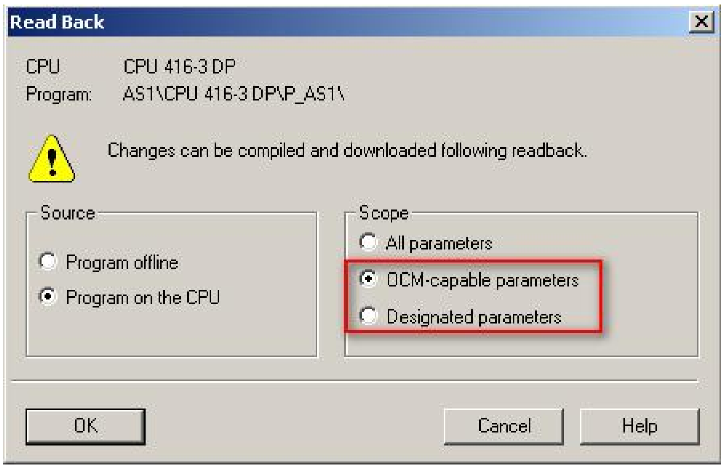
Saving the PCS 7 project

Read back process values

Process values can be changed during operation through HMI. In this case the values are saved in the AS but not in the Offline CFC. As long as you only download for changes, the actual process values are not overwritten. But in case you have to carry out a complete download for any reason, the program in the AS will be deleted and the offline data will be downloaded into the AS.

For this reason it is important to read back the process parameters regularly – and especially before you take a backup of project.

For each AS open a CFC and carry out the function Chart → Read back…



You have to carry out the Readback function **twice**:

- Once with option ‘OCM-capable parameters’ (reading the process parameters)
- Once with option ‘Designated parameters’ (e. g. for reading the Features and OS Permissions)

Archiving the project

Archiving of the Multiproject must be carried out on the Engineering Station via function "Archive…" of the SIMATIC Manager. Via this procedure the Multiproject is copressed with the Archiving program "PKZip or "WinZip", selected in the SIMATIC Manager.

⚠ Never save the Multiproject directly under Windows. Always use Archive function of the SIMATIC Manager!

All WinCC Projects must be closed before archiving. Make sure that the complete Multiproject with all partial projects and libraries is archived. The easiest way to insure this is to open the Multiproject and to select the upper most folder (the Multiproject itself) before selecting the Archive function.

⚠ Do not only save the Archive file on the Engineering Station. Copy the file additionally to a different location. Add the date (and time) to the file name and for security reason, keep the last few archives.

PCS 7 Servicing and diagnostics functions

In the PCS 7 – Configuration Manual Engineering system you find further information regarding Servicing and diagnostics with PCS 7 V8.2.

We especially refer to chapter 13.5 Archiving/Versioning and Documenting and to the add-on package "Version Trail" which offers the following options:

- You can archive objects (such as libraries, multiprojects and single projects) at a time of your choice. The saved objects are assigned versioning when it is entered into the version archive. The versioning is the unique ID for this object
- Retrieve and re-use versioned project data
- Automatic archiving
- Automatic readback
- Comparison of an archived version with an existing project or with a second arcived version. You start the Version Cross Manager (VXM) to perform the comparison.

Graphic Templates

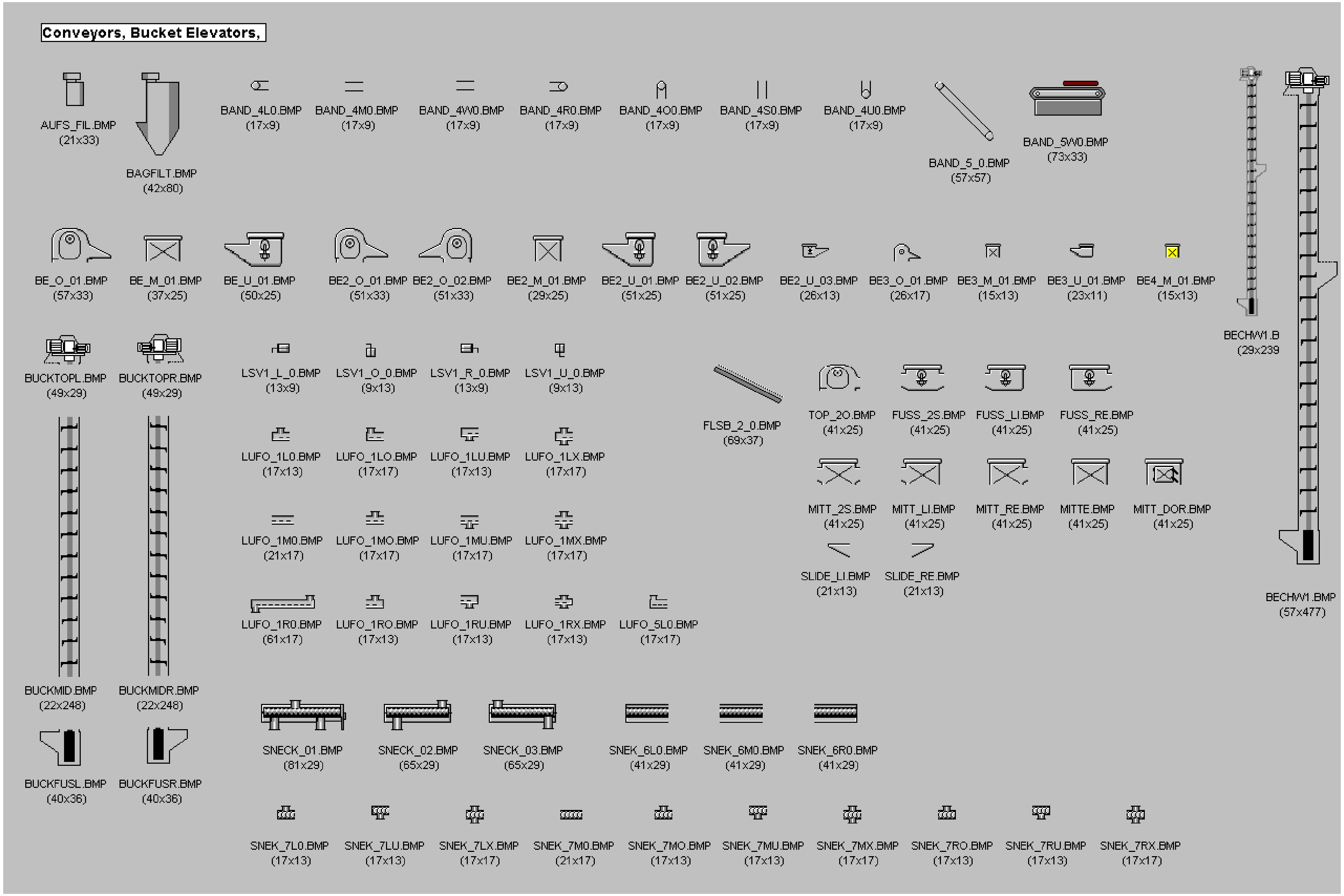
Content

With the Installation of Cemat V8.2 file Bitmaps.zip is copied to D:\CEMAT_CS\Bitmaps. This file contains a collection of bitmap samples from former projects, which you can your for the Generation of your process pictures.

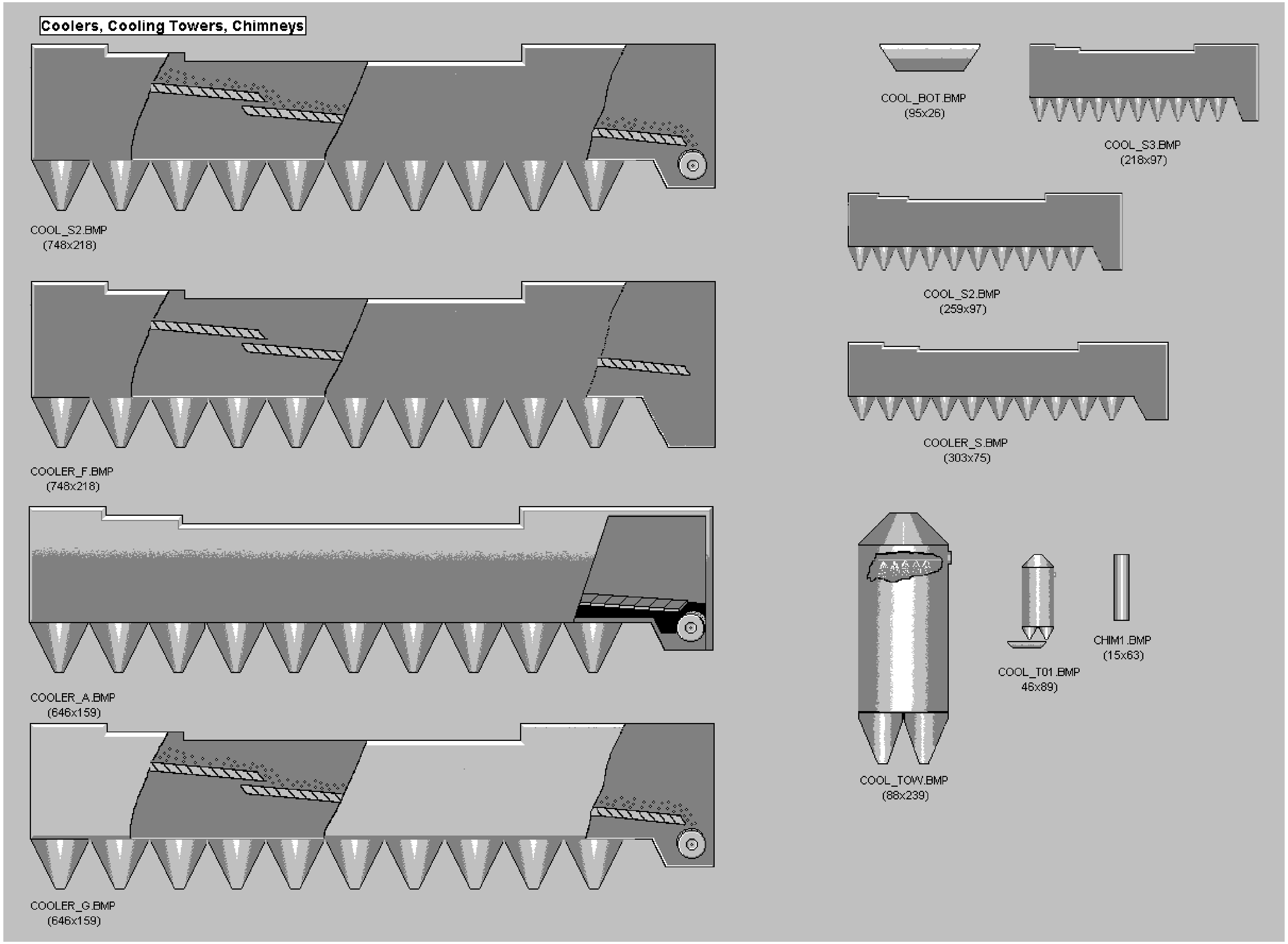
All Bitmaps which are used in the Cemat block icons and Faceplates are already copied into directory D:\CEMAT_CS\WinCC\GraCS. However from Cemat V8.0 SP1 only the symbols which are used in the template picture C_@PCS7Typicals_CemV8_000.PDL are copied to this directory.

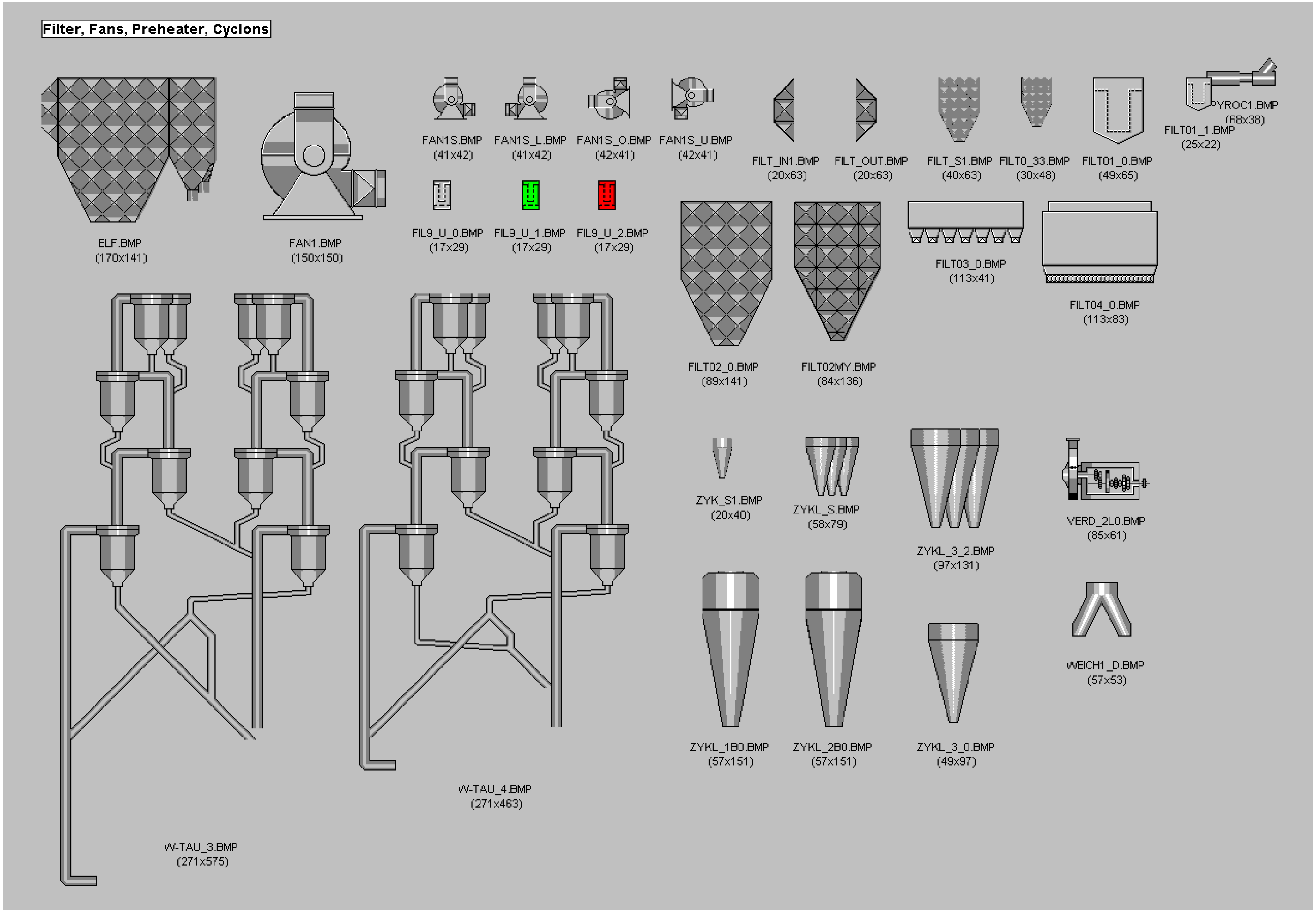
In the Subfolder CEM_DRAFTS you find zip-files with further bitmaps and example pictures (Return from existing plants) which may be used as samples.

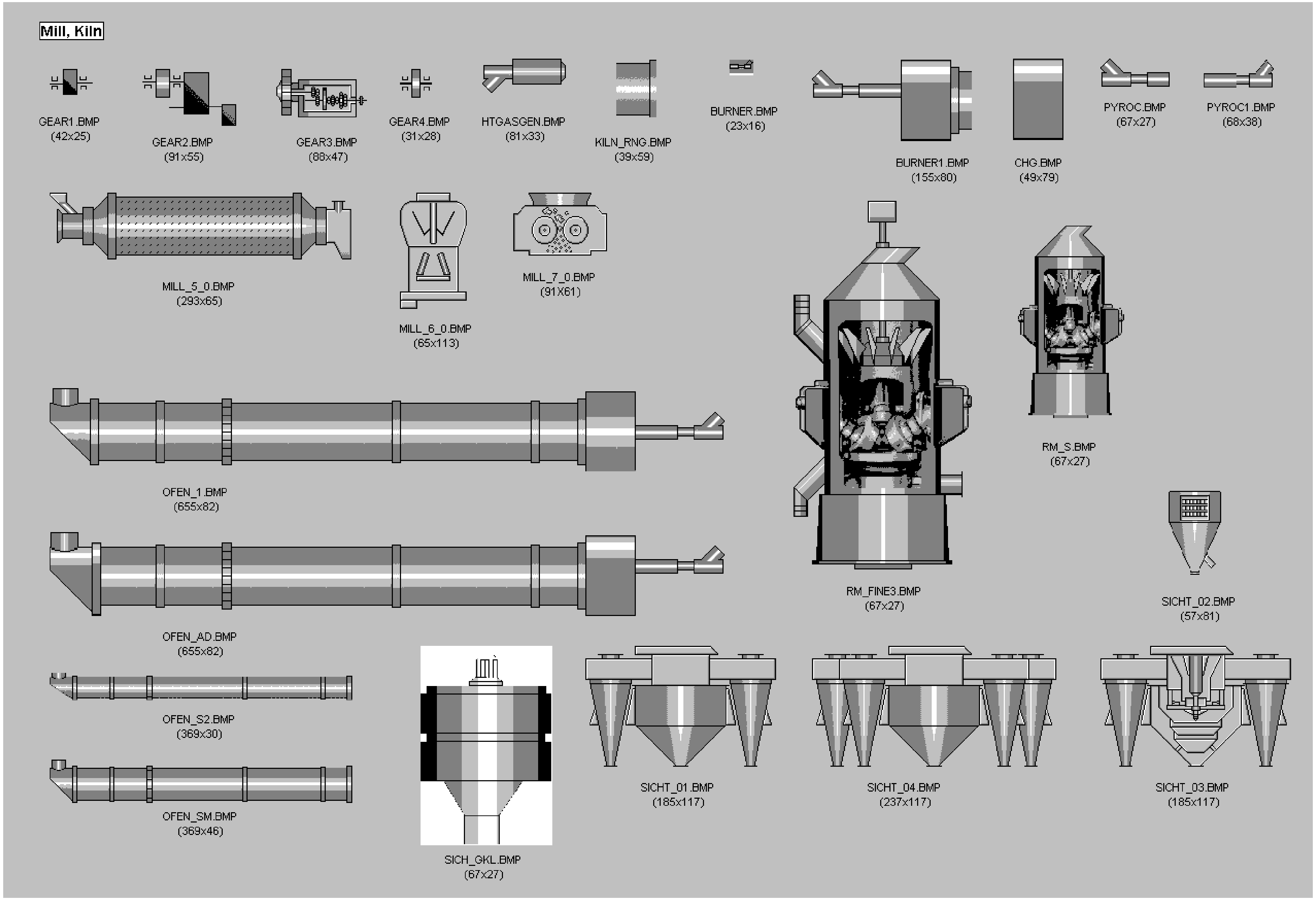
On the following pages you find the illustration of frequently used bitmap samples from directory D:\CEMAT_CS\Bitmaps.

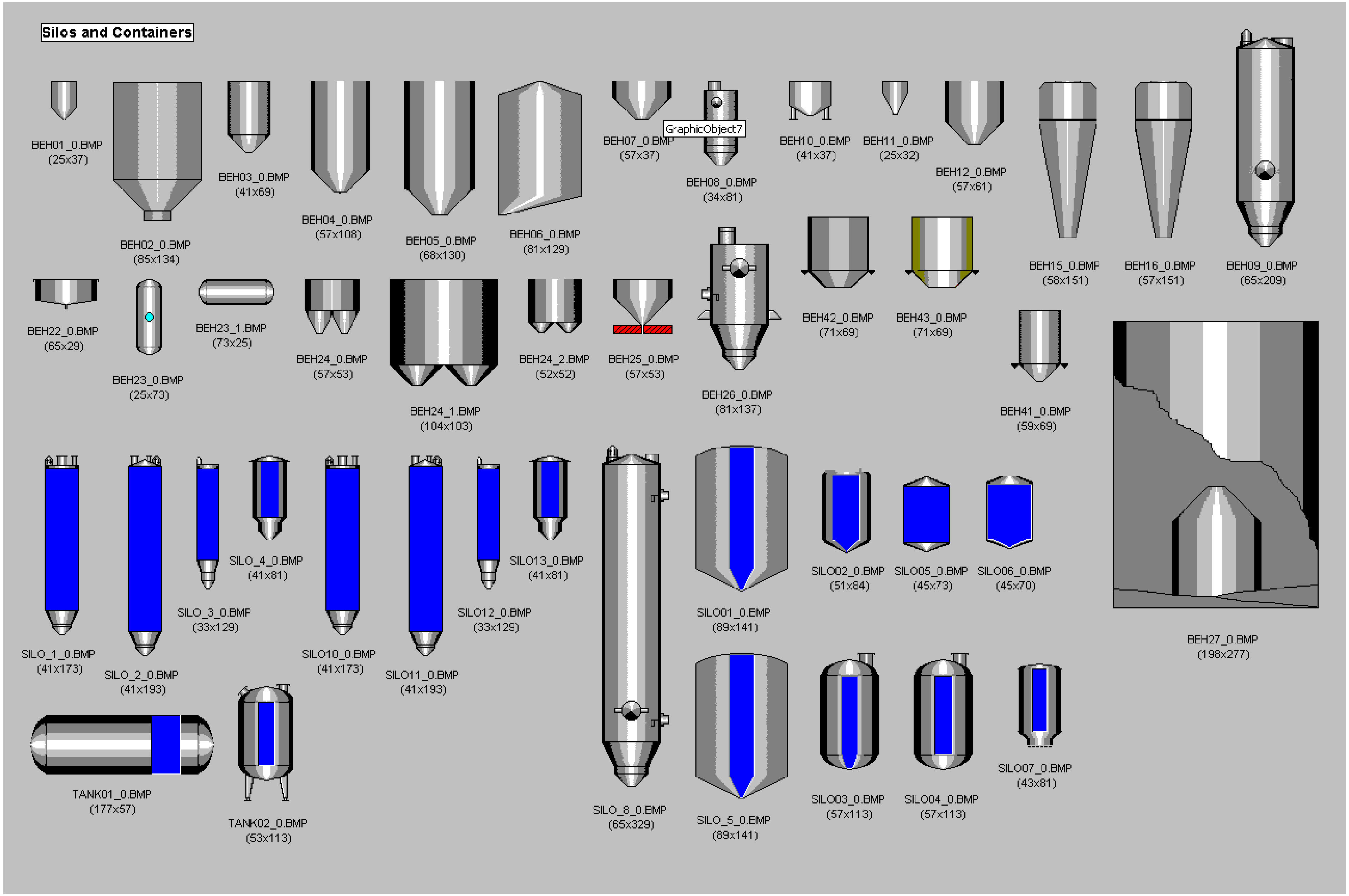


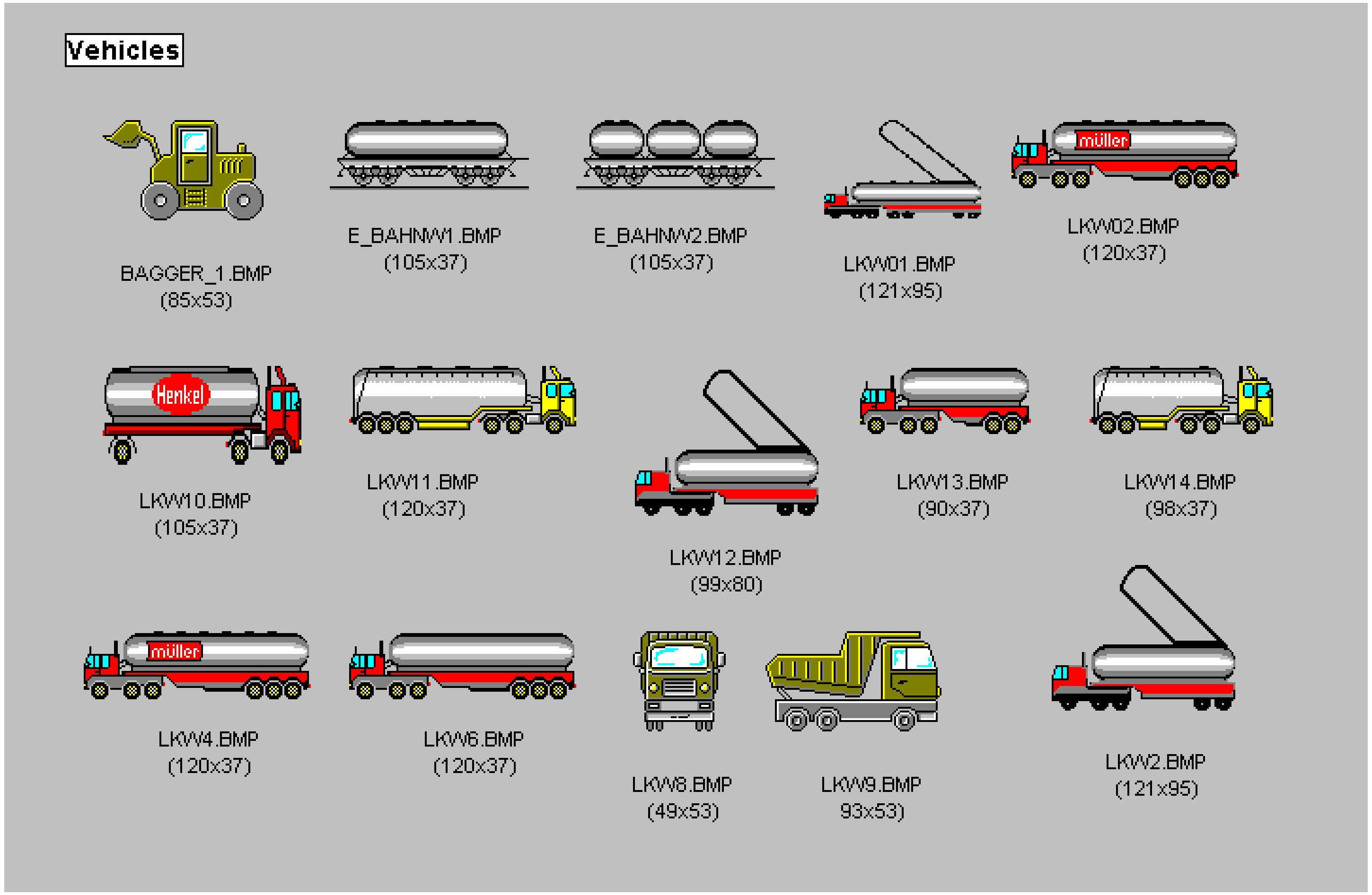
Conveyors, Bucket Elevators,
AUFS_FIL.BMP (21x33)
BAGFILT.BMP (42x80)
BAND_4L0.BMP (17x9)
BAND_4M0.BMP (17x9)
BAND_4W0.BMP (17x9)
BAND_4R0.BMP (17x9)
BAND_4O0.BMP (17x9)
BAND_4S0.BMP (17x9)
BAND_4U0.BMP (17x9)
BAND_5_0.BMP (57x57)
BAND_5W0.BMP (73x33)
BE_O_01.BMP (57x33)
BE_M_01.BMP (37x25)
BE_U_01.BMP (50x25)
BE2_O_01.BMP (51x33)
BE2_O_02.BMP (51x33)
BE2_M_01.BMP (29x25)
BE2_U_01.BMP (51x25)
BE2_U_02.BMP (51x25)
BE2_U_03.BMP (26x13)
BE3_O_01.BMP (26x17)
BE3_M_01.BMP (15x13)
BE3_U_01.BMP (23x11)
BE4_M_01.BMP (15x13)
BECHW1.B (29x239
BUCKTOPL.BMP (49x29)
BUCKTOPR.BMP (49x29)
LSV1_L_0.BMP (13x9)
LSV1_O_0.BMP (9x13)
LSV1_R_0.BMP (13x9)
LSV1_U_0.BMP (9x13)
FLSB_2_0.BMP (69x37)
TOP_2O.BMP (41x25)
FUSS_2S.BMP (41x25)
FUSS_LI.BMP (41x25)
FUSS_RE.BMP (41x25)
LUFO_1L0.BMP (17x13)
LUFO_1LO.BMP (17x17)
LUFO_1LU.BMP (17x13)
LUFO_1LX.BMP (17x17)
MITT_2S.BMP (41x25)
MITT_LI.BMP (41x25)
MITT_RE.BMP (41x25)
MITTE.BMP (41x25)
MITT_DOR.BMP (41x25)
LUFO_1M0.BMP (21x17)
LUFO_1MO.BMP (17x17)
LUFO_1MU.BMP (17x17)
LUFO_1MX.BMP (17x17)
SLIDE_LI.BMP (21x13)
SLIDE_RE.BMP (21x13)
BECHW1.BMP (57x477)
LUFO_1R0.BMP (61x17)
LUFO_1RO.BMP (17x13)
LUFO_1RU.BMP (17x13)
LUFO_1RX.BMP (17x13)
LUFO_5L0.BMP (17x17)
BUCKMID.BMP (22x248)
BUCKMIDR.BMP (22x248)
SNECK_01.BMP (81x29)
SNECK_02.BMP (65x29)
SNECK_03.BMP (65x29)
SNEK_6L0.BMP (41x29)
SNEK_6M0.BMP (41x29)
SNEK_6R0.BMP (41x29)
BUCKFUSL.BMP (40x36)
BUCKFUSR.BMP (40x36)
SNEK_7L0.BMP (17x13)
SNEK_7LU.BMP (17x13)
SNEK_7LX.BMP (17x17)
SNEK_7M0.BMP (21x17)
SNEK_7MO.BMP (17x13)
SNEK_7MU.BMP (17x13)
SNEK_7MX.BMP (17x17)
SNEK_7RO.BMP (17x13)
SNEK_7RU.BMP (17x13)
SNEK_7RX.BMP (17x17)


Coolers, Cooling Towers, Chimneys
COOL_S2.BMP
(748x218)
COOLER_F.BMP
(748x218)
COOLER_A.BMP
(646x159)
COOLER_G.BMP
(646x159)
COOL_BOT.BMP
(95x26)
COOL_S3.BMP
(218x97)
COOL_S2.BMP
(259x97)
COOLER_S.BMP
(303x75)
COOL_TOW.BMP
(88x239)
COOL_T01.BMP
46x89)
CHIM1.BMP
(15x63)


Filter, Fans, Preheater, Cyclons
ELF.BMP
(170x141)
FAN1.BMP
(150x150)
FAN1S.BMP
(41x42)
FAN1S_L.BMP
(41x42)
FAN1S_O.BMP
(42x41)
FAN1S_U.BMP
(42x41)
FIL9_U_0.BMP
(17x29)
FIL9_U_1.BMP
(17x29)
FIL9_U_2.BMP
(17x29)
FILT_IN1.BMP
(20x63)
FILT_OUT.BMP
(20x63)
FILT_S1.BMP
(40x63)
FILT0_33.BMP
(30x48)
FILT01_0.BMP
(49x65)
FILT01_1.BMP
(25x22)
PYROC1.BMP
(68x38)
FILT02_0.BMP
(89x141)
FILT02MY.BMP
(84x136)
FILT03_0.BMP
(113x41)
FILT04_0.BMP
(113x83)
ZYK_S1.BMP
(20x40)
ZYKL_S.BMP
(58x79)
ZYKL_3_2.BMP
(97x131)
VERD_2L0.BMP
(85x61)
ZYKL_1B0.BMP
(57x151)
ZYKL_2B0.BMP
(57x151)
ZYKL_3_0.BMP
(49x97)
WEICH1_D.BMP
(57x53)
W-TAU_4.BMP
(271x463)
W-TAU_3.BMP
(271x575)


Mill, Kiln
GEAR1.BMP
(42x25)
GEAR2.BMP
(91x55)
GEAR3.BMP
(88x47)
GEAR4.BMP
(31x28)
HTGASGEN.BMP
(81x33)
KILN_RNG.BMP
(39x59)
BURNER.BMP
(23x16)
BURNER1.BMP
(155x80)
CHG.BMP
(49x79)
PYROC.BMP
(67x27)
PYROC1.BMP
(68x38)
MILL_5_0.BMP
(293x65)
MILL_6_0.BMP
(65x113)
MILL_7_0.BMP
(91X61)
RM_FINE3.BMP
(67x27)
RM_S.BMP
(67x27)
OFEN_1.BMP
(655x82)
OFEN_AD.BMP
(655x82)
SICHT_02.BMP
(57x81)
OFEN_S2.BMP
(369x30)
OFEN_SM.BMP
(369x46)
SICH_GKL.BMP
(67x27)
SICHT_01.BMP
(185x117)
SICHT_04.BMP
(237x117)
SICHT_03.BMP
(185x117)


Silos and Containers
BEH01_0.BMP
(25x37)
BEH02_0.BMP
(85x134)
BEH03_0.BMP
(41x69)
BEH04_0.BMP
(57x108)
BEH05_0.BMP
(68x130)
BEH06_0.BMP
(81x129)
BEH07_0.BMP
(57x37)
GraphicObject7
BEH08_0.BMP
(34x81)
BEH10_0.BMP
(41x37)
BEH11_0.BMP
(25x32)
BEH12_0.BMP
(57x61)
BEH15_0.BMP
(58x151)
BEH16_0.BMP
(57x151)
BEH09_0.BMP
(65x209)
BEH22_0.BMP
(65x29)
BEH23_0.BMP
(25x73)
BEH23_1.BMP
(73x25)
BEH24_0.BMP
(57x53)
BEH24_1.BMP
(104x103)
BEH24_2.BMP
(52x52)
BEH25_0.BMP
(57x53)
BEH26_0.BMP
(81x137)
BEH42_0.BMP
(71x69)
BEH43_0.BMP
(71x69)
BEH41_0.BMP
(59x69)
BEH27_0.BMP
(198x277)
SILO_1_0.BMP
(41x173)
SILO_2_0.BMP
(41x193)
SILO_3_0.BMP
(33x129)
SILO_4_0.BMP
(41x81)
SILO10_0.BMP
(41x173)
SILO11_0.BMP
(41x193)
SILO12_0.BMP
(33x129)
SILO13_0.BMP
(41x81)
SILO_8_0.BMP
(65x329)
SILO01_0.BMP
(89x141)
SILO02_0.BMP
(51x84)
SILO05_0.BMP
(45x73)
SILO06_0.BMP
(45x70)
SILO_5_0.BMP
(89x141)
SILO03_0.BMP
(57x113)
SILO04_0.BMP
(57x113)
SILO07_0.BMP
(43x81)
TANK01_0.BMP
(177x57)
TANK02_0.BMP
(53x113)


Vehicles
BAGGER_1.BMP
(85x53)
E_BAHNW1.BMP
(105x37)
E_BAHNW2.BMP
(105x37)
LKW01.BMP
(121x95)
müller
LKW02.BMP
(120x37)
Henkel
LKW10.BMP
(105x37)
LKW11.BMP
(120x37)
LKW12.BMP
(99x80)
LKW13.BMP
(90x37)
LKW14.BMP
(98x37)
müller
LKW4.BMP
(120x37)
LKW6.BMP
(120x37)
LKW8.BMP
(49x53)
LKW9.BMP
93x53)
LKW2.BMP
(121x95)

Tips&Tricks

Table of Contents

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\14_Tips_Tricks_009.doc

Notes and Tips for the Use of WinCC / PCS 7 V8.2

The following notes and tips for the use of WinCC/PCS 7 V8.2 do not directly refer to CEMAT. This is just a collection of information, which may be useful for engineering and/or commissioning.

Further Information can be found in the PCS 7 V8.2 Manuals
- PCS 7 Readme
- PCS 7 PC Configuration and Authorization

and in the FAQ.

Time synchronization

A time synchronization of the OS Servers and the connected AS components is necessary for a correct annunciation processing (archiving, display, redundancy balancing).

CEMAT Functions

CEMAT Message System

Remember Filter options

The CEMAT Message System provides multiple filter options. Closing the CEMAT Message Window stores the actual selected filter options and after a picture change (when opening the message window again) the previously defined settings are available.

If the filter settings inthe CEMAT Message System shall not be remembered, this can be parameterized accordingly through a modification in the configuration file D:\CEMAT_CS\Config\C_Messages.cfg under `[MsgInit]`.

With Parameter `MsgReset = 1` the settings are not stored any more and when the CEMAT Message System is opened no filter is active.

```
[MsgInit]
;0 = Save actual filter settings
;1 = Reset filter settings
MsgReset=0
```

Changing Sorting Order

The sorting order in the CEMAT Message System can be changed in the configuration file D:\CEMAT_CS\Config\C_Messages.cfg. One can decide whether one wants to see the oldest message above or below in the list.

With Parameter `MsgSort=1` the oldest message is shown at the bottom.

```
[MsgInit]
;0 = Sortorder up -> default (oldest message at the top)
;1 = Sortorder down (oldest message at the bottom)
MsgSort=0
```

Display period

In the CEMAT Message System as standard the annunciations of the last 24 hours are displayed. If this time is too short the value can be changed in the configuration file D:\CEMAT_CS\Config\C_Messages.cfg under `[MsgProperty]`.

The default value `MinusSec=86400` corresponds to 86400 seconds = 24 hours.

```
[MsgProperty]
;MinusSec Valud is subtracted from actual date and time
;e. g.: 86400 = 1 day; 28800 = 8 hours; 3600 = 1 hour
MinusSec=86400
```

Engineering of the Messages in CFC

The Message class "Operating Message" is not available when in the column "with acknowledgement" the box is selected.

For example:

When your project has been migrated to version 6 please check all messages of CEMAT Groups in the CFC. There is a new message "Quick Stop". Deselect in the column "With acknowledgement" all boxes.

After that in CFC, select Options → Block Types. Select FB1010 and press 'New Version'.

After that check the messages of one Group again. The messages must have the message class "Operating Messages – Standard".

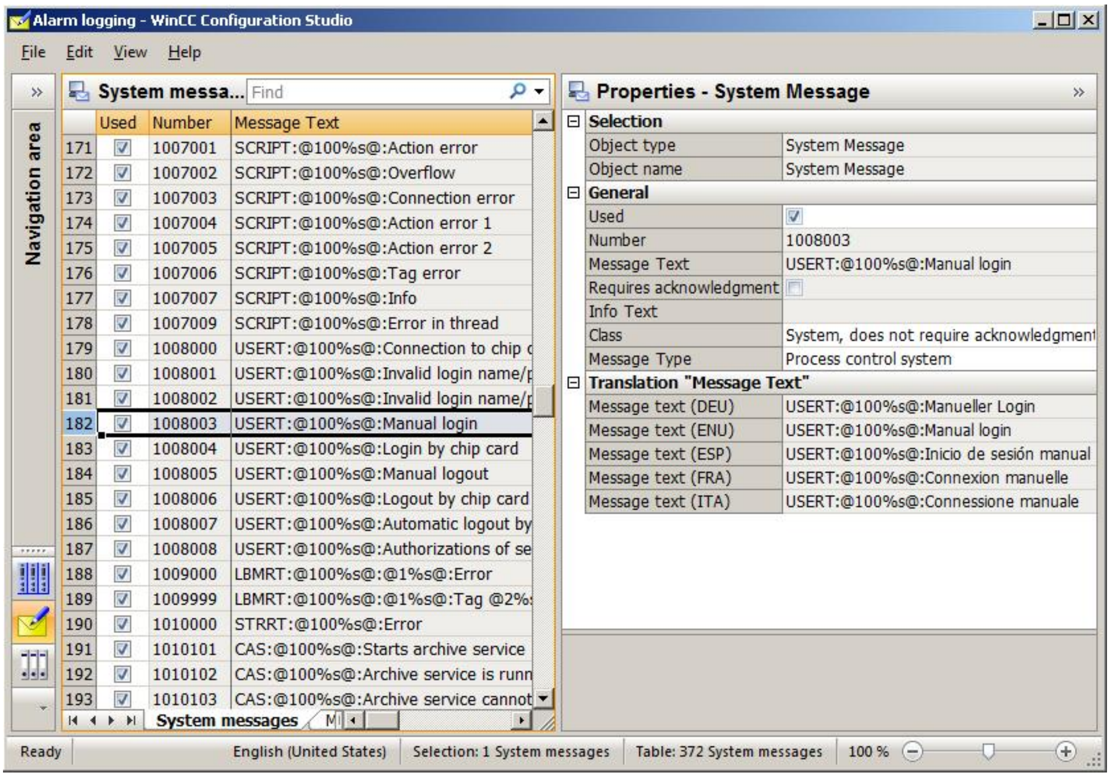
Message System

System Message for Manual Login

The WinCC System Message for Manual Login normally does not contain the User Name. In order to show the user name, in the WinCC Configuration Studio for Alarm Logging the Event text of Message Number 1008003 has to be modified

from **USERT:@100%s@:Manual Login**

to **@100%s@:@102%s@:Manual Login**



Block Icons with Style Tag

In some block icons, as e. g. for groups, routes, selections, drives, the display of the faceplate buttons can enabled/disabled by setting the appropriate code in Property 'Styles', Attribute 'StyleTag'.

In order to make this attribute change available also in case of indirect faceplate calls (e. g. block jumps), for any code unequal to "0" the code is stored in D:\CEMAT_CS\config\C_RuntimeSettings.cfg.

The storing of the variable happens in the Runtime System, while opening the faceplate through click on the block icon.

Caution: If the code under StyleTag is "0" no code is stored. If you change a StyleTag which was unequal to "0" back to "0", the old value remains stored in file at line '*Tagname*_ButtonStyle'. In order to go back to the default value you have to manually delete this line.

Process Value Archive Options

The Archive tags for the measurements can be created automatically in the WinCC Tag Logging during OS Compile.
If in the CFC the block output *MV* is selected for "Archiving" or “Long-term Archiving”, during the OS Compilation an Archive Tag with the correct name is automatically inserted into a common Archive with name 'SystemArchive'.
Beside this, an internal Variable @TagArchiveName is created, which contains the name ‘SystemArchive’. This variable is used by the C_MEASUR block in order to find out the Archive name.

Acquisition cycle and Archiving/Display Cycle are set to a default value and must be adapted afterwards in WinCC Configuration Studio for Tag Logging.

Manual creation of Archive Tags and Archive splitting

Manual creation of Archive Tags or Archive splitting (one Archive per plant section) is not recommended because it means much more Engineering effort. Nevertheless it has been used in the past and may be required for existing Projects.

In this case the user must take care that the naming conventions for the Archive tags are followed. In addition, certain settings are required in order to advise the Faceplate function of the C_MEASUR block where exactly the Archive tag is located.

Naming conventions:

Measured value name: 123_BC1/J01
Archive tag name: 123_BC1/J01.MV

Three options are possible for Archive names in case of manual creation; each one requires different settings. Option 3 may have been used in existing projects, but not very useful any more.

Option 1: Common Process Value Archive with default name ‘ProcessValueArchive’

In the WinCC Configuration Studio for Tag Logging, create a common Archive with name ‘ProcessValueArchive**’**. In this process value archive you manually create the Archive tags, e. g. 123_BC1/J01.MV.
Acquisition cycle and Archiving/Display Cycle must be configured.
→ No further engineering is carried out in the CFC or in the OS.
Archiving must not be selected in the CFC!!!!

Option 2: Multiple Archives with name configuration in the CFC

In the WinCC Configuration Studio for Tag Logging create the Area Specific Archives, e. g. ‘Clinker’ or ‘Kiln’ and manually create the Archive tags such as 100_BC2/M01_I01.MV.
Acquisition cycle and Archiving/Display Cycle must be configured.
→ As the archive name is different from 'ProcessValueArchive', the proper name must be entered in the CFC, structure variable 100_BC2/M01_I01.PV_Out#Value under 'Identifier'.
Archiving must not be selected in the CFC!!!!

Option 3: Multiple Archives with name configuration at the block icon

In the WinCC Configuration Studio for Tag Logging create the Area Specific Archives, e. g. ‘Clinker’ or ‘Kiln’ and manually create the Archive tags such as 100_BC2/M01_I01.MV.
Acquisition cycle and Archiving/Display Cycle must be configured.
→ As the archive name is different from 'ProcessValueArchive', the name is entered at the block icon, Property **Styles**, Attribute **ReturnPath.**
Archiving must not be selected in the CFC!!!!

(Option 3 is not recommended because the block icon properties are only available if the Faceplate is opened via Click on the block icon.)

Comparison between automatic and manual Archive generation:

(Option 3 for Manual creation is not shown in this overview because it is not very useful.)

Common Archive (Automatically created)	**Common Archive (Manually created)**	**Multiple Archives (manually created)**
Archive Name: "SystemArchive"	Archive Name "ProcessValueArchive"	Archive Name: "Crusher", "Raw Mill", "Kiln", etc.
Select property 'Archiving' at output Tagname.MV Archives are created during OS Compile and Variable @TagArchiveName is created which contains the Archive name	Don't select property Archiving' at output Tagname.MV **&** - Create an Archive in the WinCC Tag Logging - Create Archive Tags in WinCC Tag Logging for - Connect to Tagname.MV	Don't select property 'Archiving' at output Tagname.MV **&** - Create multiple Archives in the WinCC Tag Logging - Create Archive Tags in WinCC Tag Logging - Connect to Tagname.MV **&** - Enter Archive name "Crusher", "Raw Mill" "Kiln", etc. at output Tagname.PV_Out.Value under "Identifier" - During OS Compile the internal variable Tagname.PV_Out#Valvue# Shortcut is created.

Caution: If you have a single (common) Process Value Archive, the mix of automatic generation and manual generation of Archive Variables is not possible. Once the internal variable @TagArchiveName exists (created via automatic generation) the system will use the content of this variable as Archive name.

The priority for the determination of the archive name is as follows:

Prio1: Checking the entry in the internal variable *measure.*PV_Out#Value#Shortcut
Prio2: Checking the entry in the block icon under property Styles, Attribute ReturnPath.
Prio3: Entry in Variable @TagArchiveName (if existing)
Prio4: Default setting "ProcessValueArchive"

Work-around for archive detection problems

If the Archive Curve is called from the Faceplate of C_MEASUR the Archive name is determined as follows:

Prio1: Checking the entry in the internal variable *measure.*PV_Out#Value#Shortcut
Prio2: Checking the entry in the block icon under property Styles, Attribute ReturnPath.
Prio3: Entry in Variable @TagArchiveName (if existing)
Prio4: Default setting "ProcessValueArchive"

In some projects it happened that a manually created process value archive with name 'ProcessValueArchive' existed already, and after this, through OS Compile the system has generated a second archive (e. g. with name SystemArchive. In this case the variable @TagArchiveName has been created which contains the name "SystemArchive".
As a consequence, the old Archives located under ProcessValueArchive are not found any more.

Neither the variable @TagArchiveName nor the SystemArchive itself can be deleted (WinCC refuses this persistently!). This step cannot be revoked.

There are two solutions to solve this problem:

- Create all Archive Variables from the ProcessValueArchive in the SystemArchive again and after that delete the old ProcessValueArchive. But in this case all the historic values are lost.
- By forcing the system to ignore the entry in Variable @TagArchiveName and to skip step "Prio3" in the Archive detection.

  For the second option proceed as follows:

  - Open file C_Config.cfg (under D:\Cemat_CS\Config).

  - There you find the lines:

  ```
  [CematSettings]
   ;Individual Settings for Cemat Objects (0=No, 1=Yes)
   ;C_MEASUR
   SkipTagArchiveName=0
  ```

  - Change the last line into `SkipTagArchiveName=1`
    (Step "Prio3" will be skipped)

Replace standard PCS 7 INTERLOCK by CEMAT INTERLOCK

The Interlock block from the PCS 7 standard library needs 33 OS variables. The number of OS variables are reduced to 11 variables, if the CEMAT Interlock block is used. If the Interlock block from the PCS 7 standard library is used and this block should be replaced by the CEMAT Interlock and all existing CFC connections should remain, then follow the listed sequence. Attention! In this case the AS must be completely reloaded:

1. Delete FB 75 = INTERLOK from the PCS 7 standard library from the block container. Don not delete the entry in the symbol list FB75 = INTERLOK.
2. Copy FB1075 = C_INTERL as FB75 (symbol = INTERLOK).
3. Copy @PG_C_INTERL.PDL as @PG_INTERLOK.PDL.
   Copy @PG_C_INTERL.PDL_OVERVIEW as @PG_INTERLOK_OVERVIEW.PDL.
   Copy @PG_C_INTERL.PDL_VIEWLIST as @PG_INTERLOK_VIEWLIST.PDL.
   Copy @PG_C_INTERL.PDL_STANDARD as @PG_INTERLOK_ STANDARD.PDL.
4. Change the properties of the objects in the copied file @PG_INTERLOK.PDL as follows.
   - Object @Faceplate -> Texts -> Firstview = @PG_INTERLOK_ STANDARD.PDL.
   - Object BlockType -> Output/Input -> Output Value = INTERLOK.
   - Object Viewlist -> Miscellaneous -> Picture Name = @PG_INTERLOK_VIEWLIST.PDL.
5. Invert all inverting bits for all connected inputs in the CFC.
   Please check the logic!!!!!

Detect and store the maximum runtime of the Cyclic Interrupt OBs

With the block C_MAX_CYC = FB1021 you can detect and store the maximum runtime of OB1 and of the Cyclic Interrupt Organization Blocks (OB30 to OB38). C_MAX_CYC must be called in OB1 (MAIN_TASK).

The gross OB1 runtime is determined internally and provided at output OB1_MAXT. This time considers the cycle time stabilization as well as the interrupts through cyclic interrupt OBs.

The net OB1 runtime and the runtime of the Cyclic Interrupt OBs (OB30 to OB38) have to be 'collected' via interconnection from the chart "@CPU_RT" from block "@CPU_RT" = FB128.

Via 1-Signal at parameter "RESET", all maximum runtime values of the OBs are reset. The reset time and the moment of the occurrence of the maximum runtime of each OB are stored.

Chart "@CPU_RT" with the block "@CPU_RT" is created during CFC compiling, if the option "Generate module drivers" is selected.

Input interfaces:

Reset BOOL

Via 1-Signal at parameter "RESET", the present maximum values of the OBs will be reset.

NET01CUR REAL

Actual runtime of OB1. This parameter has to be interconnected with the parameter NET01CUR of FB @CPU_RT.

NET30CUR to NET38CUR REAL

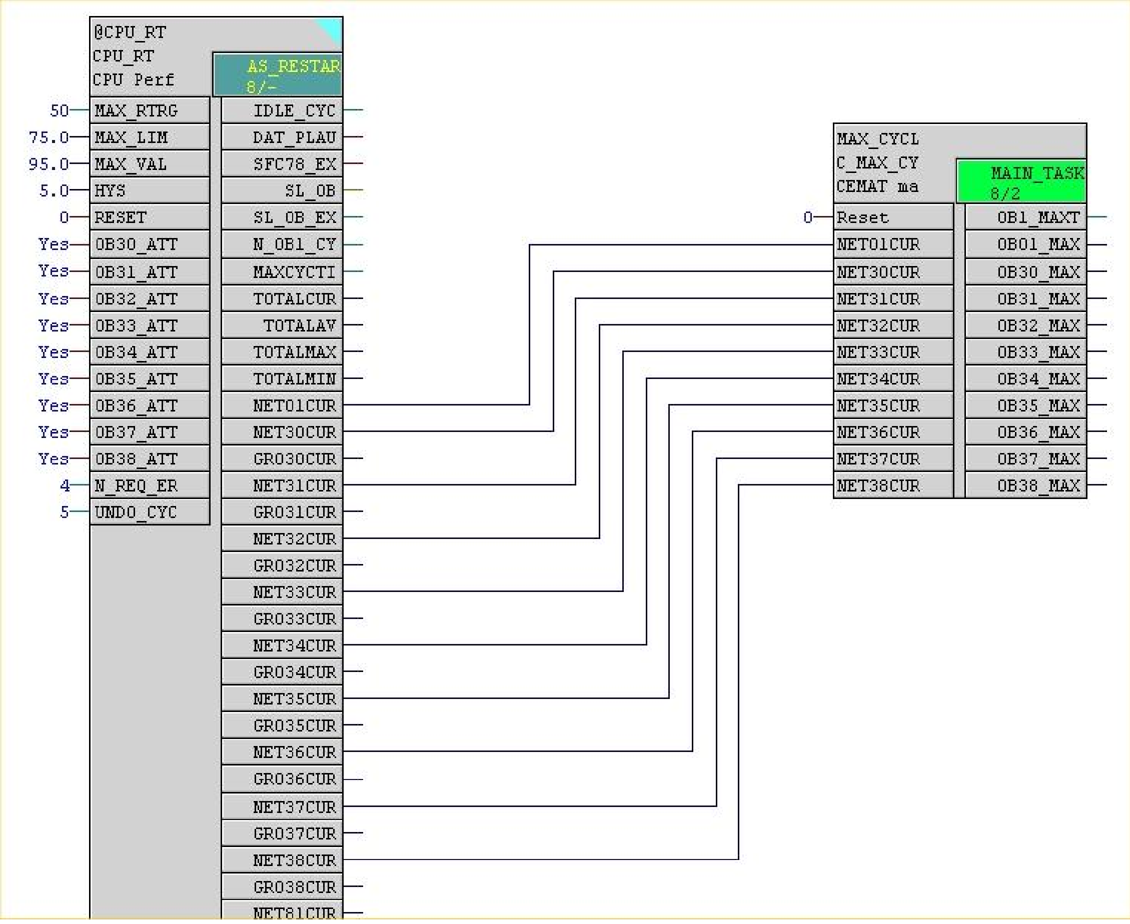
Actual runtime of the Cyclic Interrupt OBxx. This parameter has to be interconnected with the parameter NETxxCUR from FB @CPU_RT.

If you like to detect the maximum runtime of OB36 (50 ms), OB37 (20 ms) or OB38 (10 ms), then the OB1 runtime must be shorter than runtime of the cyclic interrupt OB, whose time you want to determine.

If the runtime of OB1 is longer than the runtime of the cyclic interrupt OB, then you have to make sure that the maximum runtime of the cyclic interrupt OB is still available at parameter NETxxCUR of block MAX_CYCL, when this is called in OB1.

In a small user program, you have to compare the actual detected Value from output NETxxCUR of @CPU_RT with the old maximum value and save the new maximum value until it is reset. The new maximum value must be connected to input NETxxCUR of MAX_CYCL. Of course the small user program has to run in the cyclic interrupt OB.

Connection between @CPU_RT and MAX_CYCL



Output interfaces:

RES_TIM1	DWORD	Date of reset (year, month, day)
RES_TIM2	DWORD	Time of reset (hour, minute, second, millisecond)
OB1_ACT	INT	Last actual runtime of OB1. The runtime is inclusive the time of the cyclic interrupt OB's, which has interrupted the OB1. The runtime is identified by the SFC 6 (RD_SINFO).
OB1_MAXT inclusive	INT	Maximum OB1 runtime since last reset. The runtime is the time of the cyclic interrupt OB's, which has interrupted the OB1. The runtime is identified by the SFC 6 (RD_SINFO).
MAXT_T1	DWORD	Time of OB1_MAXT (year, month, day)
MAXT_T2	DWORD	Time of OB1_MAXT (hour, minute, second, millisecond)
OB01_MAX	REAL	Maximum OB1 runtime since last reset. The actual runtime of the OB1 has to be connected to parameter NET01CUR.
MAX01_T1	DWORD	Time of OB01_MAX (year, month, day)
MAX01_T2	DWORD	Time of OB01_MAX (hour, minute, second, millisecond).
OB30_MAX	REAL	Maximum OB30 runtime since last reset. The actual runtime of the OB30 has to be connected to parameter NET30CUR.
MAX30_T1	DWORD	Time of OB30_MAX (year, month, day)
MAX30_T2	DWORD	Time of OB30_MAX (hour, minute, second, millisecond).
OB31_MAX	REAL	Maximum OB31 runtime since last reset. The actual runtime of the OB31 has to be connected to parameter NET31CUR.
MAX31_T1	DWORD	Time of OB31_MAX (year, month, day)
MAX31_T2	DWORD	Time of OB31_MAX (hour, minute, second, millisecond).
OB32_MAX	REAL	Maximum OB32 runtime since last reset. The actual runtime of the OB32 has to be connected to parameter NET32CUR.
MAX32_T1	DWORD	Time of OB32_MAX (year, month, day)
MAX32_T2	DWORD	Time of OB32_MAX (hour, minute, second, millisecond).
OB33_MAX	REAL	Maximum OB33 runtime since last reset. The actual runtime of the OB33 has to be connected to parameter NET33CUR.
MAX33_T1	DWORD	Time of OB33_MAX (year, month, day)
MAX33_T2	DWORD	Time of OB33_MAX (hour, minute, second, millisecond).
OB34_MAX	REAL	Maximum OB34 runtime since last reset. The actual runtime of the OB34 has to be connected to parameter NET34CUR.
MAX34_T1	DWORD	Time of OB34_MAX (year, month, day)
MAX34_T2	DWORD	Time of OB34_MAX (hour, minute, second, millisecond).
OB35_MAX	REAL	Maximum OB35 runtime since last reset. The actual runtime of the OB35 has to be connected to parameter NET35CUR.
MAX35_T1	DWORD	Time of OB35_MAX (year, month, day)
MAX35_T2	DWORD	Time of OB35_MAX (hour, minute, second, millisecond).
OB36_MAX	REAL	Maximum OB36 runtime since last reset. The actual runtime of the OB36 has to be connected to parameter NET36CUR.
MAX36_T1	DWORD	Time of OB36_MAX (year, month, day)
MAX36_T2	DWORD	Time of OB36_MAX (hour, minute, second, millisecond).
OB37_MAX	REAL	Maximum OB37 runtime since last reset. The actual runtime of the OB37 has to be connected to parameter NET37CUR.
MAX37_T1	DWORD	Time of OB37_MAX (year, month, day)
MAX37_T2	DWORD	Time of OB37_MAX (hour, minute, second, millisecond).
OB38_MAX	REAL	Maximum OB38 runtime since last reset. The actual runtime of the OB38 has to be connected to parameter NET38CUR.
MAX38_T1	DWORD	Time of OB38_MAX (year, month, day)
MAX38_T2	DWORD	Time of OB38_MAX (hour, minute, second, millisecond).

Run CEMAT blocks in a cyclic interrupt OB

One of the programming rules for Cemat blocks is that they have to be called in OB1 = MAIN_TASK, between the two run time groups OB1_START and OB1_END.

This has the following reasons:

- OB1 processes the blocks as fast as possible. The program can "respire", which means if in case of process faults of a lot of operation demand the program needs a little more time for some OB cycles, this does not matter at all.
- Certain functions like "Group status call", "Group instance list", "Show related Objects", "Show objects with Simulations", require the execution if the block within the same cycle.
- Also the acknowledgement of the faults only works if the programming order is followed and if the complete user program is called in the same cycle.

If you still want to call the Cemat Blocks in a cyclic interrupt OB (e.g. 100 ms task or 200 ms task), this only works properly if all Cemat blocks are called in the same cycle.

⚠ If you split the call-up of the Cemat blocks into different cyclic interrupt OBs or OB1, then the above mentioned functions are no longer guarantied.

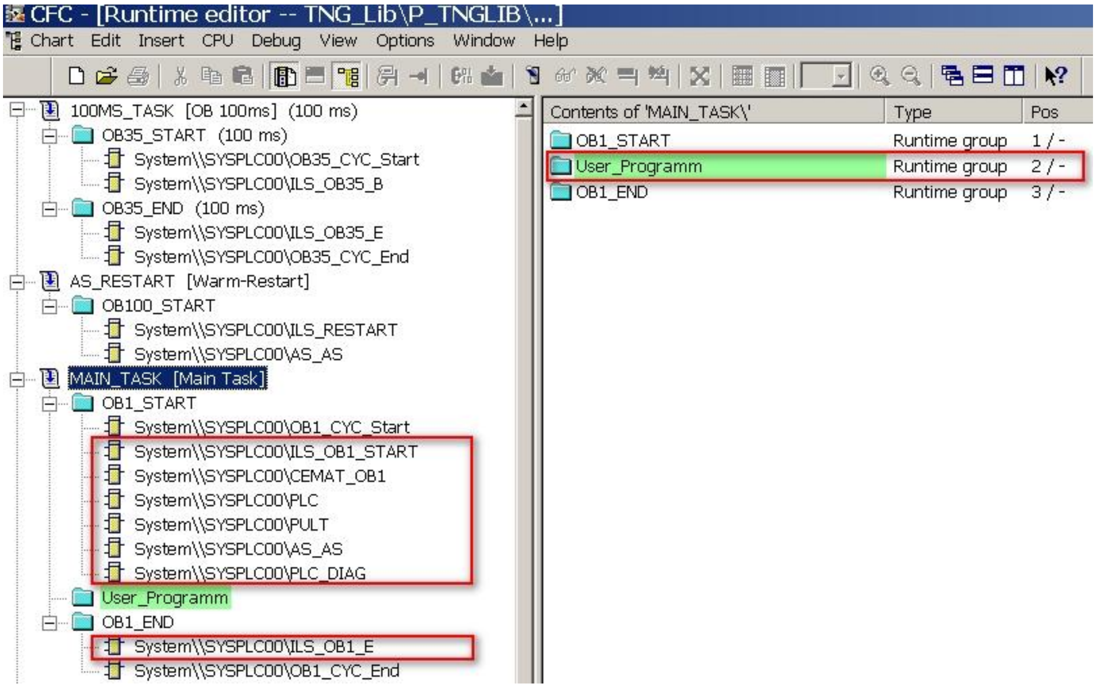
⚠ When moving the user program to a cyclic interrupt OB you also have to move some parts from runtime group OB1_START and OB1_END to the beginning and to the end of this cyclic interrupt OB.

⚠ You must make sure that the runtime of the User Program does not exceed the acquisition time of the cyclic interrupt OB, which means you have to insert it into the proper interrupt OB and to actualize the process image accordingly.

We think that it is not required to move the Cemat User program into a cyclic interrupt OB. If you still want to do this, please consider the following pages.

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\14_Tips_Tricks_009.doc

Runtime sequence if everything remains in OB1:



When moving the program into a cyclic interrupt OB, please proceed as follows:

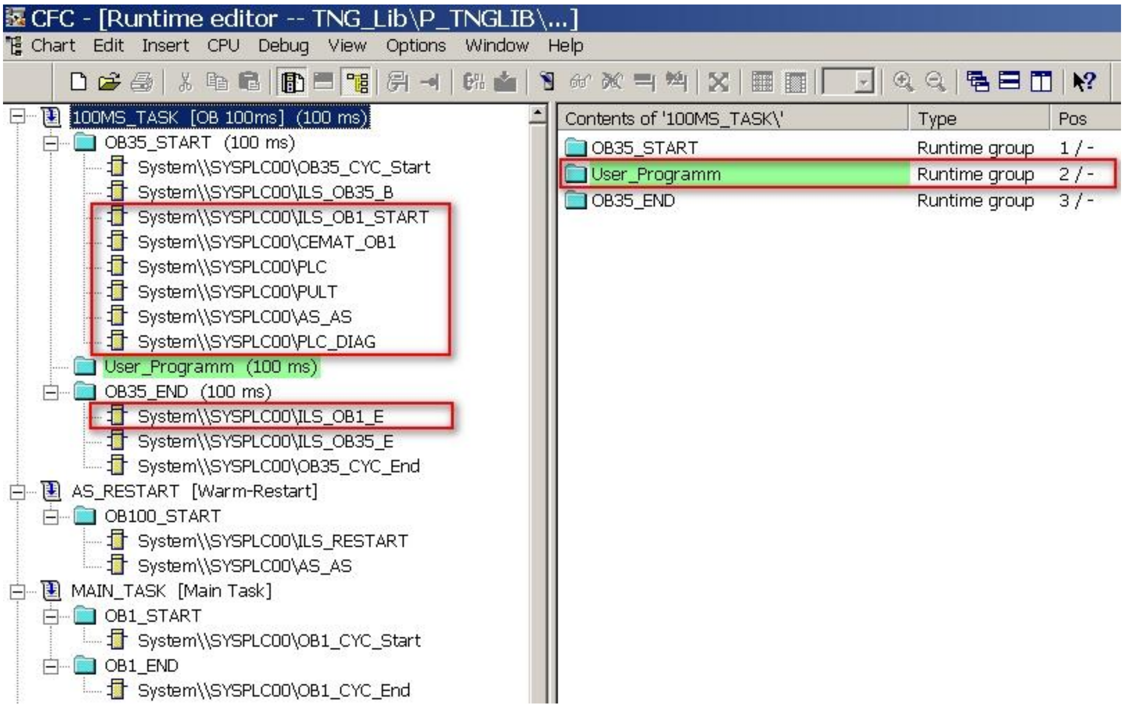
The red marked program parts under OB1_START and OB1_END (left side) must be moved to the cyclic interrupt OB.

The blocks OB1_CYC_Start and OB1_CYC_End are used for cycle time calculation and must remain in OB1.

The runtime group "User Programm" stands for the runtime groups of the application program which must be moved to the same cyclic interrupt OB (between START and END).

Follow always the Cemat Runtime Sequence:
1. possibly C_MUX blocks
2. Annunciations and Drives
3. Corresponding Routes
4. Corresponding Group

Runtime sequence after moving to OB35 (100ms):



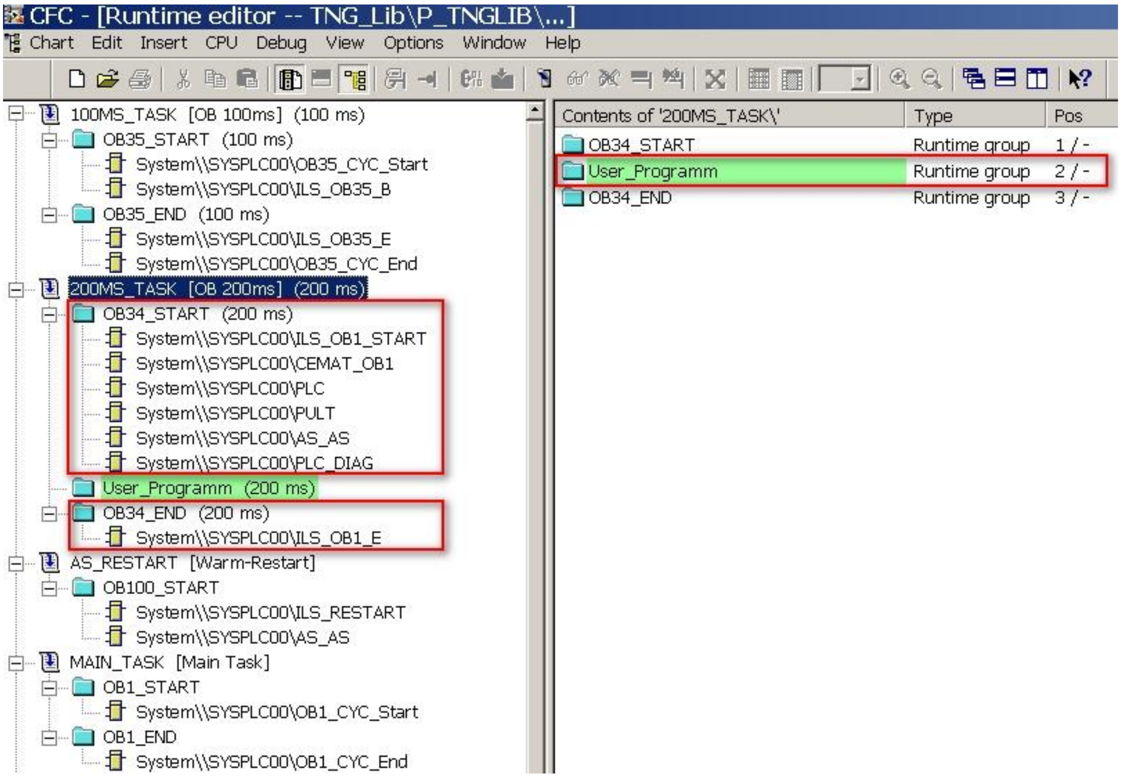
The red marked program parts have been moved from OB1 into OB35.

The blocks OB1_CYC_Start and OB1_CYC_End are used for cycle time calculation and must remain in OB1.

The runtime group "User Programm" stands for the runtime groups of the application program which must now be inserted between OB35_START and OB35_END.

Follow always the Cemat Runtime Sequence:
1. possibly C_MUX blocks
2. Annunciations and Drives
3. Corresponding Routes
4. Corresponding Group

Runtime sequence after moving to OB34 (200ms):



The red marked program parts have been moved from OB1 into OB34.

The blocks OB1_CYC_Start and OB1_CYC_End are used for cycle time calculation and must remain in OB1.

The runtime group "User Programm" stands for the runtime groups of the application program which must now be inserted between OB34_START and OB34_END.

Follow always the Cemat Runtime Sequence:
1. possibly C_MUX blocks
2. Annunciations and Drives
3. Corresponding Routes
4. Corresponding Group

Update Information

Contents

Edition: 07/16
N:\00_Cemat_Entwicklung\V8\V8200\30_Doku\Doku_Englisch\Engineering\000_Normal\15_Update_Information_009.doc

General

This document will help you to upgrade Cemat Projects of an older Version into the actual PCS 7 and Cemat Version. The document only describes the steps required for Cemat, regarding PCS 7 Update we like to refer to the corresponding PCS 7 Manual.

For the Cemat Update we assume that the Cemat engineering rules have been followed. If your project contains additional functions and own applications, you have to consider these in addition.

At this point we would like to refer to the document readme_V82.pdf, which you find on the Cemat Installation-DVD. There you find detailed information about the modifications from Version to Version.

We also like to refer to the Cemat Documentation. This you find in the Intranet and after the Cemat Installation the documents are available online on your PC.

Remarks to the Document

This document describes different upgrade variants: What has to be done in order to upgrade from the actual Version to PCS 7 V8.2 and Cemat V8.2? The details you find in the individual chapters.

If you want to upgrade from V5 or V6.0 some additional points have to be considered. These points you find in the chapters “Additional information for the Update of CEMAT V5 Projects“ and “Additional information for the Update of CEMAT V6.0 Projects “.

It is not yet clarified if the upgrade of OS Projects with migrated S5 PLC’s (CEMAT V2, V3, V4) is still possible or not. Maybe finally now you have to replace your S5 PLC by S7 PLCs.

Working steps

Each upgrade basically consists of the following steps:

- Check Hardware- and Software Requirements
- Save the project data
  - For saving the PCS 7 Multiproject use the SIMATIC Manager
  - For saving the User Archives use Export Function
  - For saving the Modifications in the CEMAT_CS Configuration copy the folders Config, Multimedia und Sounds.
- Check for changes/new Functions in the readme_V82.pdf and in the Cemat Object descriptions.
- Upgrade the PCS 7 Project to PCS 7 V8.2.
  In some cases intermediate steps are required. Refer to the descriptions in the PCS 7 Manuals.
- Eventually you have to adapt your Project structure (z. B. Multiproject)
- Installation of Cemat V8.2
- Adaptations in the Config files
- Update of the S7 Programs (Master data Library and AS)
- Update of the OS-Projects
- Compile and Download

Languages

The blocks of the APL Library use the following standard languages:

- German (Germany)
- English (US)
- French (France)
- Italian (Italy)
- Spanish (Spain, international sort)

For this reason the CEMAT blocks (messages) are provided with the same languages.

⚠ **Caution:** In former CEMAT Versions *Spanish (Spain, traditional sort)* has been used. The new blocks have been changed to *Spanish (Spain, International sort)*. You have to change the "Installed languages in Project" accordingly. In case of a Spanish project with default language in Spanish make sure that you don't lose your block comments.

Additional information for the Update of CEMAT V5 Projects

Caution: For the migration of a PCS 7 project V5.x you need an engineering station with PCS 7 V6.0. Only in this version you have an OS migrator, which is able to convert the V5 Sybase database into a MS SQL database.

For the migration of a PCS 7 Project V6.0 / V6.1 /V7.0 you need additionally an engineering station with PCS 7 V7.1 SP3 or PCS 7 V7.1 SP4. Only in this version you have an OS migrator, which is able to convert all older versions with a MS SQL database to the version of V7.1 SP3 (or V7.1 SP4).
The OS Migrator of PCS 7 V8.2 only converts Projects with Version >= V7.1 SP3 to Version V8.2. WinCC converts the Project automatically the first time you open the OS Project. Please check the PCS 7 manual "Software Update with utilization of the new functions”.

If you want to update your project from Cemat V5 to Cemat V8.2, first carry out a PCS 7 Migration to V6.0 (following the PCS 7 Manual "Software-Actualization from PCS 7 V5.x to PCS 7 V6.0").

Then follow the steps described in chapter “Update CEMAT V6.1 to CEMAT V8.2” in order to convert the Project to CEMAT V8.2.

The replacement of the Cemat blocks by the new blocks is carried out at the end (see chapter “ update to V8.2), if the PCS 7 Project is already converted to Version 8.2.

The following additional aspects have to be considered in case of the migration of a V5 Project:

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

At this point we would like to refer to the fact that since CEMAT V6 SP2 CEMAT does not contain the controller functions of the PTE library any more.
They have been replaced by the standard controller functions of the PCS 7 library.
We recommend for the future to use the standard controller functions of the PCS 7 library or to use the standard controller functions of the PCS 7 APL library.
If you like to use still the PTE controllers for existing applications, you have to take care, that all the old faceplates, OCX files are installed in your project and on your new PCs.

Information to the Update of S7 Program

Create a Multiproject

In PCS 7 V5 only Projects exist, but PCS 7 V8.2 works with Multiprojects. You have to convert your project into a Multiproject. The best is to create one partial project per AS, one partial project for each Engineering Station and one partial project for Servers and Clients. In Addition you have to create a Project Library (Master Data Library).

If you are lucky you can copy the AS from your old Project into the new project structure. If this is not possible, you have to create new AS, then export the HW Configuration from the old AS and import it into the new AS. Then also copy the symbols, blocks and charts.
Caution: Symbols with Attribute S7_m_c don't keep this information during the copy function. The Attribute must be set again.

In the HW Configuration you may have to increase the range for local data because the new blocks need more local data memory.

The OS-Projects must be created again. The migration of the old OS-Projects does not make sense.

Create a Master data library

Create an (empty) Master data library which at the end must contain all S7 blocks used in the Project.

Message numbers

If this is not yet done you have to change the message numbering definition to CPU-oriented unique message numbers:
Customize -> Message numbers -> Select "Always assign CPU oriented unique message numbers". With the function "Project Save As" and the option "With reorganization (slow)" will all CPU programs are converted to CPU-oriented unique message numbers.

New message texts

From Cemat V6.1 the blocks don't use abbreviations in the Message text any more, they use wording instead, e. g. "EBM" was changed into "Overload". If you have modified the message texts in your project you have to carry out these adaptations again after the update of the blocks.

Plant Hierarchy

The picture hierarchy and the summarizing indications are derived from the plant hierarchy. For this reason the charts as well as the pictures must be assigned to the correct hierarchy folders, even if the block icons are not created automatically.

CFC settings

The FC-Area reserved for "Other applications" has been reduced and ends with FC1399, in order to gain more space for the FCs created from the CFC. Probably you have to move some of your User FCs.

Hardware Configuration

In der Hardware Configuration of each CPU the minimum cycle time must be set to 100 ms.

Block modifications

In case of the migration of a V5 Project some special points have to be considered when exchanging the blocks:

- C_DRV_1D, C_DRV_2D: Additional block C_SPEEDM has been removed. The Pulse input must directly be configured at the motor block.
- C_DRV_1D, C_DRV_2D: If there is a C_MEASUR connected for the motor current, then the parameter REL_MVC must be set to "1".
- The blocks C_DRV_S2 and C_DAMP_S are no longer supported. Please use the blocks C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE with the SIMOCODE adapter block C_SIMOS.
- The block C_ANNUN8 is reduced to 7 alarms. The function / reaction of the block has also been changed. Please check your interconnections.
- C_SILOP: Additional block C_ SPCNT has been removed. The Pulse input must directly be connected to the Silo pilot block
- C_MEASUR: Only for HZ or if you are using the service function. Configure the parameter BYPB_ACT.
- All CEMAT blocks: If you have used block outputs, like REL_OS, INTFC_OS, MODFL_OS, PAR_OS, STATUS, etc. for interconnection, then you have to connect

to the new status words. The old output parameters do not exist anymore or the assignment has been changed.

- For all C_ANNUNC blocks, copy the 'Message text' into Parameter IN_DEL under 'Identifier'. The text length is limited to a maximum of 16 characters.
- In the C_MEASUR block we internally use ALARM8_P instead of ALARM8. This results in a loss of the block comment under 'Free Text 1'. You have to copy the block comment again (can be done in the process object view).

Note: Additional information regarding modifications in the S7 blocks you find in the readme_V82.pdf and in the object descriptions.

Information to the Update of the WinCC Project

Using of the Plant Hierarchy

The old area selection method in the message line, which was used until Cemat V6.0 is no longer possible in Cemat V8.2. For the area selection it is now absolutely necessary to derive the picture tree from the plant view, which means the process pictures have to be assigned to the correct place in the plant view.

Migration of the Process Pictures

The OS-Projects have to be created again and must be configured according to the rules described in the Engineering Manual Chapter 3. From the old process pictures you may use the background picture and the symbols, but the block icons must be created and positioned again.

Eventually it is necessary to create a new template picture with block icons.

Note: If you have written your own scripts you have to convert these as well. The same applies to your own Faceplates.

For further stops follow the chapter "Update CEMAT V6.1 to CEMAT V8.2".

Additional information for the Update of CEMAT V6.0 Projects

Caution: For the migration of a PCS 7 Project V6.0 / V6.1 /V7.0 you need additionally an engineering station with PCS 7 V7.1 SP3 or PCS 7 V7.1 SP4. Only in this version you have an OS migrator, which is able to convert all older versions with a MS SQL database to the version of V7.1 SP3 (or V7.1 SP4).
The OS Migrator of PCS 7 V8.2 only converts Projects with Version >= V7.1 SP3 to Version V8.2. WinCC converts the Project automatically the first time you open the OS Project.
Please check the PCS 7 manual "Software Update with utilization of the new functions".

If you want to update your project from Cemat V6.0 to Cemat V8.2, first carry out a PCS 7 Migration to V6.1 (following the PCS 7 Manual "Software-Actualization from PCS 7 6.0 to PCS 7 V6.1").

Then follow the steps described in chapter “Update CEMAT V6.1 to CEMAT V8.2” in order to convert the Project to CEMAT V8.2.

The replacement of the Cemat blocks by the new blocks is carried out at the end (see chapter “ update to V8.2), if the PCS 7 Project is already converted to Version 8.1.

The following additional aspects have to be considered in case of the migration of a V6.0 Project:

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT installation DVD.

At this point we would like to refer to the fact that since CEMAT V6 SP2 CEMAT does not contain the controller functions of the PTE library any more.
They have been replaced by the standard controller functions of the PCS 7 library.
We recommend for the future to use the standard controller functions of the PCS 7 library or to use the standard controller functions of the PCS 7 APL library.
If you like to use still the PTE controllers for existing applications, you have to take care, that all the old faceplates, OCX files are installed in your project and on your new PCs.

Information to the Update of S7 Program

Message numbers

If this is not yet done you have to change the message numbering definition to CPU-oriented unique message numbers:
Customize -> Message numbers -> Select "Always assign CPU oriented unique message numbers". With the function "Project Save As" and the option "With reorganization (slow)" will all CPU programs are converted to CPU-oriented unique message numbers.

New message texts

From Cemat V6.1 the blocks don’t use abbreviations in the Message text any more, they use wording instead, e. g. "EBM" was changed into "Overload". If you have modified the message texts in your project you have to carry out these adaptations again after the update of the blocks.

Create a Master data library

Create – if not yet existing - an (empty) Master data library which at the end must contain all S7 blocks used in the Project.

Hardware Configuration

In der Hardware Configuration of the CPU the minimum cycle time must be set to 100 ms.

Block modifications

In case of the migration of a V6 Project some special points have to be considered when exchanging the blocks:

- C_DRV_1D, C_DRV_2D: Additional block C_SPEEDM has been removed. The Pulse input must directly be configured at the motor block.
- C_DRV_1D, C_DRV_2D: If there is a C_MEASUR connected for the motor current, then the parameter REL_MVC must be set to “1”.
- The blocks C_DRV_S2 and C_DAMP_S are no longer supported. Please use the blocks C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE with the SIMOCODE adapter block C_SIMOS.
- The block C_ANNUN8 is reduced to 7 alarms. The function / reaction of the block has also been changed. Please check your interconnections.
- C_SILOP: Additional block C_ SPCNT has been removed. The Pulse input must directly be connected to the Silo pilot block
- C_MEASUR: Only for HZ or if you are using the service function. Configure the parameter BYPB_ACT.
- All CEMAT blocks: If you have used block outputs, like REL_OS, INTFC_OS, MODFL_OS, PAR_OS, STATUS, etc. for interconnection, then you have to connect to the new status words. The old output parameters do not exist anymore or the assignment has been changed.

Note: Additional information regarding modifications in the S7 blocks you find in the readme_V82.pdf and in the object descriptions.

Information to the Update of the WinCC Project

Using of the Plant Hierarchy

The old area selection method in the message line, which was used until Cemat V6.0 is no longer possible in Cemat V8.2. For the area selection it is now absolutely necessary to derive the picture tree from the plant view, which means the process pictures have to be assigned to the correct place in the plant view.

Migration of the OS Projects

Remark: We recommend creating the OS-Projects again. This is much easier than making all the necessary modifications in the existing OS-Projects. In this case you lose your archive data but at the end you have a proper project structure.

If you still want to continue with your existing OS-Projects, you have to carry out these steps additionally to the steps which are explained in the following chapter:

1. Delete all CEMAT Standard Scripts from the PCS 7 Project '<OS>\Library'. Keep only your own Scripts in PCS 7 Project '<OS>\Library'.
2. The CEMAT Standard Global Scripts in PCS 7 Projekt '<OS>\Pas' are no longer required and must be deleted.
3. The VBS Actions in PCS 7 Project '<OS>\ ScriptAct' are no longer required and must be deleted.
4. Open the WinCC Explorer.

   We recommend going through the project settings described in the Engineering Manual chapter 3 and correct your OS-Projects accordingly. This is just a bullet point list and does not describe the details:

   - Change WinCC Design from 'Classic' to '3D'
   - Correct settings in the OS Project Editor according to the Engineering Manual, chapter 3
   - Correct the computer settings
   - Add the internal variables.
   - Check the settings in the Tag Logging and correct them if necessary
   - Check the settings in the Alarm Logging and correct them if necessary
   - The User Archive C_INFO has got some new columns.
     Export the existing date of C_INFO. Add the additional columns, using the excel tool C_INFO.xlsm.
     Delete the old C_INFO and import the new structure. After this import the data with the additional columns.
   - Add new user rights and area specific rights.
   - Check the Horn configuration
   - If any Scripts or system-PDLs have been modified in your project, you have to carry out these modifications in the new Scripts or PDLs again after the update is finished.
   - The old template pictures @PCS7Typicals.pdl and @Template.pdl cannot be used anymore, because of the additional functions of the block icons. You have to create a new library based on the new template picture.

Note: If you have written your own scripts you have to convert these as well. The same applies to your own Faceplates.

For further stops follow the chapter "Update CEMAT V6.1 to CEMAT V8.2".

Update CEMAT V6.1 to CEMAT V8.2

You have a CEMAT V6.1 project (based on PCS 7 V6.1) and want to upgrade to CEMAT V8.2 (based on PCS 7 V8.2). The following description explains the necessary steps.

Requirements

Before the installation of Cemat V8.2, you have to upgrade your PC Stations to the permitted Windows Operating System and to PCS 7 V8.2.

The following Windows Operating Systems are supported by PCS 7 V8.2, but not all versions are compatible with each Configuration.

- Windows 7 Professional SP1 (64-Bit, only English Windows Language Version)
  (permitted for ES, non-red. OS-Single Station, OS-Client, Web-Client)
- Windows 7 Ultimate/Enterprise SP1 (32-Bit)
  (permitted for OS-Client and Web-Client)
- Windows 7 Ultimate / Enterprise SP1 (64-Bit)
  (permitted for ES, OS-Single Station, OS-Client, Web-Server (only on ES or OS-Single) and Web-Client)
- Windows 10 Enterprise 2015 LTSB (64-Bit)
  (permitted for OS-Client and Web-Client)
- Windows Server 2008 R2 SP1 Standard Edition (64-Bit)
  (permitted for ES, OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client)
- Windows Server 2012 R2 Update Standard Edition (64-Bit)
  (permitted for OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client

Further details about the installation of the operating system and PCS 7 are explained in the PCS 7 manuals and in the readme and therefore not part of this description.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2
Make sure that the PCS 7 updates are installed before updating to CEMT V8.2!

Caution: The update to CEMAT V8.2 requires a complete reloading of the PLC and consequently **cannot** be carried out while the plant is running.

Caution: For the migration of a PCS 7 Project V6.0 / V6.1 /V7.0 you need additionally an engineering station with PCS 7 V7.1 SP3 or PCS 7 V7.1 SP4. Only in this version you have an OS migrator, which is able to convert all older versions with a MS SQL database to the version of V7.1 SP3 (or V7.1 SP4).
The OS Migrator of PCS 7 V8.2 only converts Projects with Version >= V7.1 SP3 to Version V8.2. WinCC converts the Project automatically the first time you open the OS Project.
Please check the PCS 7 manual "Software Update with utilization of the new functions”.

Follow the actual Manual "PCS 7 Software Update with utilization of the new functions" from the PCS 7 V8.2 Documentation. The following description refers to this manual.
Please also pay attention to the hints for the OS-Migration. In order to maintain the runtime data (curves and messages), the Server Projects have to be migrated on the Server-PCs directly.

Consider also the project rules from the CEMAT manual 03_PCS7_Project_009.pdf, "Installation of a PCS 7 Project" and the Cemat Object descriptions.

Save Project data

Save your project data. Beside the Multiproject this comprises all settings which you have carried out in the CEM_V6 Directory, as well as modifications in the block library ILS_CEM or PRO_CEM.

- Archive the Multiproject with the SIMATIC Manager
- Export the Data of the User Archive C_INFO if you want to continue to use it after the upgrade. The structure of the User Archive C_INFO has been changed.
- You can copy the modifications under CEM_V6 (Config, Multimedia, Sounds) to a backup-directory.

Caution: Never save directory „CEM_V6/BIN"!!!
The files from the BIN directory are entered in the registry and must never be moved or deleted.

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

Please consider that the blocks C_SINA and C_ROBI (AddOn) are completely new designed. The existing blocks must not be used anymore. <u>They must be removed and replaced by a C_DRV_1D or C_DRV_2D block plus the new adapter block C_SINA or C_ROBI.</u>

Migration to PCS 7 V7.1 SP3

Open your project on a PCS 7 V7.1 SP3 Engineering Station.

Migration of the CFC Charts

Open any CFC chart and slightly move a block. A dialog box appears, asking you for conversion of the CFC PCS 7 V7.1 SP3. Acknowledge this dialog that appears with "OK". The Charts will be converted to V7.1 SP3.
Repeat this step for all chart folders of your Multiproject.

The block container still has the old S7 blocks. Block exchange is carried out later with the migration to V8.2.

OS-Migration

Start the OS Migrator and choose the MCP file of the WinCC project. Do this for all WinCC projects in your multi project.

Archive the multi project with the SIMATIC Manager and copy it to the V8.2 Engineering Station.

Migration to PCS 7 V8.2

OS-Migration

Again refer to the OS-Migration in the Manual "PCS 7 - Software Update with utilization of the new functions".

You have to migrate all OS Projects in your Multiproject. This is carried out under:
Start → SIMATIC → WinCC → Tools → Project Migrator.

Plant View

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

Caution: If your picture tree was nevertheless generated manually you have change it. Now you have to generate the picture tree from the plant view.

Installation of CEMAT V8.2

With the following steps please proceed on a Station with PCS 7 V8.2!

Start the Setup program from the DVD CEMAT V8.2 and follow the instructions.

The Installation path is 'D:\CEMAT_CS'. This cannot be changed!

Keys for project standards

000 = CEMAT
001 = Minerals Cemat
004 = Holcim
006 = Dyckerhoff
007 = Heidelberg Cement
023 = Vigier
024 = Bushehr
025 = Caima
026 = Alsen
027 = Lafarge
028 = Rossi

After the installation, for all PC stations (ES, OS SINGLE STATION, OS SERVER and OS CLIENT) the following steps have to be carried out:

1. The config files contain the default settings, which means if you changed them for your project you have to make this adaptations again. (C_Config.cfg)
   Also consider the settings in file C_Messages.cfg.
2. Probably you also added for your project some files under 'Multimedia', 'Sounds' ….
   After the CEMAT installation these files can be restored from the backup directories and copied into the new additional CEMAT_CS\MULTIMEDIA directories.
3. If you want to use the PTE-Controller Function, then you have to install the PTE-Controller OS part (OCX files, PDLs, scripts, CFG files) before you migrate your PCS 7 Project. Otherwise the standard PDLs (@PG_C_??????.PDL) will be wrong converted and cannot be used any more.
   Copy all necessary files for the PTE controller function to D:\CEMAT_CS\ into the corresponding directories. Register the OCX files using the BAT file. Refer to the PTE-Controller instructions.
   The use of the old PTE controllers you do in your own responsibility.

Update of S7 Program

Please also consider the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“.

Display languages

Caution: In case of using Spanish language, you have to change the language of all partial projects and libraries into "Spanish (Spain, international sort)". Special caution is needed if "Spanish (Spain, traditional sort)" was used as default language. Make sure that you don't loose any texts.

New CEMAT Libraries

With the installation of CEMAT V8.2 a new version of the CMEAT Library ILS_CEM has been installed in directory C:\Program Files\SIEMENS\STEP7\S7libs.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (project key other than “000”) the library PRO_CEM is also transferred into C:\Program Files\SIEMENS\STEP7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Program folder **Cemat_AddOn** contains Open blocks as e. g. C_SIMOS, C_SINA, C_ROBI. They are delivered in a separate folder in order not to overwrite the existing program during update procedure.

In order to avoid overlapping with other libraries, in CEMAT V8.2 some AS blocks of the ILS_CEM library have been transferred to new addresses (see list below).
The transfer concerns all CEMAT Project Standards with exception of Project Standard 001 (In the Minerals Automation Standard the new block numbers have been used from the beginning).

Block name	**old Number**	**→ new Number**
C_PID3	FB1008	→ FB1018
C_ODA	FB1060	→ FB1046
C_ODA_R	FB1062	→ FB1047
C_INTERL	FB1075	→ FB1057
C_INTER5	FB1076	→ FB1058
C_RelMod	FB1077	→ FB1059
AG_LSEND	FC1050	→ FC1033
AG_LRECV	FC1060	→ FC1034
FC_GALA-WALA	FC1061	→ FC1021
FC_GRUZU_WEGZU	FC1062	→ FC1022
FC_GRINZ	FC1063	→ FC1023
StruInIn	FC1065	→ FC1025
StruInOu	FC1066	→ FC1026
C_SelQC	FC1067	→ FC1027
C_PUSHBT	FC1088	→ FC1028
OB1_SYS1	FC1102	→ FC1031
C_OB1SY1	FC1103	→ FC1032

Furthermore in the ILS_CEM all non-used CEMAT blocks have been removed. These are:

FB128	
FC6	DT_DATE
FC509	BCD_SKCNT_CHECK
FC510	COPY_ELEM
FC516	FC_AUTO_APPEND
FC517	FC_GET_STRUCT_LEN
FC519	VAR_TO_DBNR
FC520	VAR_TO_BYNR
FC521	DIAGNOSE
FC522	BYTE_TO_INT
FC523	INT_TO_BYTE
FC524	BCD_TO_BYTE
FC525	BLOCK_DB_TO_INT
FC1020	C_SIMU_L
FC1064	FC_SIMU_LIST
DB676	C_SIMU_DB
DB679	DB_FATAL_ERROR
DB681	DB_DIAGNOSTIC
DB682	DB_EVENT
UDT1016	DIAGNOSE_ENTRY

Caution: The CEMAT Object blocks don’t use the Abbreviations like „ESB“ in the event texts any more. Instead of this texts like „Available“ are used.
If you have adapted the Event texts for your Project you have to save these texts in order to manually correct it after the block update is completed.

New PCS 7 Libraries

With the installation of PCS 7 V8.2 also some new block libraries were installed:
- PCS 7 Basis Library V82
- PCS 7 AP Library V82.

A PCS 7 Library V82 doesn't exist!!!!!!!!
If you don't like to use the new APL blocks and you like still to use some blocks of the Basis Library and PCS 7 Library V71, then you have to install the newest version of the V7.1 libraries. You find the setup for the V7.1 libraries on the on the PCS 7 V8.2 Installation DVD under "Additional Products".

The newest PCS 7 Library V7.1 SP3 contains 2 versions of some channel driver blocks (e. g. CH_AI). The 2 blocks are identical, but they have different version numbers (V6.0 or V7.0).
The channel / driver blocks of the V7.1 have the version number V6.0.
The channel / driver blocks of the V8.0 have the version number V7.0.
A mix of the version number of the used channel / driver blocks is not allowed!!!!

Update Procedure for the S7-Program

For the update of your S7-Programs please proceed as follows:

- First update the Master data Library
- Then update the S7-Program containers for each AS
- Update the CFC for each AS

Actualize the Master data Library

The Master data library must contain all blocks which are used in the different S7-Projects of the Multiproject. These are the CEMAT blocks as well as block from other PCS 7 libraries which are used in the Multiproject

1. Open the block folder of the Master data library.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Delete blocks TE_STWD und TE_LIMHL from the symbol list as well.
4. Open the block folder CEM_ALL of the CEMAT library ILS_CEM.
5. Select all blocks Copy all blocks from CEM_ALL and copy these into the block folder of your Master data library.
6. If a CEMAT Project Standard was installed, open library PRO_CEM and copy all blocks (overwrite = yes) into the block folder of your Master data library.
7. Your Master data library should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks cannot be used any more, but do not replace it by the new Adapter block now; otherwise you will lose your connections during the update.
8. If in your project system attributes or message attributes in the standard blocks have been modified, you have to make the adaptations after the update of the block folder manually again.
9. In the new CEMAT blocks in the Event texts the abbreviations like “ESB” have been replaced by long text like “Available”. If have adapted the Event text in your Project, you must make these modifications after the update of the block folder manually again.
10. At this time you should also check,
    - if all other blocks (e. g. with Author BASISxx, TECHNxx or DRIVERxx, etc) are actualized.
    - that the Master data library does not contain any block with Author ES_MAP
    - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!)..
11. Open the CFC editor by opening any chart.
12. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
13. In the System Chart SYSPLC00 add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.

After this the Actualization of the Master data library is completed.

Actualization of the S7-Program for each AS

For each AS you have to actualize the **S7-Program container** from the Master data library:

1. Open the block folder of your AS.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Delete blocks TE_STWD und TE_LIMHL from the symbol list as well.
4. Then copy all Blocks from your Master data library into the block container of the AS.
5. Your AS block folder should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks must be manually replaced after the CFC update by a C_DRV_1D or C_DRV_2D plus the new adapter block C_SINA or C_ROBI. Automatic replacement of C_SINA or C_ROBI is not possible, existing connections get lost!
6. If required, actualize further blocks of the S7-Program (e.g. according to the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“, from 01/2013, chapter 7.5.15).

Actualization of the CFC Charts

Through the change of existing parameters to structure format, during the update procedure some connections will be converted to Textual connections.
Some connections may get lost and must be engineered again. Using the Excel Tool C_MIG_V80.XLS the existing connections can be rewired.
For this purpose you have to make some preparations before the actualization of the CFCs:

Carry out the following Steps for each AS:

1. In the Process Object view, select the complete AS and actualize the data with F5. Convert the CFCs at the same time into the format of PCS 7 V8.2. After this export the data as follows:
2. Carry out the function *Process Objects → Export all IOs*… As file name choose e. g. AS1_PS_old.csv.
3. Carry out the function *Process Objects → Export Blocks* … As file name choose e. g. AS1_B_csv.

For each AS, carry out the following steps in order to actualize the CFC Charts:

4. Open the CFC editor by opening any chart.
5. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
6. Please check the log file after the CFC update:
   The LOG file must be saved, e. g. with name AS1_Textual Interconnections.txt!
   - Connections to Parameters which could not been found in the new version (e. g. for C_SIMDMP) have been removed or converted to textual connections.
   - Please also check for textual connections, using the function *Options → Close Textual interconnections*.
   - After closing the textual interconnections the list should be empty. Otherwise you need to correct the program manually!
7. For finding the removed connections the Excel Tool C_MIG_V80.XLS may be used. The Tool compares the program BEFORE and AFTER the block update and creates a list of differences. Some connections can also be corrected automatically with the tool.

   If you want to use the tool, select in the process object view the complete AS and actualize the data with F5. Again carry out the function *Process Objects → Export all IOs…*
   As file name choose e. g. AS1_PS_V82.csv.

   The description for the Excel Macro you find in an extra chapter.
   Refer to “CEMAT Tools C_MIG_V80.XLS and C_SIM_RIGHT.XLS” at the end of this document.

   The Excel tool generates an Import file which you can import in the process object view, using the function *Process Objects → Import IOs…*
8. The user rights for Simulation of C_ANNUNC, C_ANNUN8, C_MEASUR and C_PROFB could be modified at the block icon before and now this is done via block parameter SimRight.
   The Excel Tool C_SIM_RIGHT.XLS evaluates the settings at the block icons and allows importing this into the CFC.
   If you did not change the attribute "HigherProcesscontrolling" on the block icons, i.e. the attribute has the value 24, then skip this part and proceed with the next point.
   In case of a modification, use Excel Macro C_SIM_RIGHT.XLS, which is explained in a separate chapter.
   Refer to “CEMAT Tools C_MIG_V80.XLS and C_SIM_RIGHT.XLS” at the end of this document.
9. Through the change of the variable format (e. g. from REAL to STRUCT) it may happen that a parameter which was switched to visible before is invisible in the CFC after the

upgrade. If an interconnection exists for an invisible parameter you can see this at the blue triangle in the header of the CFC block. If required you have to switch it to visible again.

10. In the new version the size of the blocks may be different from the version before. This may lead to overlapping in the CFC. Eventually it is necessary to move some of the blocks in the CFC.
11. **Only Holcim**: Add a new OR gate with the dynamic faults (EST, KST, VST) of all drives which are involved in the material transport and connected it to FLT_MAT of the C_GROUP.
    The output N_EMPTY of the C_GROUP module is no longer working like before.
    Function until V7.1: All dynamic faults of the related objects set the output N_EMPTY. A normal stop reset the output N_EMPTY, even if there is still material on a belt.
    New function since V7.1 SP1: The new input FLT_MAT set the output N_EMPTY. A normal stop doesn't reset the output N_EMPTY. N_EMPTY must be reset by the Operator after the conveying line is cleared.
    To get this new function, you need an OR gate with the dynamic faults (EST, KST, VST) of all drives which are involved in the material transport. This new OR gate have to be connected to the new input FLT_MAT.
12. Because of extended possibilities for the quality evaluation the rules for connecting Cemat Block and Driver Block have been changed.
    If you use module driver together with C_ANNUNC or C_MEASUR please check the connection between the blocks (see AS_Engineering.pdf).
13. At this time you should also check,
    - if all blocks (e. g. with Author BASISxx, TECHNxx, CHNxx or DRIVERxx, etc) are actualized.
    - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
14. Adaptations for C_SINA and C_ROBI
    You have to replace the blocks by a C_DRV_1D/C_DRV_2D block + the new Adapter bocks C_SINA or C_ROBI. New Engineering required!
15. In the System Chart SYSPLCxx add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.
    Especially for Holcim plants: please check parameter REL_WSTP of block C_FB_PLC. REL_WSTP must be “0” in order not to interrupt the group start in case of a warning!
16. In folder D:\CEMAT_CS\TOOLS\CheckTool you find the new CEMAT Check tool “CematCheckTool.exe”. Please make use of this tool after upgrading your project.
    The check tool will tell you if your project structure is correct; e. g. if the CEMAT blocks are inserted at the correct installation position or if the Group links and C_MUX interconnections are correct.
    For details refer the CEMAT manual „10_Checkliste_009.pdf“.

After this block update a complete compile of all charts is required.

Carry out the same steps for all S7 Program folders of your Multiproject.

Update of the OS-Projects

Before the Cemat Update all OS Projects must be migrated to PCS 7 V8.2.

CEMAT Update

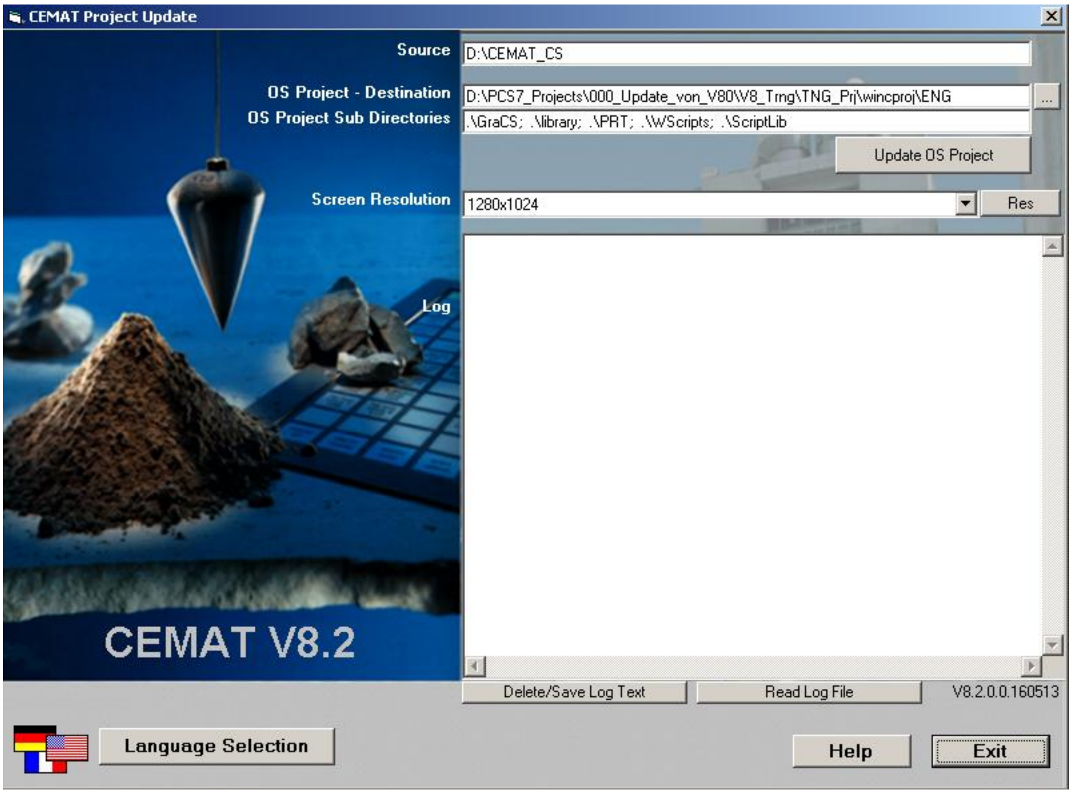
After the CEMAT installation, the D:\CEMAT_CS\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

In all WinCC Projects carry out the following steps:

1. Open the WinCC Explorer.
2. Change the WinCC Design from 'Classic' to '3D':
   With right mouse click on the OS Project, select the Project Properties. Open Tag "User Interface and Design" and switch the Active Design from 'WinCC Classic' to WinCC '3D'.
3. Open the OS Project Editor. Check the settings according to the Engineering Manual Chapter 3 "Installation of a PCS 7 Project".
   Close the OS Project Editor with "OK". All standard pictures (Overview, buttons, alarm line, etc.) are replaced by the new standard PDLs of PCS 7 V8.2.
4. Close WinCC Explorer.
5. Delete all CEMAT standard scripts in your PCS 7 project '<OS>\Library'. Keep your own scripts in PCS 7 project '<OS>\Library'.
   Since V7.0 the Cemat scripts are located in subfolders and must be deleted from the main folder.
6. Also delete the VBS Action 'Cem_Tags.bac' from the PCS 7 Project '<OS>\ScriptAct'. It is no longer required.
7. During the Cemat installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Library, PRT and WScipts. These files must now be additionally copied to the equivalent folders of the OS Project.
   With the program 'CematProjectUpdate.exe', which you find under D:\CEMAT_CS, the update can be carried out automatically.

8. Start the application 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project) Default directory is the last closed WinCC project.

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1680x1050, 1920x1080, 1920x1200 or 1600x1200) is important for all OS Projects with Operator Interface. The Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl, @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

9. Open the WinCC project.
10. Open 'Global Script', ‘C_Editor’ and regenerate the header. (Menu/Options/Regenerate Header)
11. Adapt the Computer settings under Startup for the "Additional Tasks/Applications" as follows:
    Change the entry D:\CEM_V6\BIN\Scan_CS.exe into D:\CEMAT_CS\BIN\CEMATRS.exe. Change the working directory from D:\CEM_V6\BIN to D:\CEMAT_CS\BIN.

12. Cemat needs a number of internal variables in the Tag Management. Via WinCC Configuration Studio the variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

    - Open the WinCC Configuration Studio for the Tag Management
    - Use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

    Further information can be found in the Engineering manual, chapter 03_PCS7_Project under Tag Management.

13. During the Update of the OS Project the settings in the Alarm logging have been changed to the default settings of PCS 7 which need to be corrected for Cemat regarding color definitions and acknowledgement/flashing behavior.

    Message blocks and Message configuration can be imported via WinCC configuration Studio for Alarm Logging from already prepared Excel files Import_MessageBlocks.xlsx and Import_MessageConfiguration.xlsx which you find under D:\CEMAT_CS\IMPORT.

    For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

    For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

    Further information you will find in the Engineering Manual 03_PCS7_Project under Alarm logging.

14. During the update procedure the following GraCS files are overwritten by Cemat files. These are:

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

    For Single User Stations and for OS Clients this is correct. For OS Servers the OS Project Editor must be started again, because in this case the standard pictures delivered with PCS 7 have to be used (the Server has no Visualization). Please select the above listed Pictures in order to be overwritten and confirm with OK.

    For all other Station types the files have to be deselected if you run the OS Project Editor again.
    <u>Caution:</u> The files @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must therefore be copied again after running the OS Project Editor.
    (Start the program D:\CEMAT_CS\CematProjectUpdate.exe; the button “Res” copies the CEMAT “resolutions” pictures to the destination again).

15. During the Update of the GraCS Directory the new template picture C_@PCS7Typicals_CemV8_000.pdl has been copied. The template picture is suitable for all CEMAT screen resolutions.
    As all Block Icons have been extended by new functions, we recommend using the new template picture, but you have to add your additionally created/modified block icons.

    (After updating the template picture use Dynamic Wizard *Picture Functions→ Update of the Picture objects* in order to update the block icons of your process pictures.
    Select your template picture name and config file @PCS7Typicals.cfg.)
16. If you have modified for your project system PDL's and scripts you have to do the modification again in the new system files. Refer to your backup.
17. Open the User Administrator.
    Add right 29 (Maintenance).
18. In the Horn configuration, in folder 2 the sound files from directory D:\CEM_V6\Sounds have been selected. Change the directory into D:\CEMAT_CS\Sounds. If you have used additional Sounds you must copy it to the CEMAT directory.
19. The Structure of the User Archive C_INFO has been changed. If you did not save the user archive data for C_INFO yet, please export it now and use the Excel Tool C_INFO.xlsm in order to add the additional columns.
    The Excel tool C_INFO.xlsm you find under D:\CEMAT_CS\TOOLS and the description you find in the Engineering Manual, chapter 09_Engineering_Tools.
20. The Structure of the User Archives C_INFO and C_DriveList has been extended und must therefore be replaced. The new structure can be imported via the WinCC Configuration Studio.
    - Open the WinCC Configuration Studio for the User Archive and delete the Archives C_INFO and C_DriveList.
    - In order to import the new structure use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_UserArchives.xlsx and press Button *Import.*
21. After this you have to import the C_INFO data again.

Procedure, if PTE Controllers are used

Because of the early conversion of Process pictures by the OS migrator it can happen, that some PDL files are converted wrong. E. g. if OCX controls for PTE Objects are used in the Pictures and the OCX was not yet registered. The failure is: The OCX controls are not present.
Workaround: After the migration described above you have to copy the process pictures from the backup in the V6 version again the GraCS directory. In the WinCC explorer start the conversion of the die PDL files again. After that the PTE-Controllers are integrated correctly.

If you like to use still the PTE controllers for existing applications, you have to take care, that all the old faceplates, OCX files are installed in your project and on your PCs.

Compile and Download

1. For all PLCs: Compile CFC (complete program).
2. For all Single Stations and primary Servers: OS Compile (complete OS with memory reset). In the background the Server data (package) will be automatically created and loaded into the OS Client Project. Please double-check!
3. Load the complete program into the PLC.
4. Actualize the Server and OS Clients with Project Download.

Update CEMAT V7.0 to CEMAT V8.2

You have a CEMAT V7.0 project (based on PCS 7 V7.0) and want to upgrade to CEMAT V8.2 (based on PCS 7 V8.2). The following description explains the necessary steps.

Requirements

Before the installation of Cemat V8.2, you have to upgrade your PC Stations to the permitted Windows Operating System and to PCS 7 V8.2.

The following Windows Operating Systems are supported by PCS 7 V8.2, but not all versions are compatible with each Configuration.

- Windows 7 Professional SP1 (64-Bit, only English Windows Language Version)
  (permitted for ES, non-red. OS-Single Station, OS-Client, Web-Client)
- Windows 7 Ultimate/Enterprise SP1 (32-Bit)
  (permitted for OS-Client and Web-Client)
- Windows 7 Ultimate / Enterprise SP1 (64-Bit)
  (permitted for ES, OS-Single Station, OS-Client, Web-Server (only on ES or OS-Single) and Web-Client)
- Windows 10 Enterprise 2015 LTSB (64-Bit)
  (permitted for OS-Client and Web-Client)
- Windows Server 2008 R2 SP1 Standard Edition (64-Bit)
  (permitted for ES, OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client)
- Windows Server 2012 R2 Update Standard Edition (64-Bit)
  (permitted for OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client

Further details about the installation of the operating system and PCS 7 are explained in the PCS 7 manuals and in the readme and therefore not part of this description.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2
Make sure that the PCS 7 updates are installed before updating to CEMT V8.2!

Caution: The update to CEMAT V8.2 requires a complete reloading of the PLC and consequently **cannot** be carried out while the plant is running.

Caution: For the migration of a PCS 7 Project V6.0 / V6.1 /V7.0 you need additionally an engineering station with PCS 7 V7.1 SP3 or PCS 7 V7.1 SP4. Only in this version you have an OS migrator, which is able to convert all older versions with a MS SQL database to the version of V7.1 SP3 (or V7.1 SP4).
The OS Migrator of PCS 7 V8.2 only converts Projects with Version >= V7.1 SP3 to Version V8.2. WinCC converts the Project automatically the first time you open the OS Project.
Please check the PCS 7 manual "Software Update with utilization of the new functions”.

Follow the actual Manual "PCS 7 Software Update with utilization of the new functions" from the PCS 7 V8.2 Documentation. The following description refers to this manual.
Please also pay attention to the hints for the OS-Migration. In order to maintain the runtime data (curves and messages), the Server Projects have to be migrated on the Server-PCs directly.

Consider also the project rules from the CEMAT manual 03_PCS7_Project_009.pdf, “Installation of a PCS 7 Project” and the Cemat Object descriptions.

Save Project data

Save your project data. Beside the Multiproject this comprises all settings which you have carried out in the CEMAT_CS Directory, as well as modifications in the block library ILS_CEM or PRO_CEM.

- Archive the Multiproject with the SIMATIC Manager
- Export the Data of the User Archive C_INFO if you want to continue to use it after the upgrade.
- You can copy the modifications under CEMAT_CS (Config, Multimedia, Sounds) to a backup-directory.

Caution: Never save directory „CEMAT_CS/BIN“!!!
The files from the BIN directory are entered in the registry and must never be moved or deleted.

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

Please consider that the blocks C_SINA and C_ROBI (AddOn) are completely new designed. The existing blocks must not be used anymore. <u>They must be removed and replaced by a C_DRV_1D or C_DRV_2D block plus the new adapter block C_SINA or C_ROBI.</u>

Migration to PCS 7 V7.1 SP3

Open your project on a PCS 7 V7.1 SP3 Engineering Station.

Migration of the CFC Charts

Open any CFC chart and slightly move a block. A dialog box appears, asking you for conversion of the CFC PCS 7 V7.1 SP3. Acknowledge this dialog that appears with "OK". The Charts will be converted to V7.1 SP3.
Repeat this step for all chart folders of your Multiproject.

The block container still has the old S7 blocks. Block exchange is carried out later with the migration to V8.2.

OS-Migration

Start the OS Migrator and choose the MCP file of the WinCC project. Do this for all WinCC projects in your multi project.

Archive the multi project with the SIMATIC Manager and copy it to the V8.2 Engineering Station.

Migration to PCS 7 V8.2

OS-Migration

Again refer to the OS-Migration in the Manual "PCS 7 - Software Update with utilization of the new functions".

You have to migrate all OS Projects in your Multiproject. This is carried out under:
Start → SIMATIC → WinCC → Tools → Project Migrator.

Plant View

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

Caution: If your picture tree was nevertheless generated manually you have change it. Now you have to generate the picture tree from the plant view

---

Installation of CEMAT V8.2

With the following steps please proceed on a Station with PCS 7 V8.2!

Start the Setup program from the DVD CEMAT V8.2 and follow the instructions.

The Installation path is 'D:\CEMAT_CS'. This cannot be changed!

Keys for project standards

000 = CEMAT
001 = Minerals Cemat
004 = Holcim
006 = Dyckerhoff
007 = Heidelberg Cement
023 = Vigier
024 = Bushehr
025 = Caima
026 = Alsen
027 = Lafarge
028 = Rossi

After the installation, for all PC stations (ES, OS SINGLE STATION, OS SERVER and OS CLIENT) the following steps have to be carried out:

1. The config files contain the default settings, which means if you changed them for your project you have to make this adaptations again. (C_Config.cfg)
   Also consider the settings in file C_Messages.cfg.
2. Probably you also added for your project some files under 'Multimedia', 'Sounds' ….
   After the CEMAT installation these files can be restored from the backup directories and copied into the new additional CEMAT_CS\MULTIMEDIA directories.
3. If you want to use the PTE-Controller Function, then you have to install the PTE-Controller OS part (OCX files, PDLs, scripts, CFG files) before you migrate your PCS 7 Project. Otherwise the standard PDLs (@PG_C_??????.PDL) will be wrong converted and cannot be used any more.
   Copy all necessary files for the PTE controller function to D:\CEMAT_CS\ into the corresponding directories. Register the OCX files using the BAT file. Refer to the PTE-Controller instructions.
   The use of the old PTE controllers you do in your own responsibility.

Update of S7 Program

Please also consider the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“.

Display languages

Caution: In case of using Spanish language, you have to change the language of all partial projects and libraries into "Spanish (Spain, international sort)". Special caution is needed if "Spanish (Spain, traditional sort)" was used as default language. Make sure that you don't lose any texts.

New CEMAT Libraries

With the installation of CEMAT V8.2 a new version of the CMEAT Library ILS_CEM has been installed in directory C:\Program Files\SIEMENS\STEP7\S7libs.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (project key other than “000”) the library PRO_CEM is also transferred into C:\Program Files\SIEMENS\STEP7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Program folder **Cemat_AddOn** contains Open blocks as e. g. C_SIMOS, C_SINA, C_ROBI. They are delivered in a separate folder in order not to overwrite the existing program during update procedure.

In order to avoid overlapping with other libraries, in CEMAT V8.2 some AS blocks of the ILS_CEM library have been transferred to new addresses (see list below).
The transfer concerns all CEMAT Project Standards with exception of Project Standard 001 (In the Minerals Automation Standard the new block numbers have been used from the beginning).

Block name	**old Number**	**→ new Number**
C_ODA	FB1060	→ FB1046
C_ODA_R	FB1062	→ FB1047
C_INTERL	FB1075	→ FB1057
C_INTER5	FB1076	→ FB1058
C_RelMod	FB1077	→ FB1059
AG_LSEND	FC1050	→ FC1033
AG_LRECV	FC1060	→ FC1034
FC_GALA-WALA	FC1061	→ FC1021
FC_GRUZU_WEGZU	FC1062	→ FC1022
FC_GRINZ	FC1063	→ FC1023
StruInIn	FC1065	→ FC1025
StruInOu	FC1066	→ FC1026
C_SelQC	FC1067	→ FC1027
C_PUSHBT	FC1088	→ FC1028
OB1_SYS1	FC1102	→ FC1031
C_OB1SY1	FC1103	→ FC1032

Furthermore in the ILS_CEM all non-used CEMAT blocks have been removed. These are:

FC6	DT_DATE
FC509	BCD_SKCNT_CHECK
FC510	COPY_ELEM
FC516	FC_AUTO_APPEND
FC517	FC_GET_STRUCT_LEN
FC519	VAR_TO_DBNR
FC520	VAR_TO_BYNR
FC521	DIAGNOSE
FC522	BYTE_TO_INT
FC523	INT_TO_BYTE
FC524	BCD_TO_BYTE
FC525	BLOCK_DB_TO_INT
FC1020	C_SIMU_L
FC1064	FC_SIMU_LIST
DB676	C_SIMU_DB
DB679	DB_FATAL_ERROR
DB681	DB_DIAGNOSTIC
DB682	DB_EVENT
UDT1016	DIAGNOSE_ENTRY

New PCS 7 Libraries

With the installation of PCS 7 V8.2 also some new block libraries were installed:
- PCS 7 Basis Library V82
- PCS 7 AP Library V82.

A PCS 7 Library V82 doesn't exist!!!!!!!!
If you don't like to use the new APL blocks and you like still to use some blocks of the Basis Library and PCS 7 Library V71, then you have to install the newest version of the V7.1 libraries. You find the setup for the V7.1 libraries on the on the PCS 7 V8.2 Installation DVD under "Additional Products".

The newest PCS 7 Library V7.1 SP3 contains 2 versions of some channel driver blocks (e. g. CH_AI). The 2 blocks are identical, but they have different version numbers (V6.0 or V7.0).
The channel / driver blocks of the V7.1 have the version number V6.0.
The channel / driver blocks of the V8.0 have the version number V7.0.
A mix of the version number of the used channel / driver blocks is not allowed!!!!

Update Procedure for the S7-Program

For the update of your S7-Programs please proceed as follows:

- First update the Master data Library
- Then update the S7-Program containers for each AS
- Update the CFC for each AS

Actualize the Master data Library

The Master data library must contain all blocks which are used in the different S7-Projects of the Multiproject. These are the CEMAT blocks as well as block from other PCS 7 libraries which are used in the Multiproject

1. Open the block folder of the Master data library.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Open the block folder CEM_ALL of the CEMAT library ILS_CEM.
4. Select all blocks Copy all blocks from CEM_ALL and copy these into the block folder of your Master data library.
5. If a CEMAT Project Standard was installed, open library PRO_CEM and copy all blocks (overwrite = yes) into the block folder of your Master data library.
6. Your Master data library should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks cannot be used any more, but do not replace it by the new Adapter block now; otherwise you will lose your connections during the update.
7. If in your project system attributes or message attributes in the standard blocks have been modified, you have to make the adaptations after the update of the block folder manually again.
8. At this time you should also check,
   - if all other blocks (e. g. with Author BASISxx, TECHNxx or DRIVERxx, etc) are actualized.
   - that the Master data library does not contain any block with Author ES_MAP
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!)..
9. Open the CFC editor by opening any chart.
10. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
11. In the System Chart SYSPLC00 add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.

After this the Actualization of the Master data library is completed.

Actualization of the S7-Program for each AS

For each AS you have to actualize the **S7-Program container** from the Master data library:

1. Open the block folder of your AS.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Then copy all Blocks from your Master data library into the block container of the AS.
4. Your AS block folder should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks must be manually replaced after the CFC update by a C_DRV_1D or C_DRV_2D plus the new adapter block C_SINA or C_ROBI. Automatic replacement of C_SINA or C_ROBI is not possible, existing connections get lost!
5. If required, actualize further blocks of the S7-Program (e.g. according to the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“, from 01/2013, chapter 7.5.15).

Actualization of the CFC Charts

Through the change of existing parameters to structure format, during the update procedure some connections will be converted to Textual connections.
Some connections may get lost and must be engineered again. Using the Excel Tool C_MIG_V80.XLS the existing connections can be rewired.
For this purpose you have to make some preparations before the actualization of the CFCs:

Carry out the following Steps for each AS:

1. In the Process Object view, select the complete AS and actualize the data with F5. Convert the CFCs at the same time into the format of PCS 7 V8.2. After this export the data as follows:
2. Carry out the function *Process Objects → Export all IOs…* As file name choose e. g. AS1_PS_old.csv.
3. Carry out the function *Process Objects → Export Blocks …* As file name choose e. g. AS1_B_csv.

For each AS, carry out the following steps in order to actualize the CFC Charts:

4. Open the CFC editor by opening any chart.
5. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
6. Please check the log file after the CFC update:
   The LOG file must be saved, e. g. with name AS1_Textual Interconnections.txt!
   - Connections to Parameters which could not been found in the new version (e. g. for C_SIMDMP) have been removed or converted to textual connections.
   - Please also check for textual connections, using the function *Options → Close Textual interconnections*.
   - After closing the textual interconnections the list should be empty. Otherwise you need to correct the program manually!
7. For finding the removed connections the Excel Tool C_MIG_V80.XLS may be used. The Tool compares the program BEFORE and AFTER the block update and creates a list of differences. Some connections can also be corrected automatically with the tool.

   If you want to use the tool, select in the process object view the complete AS and actualize the data with F5. Again carry out the function *Process Objects → Export all IOs…*
   As file name choose e. g. AS1_PS_V82.csv.

   The description for the Excel Macro you find in an extra chapter.
   Refer to "CEMAT Tools C_MIG_V80.XLS and C_SIM_RIGHT.XLS" at the end of this document.

   The Excel tool generates an Import file which you can import in the process object view, using the function *Process Objects → Import IOs…*
8. The user rights for Simulation of C_ANNUNC, C_ANNUN8, C_MEASUR and C_PROFB could be modified at the block icon before and now this is done via block parameter SimRight.
   The Excel Tool C_SIM_RIGHT.XLS evaluates the settings at the block icons and allows importing this into the CFC.
   If you did not change the attribute "HigherProcesscontrolling" on the block icons, i.e. the attribute has the value 24, then skip this part and proceed with the next point.
   In case of a modification, use Excel Macro C_SIM_RIGHT.XLS, which is explained in a separate chapter.
   Refer to "CEMAT Tools C_MIG_V80.XLS and C_SIM_RIGHT.XLS" at the end of this document.
9. Through the change of the variable format (e. g. from REAL to STRUCT) it may happen that a parameter which was switched to visible before is invisible in the CFC after the

upgrade. If an interconnection exists for an invisible parameter you can see this at the blue triangle in the header of the CFC block. If required you have to switch it to visible again.

10. In the new version the size of the blocks may be different from the version before. This may lead to overlapping in the CFC. Eventually it is necessary to move some of the blocks in the CFC.
11. **Only Holcim**: Add a new OR gate with the dynamic faults (EST, KST, VST) of all drives which are involved in the material transport and connected it to FLT_MAT of the C_GROUP.
    The output N_EMPTY of the C_GROUP module is no longer working like before.
    Function until V7.1: All dynamic faults of the related objects set the output N_EMPTY. A normal stop reset the output N_EMPTY, even if there is still material on a belt.
    New function since V7.1 SP1: The new input FLT_MAT set the output N_EMPTY. A normal stop doesn't reset the output N_EMPTY. N_EMPTY must be reset by the Operator after the conveying line is cleared.
    To get this new function, you need an OR gate with the dynamic faults (EST, KST, VST) of all drives which are involved in the material transport. This new OR gate have to be connected to the new input FLT_MAT.
12. Because of extended possibilities for the quality evaluation the rules for connecting Cemat Block and Driver Block have been changed.
    If you use module driver together with C_ANNUNC or C_MEASUR please check the connection between the blocks (see AS_Engineering.pdf).
13. At this time you should also check,
    - if all blocks (e. g. with Author BASISxx, TECHNxx, CHNxx or DRIVERxx, etc) are actualized.
    - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
14. Adaptations for C_SINA and C_ROBI
    You have to replace the blocks by a C_DRV_1D/C_DRV_2D block + the new Adapter bocks C_SINA or C_ROBI. New Engineering required!
15. In the System Chart SYSPLCxx add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.
    Especially for Holcim plants: please check parameter REL_WSTP of block C_FB_PLC. REL_WSTP must be “0” in order not to interrupt the group start in case of a warning!
16. In folder D:\CEMAT_CS\TOOLS\CheckTool you find the new CEMAT Check tool “CematCheckTool.exe”. Please make use of this tool after upgrading your project.
    The check tool will tell you if your project structure is correct; e. g. if the CEMAT blocks are inserted at the correct installation position or if the Group links and C_MUX interconnections are correct.
    For details refer the CEMAT manual „10_Checkliste_009.pdf“.

After this block update a complete compile of all charts is required.

Carry out the same steps for all S7 Program folders of your Multiproject.

Update of the OS-Projects

Before the Cemat Update all OS Projects must be migrated to PCS 7 V8.2.

CEMAT Update

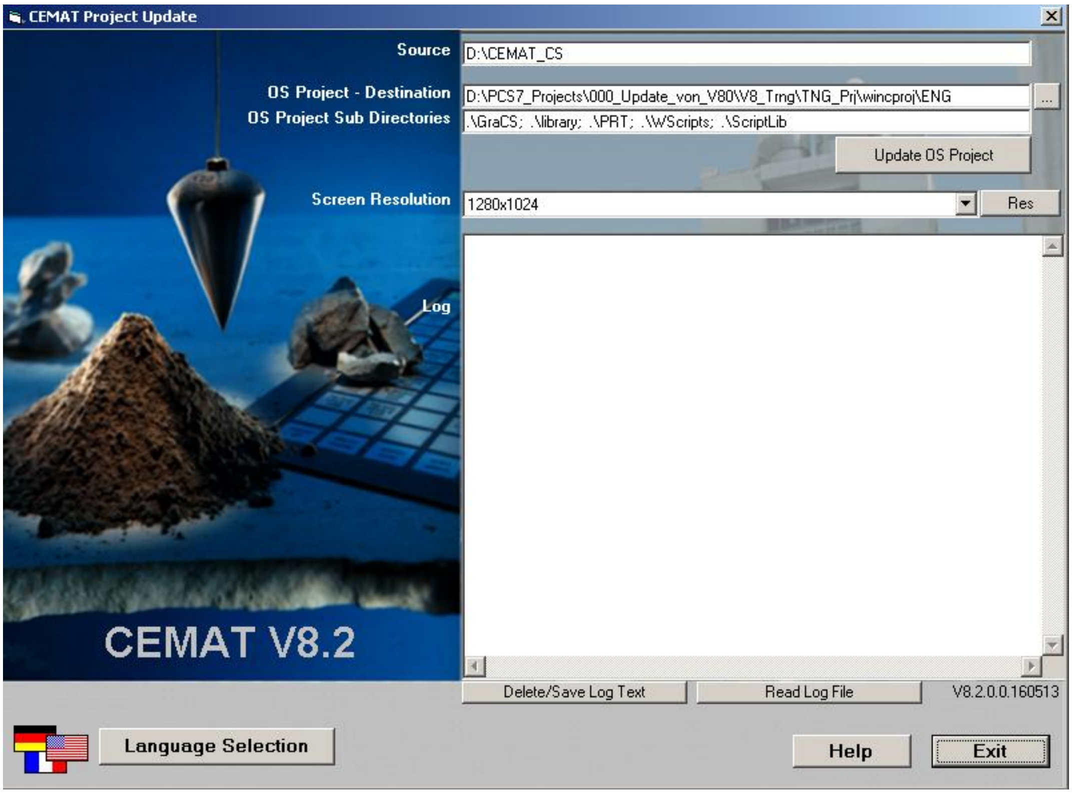
After the CEMAT installation, the D:\CEMAT_CS\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

In all WinCC Projects carry out the following steps:

1. Open the WinCC Explorer.
2. Change the WinCC Design from 'Classic' to '3D':
   With right mouse click on the OS Project, select the Project Properties. Open Tag "User Interface and Design" and switch the Active Design from 'WinCC Classic' to WinCC '3D'.
3. Open the OS Project Editor. Check the settings according to the Engineering Manual Chapter 3 "Installation of a PCS 7 Project".
   Close the OS Project Editor with "OK". All standard pictures (Overview, buttons, alarm line, etc.) are replaced by the new standard PDLs of PCS 7 V8.2.
4. Close WinCC Explorer.
5. During the Cemat installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Library, PRT and WScipts. These files must now be additionally copied to the equivalent folders of the OS Project.
   With the program 'CematProjectUpdate.exe', which you find under D:\CEMAT_CS, the update can be carried out automatically.

6. Start the application 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project) Default directory is the last closed WinCC project.

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1680x1050, 1920x1080, 1920x1200 or 1600x1200) is important for all OS Projects with Operator Interface. The Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl, @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

7. Open the WinCC project.
8. Open 'Global Script', ‘C_Editor’ and regenerate the header. (Menu/Options/Regenerate Header)
9. Adapt the Computer settings under Startup for the "Additional Tasks/Applications" as follows:
   Change the entry D:\CEMAT_CS\BIN\Scan_CS.exe into D:\CEMAT_CS\BIN\CEMATRS.exe. The working directory remains D\CEMAT_CS\BIN.

10. Cemat needs a number of internal variables in the Tag Management. Via WinCC Configuration Studio the variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

    - Open the WinCC Configuration Studio for the Tag Management
    - Use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

    Further information can be found in the Engineering manual, chapter 03_PCS7_Project under Tag Management.

11. During the Update of the OS Project the settings in the Alarm logging have been changed to the default settings of PCS 7 which need to be corrected for Cemat regarding color definitions and acknowledgement/flashing behavior.

    Message blocks and Message configuration can be imported via WinCC configuration Studio for Alarm Logging from already prepared Excel files Import_MessageBlocks.xlsx and Import_MessageConfiguration.xlsx which you find under D:\CEMAT_CS\IMPORT.

    For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

    For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

    Further information you will find in the Engineering Manual 03_PCS7_Project under Alarm logging.

12. During the update procedure the following GraCS files are overwritten by Cemat files. These are:

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

For Single User Stations and for OS Clients this is correct. For OS Servers the OS Project Editor must be started again, because in this case the standard pictures delivered with PCS 7 have to be used (the Server has no Visualization). Please select the above listed Pictures in order to be overwritten and confirm with OK.

For all other Station types the files have to be deselected if you run the OS Project Editor again.
Caution: The files @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must therefore be copied again after running the OS Project Editor.
(Start the program D:\CEMAT_CS\CematProjectUpdate.exe; the button “Res” copies the CEMAT “resolutions” pictures to the destination again).

13. During the Update of the GraCS Directory the new template picture C_@PCS7Typicals_CemV8_000.pdl has been copied. The template picture is suitable for all CEMAT screen resolutions.
    As all Block Icons have been extended by new functions, we recommend using the new template picture, but you have to add your additionally created/modified block icons.

    (After updating the template picture use Dynamic Wizard *Picture → Update of the Picture objects* in order to update the block icons of your process pictures.
    Select your template picture name and config file @PCS7Typicals.cfg.)
14. If you have modified for your project system PDL's and scripts you have to do the modification again in the new system files. Refer to your backup.
15. The Structure of the User Archive C_DriveList has been extended und must be replaced. The new structure can be imported via the WinCC Configuration Studio-
    - Open the WinCC Configuration Studio for the User Archive and delete Archive C_DriveList.
    - In order to import the new structure use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_UserArchives.xlsx and press Button *Import.*

Procedure, if PTE Controllers are used

Because of the early conversion of Process pictures by the OS migrator it can happen, that some PDL files are converted wrong. E. g. if OCX controls for PTE Objects are used in the Pictures and the OCX was not yet registered. The failure is: The OCX controls are not present.
Workaround: After the migration described above you have to copy the process pictures from the backup in the V7 version again the GraCS directory. In the WinCC explorer start the conversion of the die PDL files again. After that the PTE-Controllers are integrated correctly.

If you like to use still the PTE controllers for existing applications, you have to take care, that all the old faceplates, OCX files are installed in your project and on your PCs.

Compile and Download

1. For all PLCs: Compile CFC (complete program).
2. For all Single Stations and primary Servers: OS Compile (complete OS with memory reset). In the background the Server data (package) will be automatically created and loaded into the OS Client Project. Please double-check!
3. Load the complete program into the PLC.
4. Actualize the Server and OS Clients with Project Download.

Update CEMAT V7.1 to CEMAT V8.2

You have a CEMAT V7.1 project (with or without SP1) and want to upgrade to CEMAT V8.2 (based on PCS 7 V8.2). The following description explains the necessary steps.

Requirements

Before the installation of Cemat V8.2, you have to upgrade your PC Stations to the permitted Windows Operating System and to PCS 7 V8.2.

The following Windows Operating Systems are supported by PCS 7 V8.2, but not all versions are compatible with each Configuration.

- Windows 7 Professional SP1 (64-Bit, only English Windows Language Version) (permitted for ES, non-red. OS-Single Station, OS-Client, Web-Client)
- Windows 7 Ultimate/Enterprise SP1 (32-Bit) (permitted for OS-Client and Web-Client)
- Windows 7 Ultimate / Enterprise SP1 (64-Bit) (permitted for ES, OS-Single Station, OS-Client, Web-Server (only on ES or OS-Single) and Web-Client)
- Windows 10 Enterprise 2015 LTSB (64-Bit) (permitted for OS-Client and Web-Client)
- Windows Server 2008 R2 SP1 Standard Edition (64-Bit) (permitted for ES, OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client)
- Windows Server 2012 R2 Update Standard Edition (64-Bit) (permitted for OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client

Further details about the installation of the operating system and PCS 7 are explained in the PCS 7 manuals and in the readme and therefore not part of this description.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2
Make sure that the PCS 7 updates are installed before updating to CEMT V8.2!

Caution: The update to CEMAT V8.2 requires a complete reloading of the PLC and consequently **cannot** be carried out while the plant is running.

Caution: For the migration of a PCS 7 Project V.1 (without SP3 or SP4) you need additionally an engineering station with PCS 7 V7.1 SP3 or PCS 7 V7.1 SP4. Only in this version you have an OS migrator, which is able to convert all older versions with a MS SQL database to the version of V7.1 SP3 (or V7.1 SP4).
The OS Migrator of PCS 7 V8.2 only converts Projects with Version >= V7.1 SP3 to Version V8.2. WinCC converts the Project automatically the first time you open the OS Project. Please check the PCS 7 manual "Software Update with utilization of the new functions”.

Follow the actual Manual "PCS 7 Software Update with utilization of the new functions" from the PCS 7 V8.2 Documentation. The following description refers to this manual.
Please also pay attention to the hints for the OS-Migration. In order to maintain the runtime data (curves and messages), the Server Projects have to be migrated on the Server-PCs directly.

Consider also the project rules from the CEMAT manual 03_PCS7_Project_009.pdf, “Installation of a PCS 7 Project” and the Cemat Object descriptions.

Save Project data

Save your project data. Beside the Multi Project this comprises all settings which you have carried out in the CEMAT_CS Directory, as well as modifications in the block library ILS_CEM or PRO_CEM.

- Archive the Multi Project with the SIMATIC Manager
- Export the Data of the User Archive C_INFO if you want to continue to use it after the upgrade.
- You can copy the modifications under CEMAT_CS (Config, Multimedia, Sounds) to a backup-directory.

Caution: Never save directory ”CEMAT_CS/BIN“!!!
The files from the BIN directory are entered in the registry and must never be moved or deleted.

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

Please consider that the blocks C_SINA and C_ROBI (AddOn) are completely new designed. The existing blocks must not be used anymore. They must be removed and replaced by a C_DRV_1D or C_DRV_2D block plus the new adapter block C_SINA or C_ROBI.

Migration to PCS 7 V7.1 SP3

If your project is not yet migrated to PCS 7 V7.1 SP3, open your project on a PCS 7 V7.1 SP3 Engineering Station.

Migration of the CFC Charts

Open any CFC chart and slightly move a block. A dialog box appears, asking you for conversion of the CFC PCS 7 V7.1 SP3. Acknowledge this dialog that appears with "OK". The Charts will be converted to V7.1 SP3.
Repeat this step for all chart folders of your Multiproject.

The block container still has the old S7 blocks. Block exchange is carried out later with the migration to V8.2.

OS-Migration

Start the OS Migrator and choose the MCP file of the WinCC project. Do this for all WinCC projects in your multi project.

Archive the multi project with the SIMATIC Manager and copy it to the V8.2 Engineering Station.

Migration to PCS 7 V8.2

OS-Migration

Again refer to the OS-Migration in the Manual "PCS 7 - Software Update with utilization of the new functions".

Maybe you have to migrate all OS Projects in your Multiproject. This is carried out under:: Start → SIMATIC → WinCC → Tools → Project Migrator.

Plant View

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

Caution: If your picture tree was nevertheless generated manually you have change it. Now you have to generate the picture tree from the plant view

Installation of CEMAT V8.2

With the following steps please proceed on a Station with PCS 7 V8.2!

Start the Setup program from the DVD CEMAT V8.2 and follow the instructions.

The Installation path is 'D:\CEMAT_CS'. This cannot be changed!

Keys for project standards

000 = CEMAT
001 = Minerals Cemat
004 = Holcim
006 = Dyckerhoff
007 = Heidelberg Cement
023 = Vigier
024 = Bushehr
025 = Caima
026 = Alsen
027 = Lafarge
028 = Rossi

After the installation, for all PC stations (ES, OS SINGLE STATION, OS SERVER and OS CLIENT) the following steps have to be carried out:

1. The config files contain the default settings, which means if you changed them for your project you have to make this adaptations again. (C_Config.cfg)
   Also consider the settings in file C_Messages.cfg.
2. Probably you also added for your project some files under 'Multimedia', 'Sounds' ….
   After the CEMAT installation these files can be restored from the backup directories and copied into the new additional CEMAT_CS\MULTIMEDIA directories.
3. If you want to use the PTE-Controller Function, then you have to install the PTE-Controller OS part (OCX files, PDLs, scripts, CFG files) before you migrate your PCS 7 Project. Otherwise the standard PDLs (@PG_C_??????.PDL) will be wrong converted and cannot be used any more.
   Copy all necessary files for the PTE controller function to D:\CEMAT_CS\ into the corresponding directories. Register the OCX files using the BAT file. Refer to the PTE-Controller instructions.
   The use of the old PTE controllers you do in your own responsibility.

Update of S7 Program

Please also consider the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“.

Display languages

Caution: In case of using Spanish language, you have to change the language of all partial projects and libraries into "Spanish (Spain, international sort)". Special caution is needed if "Spanish (Spain, traditional sort)" was used as default language. Make sure that you don't loose any texts.

New CEMAT Libraries

With the installation of CEMAT V8.2 a new version of the CMEAT Library ILS_CEM has been installed in directory C:\Program Files\SIEMENS\STEP7\S7libs.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (project key other than “000”) the library PRO_CEM is also transferred into C:\Program Files\SIEMENS\STEP7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Program folder **Cemat_AddOn** contains Open blocks as e. g. C_SIMOS, C_SINA, C_ROBI. They are delivered in a separate folder in order not to overwrite the existing program during update procedure.

In order to avoid overlapping with other libraries, in CEMAT V8.2 some AS blocks of the ILS_CEM library have been transferred to new addresses (see list below).
The transfer concerns all CEMAT Project Standards with exception of Project Standard 001 (In the Minerals Automation Standard the new block numbers have been used from the beginning).

Block name	old Number	→ new Number
C_ODA	FB1060	→ FB1046
C_ODA_R	FB1062	→ FB1047
C_INTERL	FB1075	→ FB1057
C_INTER5	FB1076	→ FB1058
C_RelMod	FB1077	→ FB1059
AG_LSEND	FC1050	→ FC1033
AG_LRECV	FC1060	→ FC1034
FC_GALA-WALA	FC1061	→ FC1021
FC_GRUZU_WEGZU	FC1062	→ FC1022
FC_GRINZ	FC1063	→ FC1023
StruInIn	FC1065	→ FC1025
StruInOu	FC1066	→ FC1026
C_SelQC	FC1067	→ FC1027
C_PUSHBT	FC1088	→ FC1028
OB1_SYS1	FC1102	→ FC1031
C_OB1SY1	FC1103	→ FC1032

Furthermore in the ILS_CEM all non-used CEMAT blocks have been removed. These are:

FC6	DT_DATE
FC509	BCD_SKCNT_CHECK
FC510	COPY_ELEM
FC516	FC_AUTO_APPEND
FC517	FC_GET_STRUCT_LEN
FC519	VAR_TO_DBNR
FC520	VAR_TO_BYNR
FC521	DIAGNOSE
FC522	BYTE_TO_INT
FC523	INT_TO_BYTE
FC524	BCD_TO_BYTE
FC525	BLOCK_DB_TO_INT
FC1020	C_SIMU_L
FC1064	FC_SIMU_LIST
DB676	C_SIMU_DB
DB679	DB_FATAL_ERROR
DB681	DB_DIAGNOSTIC
DB682	DB_EVENT
UDT1016	DIAGNOSE_ENTRY

New PCS 7 Libraries

With the installation of PCS 7 V8.2 also some new block libraries were installed:
- PCS 7 Basis Library V82
- PCS 7 AP Library V82.

A PCS 7 Library V82 doesn't exist!!!!!!!!
If you don't like to use the new APL blocks and you like still to use some blocks of the Basis Library and PCS 7 Library V71, then you have to install the newest version of the V7.1 libraries. You find the setup for the V7.1 libraries on the on the PCS 7 V8.2 Installation DVD under "Additional Products".

The newest PCS 7 Library V7.1 SP3 contains 2 versions of some channel driver blocks (e. g. CH_AI). The 2 blocks are identical, but they have different version numbers (V6.0 or V7.0).
The channel / driver blocks of the V7.1 have the version number V6.0.
The channel / driver blocks of the V8.0 have the version number V7.0.
A mix of the version number of the used channel / driver blocks is not allowed!!!!

Update Procedure for the S7-Program

For the update of your S7-Programs please proceed as follows:

- First update the Master data Library
- Then update the S7-Program containers for each AS
- Update the CFC for each AS

Actualize the Master data Library

The Master data library must contain all blocks which are used in the different S7-Projects of the Multiproject. These are the CEMAT blocks as well as block from other PCS 7 libraries which are used in the Multiproject

1. Open the block folder of the Master data library.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Open the block folder CEM_ALL of the CEMAT library ILS_CEM.
4. Select all blocks Copy all blocks from CEM_ALL and copy these into the block folder of your Master data library.
5. If a CEMAT Project Standard was installed, open library PRO_CEM and copy all blocks (overwrite = yes) into the block folder of your Master data library.
6. Your Master data library should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks cannot be used any more, but do not replace it by the new Adapter block now; otherwise you will lose your connections during the update.
7. If in your project system attributes or message attributes in the standard blocks have been modified, you have to make the adaptations after the update of the block folder manually again.
8. At this time you should also check,
   - if all other blocks (e. g. with Author BASISxx, TECHNxx or DRIVERxx, etc) are actualized.
   - that the Master data library does not contain any block with Author ES_MAP
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!)..
9. Open the CFC editor by opening any chart.
10. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
11. In the System Chart SYSPLC00 add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.

After this the Actualization of the Master data library is completed.

Actualization of the S7-Program for each AS

For each AS you have to actualize the **S7-Program container** from the Master data library:

1. Open the block folder of your AS.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Then copy all Blocks from your Master data library into the block container of the AS.
4. Your AS block folder should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks must be manually replaced after the CFC update by a C_DRV_1D or C_DRV_2D plus the new adapter block C_SINA or C_ROBI. Automatic replacement of C_SINA or C_ROBI is not possible, existing connections get lost!

5. If required, actualize further blocks of the S7-Program (e.g. according to the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“, from 01/2013, chapter 7.5.15).

Actualization of the CFC Charts

For each AS, carry out the following steps in order to actualize the CFC Charts:

1. Open the CFC editor by opening any chart.
2. Open *Reference data → Block types* and search for C_SIMU_L blocks. This function no longer exists and the blocks must be deleted.
3. Also check if C_SIMDMP (damper simulation) is has been used. During the update this block is replaced by a block with different parameter names and connections have to be rewired after the update.
4. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
5. Please check the log file after the update:
   - Connections to Parameters which could not been found in the new version (e. g. for C_SIMDMP) have been removed or converted to textual connections.
   - Please also check for textual connections, using the function *Options → Close Textual interconnections*.
   - After closing the textual interconnections the list should be empty. Otherwise you need to correct the program manually!
6. The user rights for Simulation of C_ANNUNC, C_ANNUN8, C_MEASUR and C_PROFB could be modified at the block icon before and now this is done via block parameter SimRight.
   The Excel Tool C_SIM_RIGHT.XLS evaluates the settings at the block icons and allows importing this into the CFC.
   If you did not change the attribute "HigherProcesscontrolling" on the block icons, i.e. the attribute has the value 24, then skip this part and proceed with the next point.
   In case of a modification, use Excel Macro C_SIM_RIGHT.XLS, which is explained in a separate chapter.
   Refer to “CEMAT Tools C_MIG_V80.XLS and C_SIM_RIGHT.XLS” at the end of this document.
7. In the new version the size of the blocks may be different from the version before. This may lead to overlapping in the CFC. Eventually it is necessary to move some of the blocks in the CFC.
8. Because of extended possibilities for the quality evaluation the rules for connecting Cemat Block and Driver Block have been changed.
   If you use module driver together with C_ANNUNC or C_MEASUR please check the connection between the blocks (see AS_Engineering.pdf).
9. At this time you should also check,
   - if all blocks (e. g. with Author BASISxx, TECHNxx, CHNxx or DRIVERxx, etc) are actualized.
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
10. Adaptations for C_SINA and C_ROBI
    You have to replace the blocks by a C_DRV_1D/C_DRV_2D block + the new Adapter bocks C_SINA or C_ROBI. New Engineering required!
11. In the System Chart SYSPLCxx add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.
    Especially for Holcim plants: please check parameter REL_WSTP of block C_FB_PLC. REL_WSTP must be “0” in order not to interrupt the group start in case of a warning!

12. In folder D:\CEMAT_CS\TOOLS\CheckTool you find the new CEMAT Check tool “CematCheckTool.exe”. Please make use of this tool after upgrading your project.
    The check tool will tell you if your project structure is correct; e. g. if the CEMAT blocks are inserted at the correct installation position or if the Group links and C_MUX interconnections are correct.
    For details refer the CEMAT manual „10_Checkliste_009.pdf“.

After this block update a complete compile of all charts is required.

Carry out the same steps for all S7 Program folders of your Multiproject.

Update of the OS-Projects

Before the Cemat Update all OS Projects must be migrated to PCS 7 V8.2.

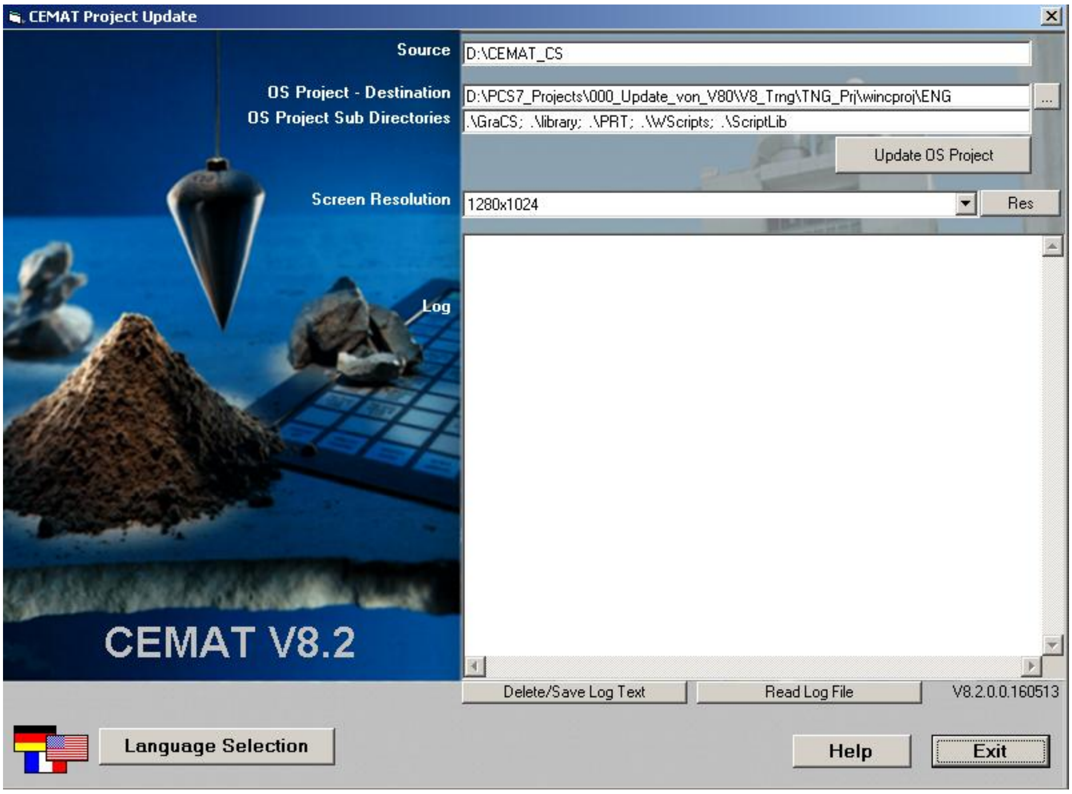
CEMAT Update

After the CEMAT installation, the D:\CEMAT_CS\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

In all WinCC Projects carry out the following steps:

1. Open the WinCC Explorer.
2. Open the OS Project Editor. Check the settings according to the Engineering Manual Chapter 3 "Installation of a PCS 7 Project".
   Close the OS Project Editor with "OK". All standard pictures (Overview, buttons, alarm line, etc.) are replaced by the new standard PDLs of PCS 7 V8.2.
3. Close WinCC Explorer.
4. During the Cemat V8.2 installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Library, PRT and WScipts. These files must now be additionally copied to the equivalent folders of the OS Project.
   With the program 'CematProjectUpdate.exe', which you find under D:\CEMAT_CS, the update can be carried out automatically.
5. Start the application 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project). Default directory is the last closed WinCC project.

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1680x1050, 1920x1050, 1920x1200 or 1600x1200) is important for all OS Projects with Operator Interface. The Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl, @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

6. Open the WinCC project.
7. Open 'Global Script', 'C_Editor' and regenerate the header. (Menu/Options/Regenerate Header)
8. Cemat needs a number of internal variables in the Tag Management. Via WinCC Configuration Studio the variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

   - Open the WinCC Configuration Studio for the Tag Management
   - Use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

   Further information can be found in the Engineering manual, chapter 03_PCS7_Project under Tag Management.
9. During the Update of the OS Project the settings in the Alarm logging have been changed to the default settings of PCS 7 which need to be corrected for Cemat regarding color definitions and acknowledgement/flashing behavior.

Message blocks and Message configuration can be imported via WinCC configuration Studio for Alarm Logging from already prepared Excel files Import_MessageBlocks.xlsx and Import_MessageConfiguration.xlsx which you find under D:\CEMAT_CS\IMPORT.

For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

Further information you will find in the Engineering Manual 03_PCS7_Project under Alarm logging.

10. During the update procedure the following GraCS files are overwritten by Cemat files. These are:

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

For Single User Stations and for OS Clients this is correct. For OS Servers the OS Project Editor must be started again, because in this case the standard pictures delivered with PCS 7 have to be used (the Server has no Visualization). Please select the above listed Pictures in order to be overwritten and confirm with OK.

For all other Station types the files have to be deselected if you run the OS Project Editor again.
<u>Caution:</u> The files @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must therefore be copied again after running the OS Project Editor.
(Start the program D:\CEMAT_CS\CematProjectUpdate.exe; the button “Res” copies the CEMAT “resolutions” pictures to the destination again).

11. During the Update of the GraCS Directory the new template picture C_@PCS7Typicals_CemV8_000.pdl has been copied. The template picture is suitable for all CEMAT screen resolutions.
As all Block Icons have been extended by new functions, we recommend using the new template picture, <u>but you have to add your additionally created/modified block icons</u>.

(After updating the template picture use Dynamic Wizard *Picture → Update of the Picture objects* in order to update the block icons of your process pictures.
Select your template picture name and config file @PCS7Typicals.cfg.)

12. If you have modified for your project system PDL’s and scripts you have to do the modification again in the new system files. Refer to your backup.

13. The Structure of the User Archive C_DriveList has been extended und must be replaced. The new structure can be imported via the WinCC Configuration Studio-
- Open the WinCC Configuration Studio for the User Archive and delete Archive C_DriveList.
- In order to import the new structure use function *Edit → Import*, then browse to

D:\CEMAT_CS\IMPORT, select Excel Workbook Import_UserArchives.xlsx and press Button *Import.*

Procedure, if PTE Controllers are used

Because of the early conversion of Process pictures by the OS migrator it can happen, that some PDL files are converted wrong. E. g. if OCX controls for PTE Objects are used in the Pictures and the OCX was not yet registered. The failure is: The OCX controls are not present.
Workaround: After the migration described above you have to copy the process pictures from the backup in the V7.1 version again the GraCS directory. In the WinCC explorer start the conversion of the die PDL files again. After that the PTE-Controllers are integrated correctly.

If you like to use still the PTE controllers for existing applications, you have to take care, that all the old faceplates, OCX files are installed in your project and on your PCs.

Compile and Download

1. For all PLCs: Compile CFC (complete program).
2. For all Single Stations and primary Servers: OS Compile (complete OS with memory reset). In the background the Server data (package) will be automatically created and loaded into the OS Client Project. Please double-check!
3. Load the complete program into the PLC.
4. Actualize the Server and OS Clients with Project Download.

Update CEMAT V8.0 to CEMAT V8.2

You have a CEMAT V8.0 project and want to upgrade to CEMAT V8.2 (based on PCS 7 V8.2). The following description explains the necessary steps.

Requirements

Before the installation of Cemat V8.2, you have to upgrade your PC Stations to the permitted Windows Operating System and to PCS 7 V8.2.

The following Windows Operating Systems are supported by PCS 7 V8.2, but not all versions are compatible with each Configuration.

- Windows 7 Professional SP1 (64-Bit, only English Windows Language Version)
  (permitted for ES, non-red. OS-Single Station, OS-Client, Web-Client)
- Windows 7 Ultimate/Enterprise SP1 (32-Bit)
  (permitted for OS-Client and Web-Client)
- Windows 7 Ultimate / Enterprise SP1 (64-Bit)
  (permitted for ES, OS-Single Station, OS-Client, Web-Server (only on ES or OS-Single) and Web-Client)
- Windows 10 Enterprise 2015 LTSB (64-Bit)
  (permitted for OS-Client and Web-Client)
- Windows Server 2008 R2 SP1 Standard Edition (64-Bit)
  (permitted for ES, OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client)
- Windows Server 2012 R2 Update Standard Edition (64-Bit)
  (permitted for OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client

Further details about the installation of the operating system and PCS 7 are explained in the PCS 7 manuals and in the readme and therefore not part of this description.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2
Make sure that the PCS 7 updates are installed before updating to CEMT V8.2!

Caution: The update to CEMAT V8.2 requires a complete reloading of the PLC and consequently **cannot** be carried out while the plant is running.

Follow the actual Manual "PCS 7 Software Update with utilization of the new functions" from the PCS 7 V8.2 Documentation. The following description refers to this manual.
Please also pay attention to the hints for the OS-Migration. In order to maintain the runtime data (curves and messages), the Server Projects have to be migrated on the Server-PCs directly.

Consider also the project rules from the CEMAT manual 03_PCS7_Project_009.pdf, “Installation of a PCS 7 Project” and the Cemat Object descriptions.

Save Project data

Save your project data. Beside the Multi Project this comprises all settings which you have carried out in the CEMAT_CS Directory, as well as modifications in the block library ILS_CEM or PRO_CEM.

- Archive the Multi Project with the SIMATIC Manager
- For saving the User Archives use Export Function.
- You can copy the modifications under CEMAT_CS (Config, Multimedia, Sounds) to a backup-directory.

Caution: Never save directory "CEMAT_CS/BIN"!!!
The files from the BIN directory are entered in the registry and must never be moved or deleted.

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

Please consider that the blocks C_SINA and C_ROBI (AddOn) are completely new designed. The existing blocks must not be used anymore. They must be removed and replaced by a C_DRV_1D or C_DRV_2D block plus the new adapter block C_SINA or C_ROBI.

Migration to PCS 7 V8.2

OS-Migration

Again refer to the OS-Migration in the Manual "PCS 7 - Software Update with utilization of the new functions".

The Migration of the OS Project is carried out automatically when you open the OS Project the first time with PCS 7 V8.2

Installation of CEMAT V8.2

With the following steps please proceed on a Station with PCS 7 V8.2!

Start the Setup program from the DVD CEMAT V8.2 and follow the instructions.

The Installation path is 'D:\CEMAT_CS'. This cannot be changed!

Keys for project standards	**000 = CEMAT** 001 = Minerals Cemat **004 = Holcim** **006 = Dyckerhoff** **007 = Heidelberg Cement** **023 = Vigier** **024 = Bushehr** **025 = Caima** **026 = Alsen** 027 = Lafarge **028 = Rossi**

After the installation, for all PC stations (ES, OS SINGLE STATION, OS SERVER and OS CLIENT) the following steps have to be carried out:

1. The config files contain the default settings, which means if you changed them for your project you have to make this adaptations again. (C_Config.cfg)
   Also consider the settings in file C_Messages.cfg.
2. Probably you also added for your project some files under 'Multimedia', 'Sounds' ….
   After the CEMAT installation these files can be restored from the backup directories and copied into the new additional CEMAT_CS\MULTIMEDIA directories.

Update of S7 Program

Please also consider the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions".

New CEMAT Libraries

With the installation of CEMAT V8.2 a new version of the CMEAT Library ILS_CEM has been installed in directory C:\Program Files\SIEMENS\STEP7\S7libs.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (project key other than "000") the library PRO_CEM is also transferred into C:\Program Files\SIEMENS\STEP7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Program folder **Cemat_AddOn** contains Open blocks as e. g. C_SIMOS, C_SINA, C_ROBI. They are delivered in a separate folder in order not to overwrite the existing program during update procedure.

In order to avoid overlapping with other libraries, in CEMAT V8.2 some AS blocks of the ILS_CEM library have been transferred to new addresses (see list below).
The transfer concerns all CEMAT Project Standards with exception of Project Standard 001 (In the Minerals Automation Standard the new block numbers have been used from the beginning).

Block name	**old Number**	**→ new Number**
C_ODA	FB1060	→ FB1046
C_ODA_R	FB1062	→ FB1047
C_INTERL	FB1075	→ FB1057
C_INTER5	FB1076	→ FB1058
C_RelMod	FB1077	→ FB1059
AG_LSEND	FC1050	→ FC1033
AG_LRECV	FC1060	→ FC1034
FC_GALA-WALA	FC1061	→ FC1021
FC_GRUZU_WEGZU	FC1062	→ FC1022
FC_GRINZ	FC1063	→ FC1023
StruInIn	FC1065	→ FC1025
StruInOu	FC1066	→ FC1026
C_SelQC	FC1067	→ FC1027
C_PUSHBT	FC1088	→ FC1028
OB1_SYS1	FC1102	→ FC1031
C_OB1SY1	FC1103	→ FC1032

Furthermore in the ILS_CEM all non-used CEMAT blocks have been removed. These are:

FC6	DT_DATE
FC509	BCD_SKCNT_CHECK
FC510	COPY_ELEM
FC516	FC_AUTO_APPEND
FC517	FC_GET_STRUCT_LEN
FC519	VAR_TO_DBNR
FC520	VAR_TO_BYNR
FC521	DIAGNOSE
FC522	BYTE_TO_INT
FC523	INT_TO_BYTE
FC524	BCD_TO_BYTE
FC525	BLOCK_DB_TO_INT
FC1020	C_SIMU_L
FC1064	FC_SIMU_LIST
DB676	C_SIMU_DB
DB679	DB_FATAL_ERROR

DB681	DB_DIAGNOSTIC
DB682	DB_EVENT
UDT1016	DIAGNOSE_ENTRY

New PCS 7 Libraries

With the installation of PCS 7 V8.2 also some new block libraries were installed:
- PCS 7 Basis Library V82
- PCS 7 AP Library V82.

A PCS 7 Library V82 doesn't exist!!!!!!!!
If you don't like to use the new APL blocks and you like still to use some blocks of the Basis Library and PCS 7 Library V71, then you have to install the newest version of the V7.1 libraries. You find the setup for the V7.1 libraries on the on the PCS 7 V8.2 Installation DVD under "Additional Products".

The newest PCS 7 Library V7.1 SP3 contains 2 versions of some channel driver blocks (e. g. CH_AI). The 2 blocks are identical, but they have different version numbers (V6.0 or V7.0).
The channel / driver blocks of the V7.1 have the version number V6.0.
The channel / driver blocks of the V8.0 have the version number V7.0.
A mix of the version number of the used channel / driver blocks is not allowed!!!!

Update Procedure for the S7-Program

For the update of your S7-Programs please proceed as follows:

- First update the Master data Library
- Then update the S7-Program containers for each AS
- Update the CFC for each AS

Actualize the Master data Library

The Master data library must contain all blocks which are used in the different S7-Projects of the Multiproject. These are the CEMAT blocks as well as block from other PCS 7 libraries which are used in the Multiproject

1. Open the block folder of the Master data library.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Open the block folder CEM_ALL of the CEMAT library ILS_CEM.
4. Select all blocks Copy all blocks from CEM_ALL and copy these into the block folder of your Master data library.
5. If a CEMAT Project Standard was installed, open library PRO_CEM and copy all blocks (overwrite = yes) into the block folder of your Master data library.
6. Your Master data library should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks cannot be used any more, but do not replace it by the new Adapter block now; otherwise you will lose your connections during the update.
7. If in your project system attributes or message attributes in the standard blocks have been modified, you have to make the adaptations after the update of the block folder manually again.

8. At this time you should also check,
   - if all other blocks (e. g. with Author BASISxx, TECHNxx or DRIVERxx, etc) are actualized.
   - that the Master data library does not contain any block with Author ES_MAP
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
9. Open the CFC editor by opening any chart.
10. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
11. In the System Chart SYSPLC00 add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.

After this the Actualization of the Master data library is completed.

Actualization of the S7-Program for each AS

For each AS you have to actualize the **S7-Program container** from the Master data library:

1. Open the block folder of your AS.
2. Delete from the block folder all above mentioned blocks. (Delete both, the moved blocks and the deleted blocks, including the symbols).
3. Then copy all Blocks from your Master data library into the block container of the AS.
4. Your AS block folder should not contain blocks with Author CEMAT anymore. If they still exist, e. g. if blocks from the Cemat_AddOn Library have been used, you have to pay special attention, because these blocks may have user specific adaptations!

   Caution: Existing C_SINA and C_ROBI blocks must be manually replaced after the CFC update by a C_DRV_1D or C_DRV_2D plus the new adapter block C_SINA or C_ROBI. Automatic replacement of C_SINA or C_ROBI is not possible, existing connections get lost!
5. If required, actualize further blocks of the S7-Program (e.g. according to the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“, from 01/2013, chapter 7.5.15).

Actualization of the CFC Charts

For each AS, carry out the following steps in order to actualize the CFC Charts:

1. Open the CFC editor by opening any chart.
2. Open *Reference data → Block types* and search for C_SIMU_L blocks. This function no longer exists and the blocks must be deleted.
3. Also check if C_SIMDMP (damper simulation) is has been used. During the update this block is replaced by a block with different parameter names and connections have to be rewired after the update.
4. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
5. Please check the log file after the update:
   - Connections to Parameters which could not been found in the new version (e. g. for C_SIMDMP) have been removed or converted to textual connections.
   - Please also check for textual connections, using the function *Options → Close Textual interconnections*.
   - After closing the textual interconnections the list should be empty. Otherwise you need to correct the program manually!

6. In the new version the size of the blocks may be different from the version before. This may lead to overlapping in the CFC. Eventually it is necessary to move some of the blocks in the CFC.
7. At this time you should also check,
   - if all blocks (e. g. with Author BASISxx, TECHNxx, CHNxx or DRIVERxx, etc) are actualized.
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
8. Adaptations for C_SINA and C_ROBI
   You have to replace the blocks by a C_DRV_1D/C_DRV_2D block + the new Adapter bocks C_SINA or C_ROBI. New Engineering required!
9. In the System Chart SYSPLCxx add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.
   Especially for Holcim plants, please check parameter REL_WSTP of block C_FB_PLC. REL_WSTP must be “0” in order not to interrupt the group start in case of a warning!
10. In folder D:\CEMAT_CS\TOOLS\CheckTool you find the new CEMAT Check tool “CematCheckTool.exe”. Please make use of this tool after upgrading your project.
    The check tool will tell you if your project structure is correct; e. g. if the CEMAT blocks are inserted at the correct installation position or if the Group links and C_MUX interconnections are correct.
    For details refer the CEMAT manual „10_Checkliste_009.pdf“.

After this block update a complete compile of all charts is required.

Carry out the same steps for all S7 Program folders of your Multiproject.

Update of the OS-Projects

Before the Cemat Update all OS Projects must be migrated to PCS 7 V8.2.

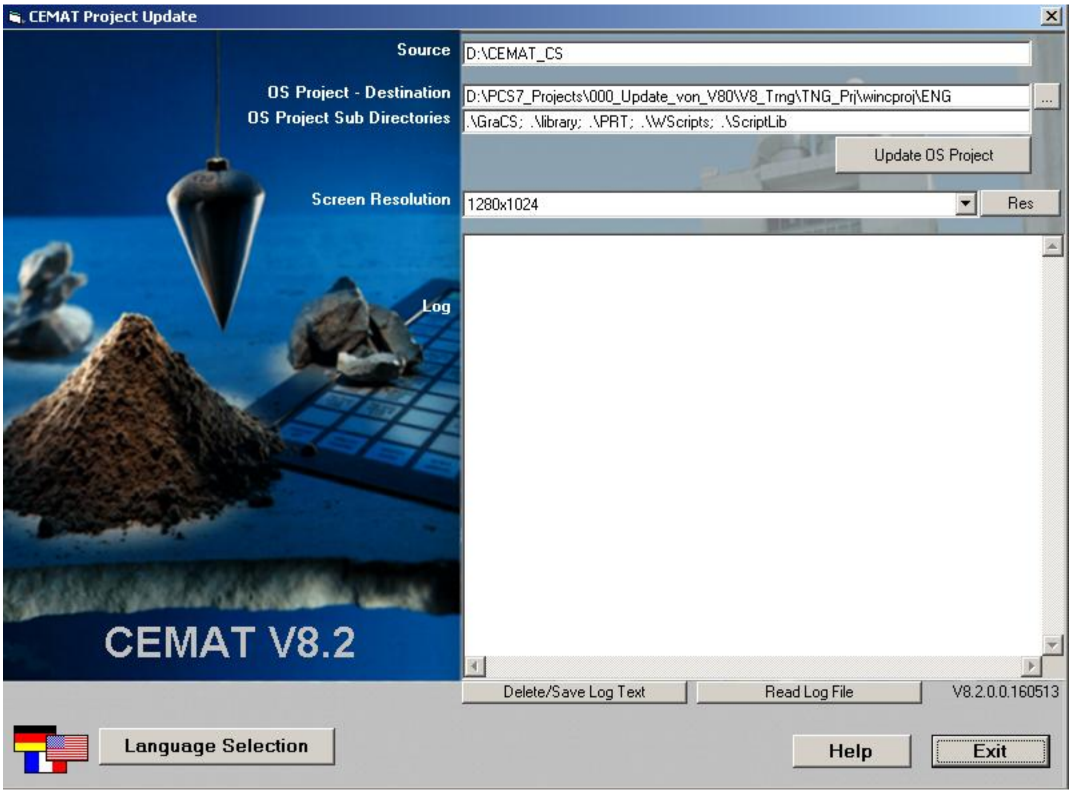
CEMAT Update

After the CEMAT installation, the D:\CEMAT_CS\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

In all WinCC Projects carry out the following steps:

1. Open the WinCC Explorer.
2. Open the OS Project Editor. Check the settings according to the Engineering Manual Chapter 3 "Installation of a PCS 7 Project".
   Close the OS Project Editor with "OK". All standard pictures (Overview, buttons, alarm line, etc.) are replaced by the new standard PDLs of PCS 7 V8.2.
3. Close WinCC Explorer.
4. During the Cemat V8.2 installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Library, PRT and WScipts. These files must now be additionally copied to the equivalent folders of the OS Project.
   With the program 'CematProjectUpdate.exe', which you find under D:\CEMAT_CS, the update can be carried out automatically.
5. Start the application 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project). Default directory is the last closed WinCC project.

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1680x1050, 1920x1050, 1920x1200 or 1600x1200) is important for all OS Projects with Operator Interface. The Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl, @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

6. Open the WinCC project.
7. Open 'Global Script', ‘C_Editor’ and regenerate the header. (Menu/Options/Regenerate Header)
8. Cemat needs a number of internal variables in the Tag Management. Via WinCC Configuration Studio the variables can be imported from file Import_InternalTags.xlsx which you find under D:\CEMAT_CS\IMPORT.

   - Open the WinCC Configuration Studio for the Tag Management
   - Use function *Edit* → *Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_InternalTags.xlsx and press Button *Import.*

   Further information can be found in the Engineering manual, chapter 03_PCS7_Project under Tag Management.
9. During the Update of the OS Project the settings in the Alarm logging have been changed to the default settings of PCS 7 which need to be corrected for Cemat regarding color definitions and acknowledgement/flashing behavior.

   Message blocks and Message configuration can be imported via WinCC configuration Studio for Alarm Logging from already prepared Excel files Import_MessageBlocks.xlsx and Import_MessageConfiguration.xlsx which you find under D:\CEMAT_CS\IMPORT.

   For updating the Message blocks regarding name, flashing and field size use function *Edit* → *Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

   For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit* → *Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

   Further information you will find in the Engineering Manual 03_PCS7_Project under Alarm logging.

10. During the update procedure the following GraCS files are overwritten by Cemat files. These are:

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

    For Single User Stations and for OS Clients this is correct. For OS Servers the OS Project Editor must be started again, because in this case the standard pictures delivered with PCS 7 have to be used (the Server has no Visualization). Please select the above listed Pictures in order to be overwritten and confirm with OK.

    For all other Station types the files have to be deselected if you run the OS Project Editor again.
    <u>Caution:</u> The files @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must therefore be copied again after running the OS Project Editor.
    (Start the program D:\CEMAT_CS\CematProjectUpdate.exe; the button “Res” copies the CEMAT “resolutions” pictures to the destination again).

11. During the Update of the GraCS Directory the new template picture C_@PCS7Typicals_CemV8_000.pdl has been copied. The template picture is suitable for all CEMAT screen resolutions.
    As all Block Icons have been extended by new functions, we recommend using the new template picture, <u>but you have to add your additionally created/modified block icons</u>.

    (After updating the template picture use Dynamic Wizard *Picture → Update of the Picture objects* in order to update the block icons of your process pictures.
    Select your template picture name and config file @PCS7Typicals.cfg.)

12. If you have modified for your project system PDL’s and scripts you have to do the modification again in the new system files. Refer to your backup.

13. The Structure of the User Archive C_DriveList has been extended und must be replaced. The new structure can be imported via the WinCC Configuration Studio
    - Open the WinCC Configuration Studio for the User Archive and delete Archive C_DriveList.
    - In order to import the new structure use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_UserArchives.xlsx and press Button *Import.*

Compile and Download

1. For all PLCs: Compile CFC (complete program).
2. For all Single Stations and primary Servers: OS Compile (complete OS with memory reset). In the background the Server data (package) will be automatically created and loaded into the OS Client Project. Please double-check!
3. Load the complete program into the PLC.
4. Actualize the Server and OS Clients with Project Download.

Update CEMAT V8.0 SP1 to CEMAT V8.2

You have a CEMAT V8.0 SP1 project and want to upgrade to CEMAT V8.2 (based on PCS 7 V8.2). The following description explains the necessary steps.

Requirements

Before the installation of Cemat V8.2, you have to upgrade your PC Stations to the permitted Windows Operating System and to PCS 7 V8.2.

The following Windows Operating Systems are supported by PCS 7 V8.2, but not all versions are compatible with each Configuration.

- Windows 7 Professional SP1 (64-Bit, only English Windows Language Version)
  (permitted for ES, non-red. OS-Single Station, OS-Client, Web-Client)
- Windows 7 Ultimate/Enterprise SP1 (32-Bit)
  (permitted for OS-Client and Web-Client)
- Windows 7 Ultimate / Enterprise SP1 (64-Bit)
  (permitted for ES, OS-Single Station, OS-Client, Web-Server (only on ES or OS-Single) and Web-Client)
- Windows 10 Enterprise 2015 LTSB (64-Bit)
  (permitted for OS-Client and Web-Client)
- Windows Server 2008 R2 SP1 Standard Edition (64-Bit)
  (permitted for ES, OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client)
- Windows Server 2012 R2 Update Standard Edition (64-Bit)
  (permitted for OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client

Further details about the installation of the operating system and PCS 7 are explained in the PCS 7 manuals and in the readme and therefore not part of this description.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2

Make sure that the PCS 7 updates are installed before updating to CEMT V8.2!

Caution: The update to CEMAT V8.2 requires a complete reloading of the PLC and consequently **cannot** be carried out while the plant is running.

Follow the actual Manual "PCS 7 Software Update with utilization of the new functions" from the PCS 7 V8.2 Documentation. The following description refers to this manual.
Please also pay attention to the hints for the OS-Migration. In order to maintain the runtime data (curves and messages), the Server Projects have to be migrated on the Server-PCs directly.

Consider also the project rules from the CEMAT manual 03_PCS7_Project_009.pdf, “Installation of a PCS 7 Project” and the Cemat Object descriptions.

Save Project data

Save your project data. Beside the Multi Project this comprises all settings which you have carried out in the CEMAT_CS Directory, as well as modifications in the block library ILS_CEM or PRO_CEM.

- Archive the Multi Project with the SIMATIC Manager
- For saving the User Archives use Export Function.
- You can copy the modifications under CEMAT_CS (Config, Multimedia, Sounds) to a backup-directory.

Caution: Never save directory "CEMAT_CS/BIN"!!!
The files from the BIN directory are entered in the registry and must never be moved or deleted.

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

Migration to PCS 7 V8.2

OS-Migration

Again refer to the OS-Migration in the Manual "PCS 7 - Software Update with utilization of the new functions".

The Migration of the OS Project is carried out automatically when you open the OS Project the first time with PCS 7 V8.2

Installation of CEMAT V8.2

With the following steps please proceed on a Station with PCS 7 V8.2!

Start the Setup program from the DVD CEMAT V8.2 and follow the instructions.

The Installation path is 'D:\CEMAT_CS'. This cannot be changed!

Keys for project standards	**000 = CEMAT** 001 = Minerals Cemat **004 = Holcim** **006 = Dyckerhoff** **007 = Heidelberg Cement** **023 = Vigier** **024 = Bushehr** **025 = Caima** **026 = Alsen** 027 = Lafarge **028 = Rossi**

After the installation, for all PC stations (ES, OS SINGLE STATION, OS SERVER and OS CLIENT) the following steps have to be carried out:

1. The config files contain the default settings, which means if you changed them for your project you have to make this adaptations again. (C_Config.cfg)
   Also consider the settings in file C_Messages.cfg.
2. Probably you also added for your project some files under 'Multimedia', 'Sounds' ….
   After the CEMAT installation these files can be restored from the backup directories and copied into the new additional CEMAT_CS\MULTIMEDIA directories.

Update of S7 Program

Please also consider the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions".

New CEMAT Libraries

With the installation of CEMAT V8.2 a new version of the CMEAT Library ILS_CEM has been installed in directory C:\Program Files\SIEMENS\STEP7\S7libs.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (project key other than “000”) the library PRO_CEM is also transferred into C:\Program Files\SIEMENS\STEP7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Program folder **Cemat_AddOn** contains Open blocks as e. g. C_SIMOS, C_SINA, C_ROBI. They are delivered in a separate folder in order not to overwrite the existing program during update procedure.

New PCS 7 Libraries

With the installation of PCS 7 V8.2 also some new block libraries were installed:
- PCS 7 Basis Library V82
- PCS 7 AP Library V82.

A PCS 7 Library V82 doesn't exist!!!!!!!!
If you don't like to use the new APL blocks and you like still to use some blocks of the Basis Library and PCS 7 Library V71, then you have to install the newest version of the V7.1 libraries. You find the setup for the V7.1 libraries on the on the PCS 7 V8.2 Installation DVD under "Additional Products".

The newest PCS 7 Library V7.1 SP3 contains 2 versions of some channel driver blocks (e. g. CH_AI). The 2 blocks are identical, but they have different version numbers (V6.0 or V7.0).
The channel / driver blocks of the V7.1 have the version number V6.0.
The channel / driver blocks of the V8.0 have the version number V7.0.
A mix of the version number of the used channel / driver blocks is not allowed!!!!

Update Procedure for the S7-Program

For the update of your S7-Programs please proceed as follows:

- First update the Master data Library
- Then update the S7-Program containers for each AS
- Update the CFC for each AS

Actualize the Master data Library

The Master data library must contain all blocks which are used in the different S7-Projects of the Multiproject. These are the CEMAT blocks as well as block from other PCS 7 libraries which are used in the Multiproject

1. Open the block folder of the Master data library.
2. Open the block folder CEM_ALL of the CEMAT library ILS_CEM.
3. Select all blocks Copy all blocks from CEM_ALL and copy these into the block folder of your Master data library.
4. If a CEMAT Project Standard was installed, open library PRO_CEM and copy all blocks (overwrite = yes) into the block folder of your Master data library.
5. If in your project system attributes or message attributes in the standard blocks have been modified, you have to make the adaptations after the update of the block folder manually again.
6. At this time you should also check,
   - if all other blocks (e. g. with Author BASISxx, TECHNxx or DRIVERxx, etc) are actualized.
   - that the Master data library does not contain any block with Author ES_MAP
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
7. Open the CFC editor by opening any chart.
8. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
9. In the System Chart SYSPLC00 add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.

After this the Actualization of the Master data library is completed.

Actualization of the S7-Program for each AS

For each AS you have to actualize the **S7-Program container** from the Master data library:

1. Open the block folder of your AS.
2. Then copy all Blocks from your Master data library into the block container of the AS.
3. If required, actualize further blocks of the S7-Program (e.g. according to the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“, from 01/2013, chapter 7.5.15).

Actualization of the CFC Charts

For each AS, carry out the following steps in order to actualize the CFC Charts:

1. Open the CFC editor by opening any chart.
2. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
3. Please check the log file after the update:
   - Connections to Parameters which could not been found in the new version have been removed or converted to textual connections.
   - Please also check for textual connections, using the function *Options → Close Textual interconnections*.
   - After closing the textual interconnections the list should be empty. Otherwise you need to correct the program manually!
4. In the new version the size of the blocks may be different from the version before. This may lead to overlapping in the CFC. Eventually it is necessary to move some of the blocks in the CFC.
5. At this time you should also check,
   - if all blocks (e. g. with Author BASISxx, TECHNxx, CHNxx or DRIVERxx, etc) are actualized.
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
6. In the System Chart SYSPLCxx add block C_CNT100 and insert this block into Runtime group OB35_START, after block OB35_CYC_Start.
   Especially for Holcim plants, please check parameter REL_WSTP of block C_FB_PLC. REL_WSTP must be “0” in order not to interrupt the group start in case of a warning!
7. In folder D:\CEMAT_CS\TOOLS\CheckTool you find the new CEMAT Check tool “CematCheckTool.exe”. Please make use of this tool after upgrading your project.
   The check tool will tell you if your project structure is correct; e. g. if the CEMAT blocks are inserted at the correct installation position or if the Group links and C_MUX interconnections are correct.
   For details refer the CEMAT manual „10_Checkliste_009.pdf“.

After this block update a complete compile of all charts is required.

Carry out the same steps for all S7 Program folders of your Multiproject.

Update of the OS-Projects

Before the Cemat Update all OS Projects must be migrated to PCS 7 V8.2.

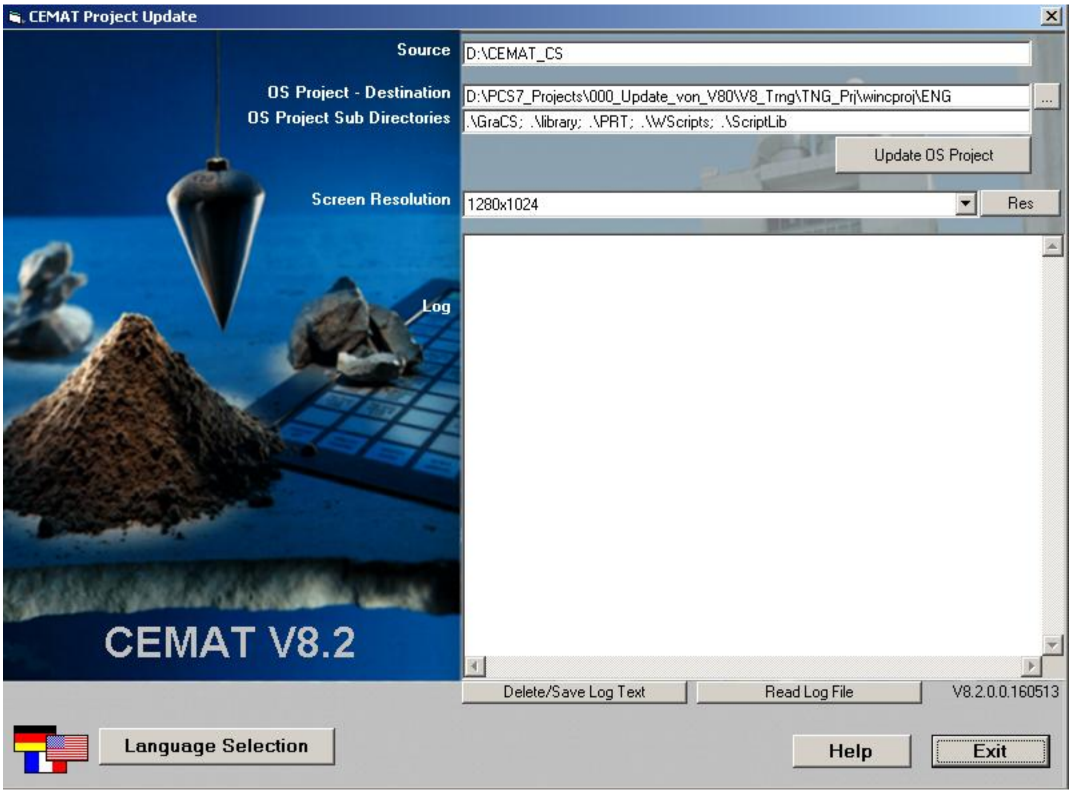
CEMAT Update

After the CEMAT installation, the D:\CEMAT_CS\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

In all WinCC Projects carry out the following steps:

1. Open the WinCC Explorer.
2. Open the OS Project Editor. Check the settings according to the Engineering Manual Chapter 3 "Installation of a PCS 7 Project".
   Close the OS Project Editor with "OK". All standard pictures (Overview, buttons, alarm line, etc.) are replaced by the new standard PDLs of PCS 7 V8.2.
3. Close WinCC Explorer.
4. During the Cemat V8.2 installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Library, PRT and WScipts. These files must now be additionally copied to the equivalent folders of the OS Project.
   With the program 'CematProjectUpdate.exe', which you find under D:\CEMAT_CS, the update can be carried out automatically.
5. Start the application 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project). Default directory is the last closed WinCC project.

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1680x1050, 1920x1050, 1920x1200 or 1600x1200) is important for all OS Projects with Operator Interface. The Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl, @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

6. Open the WinCC project.
7. Open 'Global Script', 'C_Editor' and regenerate the header. (Menu/Options/Regenerate Header)
8. During the Update of the OS Project the settings in the Alarm logging have been changed to the default settings of PCS 7 which need to be corrected for Cemat regarding color definitions and acknowledgement/flashing behavior.

   Message blocks and Message configuration can be imported in the WinCC configuration Studio from already prepared Excel files Import_MessageBlocks.xlsx and Workbook Import_MessageConfiguration.xlsx which you find under D:\CEMAT_CS\IMPORT.

   For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

   For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

   Further information you will find in the Engineering Manual 03_PCS7_Project under Alarm logging.
9. During the update procedure the following GraCS files are overwritten by Cemat files. These are:

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

   For Single User Stations and for OS Clients this is correct. For OS Servers the OS Project Editor must be started again, because in this case the standard pictures delivered with PCS 7 have to be used (the Server has no Visualization). Please select the above listed Pictures in order to be overwritten and confirm with OK.

   For all other Station types the files have to be deselected if you run the OS Project Editor again.

Caution: The files @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must therefore be copied again after running the OS Project Editor.
(Start the program D:\CEMAT_CS\CematProjectUpdate.exe; the button “Res” copies the CEMAT “resolutions” pictures to the destination again).

10. During the Update of the GraCS Directory the new template picture C_@PCS7Typicals_CemV8_000.pdl has been copied. The template picture is suitable for all CEMAT screen resolutions.
    As all Block Icons have been extended by new functions, we recommend using the new template picture, but you have to add your additionally created/modified block icons.

    (After updating the template picture use Dynamic Wizard *Picture → Update of the Picture objects* in order to update the block icons of your process pictures.
    Select your template picture name and config file @PCS7Typicals.cfg.)

11. If you have modified for your project system PDL’s and scripts you have to do the modification again in the new system files. Refer to your backup.

Compile and Download

1. For all PLCs: Compile CFC (complete program).
2. For all Single Stations and primary Servers: OS Compile (complete OS with memory reset). In the background the Server data (package) will be automatically created and loaded into the OS Client Project. Please double-check!
3. Load the complete program into the PLC.
4. Actualize the Server and OS Clients with Project Download.

Update CEMAT V8.1 to CEMAT V8.2

You have a CEMAT V8.1 project and want to upgrade to CEMAT V8.2 (based on PCS 7 V8.2). The following description explains the necessary steps.

Requirements

Before the installation of Cemat V8.2, you have to upgrade your PC Stations to the permitted Windows Operating System and to PCS 7 V8.2.

The following Windows Operating Systems are supported by PCS 7 V8.2, but not all versions are compatible with each Configuration.

- Windows 7 Professional SP1 (64-Bit, only English Windows Language Version)
  (permitted for ES, non-red. OS-Single Station, OS-Client, Web-Client)
- Windows 7 Ultimate/Enterprise SP1 (32-Bit)
  (permitted for OS-Client and Web-Client)
- Windows 7 Ultimate / Enterprise SP1 (64-Bit)
  (permitted for ES, OS-Single Station, OS-Client, Web-Server (only on ES or OS-Single) and Web-Client)
- Windows 10 Enterprise 2015 LTSB (64-Bit)
  (permitted for OS-Client and Web-Client)
- Windows Server 2008 R2 SP1 Standard Edition (64-Bit)
  (permitted for ES, OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client)
- Windows Server 2012 R2 Update Standard Edition (64-Bit)
  (permitted for OS-Single Station, OS-Server, OS-Client, Web-Server, Web-Client

Further details about the installation of the operating system and PCS 7 are explained in the PCS 7 manuals and in the readme and therefore not part of this description.

Caution: For CEMAT V8.2 the following Updates for PCS 7 are essential:
- AS-OS Eng. V8.3 Update 1
- CFC V8.2 Update 2
Make sure that the PCS 7 updates are installed before updating to CEMT V8.2!

Caution: The update to CEMAT V8.2 requires a complete reloading of the PLC and consequently **cannot** be carried out while the plant is running.

Follow the actual Manual "PCS 7 Software Update with utilization of the new functions" from the PCS 7 V8.2 Documentation. The following description refers to this manual.
Please also pay attention to the hints for the OS-Migration. In order to maintain the runtime data (curves and messages), the Server Projects have to be migrated on the Server-PCs directly.

Consider also the project rules from the CEMAT manual 03_PCS7_Project_009.pdf, “Installation of a PCS 7 Project” and the Cemat Object descriptions.

Save Project data

Save your project data. Beside the Multi Project this comprises all settings which you have carried out in the CEMAT_CS Directory, as well as modifications in the block library ILS_CEM or PRO_CEM.

- Archive the Multi Project with the SIMATIC Manager
- For saving the User Archives use Export Function.
- You can copy the modifications under CEMAT_CS (Config, Multimedia, Sounds) to a backup-directory.

Caution: Never save directory "CEMAT_CS/BIN"!!!
The files from the BIN directory are entered in the registry and must never be moved or deleted.

Modified/new functions

The modified/new functions are listed in file readme_V82.pdf on the CEMAT Installation-DVD.

Migration to PCS 7 V8.2

OS-Migration

Again refer to the OS-Migration in the Manual "PCS 7 - Software Update with utilization of the new functions".

The Migration of the OS Project is carried out automatically when you open the OS Project the first time with PCS 7 V8.2

Installation of CEMAT V8.2

With the following steps please proceed on a Station with PCS 7 V8.2!

Start the Setup program from the DVD CEMAT V8.2 and follow the instructions.

The Installation path is 'D:\CEMAT_CS'. This cannot be changed!

Keys for project standards

000 = CEMAT
001 = Minerals Cemat
004 = Holcim
006 = Dyckerhoff
007 = Heidelberg Cement
023 = Vigier
024 = Bushehr
025 = Caima
026 = Alsen
027 = Lafarge
028 = Rossi

After the installation, for all PC stations (ES, OS SINGLE STATION, OS SERVER and OS CLIENT) the following steps have to be carried out:

1. The config files contain the default settings, which means if you changed them for your project you have to make this adaptations again. (C_Config.cfg)
   Also consider the settings in file C_Messages.cfg.
2. Probably you also added for your project some files under 'Multimedia', 'Sounds' ….
   After the CEMAT installation these files can be restored from the backup directories and copied into the new additional CEMAT_CS\MULTIMEDIA directories.

Update of S7 Program

Please also consider the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions".

New CEMAT Libraries

With the installation of CEMAT V8.2 a new version of the CMEAT Library ILS_CEM has been installed in directory C:\Program Files\SIEMENS\STEP7\S7libs.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (project key other than "000") the library PRO_CEM is also transferred into C:\Program Files\SIEMENS\STEP7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Program folder **Cemat_AddOn** contains Open blocks as e. g. C_SIMOS, C_SINA, C_ROBI. They are delivered in a separate folder in order not to overwrite the existing program during update procedure.

New PCS 7 Libraries

With the installation of PCS 7 V8.2 also some new block libraries were installed:
- PCS 7 Basis Library V82
- PCS 7 AP Library V82.

A PCS 7 Library V82 doesn't exist!!!!!!!!
If you don't like to use the new APL blocks and you like still to use some blocks of the Basis Library and PCS 7 Library V71, then you have to install the newest version of the V7.1 libraries. You find the setup for the V7.1 libraries on the on the PCS 7 V8.2 Installation DVD under "Additional Products".

The newest PCS 7 Library V7.1 SP3 contains 2 versions of some channel driver blocks (e. g. CH_AI). The 2 blocks are identical, but they have different version numbers (V6.0 or V7.0).
The channel / driver blocks of the V7.1 have the version number V6.0.
The channel / driver blocks of the V8.0 have the version number V7.0.
A mix of the version number of the used channel / driver blocks is not allowed!!!!

Update Procedure for the S7-Program

For the update of your S7-Programs please proceed as follows:

- First update the Master data Library
- Then update the S7-Program containers for each AS
- Update the CFC for each AS

Actualize the Master data Library

The Master data library must contain all blocks which are used in the different S7-Projects of the Multiproject. These are the CEMAT blocks as well as block from other PCS 7 libraries which are used in the Multiproject

1. Open the block folder of the Master data library.
2. Open the block folder CEM_ALL of the CEMAT library ILS_CEM.
3. Select all blocks Copy all blocks from CEM_ALL and copy these into the block folder of your Master data library.
4. If a CEMAT Project Standard was installed, open library PRO_CEM and copy all blocks (overwrite = yes) into the block folder of your Master data library.
5. If in your project system attributes or message attributes in the standard blocks have been modified, you have to make the adaptations after the update of the block folder manually again.
6. At this time you should also check,
   - if all other blocks (e. g. with Author BASISxx, TECHNxx or DRIVERxx, etc) are actualized.
   - that the Master data library does not contain any block with Author ES_MAP
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
7. Open the CFC editor by opening any chart.
8. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".

After this the Actualization of the Master data library is completed.

Actualization of the S7-Program for each AS

For each AS you have to actualize the **S7-Program container** from the Master data library:

1. Open the block folder of your AS.
2. Then copy all Blocks from your Master data library into the block container of the AS.
3. If required, actualize further blocks of the S7-Program (e.g. according to the instructions in the PCS 7 Service Manual "PCS 7 - Software Update with utilization of the new functions“, from 01/2013, chapter 7.5.15).

Actualization of the CFC Charts

For each AS, carry out the following steps in order to actualize the CFC Charts:

1. Open the CFC editor by opening any chart.
2. In order to update the chart folder use *Options -> Block Types*, select all blocks in the chart folder (right window) and press "New Version".
3. Please check the log file after the update:
   - Connections to Parameters which could not been found in the new version have been removed or converted to textual connections.
   - Please also check for textual connections, using the function *Options → Close Textual interconnections*.
   - After closing the textual interconnections the list should be empty. Otherwise you need to correct the program manually!
4. In the new version the size of the blocks may be different from the version before. This may lead to overlapping in the CFC. Eventually it is necessary to move some of the blocks in the CFC.
5. At this time you should also check,
   - if all blocks (e. g. with Author BASISxx, TECHNxx, CHNxx or DRIVERxx, etc) are actualized.
   - that no S7-300 blocks (Author ELEM_300) are used (these blocks are not usable for S7-400 and lead to sporadic faults!).
6. In folder D:\CEMAT_CS\TOOLS\CheckTool you find the new CEMAT Check tool "CematCheckTool.exe". Please make use of this tool after upgrading your project.
   The check tool will tell you if your project structure is correct; e. g. if the CEMAT blocks are inserted at the correct installation position or if the Group links and C_MUX interconnections are correct.
   For details refer the CEMAT manual „10_Checkliste_009.pdf".

After this block update a complete compile of all charts is required.

Carry out the same steps for all S7 Program folders of your Multiproject.

Update of the OS-Projects

Before the Cemat Update all OS Projects must be migrated to PCS 7 V8.2.

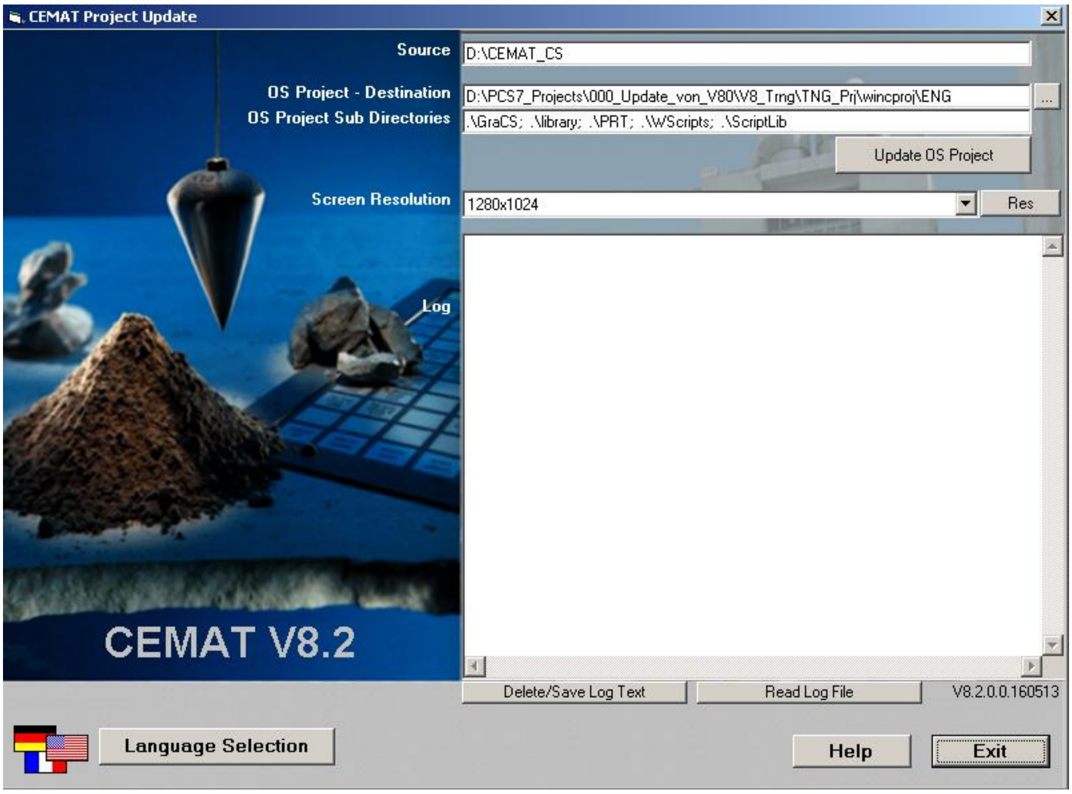
CEMAT Update

After the CEMAT installation, the D:\CEMAT_CS\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

We assume that in your project the picture tree was derived from the plant view and that the standard message selection of PCS 7 is used, which is also used since V6.1 of CEMAT.

In all WinCC Projects carry out the following steps:

1. Open the WinCC Explorer.
2. Open the OS Project Editor. Check the settings according to the Engineering Manual Chapter 3 "Installation of a PCS 7 Project".
   Close the OS Project Editor with "OK". All standard pictures (Overview, buttons, alarm line, etc.) are replaced by the new standard PDLs of PCS 7 V8.2.
3. Close WinCC Explorer.
4. During the Cemat V8.2 installation the Cemat WinCC files are copied into folder D:\CEMAT_CS\WinCC under GraCS, Library, PRT and WScipts. These files must now be additionally copied to the equivalent folders of the OS Project.
   With the program 'CematProjectUpdate.exe', which you find under D:\CEMAT_CS, the update can be carried out automatically.
5. Start the application 'CematProjectUpdate.exe'.



The **Source** must be D:\CEMAT_CS

Under **OS Project - Destination** you have to enter destination (Directory of the OS Project). Default directory is the last closed WinCC project.

Under **OS Project Sub Directories** you find the Subfolders which get copied during the update (cannot be modified).

The **Screen Resolution** (1280x1024, 1680x1050, 1920x1050, 1920x1200 or 1600x1200) is important for all OS Projects with Operator Interface. The Standard pictures @AlarmEmergency.pdl, @AlarmEmergencyOP.pdl, @AlarmOneLine.pdl, @C_AlarmListing.pdl, @Overview1.pdl, @TopAlarmNew.pdl are exchanged according to the screen resolution.

Press button '**Update OS Project'** and the files will be copied to the corresponding directories.

6. Open the WinCC project.
7. Open 'Global Script', 'C_Editor' and regenerate the header. (Menu/Options/Regenerate Header)
8. During the Update of the OS Project the settings in the Alarm logging have been changed to the default settings of PCS 7 which need to be corrected for Cemat regarding color definitions and acknowledgement/flashing behavior.

   Message blocks and Message configuration can be imported in the WinCC configuration Studio from already prepared Excel files Import_MessageBlocks.xlsx and Workbook Import_MessageConfiguration.xlsx which you find under D:\CEMAT_CS\IMPORT.

   For updating the Message blocks regarding name, flashing and field size use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageBlocks.xlsx and press Button *Import.*

   For updating the Message Configuration regarding color definition and acknowledgement, use function *Edit → Import*, then browse to D:\CEMAT_CS\IMPORT, select Excel Workbook Import_MessageConfiguration.xlsx and press Button *Import.*

   Further information you will find in the Engineering Manual 03_PCS7_Project under Alarm logging.
9. During the update procedure the following GraCS files are overwritten by Cemat files. These are:

@AlarmEmergency.pdl	Extended Alarm line
@AlarmEmergencyOP.pdl	Extended Alarm line
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@Buttons12.pdl	Button keys 2
@Language.pdl	Language Selection
@Overview1.pdl	Overview Range
@TopAlarmNew.pdl	Alarm lines
@PG_Intlk02_Standard.pdl	Interlock faceplate
@PG_Intlk04_Standard.pdl	Interlock faceplate
@PG_Intlk08_Standard.pdl	Interlock faceplate
@PG_Intlk16_Screen2.pdl	Interlock faceplate
@PG_Intlk16_Standard.pdl	Interlock faceplate

   For Single User Stations and for OS Clients this is correct. For OS Servers the OS Project Editor must be started again, because in this case the standard pictures delivered with PCS 7 have to be used (the Server has no Visualization). Please select the above listed Pictures in order to be overwritten and confirm with OK.

   For all other Station types the files have to be deselected if you run the OS Project Editor again.

Caution: The files @TopAlarmNew.pdl and @Overview1.pdl cannot be deselected and must therefore be copied again after running the OS Project Editor.
(Start the program D:\CEMAT_CS\CematProjectUpdate.exe; the button “Res” copies the CEMAT “resolutions” pictures to the destination again).

10. During the Update of the GraCS Directory the new template picture C_@PCS7Typicals_CemV8_000.pdl has been copied. The template picture is suitable for all CEMAT screen resolutions.
As all Block Icons have been extended by new functions, we recommend using the new template picture, but you have to add your additionally created/modified block icons.

(After updating the template picture use Dynamic Wizard *Picture → Update of the Picture objects* in order to update the block icons of your process pictures.
Select your template picture name and config file @PCS7Typicals.cfg.)

11. If you have modified for your project system PDL’s and scripts you have to do the modification again in the new system files. Refer to your backup.

Compile and Download

1. For all PLCs: Compile CFC (complete program).
2. For all Single Stations and primary Servers: OS Compile (complete OS with memory reset). In the background the Server data (package) will be automatically created and loaded into the OS Client Project. Please double-check!
3. Load the complete program into the PLC.
4. Actualize the Server and OS Clients with Project Download.

CEMAT Tools C_MIG_V80.XLS and C_SIM_RIGHT.XLS

Excel tool C_MIG_V80.XLS

The Excel Tool C_MIG_V80.XLS compares the Project BEFORE and AFTER the block update and refers to eventual differences. Some connection can as well be corrected with the tool.

If want to use this tool you have to export the Process Object view **BEFORE and AFTER the block update** and read in to the tool.

The following steps need to be done already in the PCS 7 Project:

1. The original, not yet migrated Project has been opened.
   In the Prozess Object view, you have selected the complete AS selected and actualized the data with F5. After this you exported the data as follows:

   - *Process Objects → Export all IOs …*
     The file name is e. g. AS1_PS_old.csv.

   - *Process Objects → Export Blocks...*
     The file name is e. g. AS1_B_csv.
2. After this the CFC charts have been migrated to V8.2.
3. Next you again opened the Process Object, selected the complete AS and actualized the data with F5. After this you exported the data as follows:

   - *Process Objects → Export all IOs…*
     The file name is e. g. AS1_PS_V82.csv.

For the next step you need the Excel Tool C_MIG_V80.XLS which you can find in folder D:\CEMAT_CS\TOOLS.

4. Copy the Excel Tool to the same folder where the exported data AS1_PS_old.csv, AS1_B_csv, AS1_Textual Interconnections.txt and AS1_PS_V82.csv are located.
   Open file C_MIG_V80.XLS and enter under ‚Presettings'the file names and the name new name of the Import file.
   Start the Macro via button "Create PO Import List".
   "Excel 2003" is limited to 65535 rows. If there are many objects in your project, it could be that the Excel macro stops by a fault because of this limit. In this case we recommend to use "Excel 2010". Store the XLS file as XLSM file. **Don't** run the file in the "compatibility mode". In the "compatibility mode" there is still the limit of 65535 rows!
5. Open table "Log text refs". In this sheet are listed all textual interconnections from the LOG file e. g. AS1_Textual Interconnections.txt. The textual interconnections which could be completed get green color. The textual interconnections which could be completed, but a rewired interconnection is stored in the import file, get yellow color (E. g. rewiring from MV to PV_Out). The lines marked in red indicate that the textual interconnection could not be completed. These interconnections have to be restored manually. Check each individual case.
6. After successful execution of the Macro you find the Import file for the Process Object view in the same directory.

The next step you have to carry out in the PCS 7 Project:

7. Import the Import file in the Process Object view, using the function *Process Objects → Import IOs…*
   If you do not like to rewire all listed interconnections, please check all entries in the generated import file. If necessary, modify the import file manually. If you get during import the following message: "The I/O attributes could not be imported. --- The connection is not selected for the process object view." In this case, please set the corresponding I/O as "Parameter."
   Process Objects → Select IOs…. Use the filter to find the corresponding I/O.

Excel tool C_SIM_RIGHT.XLS

The Excel Tool C_SIM_RIGHT.XLS is only needed if the user right for Simulation of C_ANNUNC, C_ANNUN8, C_MEASUR or C_PROFB has been individually adapted via block icon (via the Attribute "HigherProcesscontrolling").

The blocks C_ANNUNC, C_ANNUN8, C_MEASUR and C_PROFB (Holcim) have a new Parameter "SimRight". The user right for the Simulation can now be adapted via this parameter.

With the Excel Tool C_SIM_RIGHT.XLS the settings at the bock icon can be transferred to Parameter SimRight. Please proceed as follows:

1. Open a PDL file from GraCS and choose in the window "Dynamic Wizard" the tab "Picture Functions". Choose there the function "Export picture objects" (select all pictures) and store the result in a CSV file, e.g. PDL_Export.CSV.
2. In the process object view, select the complete AS and actualize the data with F5. Carry out the function *Process Objects → Export IOs…*
   As file name choose e. g. AS1_PS_V82.csv.
3. Copy the Excel tool C_SIM_RIGHT.XLS into the same directory with the exported files AS1_PS_V82.csv and PDL_Export.CSV.
4. Open the file C_SIM_RIGHT.XLS and enter under "Presettings" the file names and additionally the name for the import file.
   Start the Macro via button "Create PO Import List".
5. "Excel 2003" is limited to 65535 rows. If there are many objects in your project, it could be that the Excel macro stops by a fault because of this limit. In this case we recommend to use "Excel 2010". Store the XLS file as XLSM file. **Don't** run the file in the "compatibility mode". In the "compatibility mode" there is still the limit of 65535 rows!
6. Open the sheet "objects". In this sheet are listed all block icons of the block types C_ANNUNC, C_ANNUN8, C_MEASUR and C_PROFB (Holcim), which have a user right value (HigherProcesscontrolling), which is different from the default value 24:
   - If the macro could create an import entry for the process object view, then the complete line gets green color.
   - If there is another line for further block icon connected to the same tag, then the complete line gets yellow.
   - If the tag from the block icon was not found in the process object view, then the complete
   line gets red.
7. After completion of the macro you find in the same directory an import file for the process object view. With this import file you could change all SimRight parameters to the values of the corresponding attributes HigherProcesscontrolling.
8. Import this file in the process object view using the function
   Process Objects → Import IOs… If you do not like to change all listed SimRight parameters, please check all entries from the import file. Edit if necessary the import file manually or edit the SimRight parameters manually!

⚠ **Caution:** The function "Export picture objects" exports the data of all pictures, i.e. the data of all AS. The export of the Process Object View exports normally the data of only one AS. Either you create an Excel file for each AS. In this case, the line for the block icons of the other AS will be marked as "not found" = red color. Or you copy the export CSV files of the Process Object View together into one file. In this case you have only one Excel file. But the import file you have to import into the Process Object View for each AS. The entries for the other AS create error messages during import.