Introduction

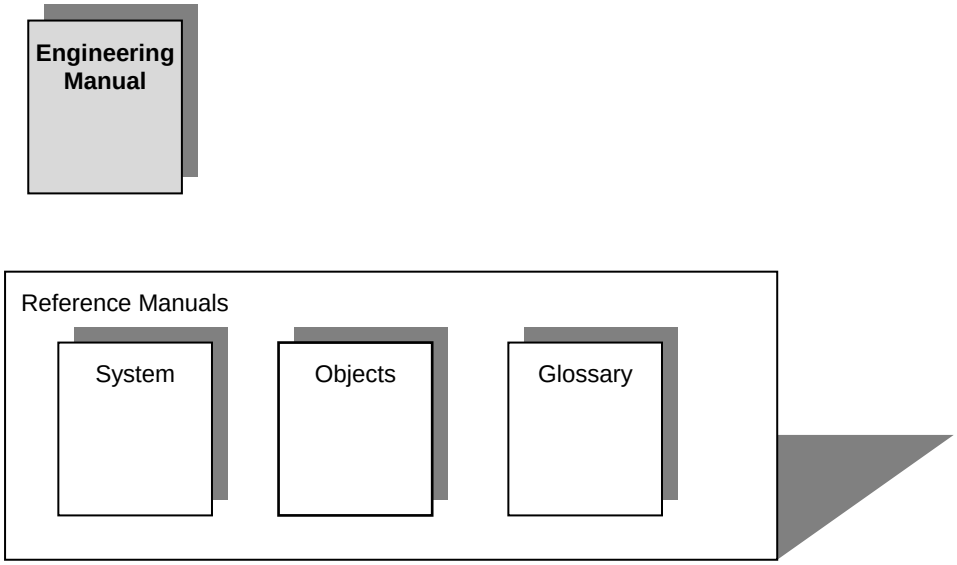
Content

Ausgabe: 01.03..2005
D:\CEM_V6\DOCU\English\Engineering\01_Introduction.doc

General

You have here the *Engineering Manual* for CEMAT V6. This manual should support you in performing the work required to configure your plant.

The *Engineering Manual* is part of a comprehensive CEMAT V6 complete documentation. In the current version V6, this consists of the volumes listed below.



After the installation of CEMAT V6 the CEMAT documentation is available as PDL in directory D:\Cem_V6\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	PLC-PLC Coupling to older CEMAT Versions
8	OS Engineering
9.	free
10	free
11	Project administration
12	free
13	Graphic Templates
14	Tips&Tricks
15	Update Information

	Reference - System
1	Introduction
2	
3	
4	System Description
5	
6	

	Reference - Objects
1	Introduction
2	General
3	Unidirectional Drive
4	Damper
5	Valve
6	Annunciation Module
7	UM-Module
8	Route Module
9.	Group Module
10	Selection
11	Silopilot
12	CNT-Module
13	RT-Module
14	PID Controller
15	Step Controller
16	Adapt
17	Measured Value Integrator
18	Bi-Directional Drive
19	Annunation Module with 8 Alarms
20	
21	
22	Adapt to Simocode
23	
24	
25	
26	
27	
28	Interlock
29	Interlock 5
30	

	Reference - Glossary
1	Introduction
2	Definitions and Instructions

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

Ausgabe : 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\02_Preparations.DOC

Verification of the Standard

Per plant the responsible engineering department must define the project definition. Henceforth these unique definitions are valid for all PLCs 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)
- 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 colour definition for the annunciations, Limit values, Status, Font etc.
- The communication (Communication type, Actualisation cycles)
- The volume of project data (Quantity of alarms, Archive values, Curves, Clients etc.)
- Message and archiving procedure

There are now various forms of the CEMAT Standard. Ensure that the project standard meets the requirements of your plant (MCC / plausibility, local switches, operation).

Caution: Subsequent changes in the function block parameters need a reload of the PLC. This means PLC Stop!

Verification of the received documents

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

- Object list
- 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 into PLCs

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

Determine the number of PLCs required

- Maximum number of objects per PLC
- Maximum number of signals per PLC
- 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, PLCs, I/O-Panels etc.)
- Electrical View (MCC)

Plant Designation System

Naming conventions for PCS 7

Depending on the language and PCS7 components, only certain characters are permitted in names. Make sure that you don't use illegal characters in your PCS7 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 PCS7 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.

Designation rules for CEMAT

Name of the PC Station

Special characters are not allowed in the PC Name. Better use only capital letters and numbers. Use upper case letters and the first character must be a letter.

Plant view

One of the most important decisions at the beginning of the engineering of a Cemat V6 Project is the structure of the plant view. You have to decide, if the name of the hierarchy folder should be included in the object tag or not.

It also has to be decided, which hierarchy level should be used as "OS Area". The OS Area is important for the Selections Buttons in the Overview Range and for the allocation of User Rights.

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: Please keep in mind that the CEMAT Message system can manage a maximum of 16 OS Areas at the moment!

Module tags

The designations of the modules depend on the settings in the plant view of the PCS 7 project.

The module tags contain at least the chart name and the instance name.

The maximum length of the module tags is 20 characters.

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).

The maximum length of the comment is 40 characters.

Example:

Motor Tag:	331_BC1/M01
Tag of the annunciation module for the rope switch:	331_BC1/M01_RS01

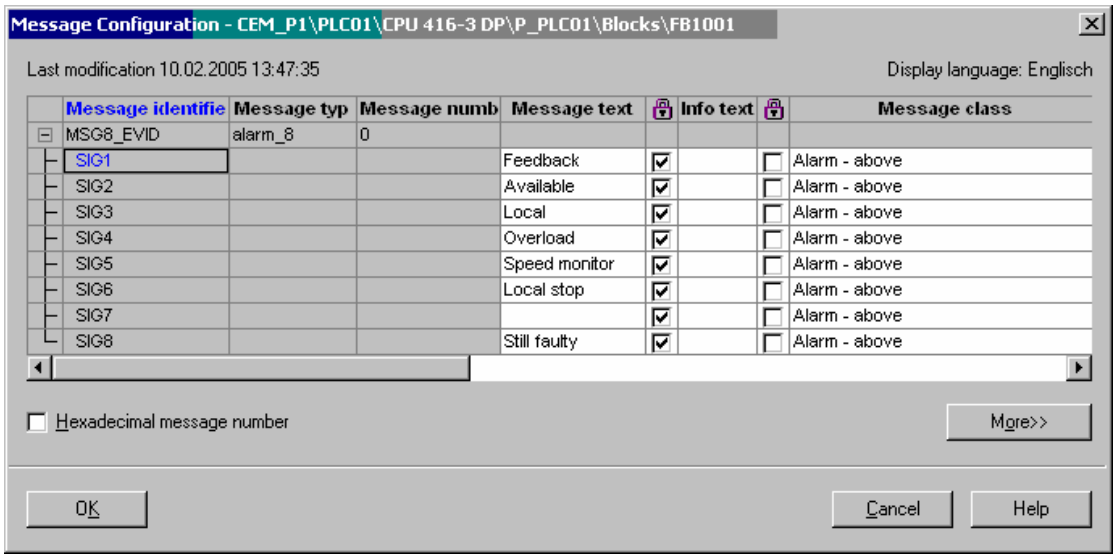
Signal names

The maximum length for global symbols in PCS 7 is 24 characters. The signal names für 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.

Fault Text (Message text)

The fault text is defined in the CFC at the function block and for the operation annunciations in the CFG file in directory D:\CEM_V6\CONFIG. For most of the standard function blocks there is a fixed fault text assignment which cannot be changed per instance.
You can adapt the message texts in the SIMATIC Manager under Special Object Properties.



For Annunciation modules C_ANNUNC and C_ANNUN8 the fault text must be entered per instance.

Definition of the signal identifications

Definition of unique abbreviations for all the signal types in the project.

Definition of 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 chapter „Engineering Examples“ of this manual you find solutions for special tasks.

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 colour 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 diagram overviews:

Representation:	Schematic representation of the diagram hierarchy with group states
Content:	Overview of the production plant with display of the operational groups/paths
Operation:	Selection of a process diagram

Grouping overview

Representation:	Tabular representation of the measured values, closed-loop controllers, curves
Content:	Overview of the grouping of the measured values, closed-loop controllers, curves either general or operator-specific
Operation:	Selection of a group diagram

Groups, routes, selection overview

Representation:	Tabular representation of the plant-specific groups, routes, selections
Content:	Overview of the states of the production plant
Operation:	Selection of the process diagram in which the groups, routes, selections are displayed

Process diagrams:

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 diagrams:

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

Designation System for the process pictures

Introduction

- **Overview area:**

  psu_xxzz.pdl = Process diagram - system overview

- **Overview diagrams:**

  pau_xxzz.pdl = Process plant overview

  pbu_xxzz.pdl = Process diagrams overview

  pmu_xxzz.pdl = Process measured values overview

  pru_xxzz.pdl = Process controller overview

  pku_xxzz.pdl = Process curves overview

  pgu_xxzz.pdl = Process groups overview

  pwu_xxzz.pdl = Process routes/selection overview

  pzu_xxzz.pdl = Process recipes overview

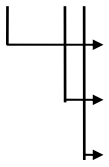
- **Process/Detail diagram:**

  pb_xxyzz.pdl = Process diagram

  d_ttxyzz.pdl = Detail diagram

Process Plant Overview

Designation: **pau**_xxzz.pdl



pau: Process plant overview

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

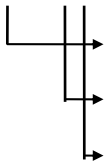
- Representation of the material flow for a complete production plant (without secondary aggregates and details).
- Display of all physical variables relevant for the process control.

Possible operating commands:

- Start and stop the plant,
- Specification of set points,
- Assignment of recipes
- Invocation of overviews for process diagram, measured value, controller or curve

Process Diagrams Overview

Designation: **pbu**_xxzz.pdl



pbu: Process diagrams overview

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

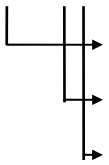
- Schematic representation of the diagram hierarchy with group states.
- Overview of the production plant with the display of the operational groups/routes.

Possible operating commands:

- Selection of a process diagram

Process Measured Values Overview

Designation: **pmu**_xxzz.pdl



pmu: Process measured values overview

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

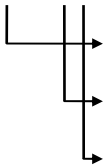
- Schematic representation of the measured value bar chart diagram hierarchy with the assignment allocation by the AKZ (plant identification)

Possible operating commands:

- Selection of a measured value bar chart diagram

Process Controller Overview Diagram

Designation: **pru**_xxzz.pdl



pru: Process controller overview diagram

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

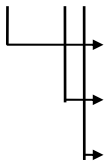
- Schematic representation of the controller bar chart diagram hierarchy with the assignment allocation by the AKZ (plant identification)

Possible operating commands:

- Selection of a controller bar chart diagram

Process Curves Overview Diagram

Designation: **pku**_xxzz.pdl



pku: Process curves overview diagram

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

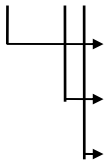
- Schematic representation of the curve diagram hierarchy with the assignment allocation by the AKZ (plant identification)

Possible operating commands:

- Selection of a curve diagram

Process Groups Overview Diagram

Designation: **pgu**_xxzz.pdl



pgu: Process groups overview diagram

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

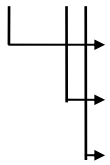
- Tabular representation of the plant-relevant groups with the group states of the production plant.

Possible operating commands:

- Selection of the process diagram that shows the group.

Process Routes/Selection Overview Diagram

Designation: **pwu**_xxzz.pdl



pwu: Process routes/selection overview diagram

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

zz: Serial diagram no.

Used for:

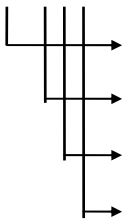
- Tabular representation of the plant-relevant routes/selections with the routes/selections-states of the production plant.

Possible operating commands:

- Selection of the flow diagram in which the routes/selections are represented.

Process Control Diagrams

Designation: **pb**_xxyzz.pdl



pb: Process control diagram

xx: Plant section identification (no. 1-99 or ZM, RM etc.)

y: Overview diagram no. 1-9

zz: Serial diagram no. 1-99

Used for:

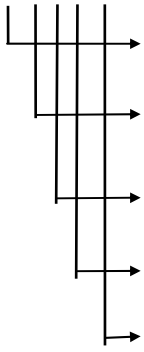
- Representation of the plant sections with all aggregates and
- Display of the operating modes, measured values and states.

Possible operating commands:

- Selection
- Group Start/Stop
- Operation of single aggregates
- Diagnostic, information and object parameter invocations

Process Detail Diagrams

Designation: **d**_tt x y zz.pdl



d: Process detail diagrams

tt: Plant section identification (no. 1-99 or ZM, RM etc.)

x: Overview diagram no. 1-9

y: Process diagram no. 1-9

zz: Serial diagram no. 1-99

If the detail diagrams are not to be nested, dialogs can also be configured instead of the diagrams.

Used for:

- Representation of single aggregates or subcontrols with operating modes, measured values and states.

Possible operating commands:

- Selection
- Group Start/Stop
- Operation of single aggregates
- Diagnostic, information and object parameter invocations

Bar Charts

System diagram with group selection mask

Curve Diagrams

System diagram with group selection mask

Installation of a PCS7 Project

Content

Ausgabe : 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\03_PCS7_Project.DOC

General

This chapter describes the installation of a PCS7 project on the Engineering Station (ES) and covers all the required settings for CEMAT based on PCS7 V6.

For general questions regarding PCS7 Engineering we refer to the Engineering Manual for PCS7 V6. These items are not included in the Engineering Manual for CEMAT.

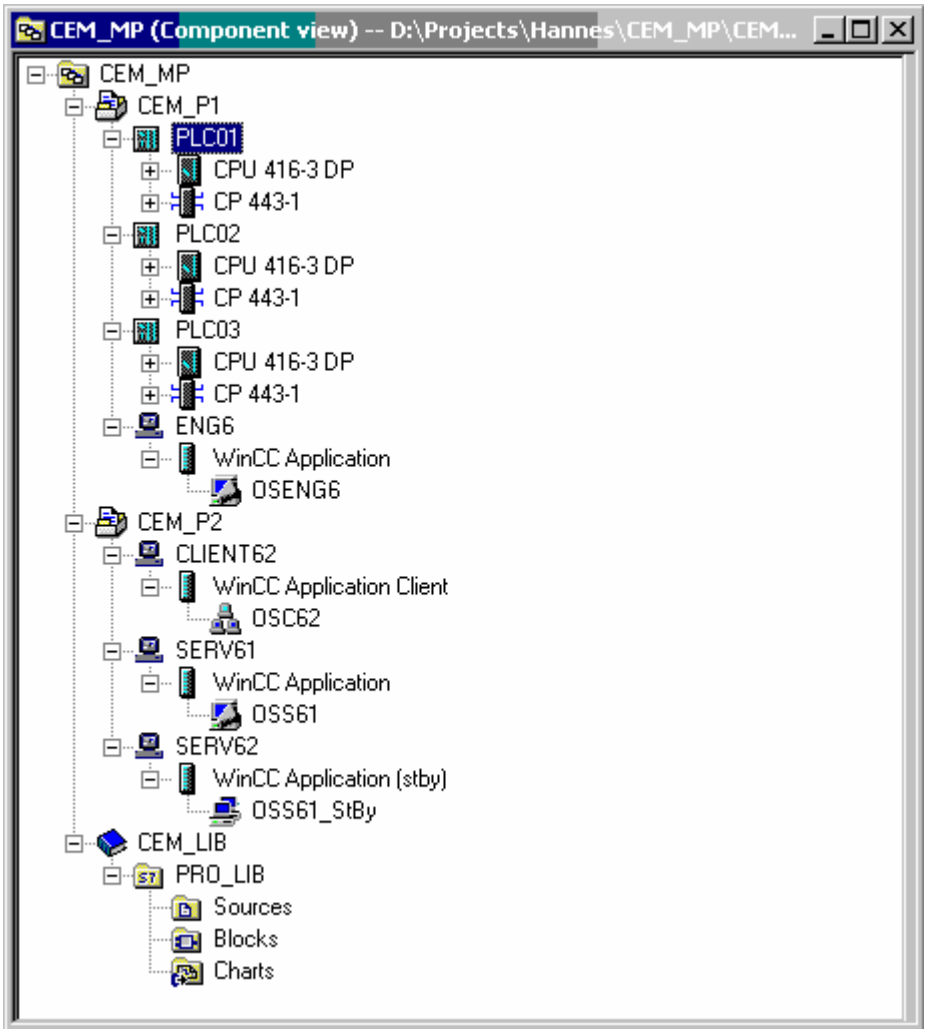
As the project structure will be different from Project to Project (depending on project size, number of objects, number of PLCs, Servers, OS Clients etc) this description can only be an example for one of many possible solutions.

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

3 PLCs
1 Engineering Station
1 redundant Server-Pair
1 OS Client.

The following example describes the engineering of a Multiproject **CEM_MP** with 2 Projects and a Library:

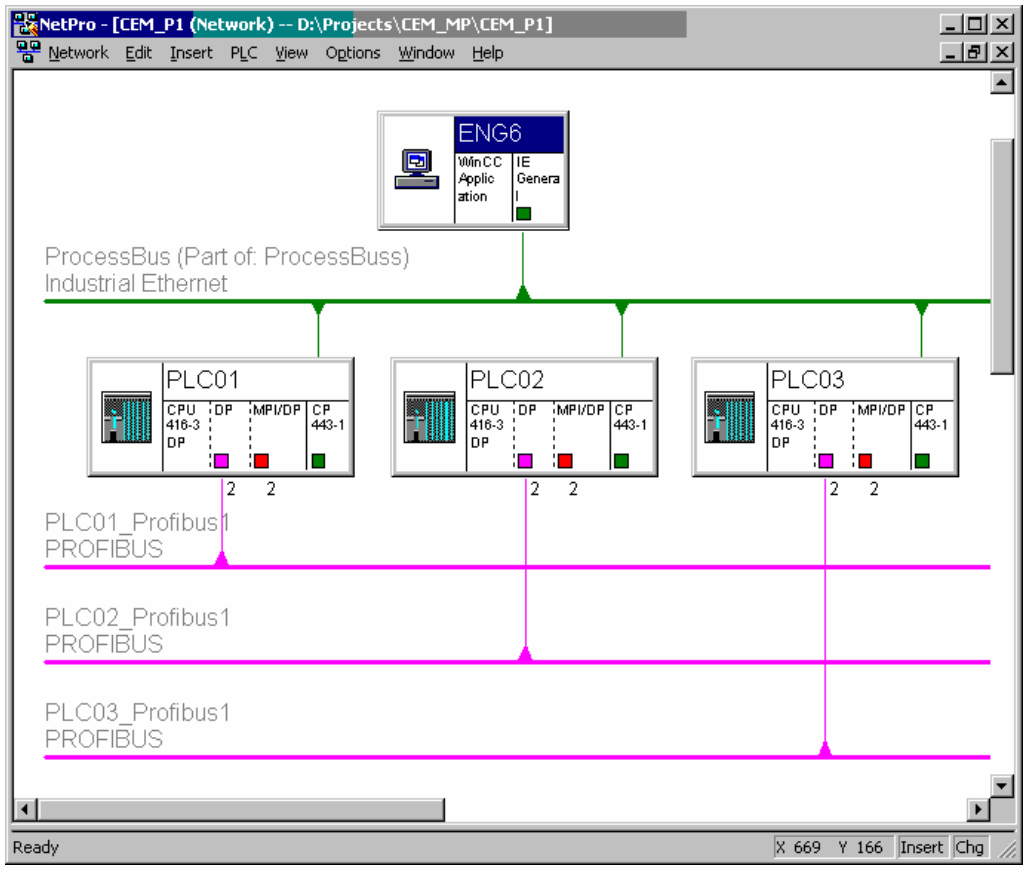
CEM_P1, including the PLC Stations and the ES
CEM_P2, including the redundant Server-Pair and the OS Client(s)
CEM_LIB, the Project Library



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

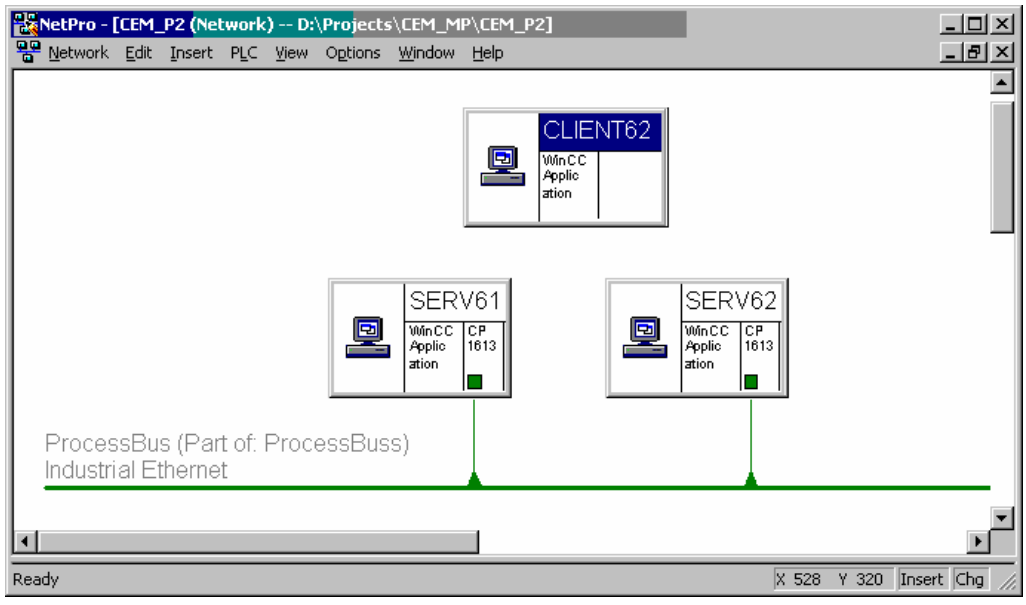
Caution: The Terminal bus and the corresponding CPs can not be displayed in the Network Configuration, otherwise there will be errors during download to the Configuration Editor.

Network Configuration for Project1 (CEM_P1)



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 Process bus can be configured!

Network Configuration for Project2 (CEM_P2)



To enable independent Engineering for the different PLCs, individual PCS7-Projects can be created for each Engineer.

There can be as many solutions as Projects exist and one of the most important steps at the beginning of the Engineering is to define the suitable project structure which fits to your requirements.

General engineering rules

Before you start creating your own PCS7 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 PCS7 V6 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.
  The only exception is the activation of the Time Synchronization for the Servers. If the Engineering Station has no CP1613, this selection is not possible on the ES and has to be carried out on the server 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 very important 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 (PCs and PLCs) get the correct time. In PCS7 V6 the PLC has always GMT!!

The following engineering manual explains all steps for the installation and configuration of the PCS7 Project, including PLCs, Engineering Station, Single Station, OS Server and OS Client:

For the Installation of PCS7 V6
→ see “Installation of PCS7” and the Engineering Manual of PCS7 V6

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 PCS7 for CEMAT”

For the creation of a PCS7 Project with PLCs and OS Stations (including Hardware Configuration and Network Configuration)
→ see “Create a new PCS7 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 PLC
→ see” PLC 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 PCS7

For each PC Station (ES, Server, Standby-Server and OS Client) the appropriate startup procedure must be carried out. Please follow the instructions in PCS7 V6 description and the readme file which is delivered with the PCS7 CDs.

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 PCS7 V6!

Screen Resolution

The best screen resolution for the CEMAT Faceplates and Pictures is 1280x1024. This is the recommended selection, especially for Single Stations and for OS Clients.

Power Options

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

Installation of the software for PCS7

Please, always use the actual installation instructions, which are delivered with the PCS7 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 „PCS7 PC Configuration and Authorization“.

Hardware definition with Commissioning Wizard

When you start your PC the first time after installation of PCS7 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 PCS7 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.

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

  000 = CEMAT
  006 = Dyckerhoff
  007 = Heidelberger Zement
  024 = Bushehr
  025 = Caima
  026 = Alsen
  028 = Rossi

- The Destination Folder for CEMAT is D:\CEM_V6
  Don't modify the destination directory from the default. To ensure that all CEMAT functions working correctly CEMAT must be installed under D:\CEM_V6!
- Select the correct Setup-Type:

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

- 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 CEM_V6.

D:\Cem_V6

\Bin	CEMAT IndustrialX Controls
\Bitmaps	CEMAT Bitmaps (zip-file)
\Config	Configuration Files
\Config_KCS	Configuration Files for KCS
\Docu	Documentation in PDF format
\Docu_KCS	Documentation for KCS in PDF format
\Multimedia	Documents for Info-Dialog
\Multimedia\AS	AS CAD Picture of Objects
\Multimedia\IO	In/output CAD Picture of Objects
\Multimedia\Manual	Manual for each Object
\Multimedia\Map	site plan
\Multimedia\MCC	MCC CAD Picture of Objects
\Multimedia\Sound	Sound-files for Infodialog
\Multimedia\Video	Video files of Objects
\Sounds	Sound files for alarms
\Tools	Software Tools for CEMAT
\ua	User Archive default
\WinCC	
\WinCC\GraCS	CEMAT Standard pictures and bitmaps
\WinCC\Library	WinCC project functions for CEMAT
\WinCC\Pas	cyclical WinCC project functions for CEMAT
\WinCC\Siemens	WinCC standard functions for CEMAT
\WinCC\WScripts	WinCC Scripts for CEMAT

The CEMAT libraries ILS_CEM and PRO_CEM (only available if project standard is not ‘000’) are installed in the PCS7 system directory and will be found after the installation under

C:\….\Siemens\Step7\S7libs

Adaptation in PCS7 for CEMAT

For all PC Stations (ES, Single Station, SERVER and CLIENT) the following steps have to be carried out:

1. Install the Fonts "wingdng3.ttf" under Control Panel -> Fonts -> "Install new Font“. You will find this fonts in directory "d:\cem_v6\tools\windng3.ttf"
2. Copy all files under d:\Cem_v6\WinCC\Siemens to c:\….\Siemens

Create a new PCS7 Project

Define a directory where all the PCS7 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.

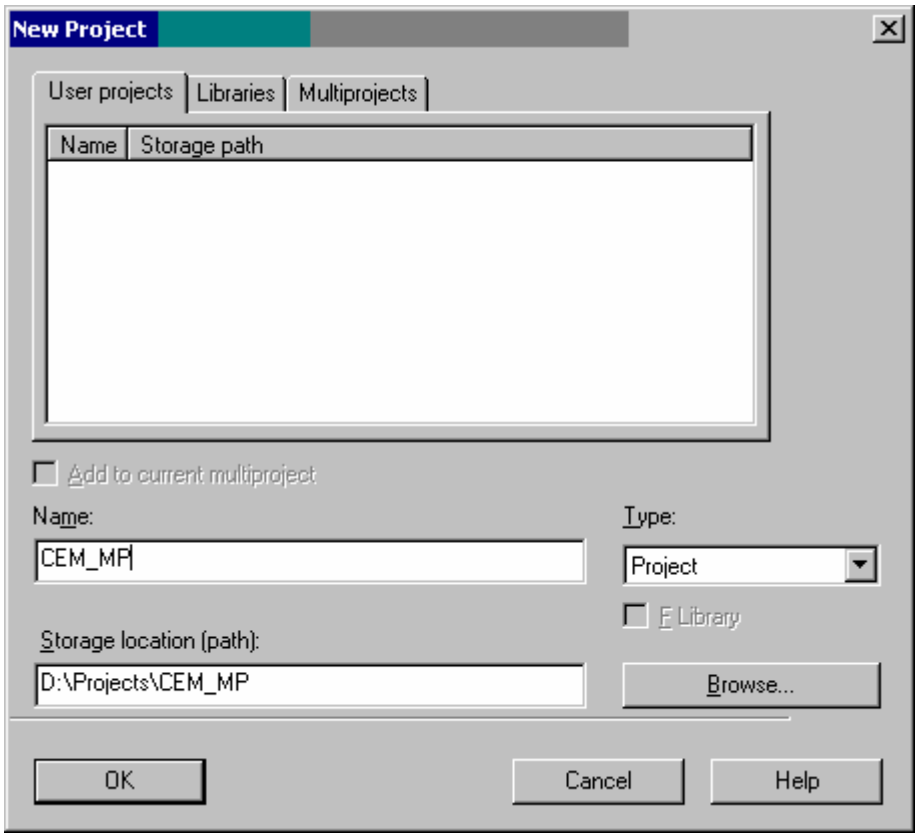
Use *Options -> Customize* to select the new directory under 'Storage location for projects'. Use directory D:\PROJECTS to save your PCS7 projects. If you use Multiprojects, create a Subfolder as Storage location for the Multiproject, e. g. CEM_MP

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:

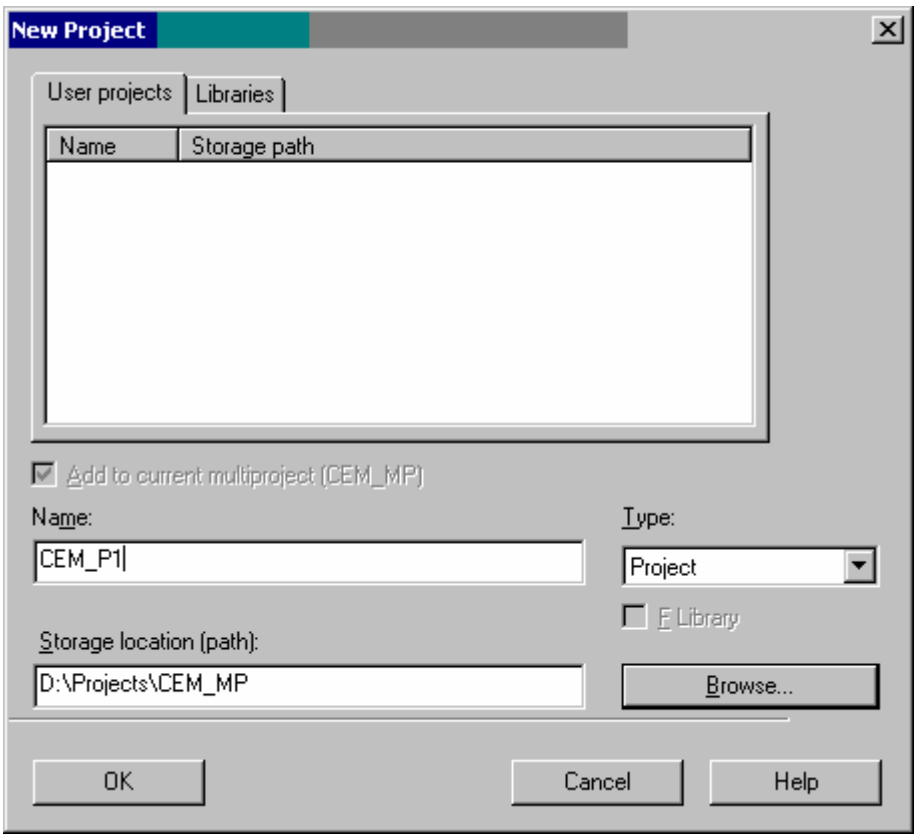
In the SIMATIC Manager create a new project using *File -> New.*

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



Confirm with OK

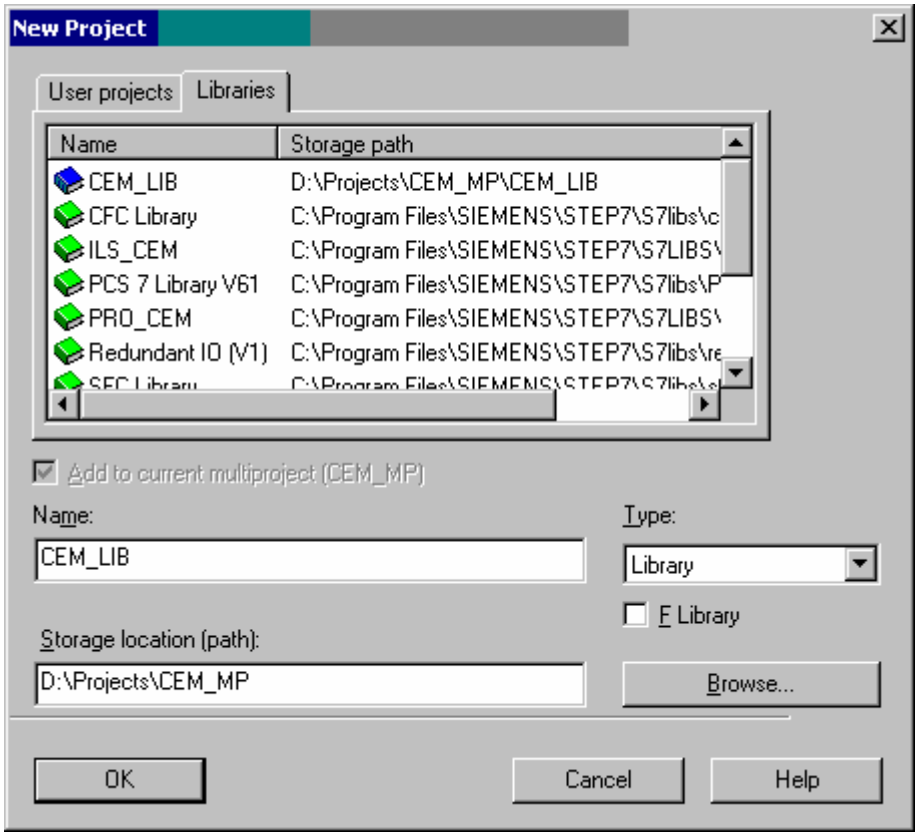
Select the Multiproject in the SIMATIC Manager and use *Multiproject -> Create in Multiproject* to create the first Project within the Multiproject.



Confirm with OK.

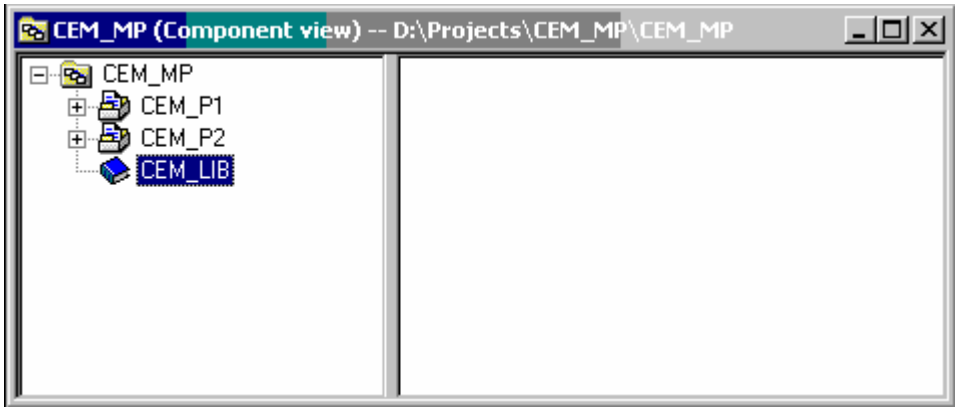
Repeat this step for all your Projects.

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 PLC-Program).

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



Once the Projects are created you have to create the PLCs and the PC Stations according to the Hardware Configuration of your plant. New PCs/PLCs 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.

After inserting a new station you have to give it a meaningful and unique name (e.g. CLIENT62, SERV61, PLC01).

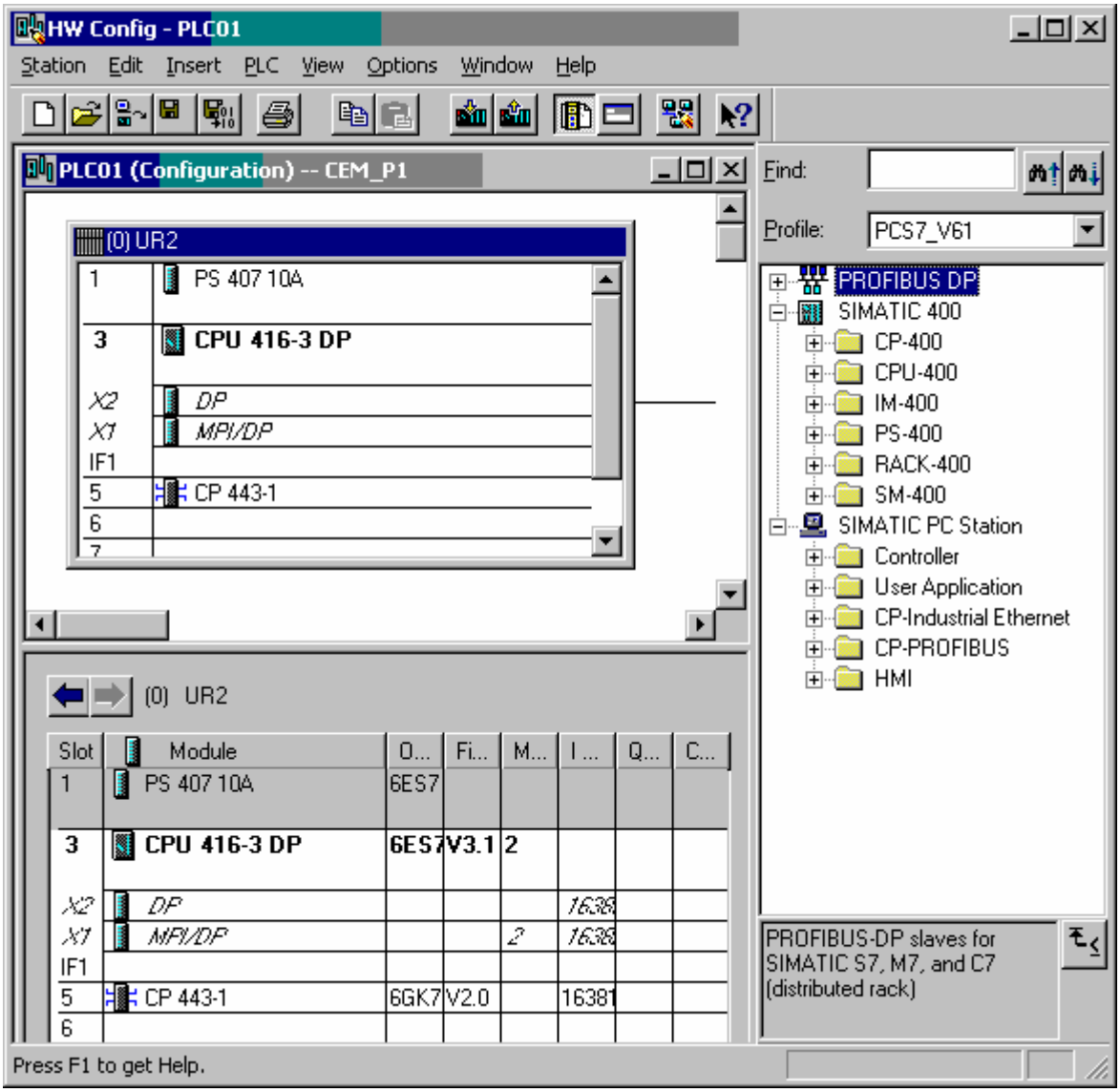
Hardware Configuration for a CEMAT PLC

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

The Screenshots in this description show the Properties of a CPU 416-3 DP and a CP443-1 in order to explain the required settings for CEMAT.
Other CPUs or CPs may have slightly different Property windows. To avoid misunderstanding the most important CEMAT settings are mentioned below each screen shot.

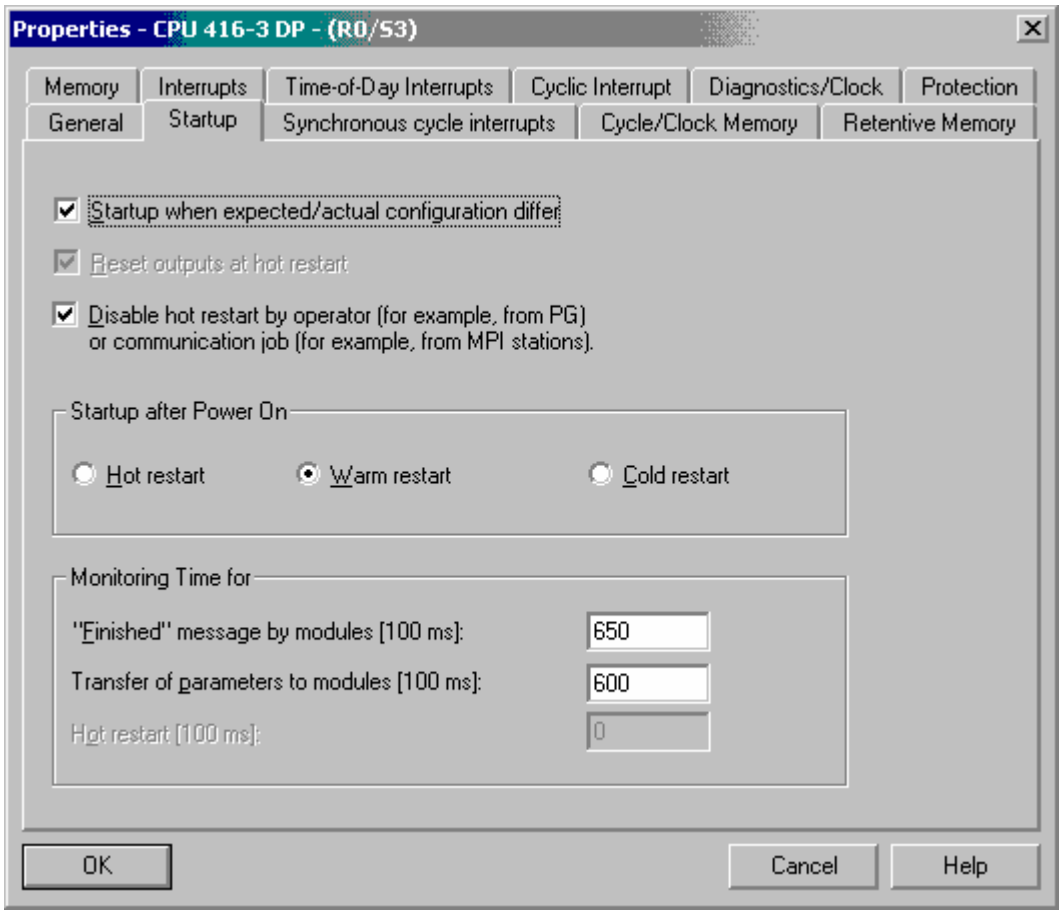
Define AS Hardware

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



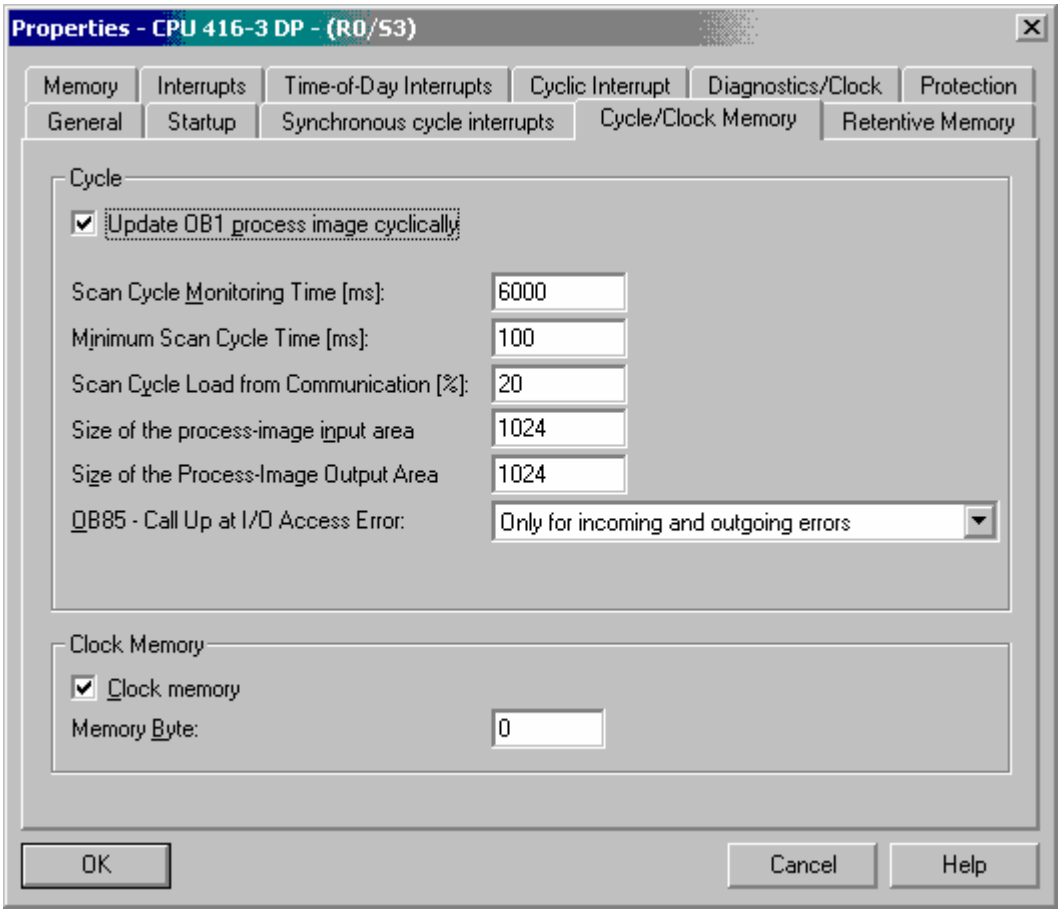
When dropping the CPU create a new PROFIBUS and connect it.

Settings in tab Start-up:



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

Settings in tab Cycle/Clock Memory:

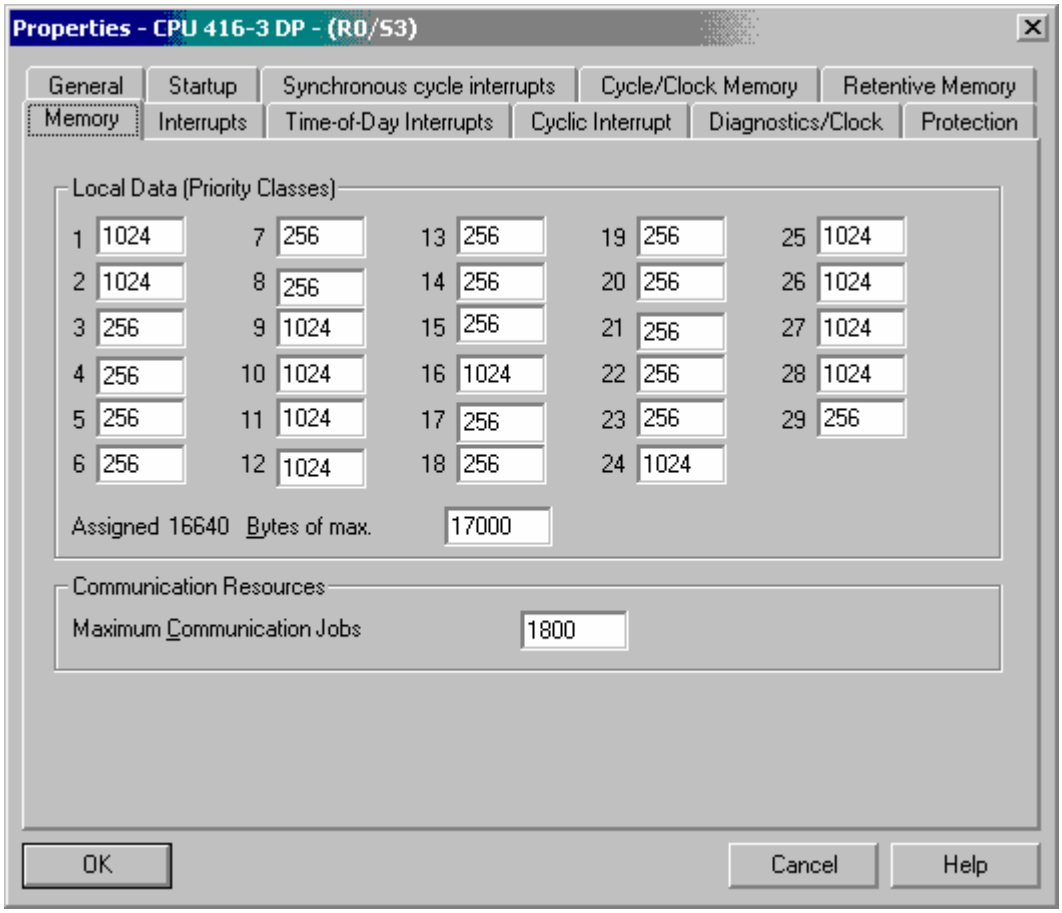


Enter the Scan Cycle Monitoring Time, minimum 2000ms.

Enter the Minimum Scan Cycle Time of 100ms

Select Clock Memory. The memory Byte must be 0

Settings in tab Memory:

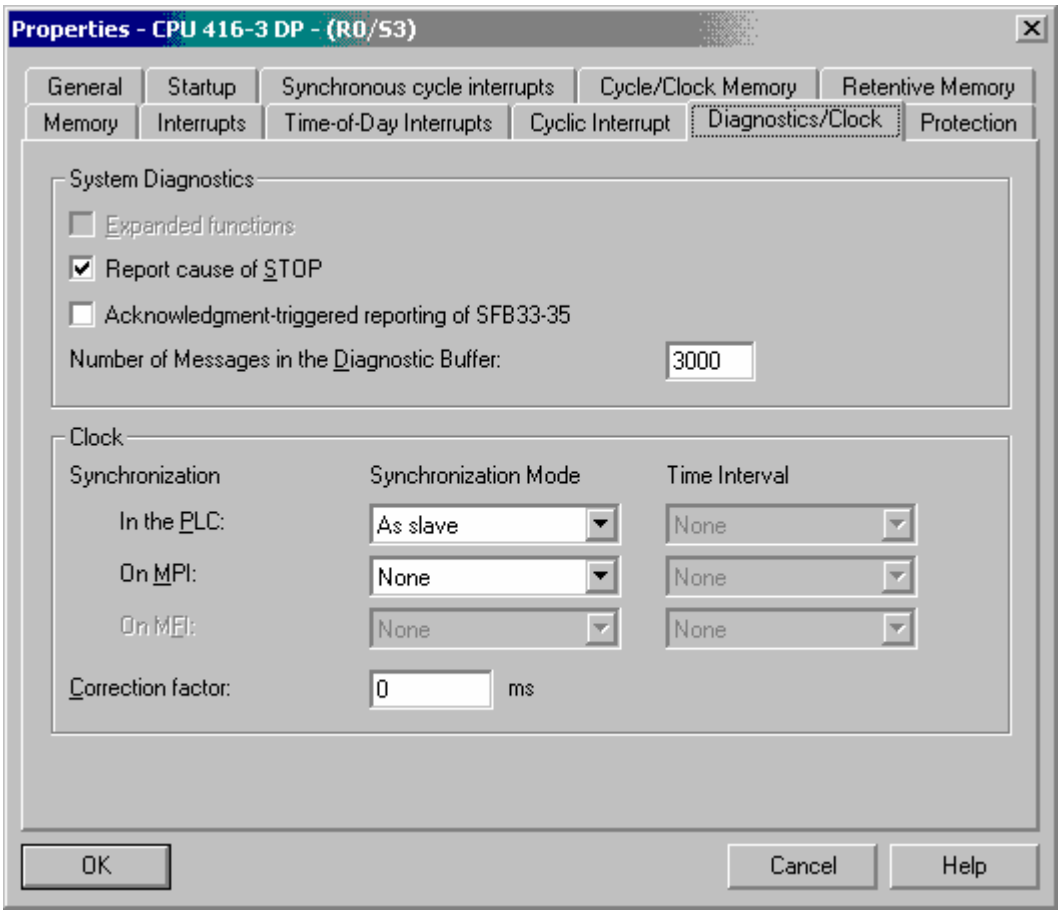


These are the minimum values for Local Data per each priority class. You have to check the required local data after completing the user program and modify the local data values if necessary.

Bigger CPUs probably have more memory for local data.

Please set the maximum Communication Jobs to 1800 if the type of CPU allows this.

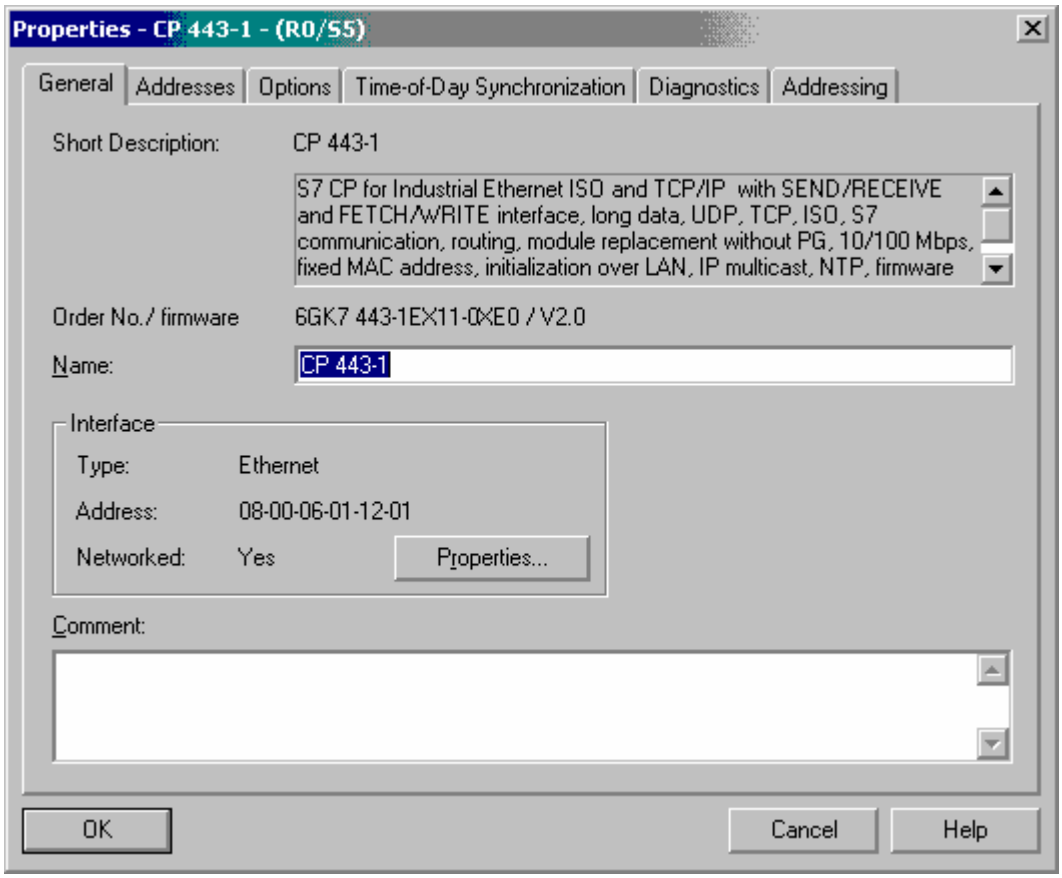
Settings in tab Memory:



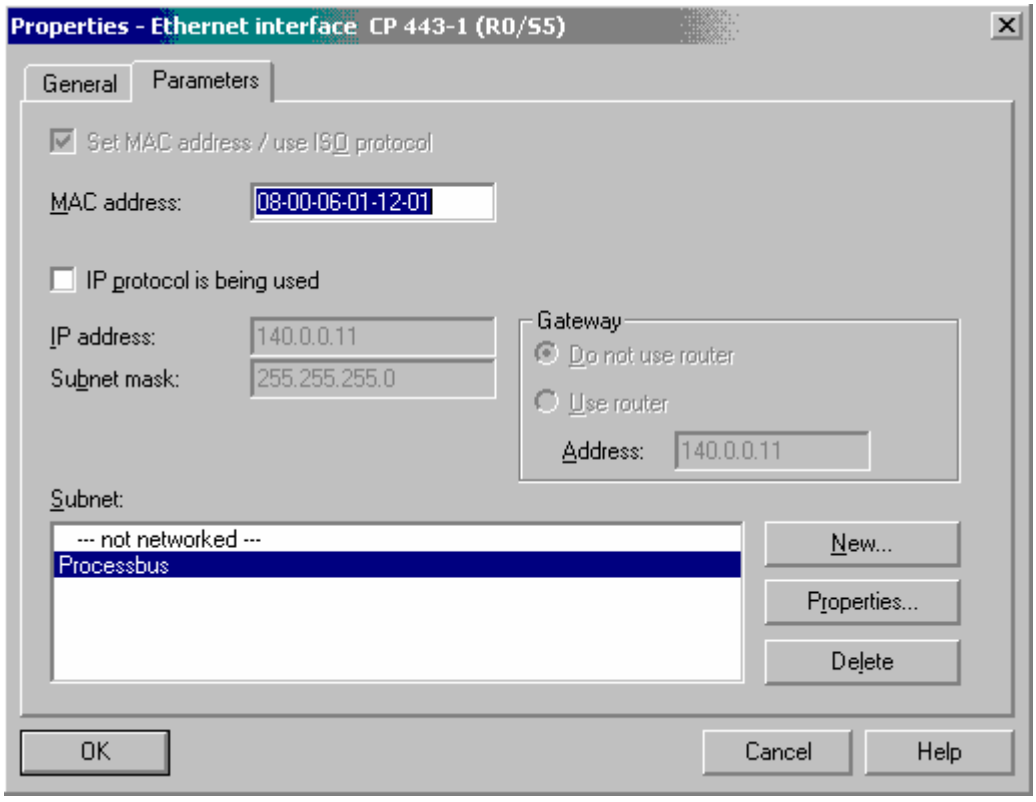
The PLC synchronization type should be “As Slave”

Note: In PCS7 V6 the PLCs always work with GMT.

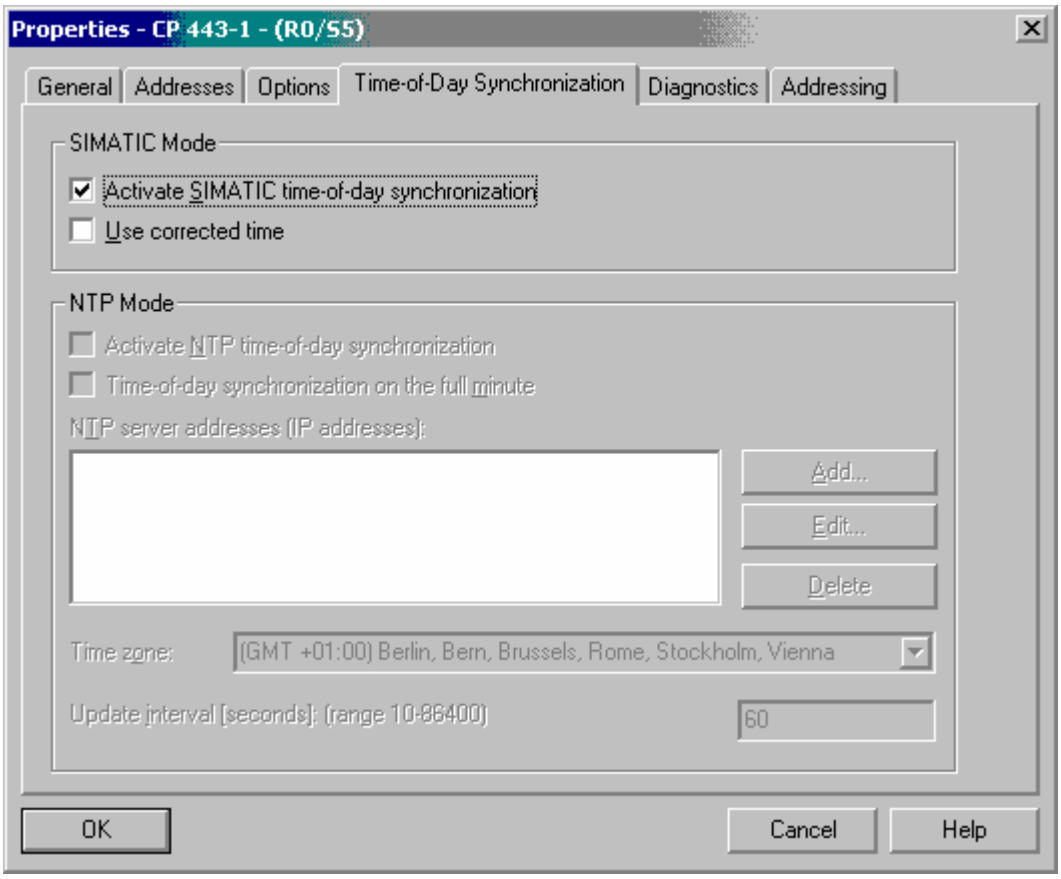
When dropping the CP443 connect the same to the Ethernet bus. If no Ethernet bus exists, create a new Ethernet bus and connect it to the CP443. Define the Ethernet Address for the PLC, e. g. 08.00.06.01.12.01 (MAC Address).



Properties for Ethernet interface:

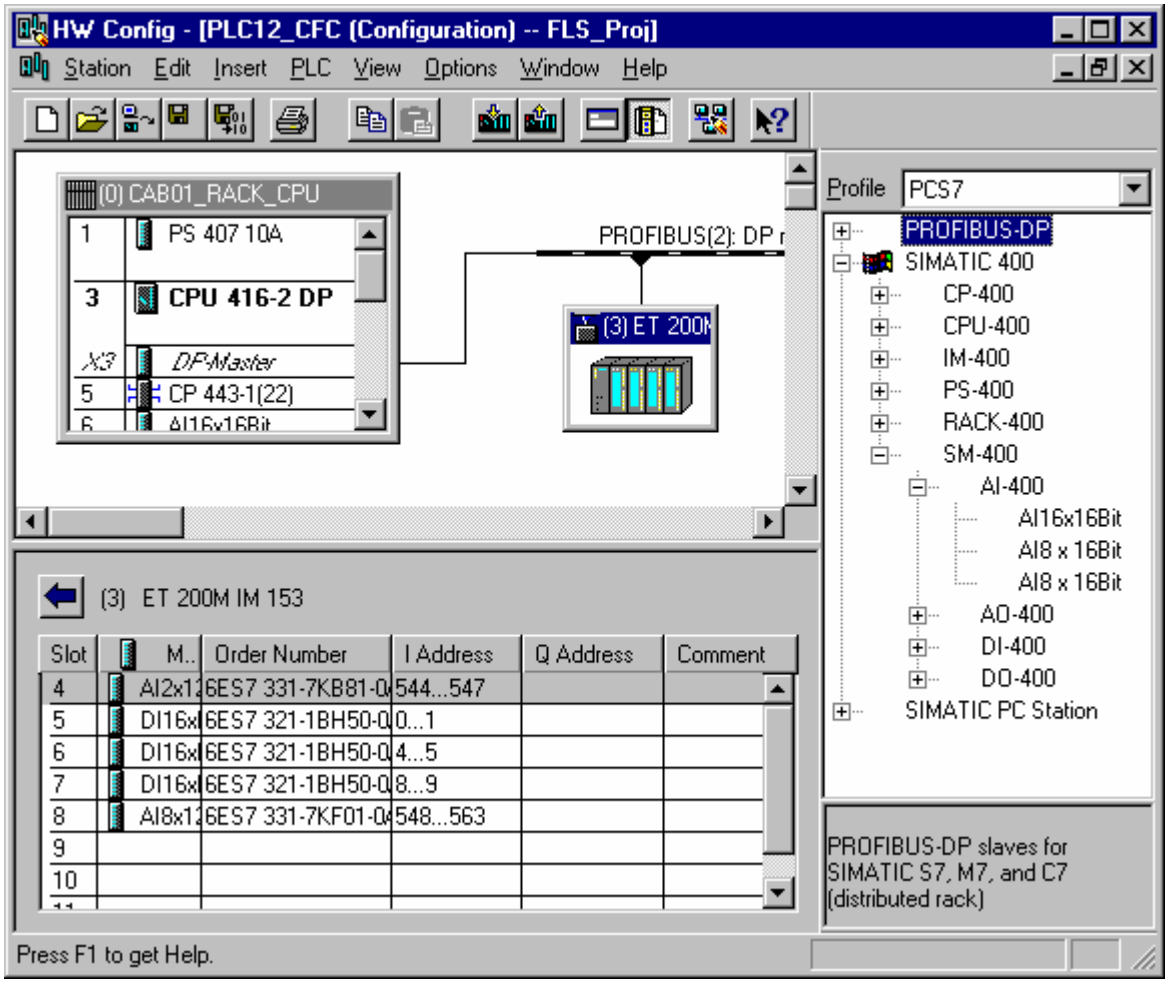


Under Options you have to activate the time-of-day synchronization:

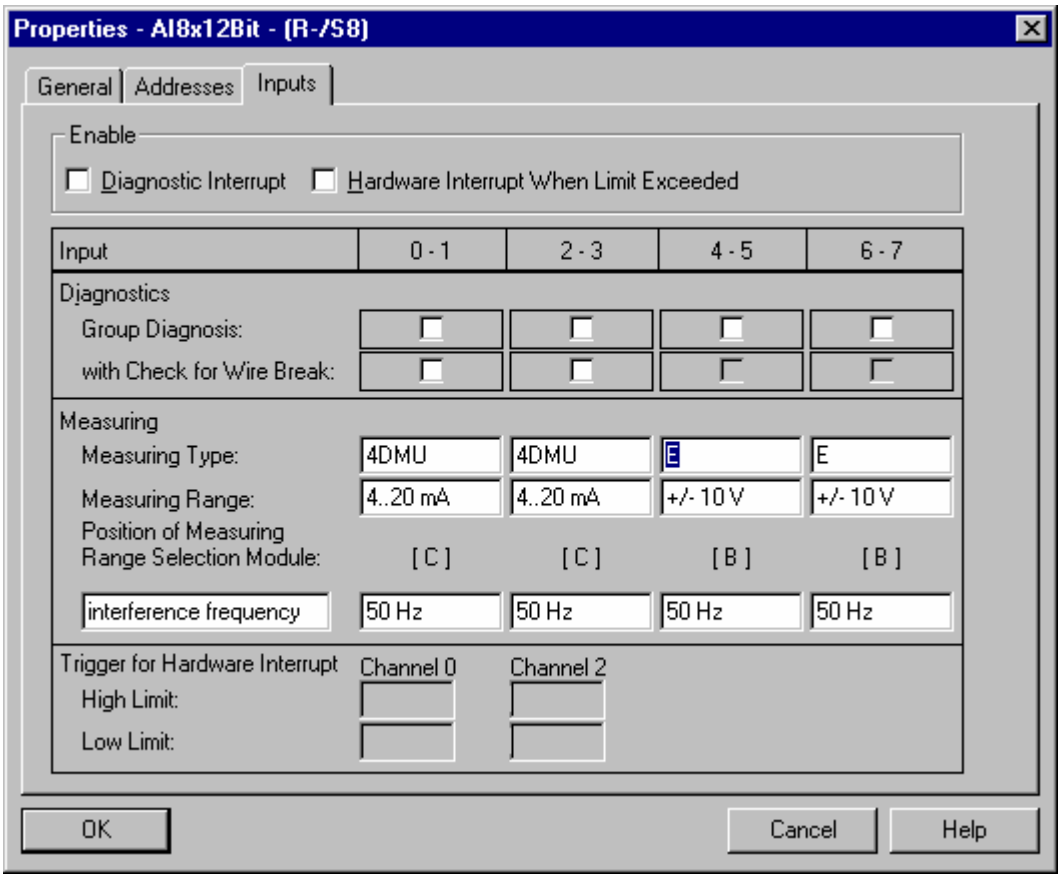


In PCS7 V6 the PLCs always work with GMT.

Define the I/O Periphery and parameterise:



Enter the properties for the analog input and analog outputs cards:



Save and Compile the hardware definitions and load it to the PLC.

Rename S7-Program

Back in the SIMATIC Manager you see the elements of the Hardware Configuration (CPU, CP) under each PLC 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_PLC03).

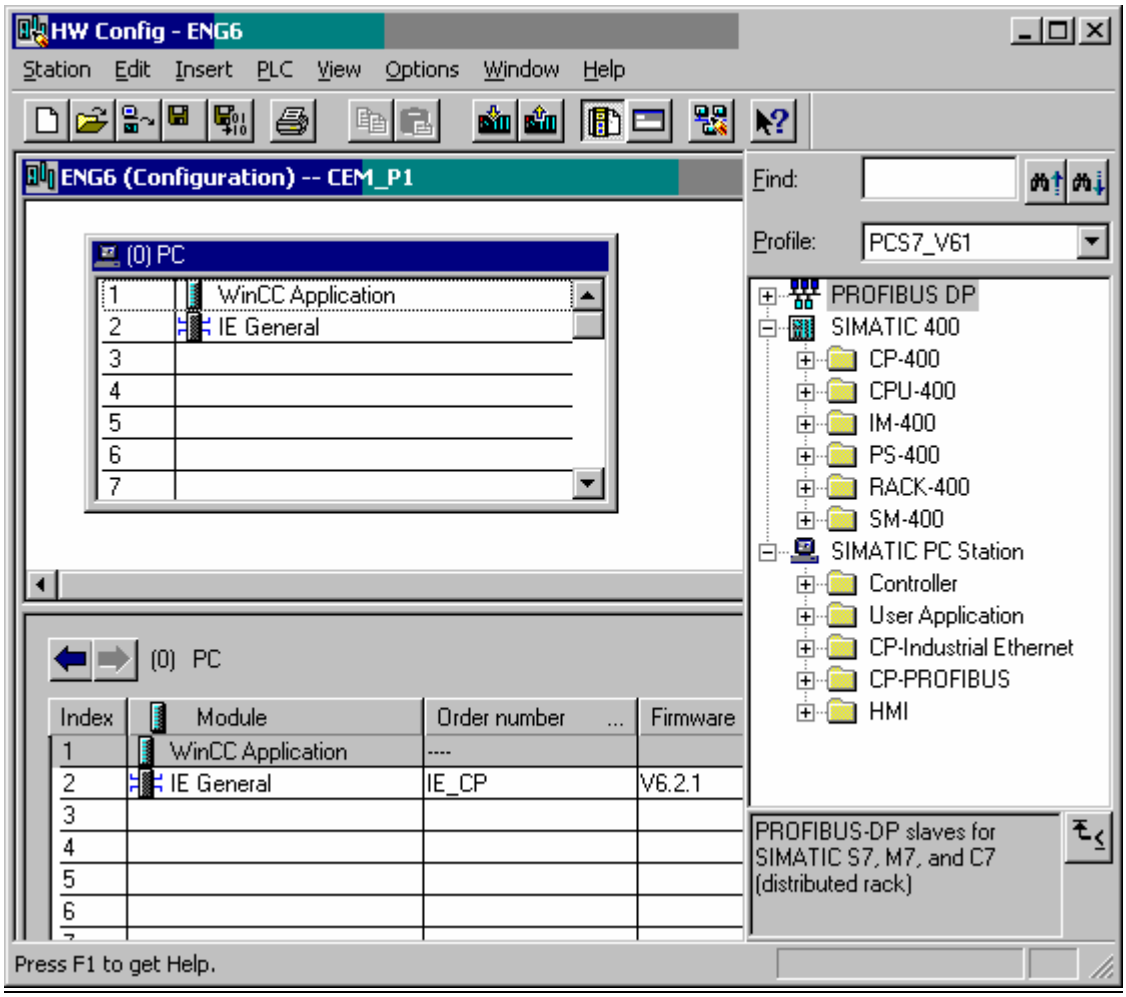
Configuration of a SIMATIC PC Station

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

Select the appropriate WinCC Application. Select the CP Industrial Ethernet for the Process Bus communication according to the Hardware of your PC (CP1613 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:

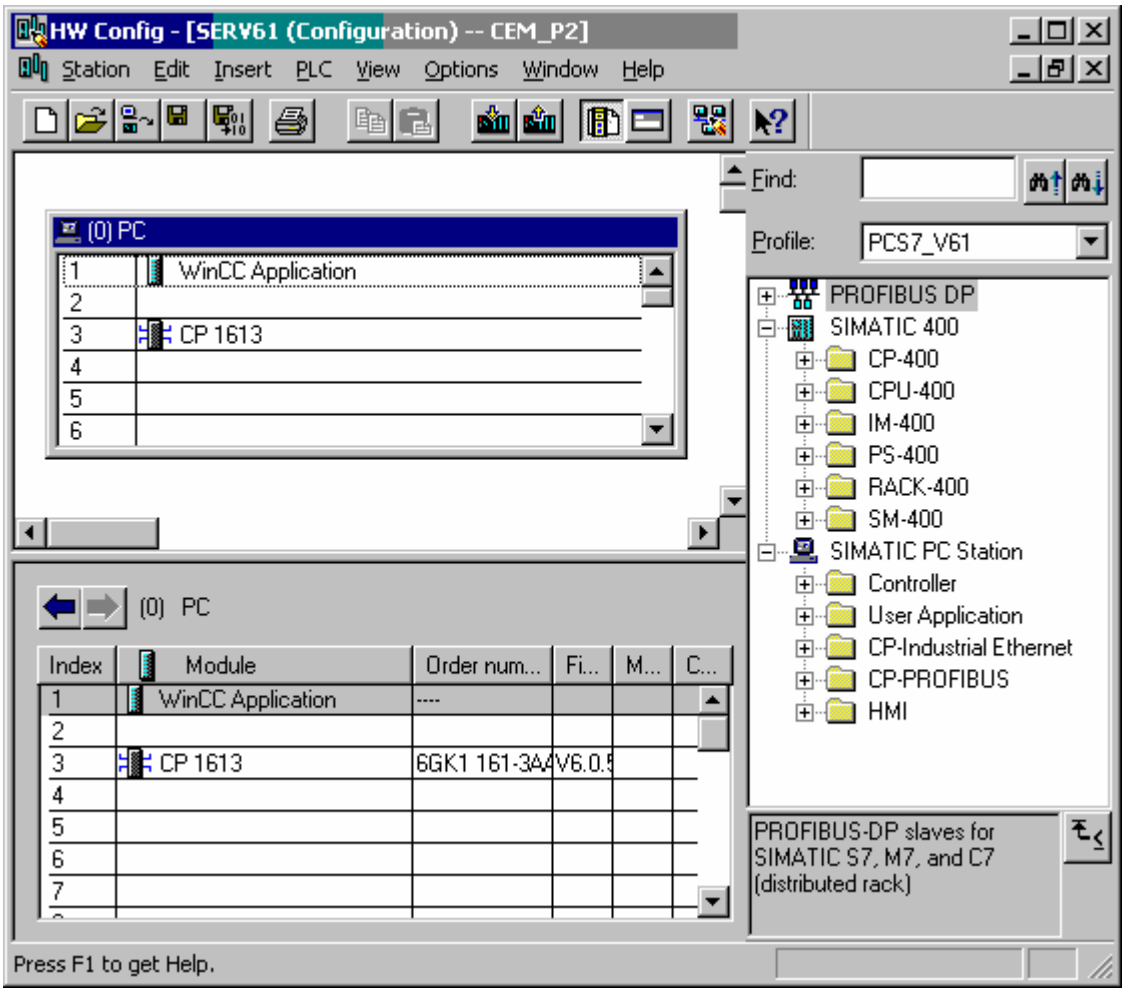


In this Example the Engineering Station has two CPs (one internal CP and one 3COM) but no CP1613. Instead of the CP1613 the 3COM is used for the communication to the process bus. In the Hardware Configuration only the CP for the connection to the process bus can be configured.

Please consider that, if the Engineering Station has no CP1613, in the Time Synchronisation the access point CP1613 can not be selected directly. You have to select the symbolic name of the access point.

When saving the HW Configuration the OS-Project for the ES is created.
Change the OS Project name according to the station name (e.g. OSENG6 for Station ENG6). Die OS Project name must be unique within the complete MultiProject.

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

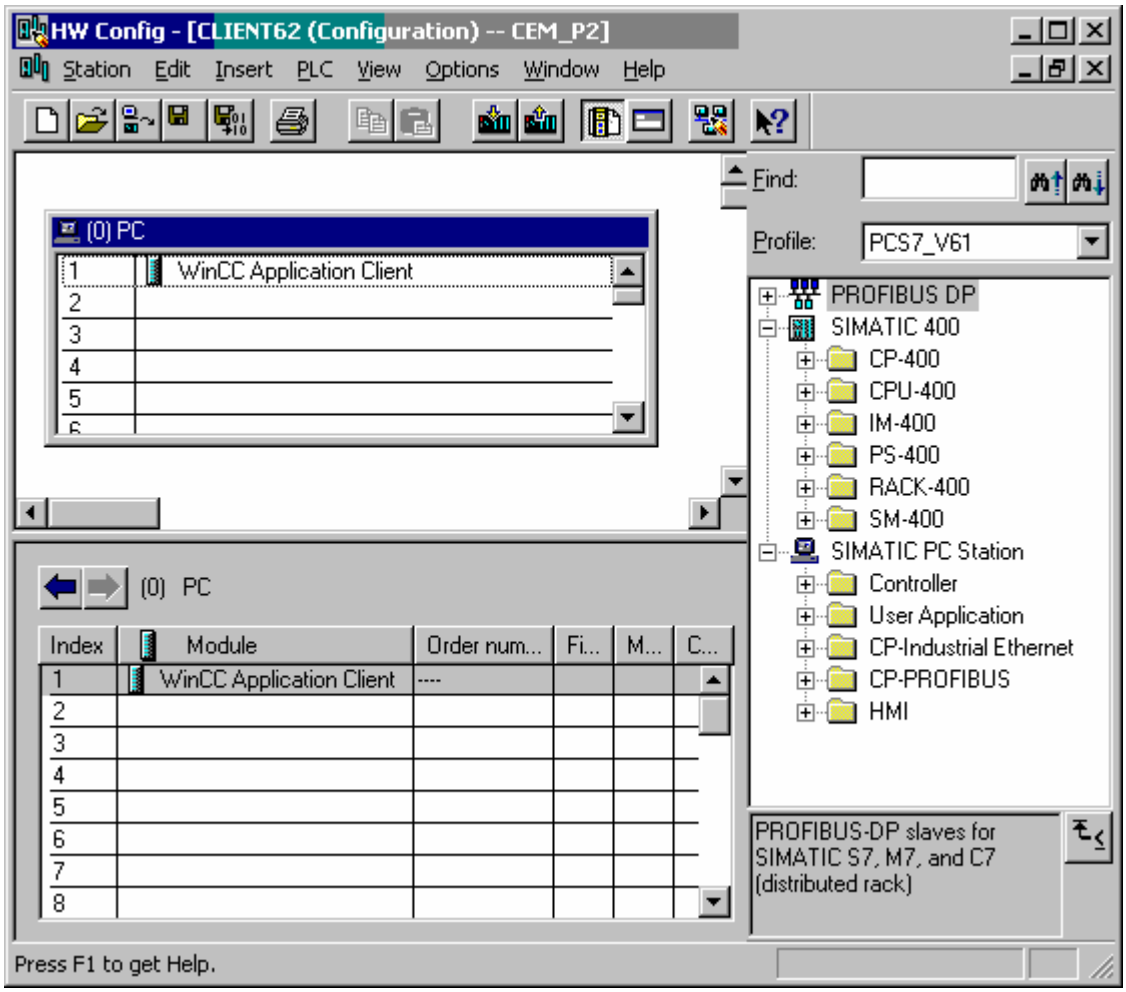


When saving the HW Configuration the OS-Project for the station is created.

Change the OS Project name according to the station name (e.g. OSS61 for Server Station SERV61). 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 according to the station name (e.g. OSC62 for OS Client Station CLIENT62). 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 PLCs and for the Communication between PLCs.

The settings for the Network Connections between the PLCs are described in Chapter PLC-PLC Communication.

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

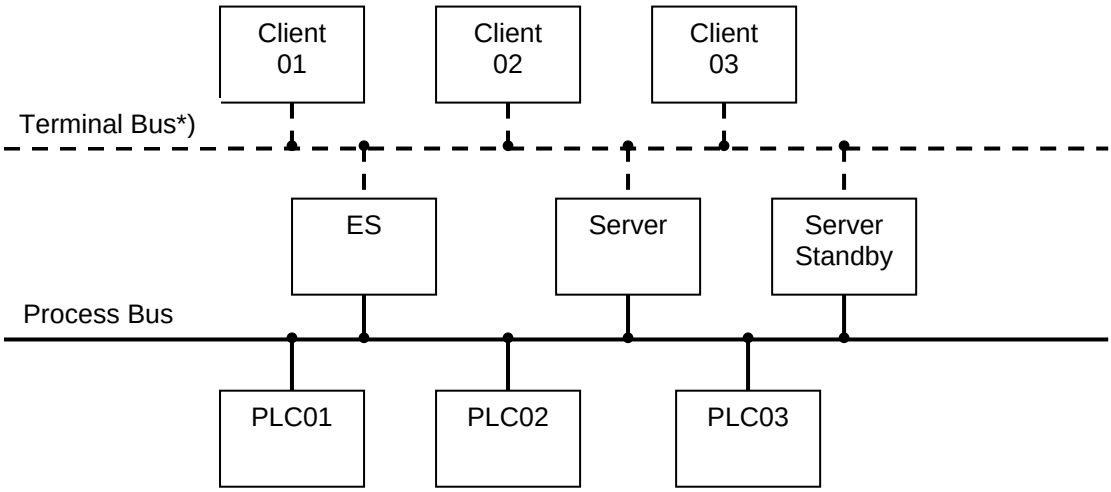
To define the connection, the CPs of the different Stations must be linked to an Ethernet Bus.

The CP of the PLCs is linked via “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 PLCs, 1 redundant Server-Pair, 1 ES and 3 OS Clients:

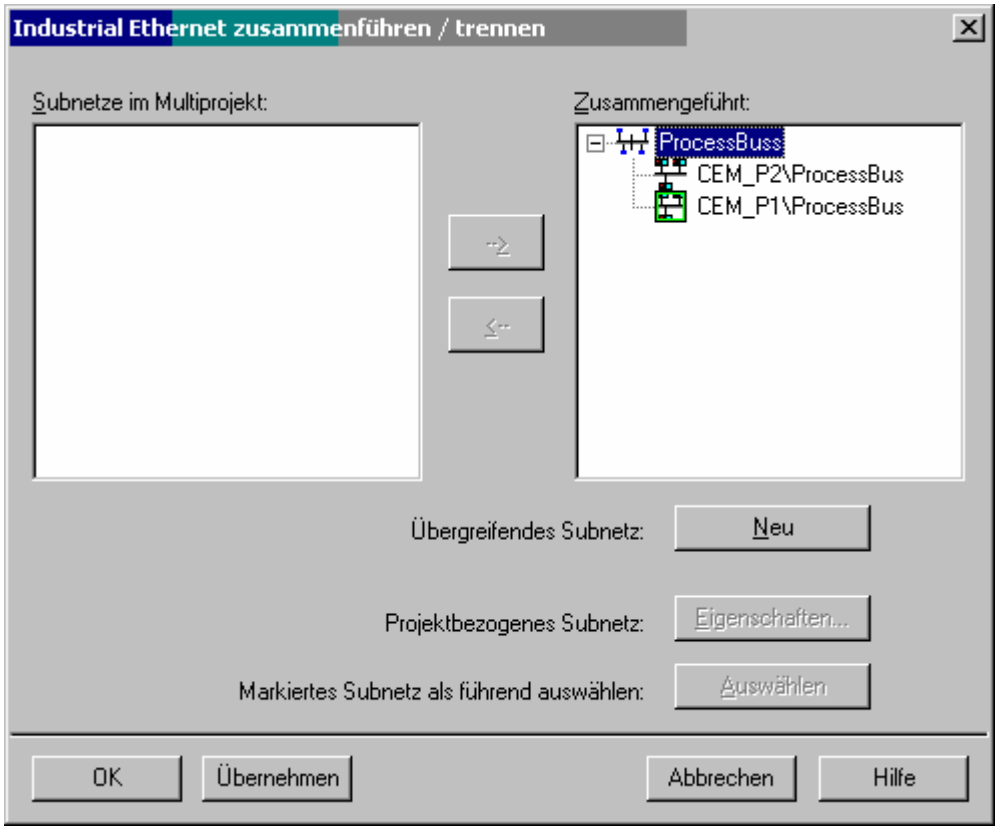


*) The CPs for the Terminal bus communication can not be displayed in the Network Configuration of PCS7, otherwise the Hardware cannot be downloaded to the PC Stations. For the communication to the Terminal bus no configuration in Netpro is needed.

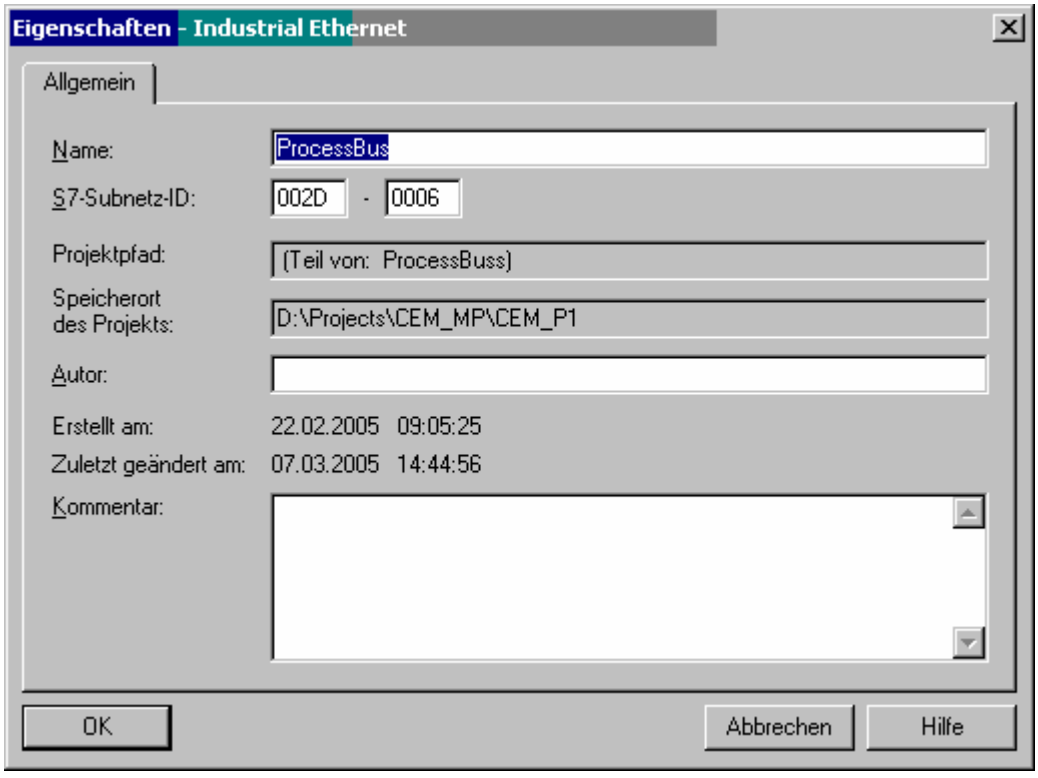
In a Multiproject the Stations are often located in different Projects. Each Project has it’s Network Configuration which contains Ethernet Connections to Process bus or Terminal bus.

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.

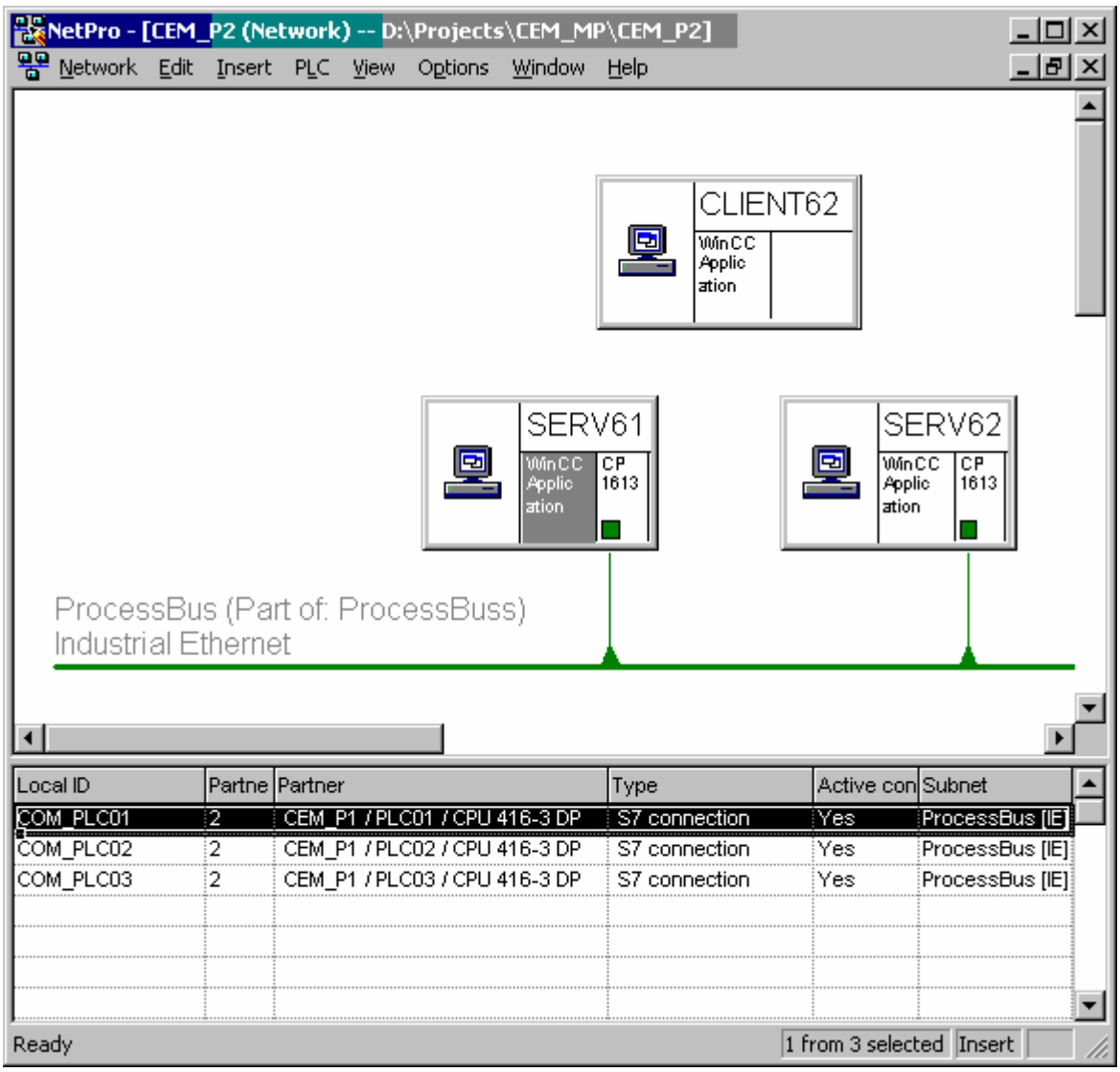
Select Ethernet and press Execute and the Window for Industrial Ethernet merge/unmerge will open and allow you to combine the Ethernet buses from different Networks.



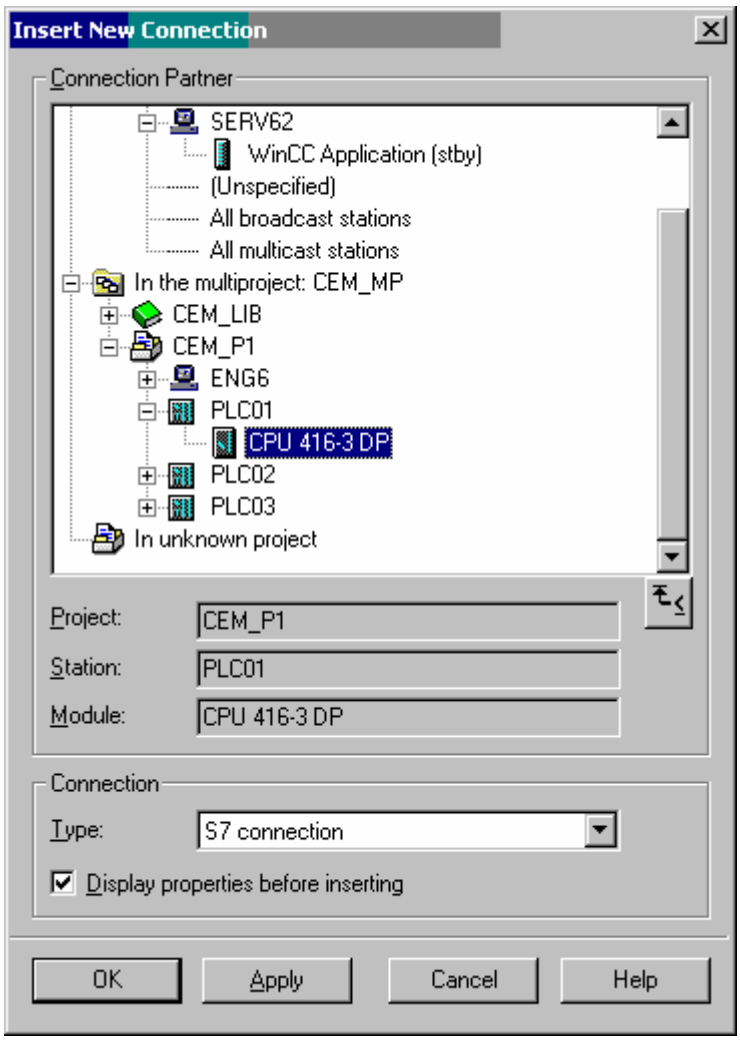
Under Properties you will find the S7 Subnet ID:



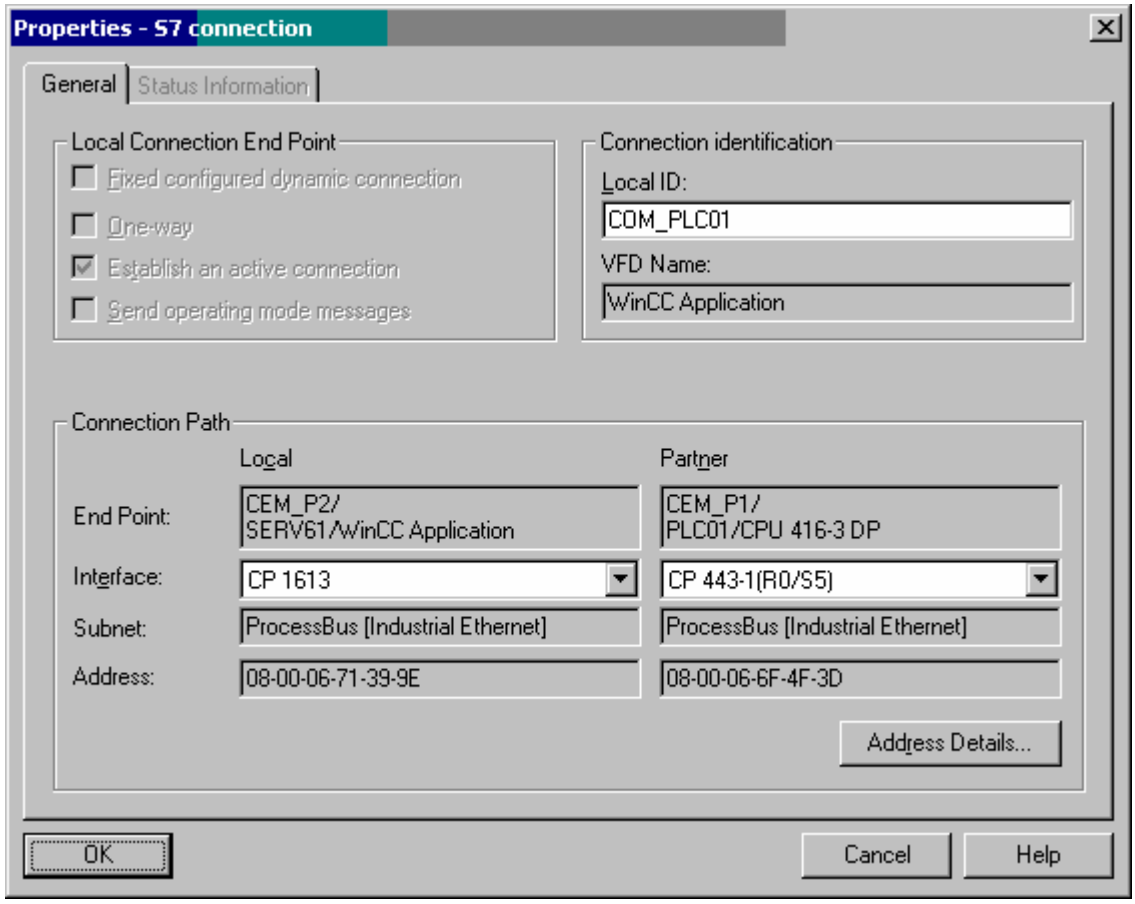
After the Network is merged the Network Connections between OS Stations and the PLCs can be defined.



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



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



We recommend to change the Connection identification for Local ID from S7 connection_1 into a more meaningful name, e. g. COM_PLC01.

Note: The Connection identification for Local ID must be identical for the connection of all OS-Projects to this PLC.
Example:
COM_PLC01 for the Communication to PLC1
COM_PLC02 for the Communication to PLC2
COM_PLC03 for the Communication to PLC3
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 PLCs.

Download for HW Configuration and Network Connections

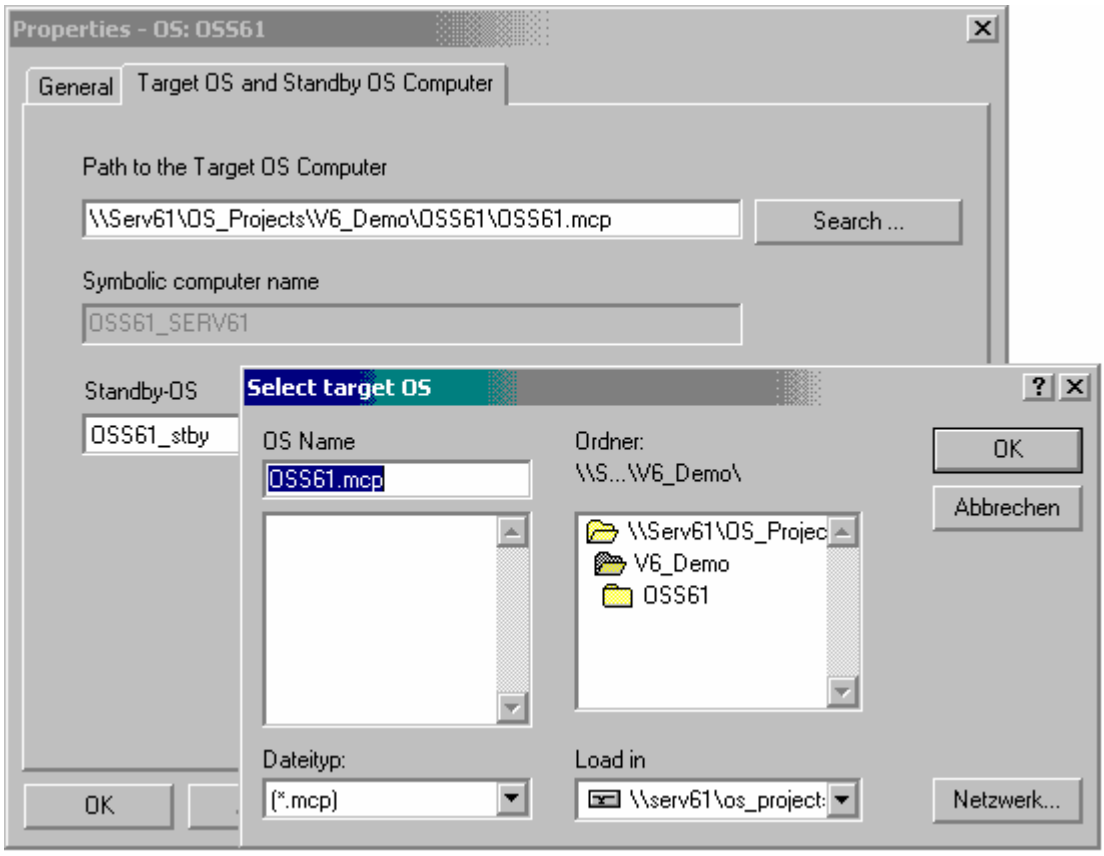
After the Station Configuration is finished and the Network Connections are complete the Configuration Data has to be transferred to the PLCs and to the OS Stations.

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 PLCs and to the other PC Stations.

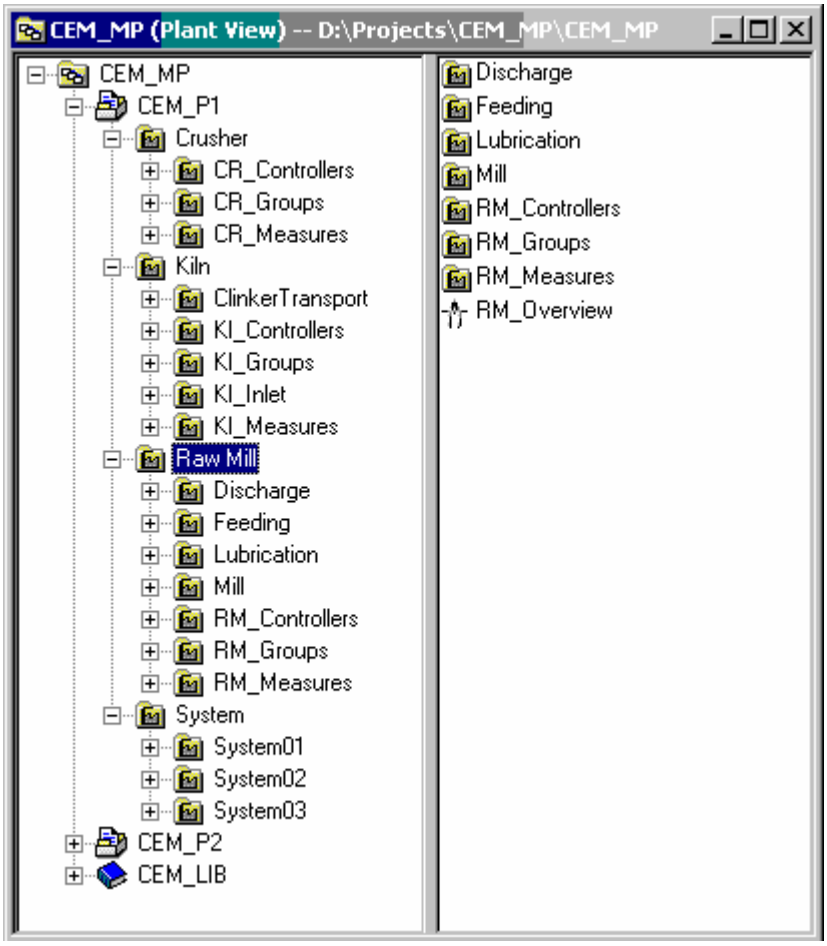
   Check the Configuration Editor to be sure that everything is configured correctly!
3. Download the Hardware Configuration to the PLCs.
4. Download the Network Configuration to the PLCs
5. Download the Hardware Configuration for each PC Station (Single Station, Server, Standby-Server) and double-check 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.

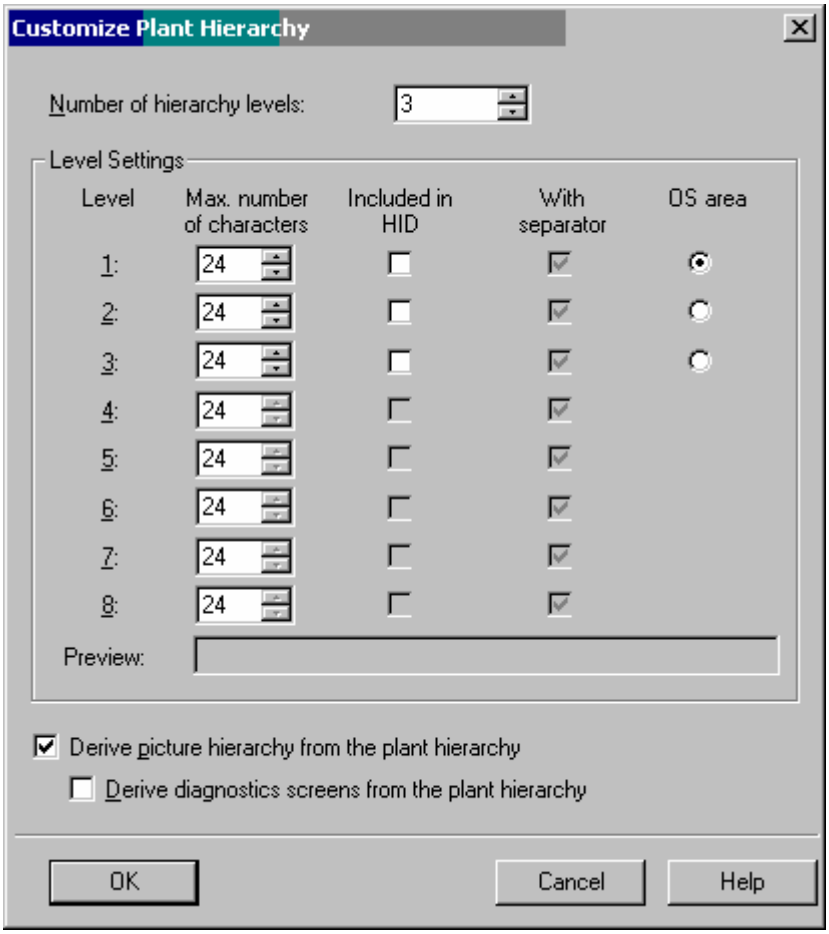
For the technological structure the plant designation system and the message philosophy are very important. Refer to point “Designation rules for CEMAT” in chapter 2 of this manual (Preparations).

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 -> Customize.*



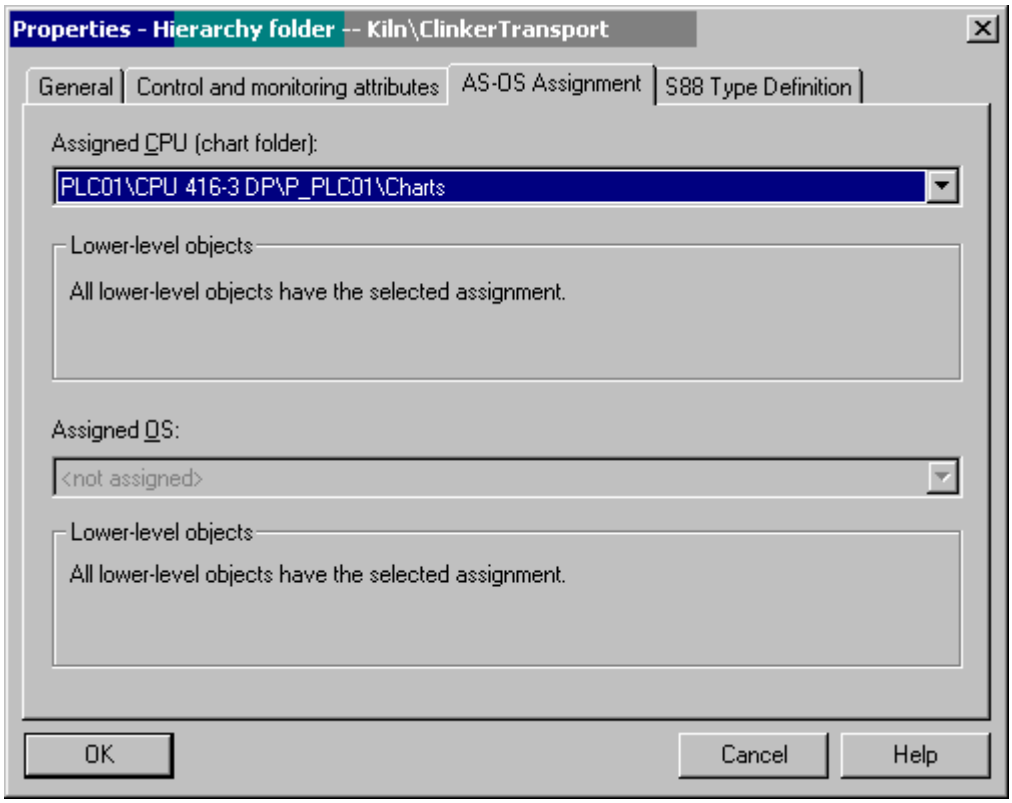
For a Cement plant 3 hierarchy folders should be (more than) enough.

The OS Area is used for the authorisations. The rights can be enabled and disabled area specific.

If you check the “Derive picture hierarchy on the plant hierarchy” check box, the OS picture hierarchy is derived entirely from the configured PH data. When it is subsequently transferred to the OS, any picture hierarchy configured in WinCC using the Picture Tree Manager is deleted and overwritten with the data generated in SIMATIC Manager.

If you want to derive the diagnostic screens from the plant hierarchy, you have to select the function. In this case 3 hierarchy levels are not sufficient. For the diagnostic of the hardware up to 8 hierarchy levels are required.

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 PLC. 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 *PLC and OS assignment* to select the chart folder for the assigned PLC and the assigned OS.



All Elements you add to this hierarchy folder will be assigned to this PLC/OS. If you pass the Selected Assignments to the Lower-level Objects, the subordinated hierarchy folders get the same PLC/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 PLC configuration, you should create a project library which contains all symbols, blocks and charts which are used in the project.

The project library contains at least all CEMAT symbols, blocks and charts. 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 as well as the S7-Program KCS, containing the Technological Functions for the CEMAT Kiln Control System.
- 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.)

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.

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.

PLC Configuration

Copy Standard Symbols and blocks into the PCS7 Project

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

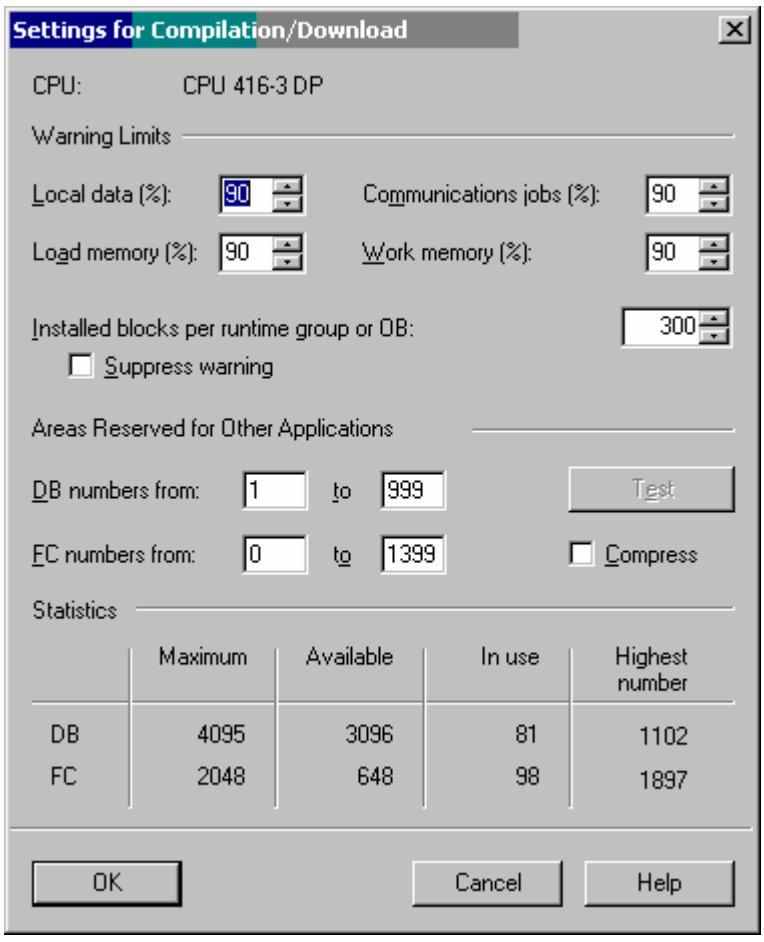
- Copy first the symbols from the project library into the S7 Program Container of your PLC.
- Copy all blocks from the project library into the block Container of your PLC.

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 PLC 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 -> Compilation/Download* you have to define the illegal ranges:

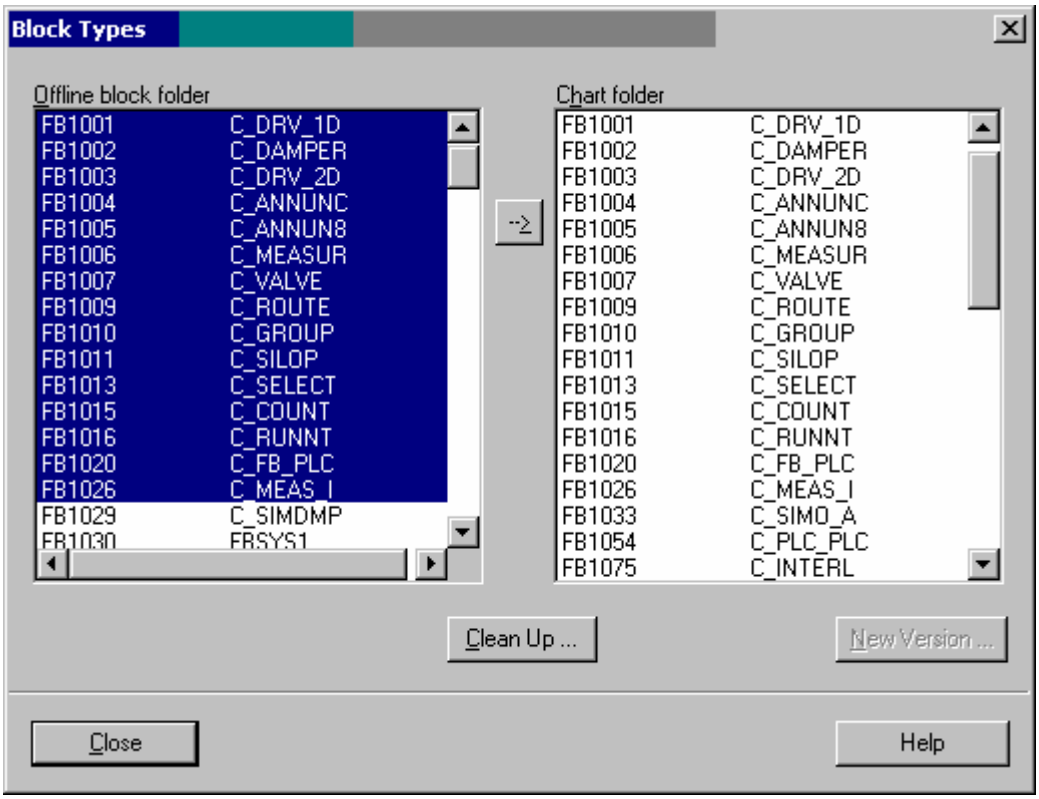


DB 1 to 999
FC 0 to 1399

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).

Copy the CEMAT blocks into the chart folder:

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
FB1009	C_ROUTE	OB1
FB1010	C_GROUP	OB1
FB1011	C_SILOP	OB1
FB1013	C_SELECT	OB1
FB1015	C_COUNT	OB35
FB1016	C_RUNNT	OB1
FB1020	C_FB_PLC	OB1
FB1026	C_MEAS_I	OB1
FB1033	C_SIMO_A	OB1
FB1052	C_PLC_SEND	OB1
FB1053	C_PLC_RECEIVE	OB1
FB1054	C_PLC_PLC	OB1/OB35

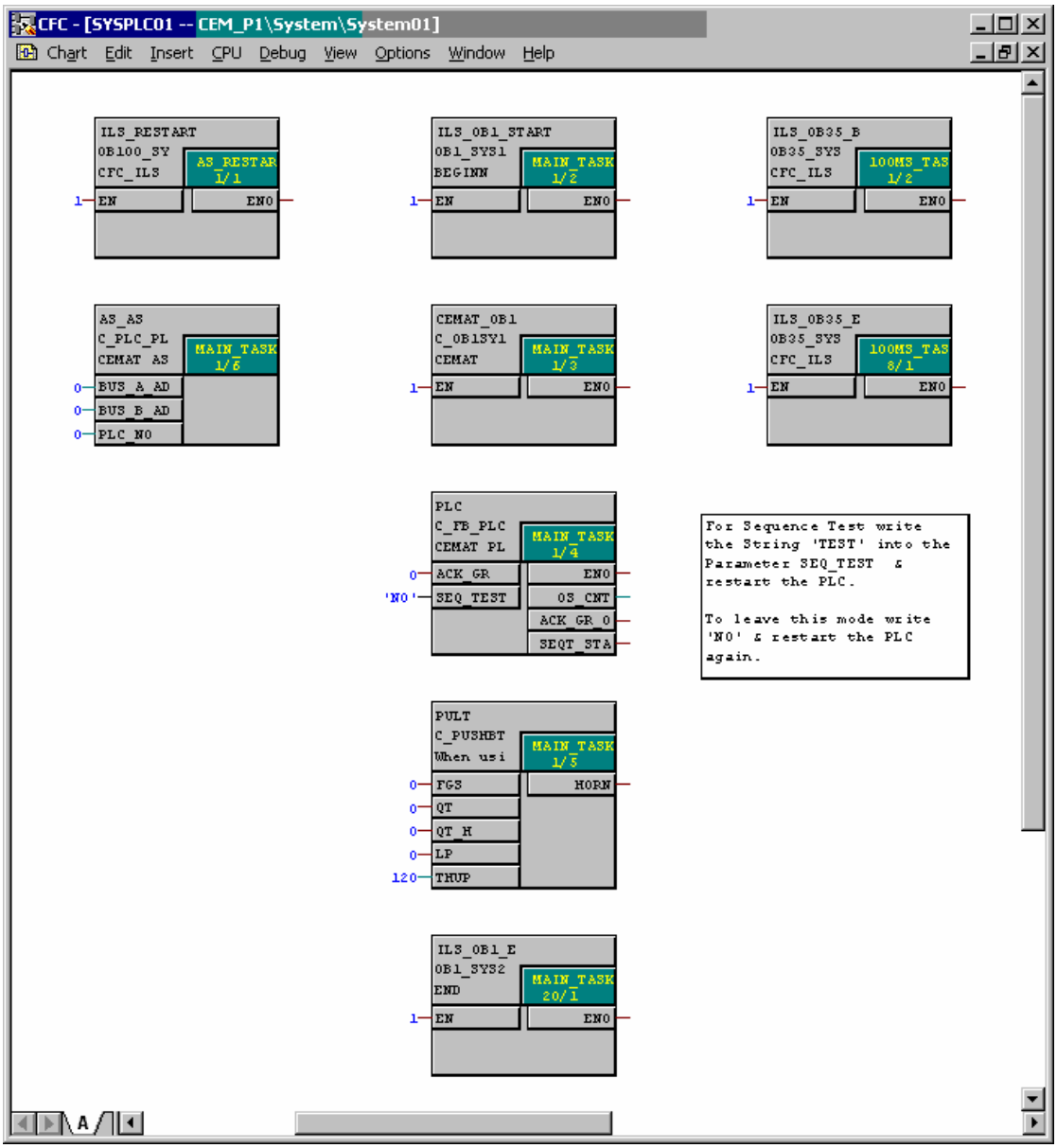
Absolute	Symbol	Task
FB1075	C_INTERL	OB1
FB1076	C_INTER5	OB1
FB61	CTRL_PID	OB35*)
FB70	RATIO_P	OB35*)
FB76	CTRL_S	OB35*)
FC527	OB100_SYS1	OB100
FC528	OB1_SYS2	OB1
FC529	OB35_SYS1	OB35
FC530	OB35_SYS2	OB35
FC1017	C_MUX	OB1
FC1018	C_ADAPT	OB1
FC1088	C_PUSHBT	OB1
FC1102	OB1_SYS1	OB1
FC1103	C_OB1SY1	OB1

*) This block can also be called from a different Task.

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 the system chart into the PCS7 Project

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



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 PLC number (e. g. SYSPLC02).

The system block C_PLC_PLC is only required, if the CEMAT PLC-PLC-Coupling is used (for communications to older CEMAT systems, e.g. CEMAT V4). In this case the CP-Address of the CP443 for Bus A and Bus B (e. g. 16372 and 16376) must be entered into the parameters BUS_A_AD and BUS_B_AD. Parameter PLC_NO must have the PLC-Number of the own PLC. The function block is called once in the restart (AS_RESTART) and once in OB1 (MAIN_TASK). For detailed description see chapter PLC-PLC-Coupling.

Compile and Download CFC

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

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

After these steps you can start with the PLC Engineering. In chapter 6 of this manual “PLC 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 Control Center 6.0*.

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 have to 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 Licence 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” or “Anwenderzyklus 1”. The default value of this is 2000ms. You have to adapt this value according to your requirement, e.g. 1000ms.

OS Project Editor

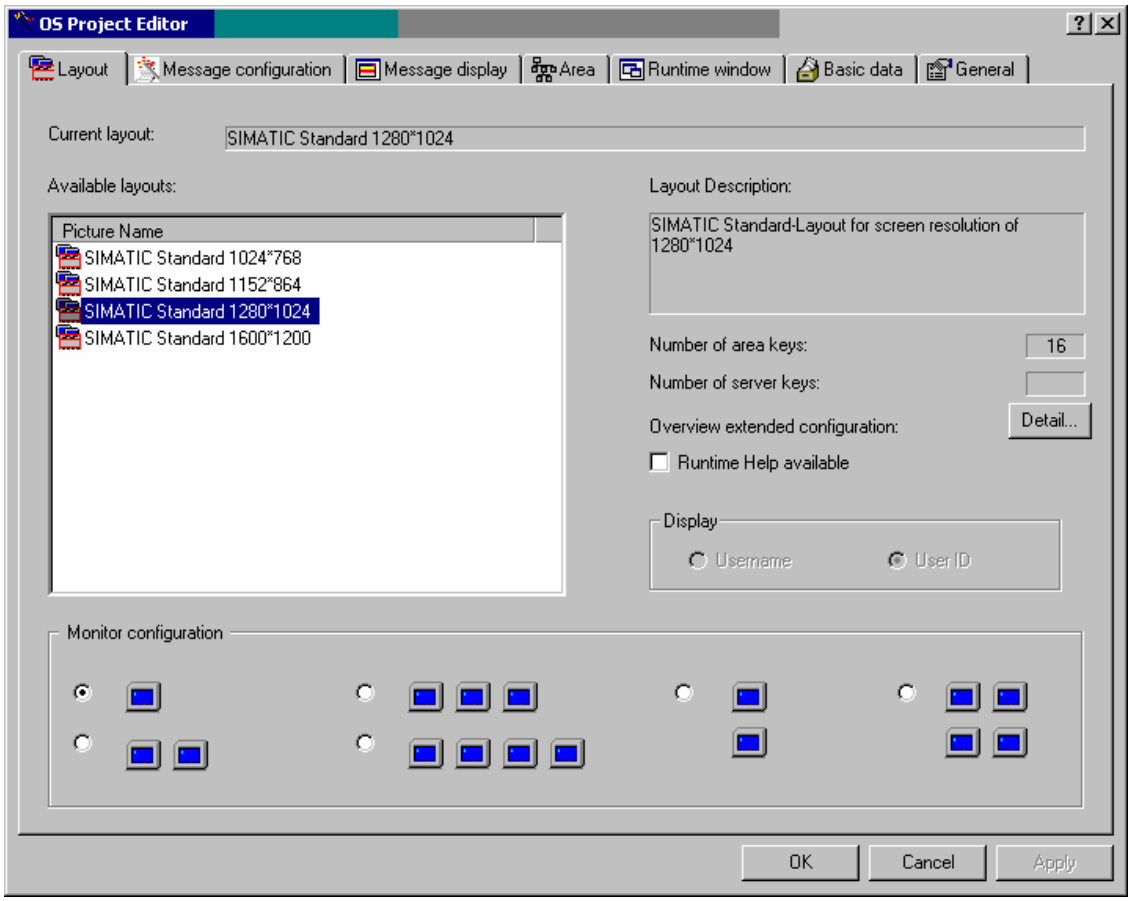
Settings for the user interface used by the plant operator for monitoring and controlling the plant during process operation In the OS Project Editor.

A detailed description to this item you find in the PCS7 Configuration Manual Operator Station. The following pages describe only the settings relevant for CEMAT.

In Tab Layout choose the right screen resolution. It may be SIMATIC Standard 1280*1024 for a Single-User Station.

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

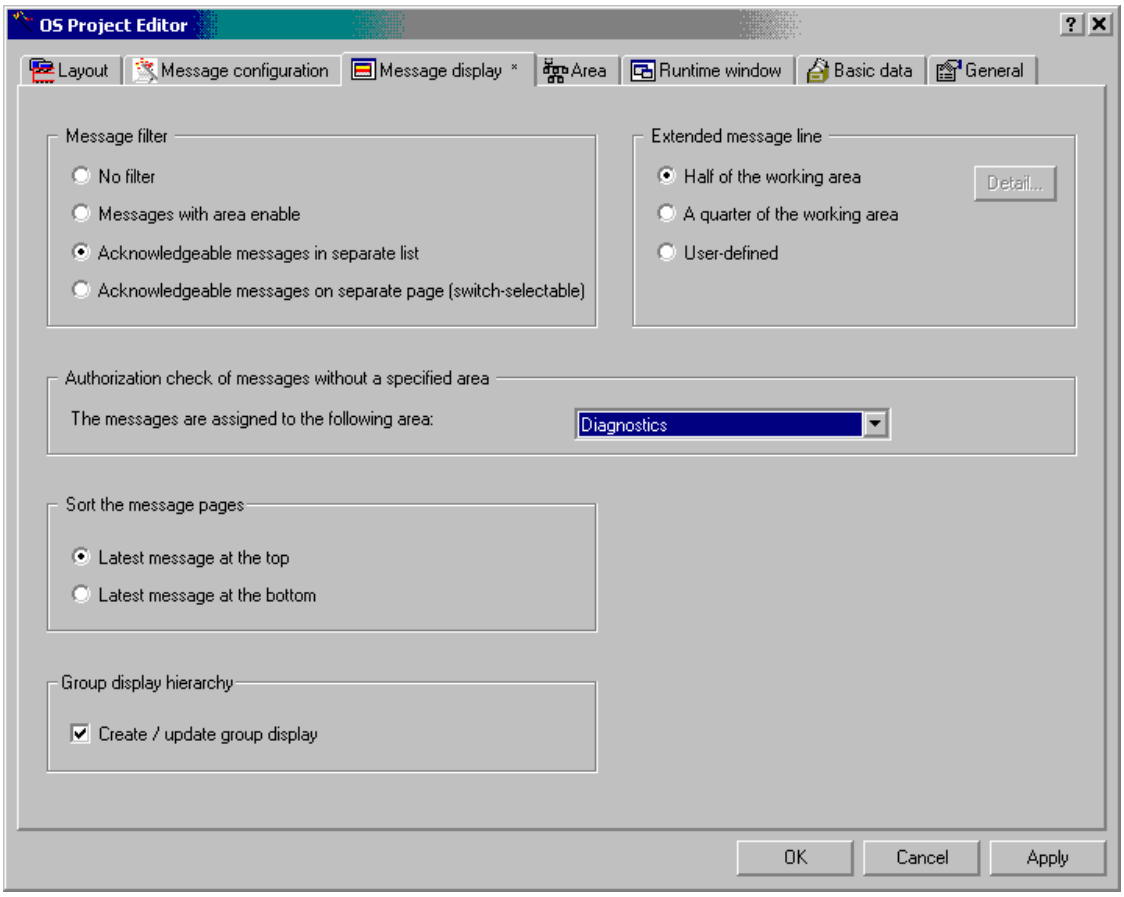
Example for the Layout of a Single Station:



The standard screen resolution for CEMAT is normally 1280*1024.

Concerning the OS Areas: Please keep in mind that the CEMAT message system can manage a maximum of 16 OS Areas at the moment!

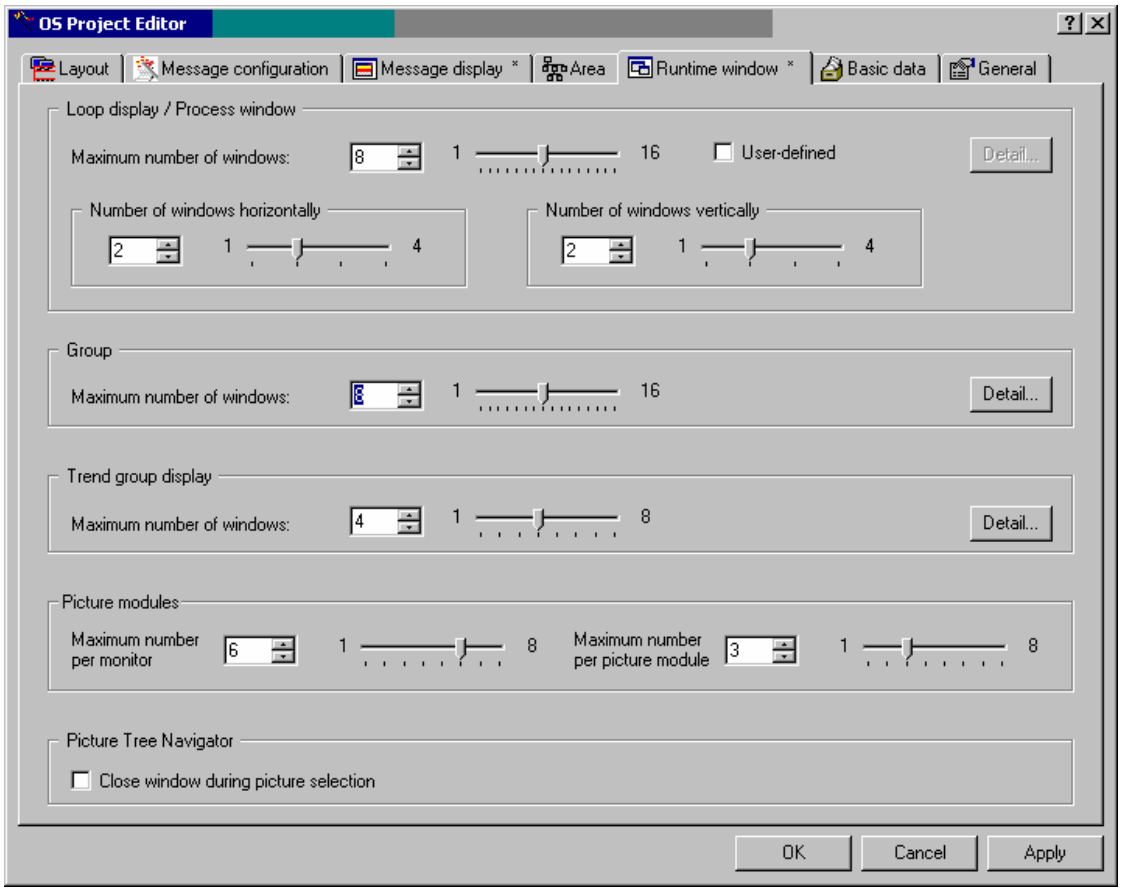
Message display for a Single Station:



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

The group display should be created and updated automatically.

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.

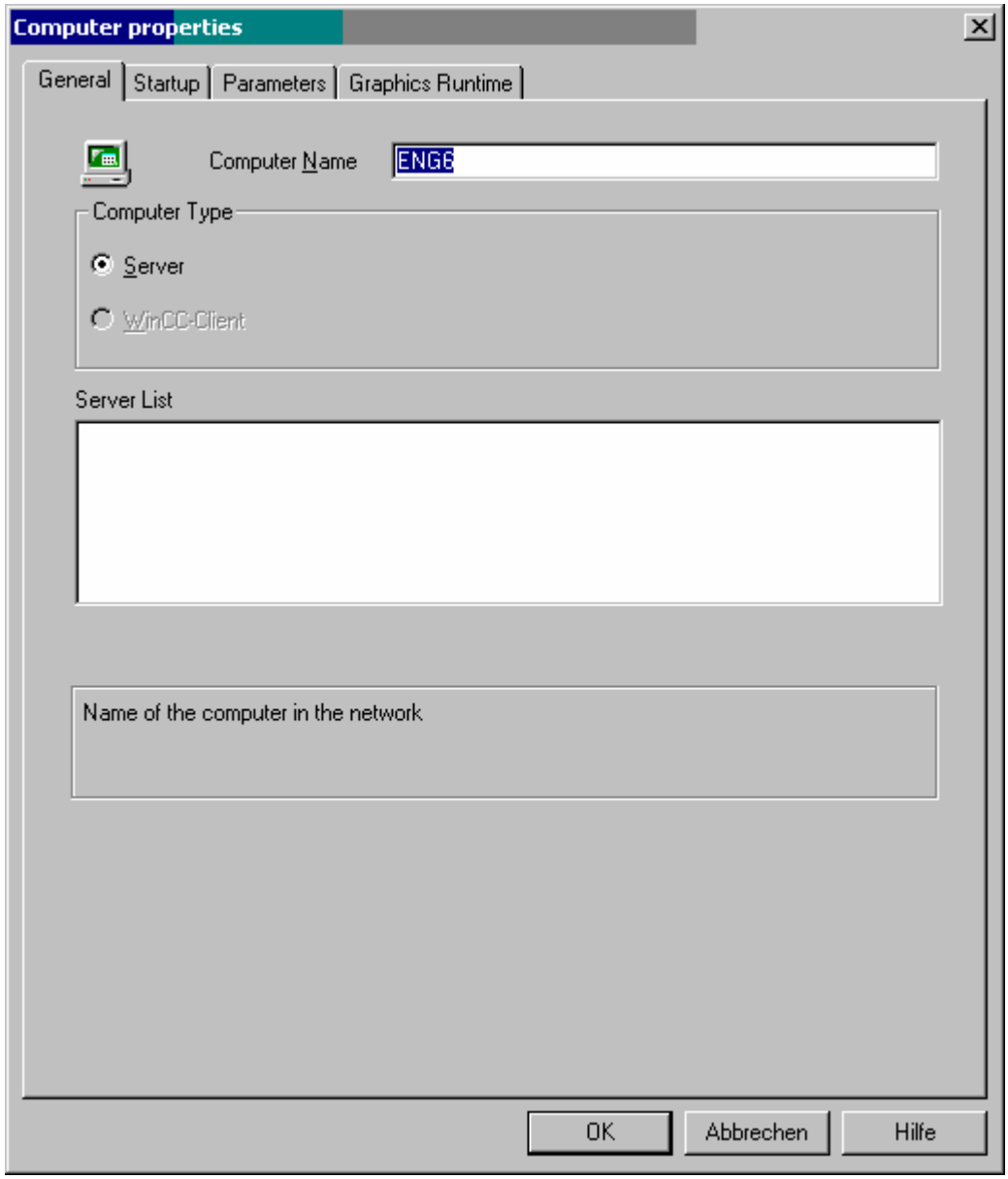
The rest of the settings in the OS Project Editor can remain as per default or can be adapted later.

With “OK” all settings are carried out.

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

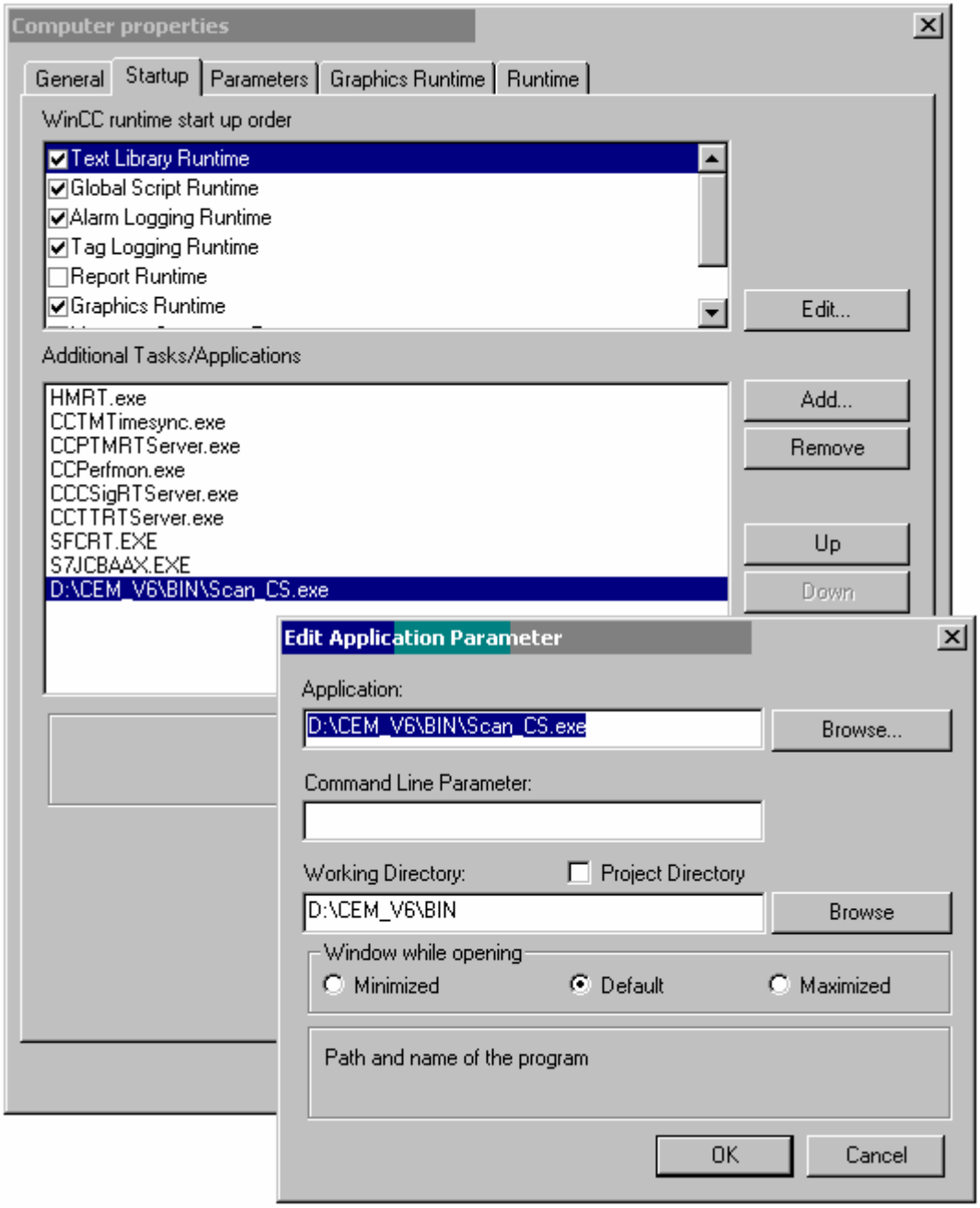


Caution: The Computer Name for all OS Projects is the name of the Engineering Station. During the download of the OS-Project to the individual PCs the name is automatically replaced by the Name of the PC Station.

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

Under Additional Tasks/Applications you have to add the Scan_CS.exe to the Startup list. The application is located in Directory D:\CEM_V6\BIN.

Scan_CS.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

Open the Tag Management and add the internal variables for CEMAT:

Add new Group “CEMAT”

Within this group add the following variables:

Name	Data Type	update
C_VIEW_TAG_A	Binary	Computer-local update
C_VIEW_TAG_D	Binary	Computer-local update
C_ServerName	Text tag 16bit	Computer-local update

If the message selection is carried out via alarm line (V6.0 principle), you additionally have to add the following variables. This is mainly used in case of migrated projects. If the message selection is carried out via user (V6.1) the following variables are not required:

Name	Data Type	update
C_AcknowledgeHorn	Unsigned 32-Bit Value	Project-wide update
C_<ComputerName>_AlarmSound	Singed 16-Bit Value	Computer-local update

Replace <ComputerName> by the station name, e. g. C_ENG6_AlarmSound.

Note: For the computer name use exact characters (Capital letters!). Otherwise the function will not work.

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!

Horn Tags

During the horn configuration additional tags are automatically created. You will find these variables in group “Horn”.

General changes in the alarm logging

Select the „Alarm Logging“ in the WinCC Explorer and use right mouse button to open the alarm logging editor.

Click on the plus symbol in front of **Message blocks**

Select **System blocks**

- Use right mouse button to click on „Date“ and select properties
- Change the following settings:
  Date: Four-Digit Day.Month.Year

- Use right mouse button to click on „Time“ and select properties
  Time: H/M/S (deselect Milliseconds)

Select **User text block**

- Use right mouse button to click on „Origin“ and select properties
- Change the following settings:
  Change „Source“ to „Tag“ Length = 22 Flashing = off

- Use right mouse button to click on „Event“ and select properties
- Change the following settings:
  Length = 20 Flashing = off

- Use right mouse button to click on „Free 1“ and select properties
- Change the following settings:
  Change „Free 1“ to „Tag Comment“ Length = 42

- Use right mouse button to click on „Free 2“ and select properties
- Change the following settings:
  Change „Free 2“ to „FCL“ (Fault class) Length = 2 Flashing = off

Select **Process value blocks**

- Use right mouse button to click on „Process_Value_1“ and select properties
- Change the following settings:
  Change „Process_Value_1“ to „Old value“

- Use right mouse button to click on „Process_Value_2“ and select properties
- Change the following settings:
  Change „Process_Value_2“ to „New value“

Message classes and message types

Click on the plus symbol in front of **Message classes**

Change the settings for „Alarm“, „Warning“, „PLC Process Control System Messages“, „Process Message“, „Operator Input Messages“ according to the following table.

- Use right mouse button to click on „Alarm“ and select properties
  Go to folder „General Information“ and select „Properties“ button to change the colors.
  Go to folder „Acknowledgement“ to change the acknowledge parameters.

The colors and message types have to be changes as follows:

Message class	Message Type	Came in		Went out		Acknowledged				
		Text color	Back-ground color	Text color	Back-ground color	Text color	Back-ground color	Must be ackn.	Come in must be acknowl.	Flashing
Alarm	H L	white	red	black	green	black	pink	yes	yes	off
Warning	H L	black	yellow	black	green	black	light yellow	yes	yes	off
Process Message	Process	white	red	black	green	black	pink	yes	yes	off
Operator Input Message		black	turquoise					no	no	off

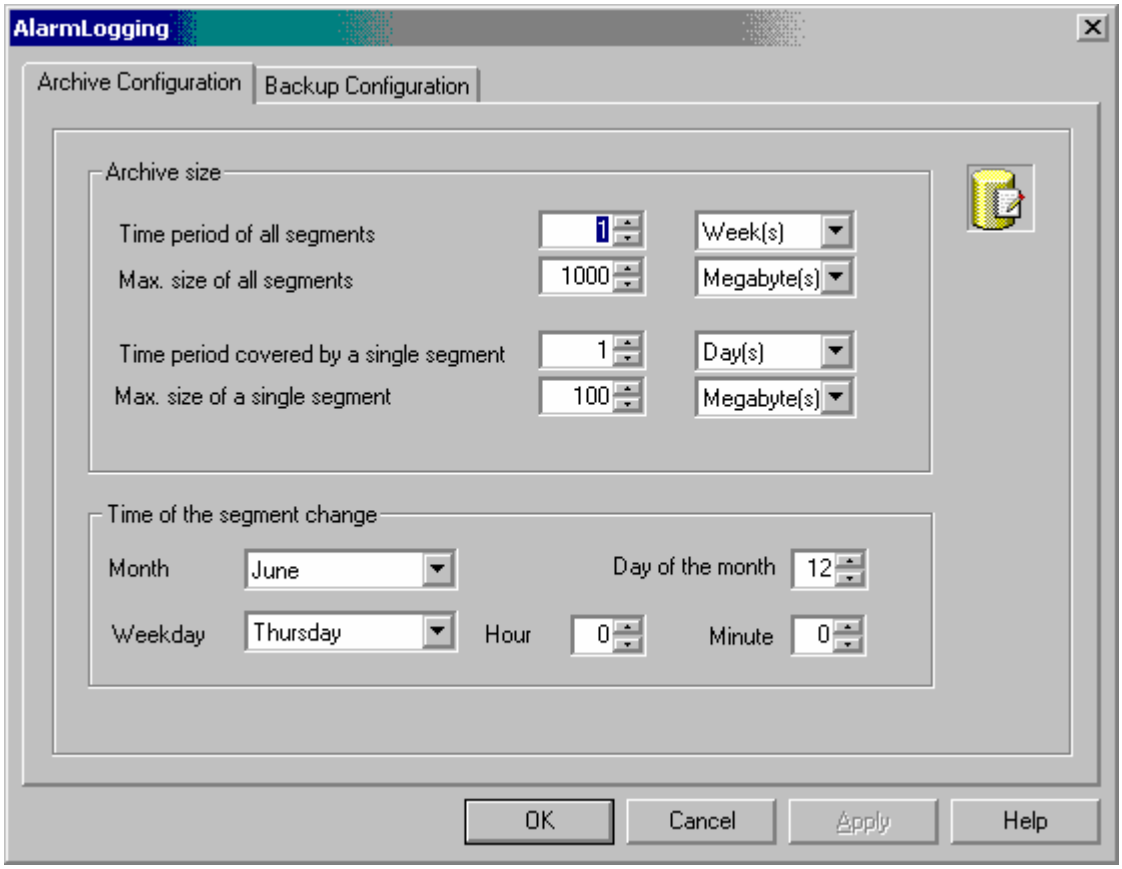
The acknowledge philosophy for the message classes „Alarm“, „Warning“ and „Process message“ has to be configured as follows:

- Acknowledge Came In: yes
- Acknowledge Went Out: no
- Flashing on: no
- Only initial value: no
- Messages without status Went Out: no
- Separate Acknowledge key: no
- By means of single acknowledge: yes

Settings for reload after power failure (short term archive)

Under “Archive Configuration” -> “Message archive” use right mouse button to make the following settings for “Reload after power failure”:

Under “Archive Configuration” -> “Message Archive” -> “Properties” you have to configure the archive according to the requirements of your project:



Exit the alarm logging editor and save the changes.

Tag logging

For each measure which needs to be archived, in the process value archive an archive tag has to be created.

How to create a process value archive and the archive tags is described in the engineering manual of PCS7 and not part of this description.

For CEMAT some additional rules must be followed:

- The Archive name for all measures, as per default, must be **ProcessValueArchive**. From CEMAT V6 SP3 the possibility exists to create multiple Archives with different names. This needs additional Engineering effort. See description under OS_Engineering chapter C_MEASUR.
- To enable the direct call of measured value archive from the faceplate of the measure, it is necessary that the name of the archive tag exactly corresponds to the name of the process tag.

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

At this step it is probably more convenient, only to prepare the process value archive (Size of the archive, archiving duration, default values for archiving cycles.) and to configure the archive later, because at this moment no measures exist and therefore no archive variables can be created.

User Archives

There are 2 user archives for CEMAT Functions.

The user archive **C_INFO** contains the object information (Info Database).

The user archive **C_DriveList** contains the Group-Object-Lists (list of objects associated to a route or group).

How to create the user archive for the Info Database

In directory D:\Cem_V6\ua you will find the default files for the Info Database.

File C_INFO.uap contains the structure of the Info Database.

Import the structure of the C_INFO and save:

1. Open the User-Archive
2. Import the structure of the Info Database from C_INFO.uap
3. Save the structure.

Through the import the Info Database is created with the following structure:

ID	integer	1
Obj_ID	double	2
C_TAG	varchar (20)	3
GL_Complex	varchar (40)	4
GL_Plant	varchar (40)	5
GL_Plant_zone	varchar (40)	6
GL_Equipment	varchar (40)	7
GL_LocationObject	varchar (40)	8
GL_Location_LS	varchar (40)	9
G_LoopA	varchar (40)	10
G_Infotext	varchar (250)	11
G_Sound_1	varchar (40)	12
G_Sound_2	varchar (40)	13
G_DRAW_POS	varchar (40)	14
G_Video_Name	varchar (40)	15
G_DOC_Name_1	varchar (40)	16
G_Spare	varchar (40)	17
G_Spare_1	varchar (40)	18
I_Building	varchar (40)	19
I_Location	varchar (40)	20
I_Area	varchar (40)	21
I_Cabinet	varchar (40)	22
I_Rack_Q	varchar (40)	23
I_Slot_Q	varchar (20)	24
I_Rack_I	varchar (40)	25
I_Slot_I	varchar (20)	26
M_Building	varchar (40)	27
M_Location	varchar (40)	28
M_Area	varchar (40)	29
M_Cabinet	varchar (40)	30
M_Feeder	varchar (40)	31
M_Fuse	varchar (20)	32
M_Bimetall	varchar (20)	33
H_Building	varchar (40)	34
H_Location	varchar (40)	35
H_Area	varchar (40)	36
H_Cabinet	varchar (40)	37
H_Rack	varchar (40)	38
H_PLC_Name	varchar (20)	39
H_Instanc_DB_NO	integer	40
LastUser	varchar (10)	41
LastAccess	timestamp	42

The Info Database must be filled by the user. The easiest way is to use an Excel Tool which is described in chapter OS Engineering.

Attention: You must create at least one line of runtime data into the Info Database. Select the input mode, in the first line enter in column C_TAG the string „Dummy“ and press Return.

How to create the user archive for the Group-Object-List

In directory D:\Cem_V6\ua you will find the default user archive for the Group-Object-List.

File DriveList.uap contains the structure of the Group-Object-List. The data for Group-Object-List are entered in runtime mode.

Through the import the Group-Object-List is created with the following structure:

GroupTAG	String	24
HasSubRoute	Number (integer)	
HierachyLevel	Number (integer)	
ObjectComment	String	50
ObjectTAG	String	24
ObjectType	String	15
OrderNo	Number (integer)	
ParentTAG	String	24

Import the structure of the C_DriveList and save:

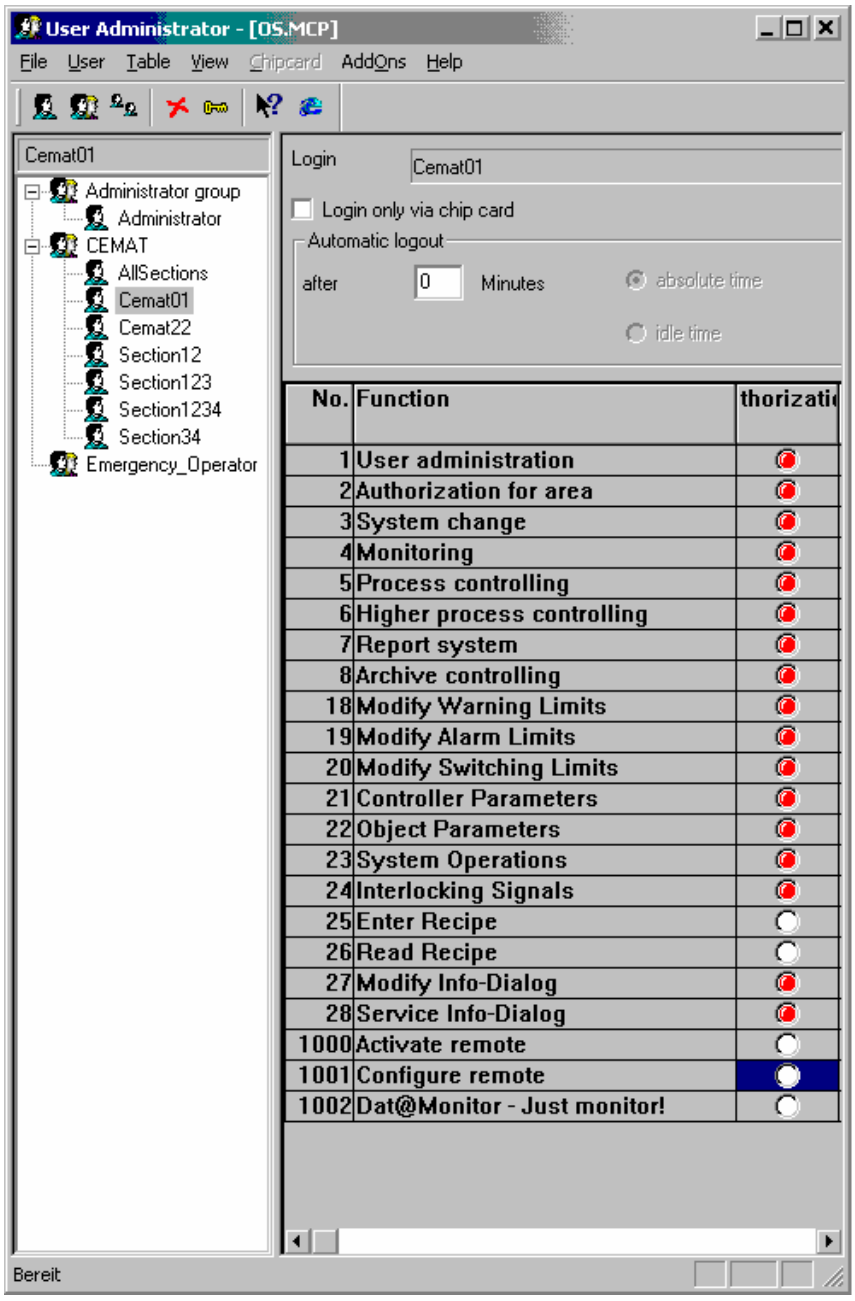
1. Open the User-Archive
2. Import the structure of the Info Database from C_INFO.uap
3. Save the structure.

You must not enter any data into the user archive Group-Object-List. The data will be entered during runtime.

Add User rights

In CEMAT Standard additional User rights must be added to the PCS7 User rights in order to define more detailed who is authorized for which action.

In the WinCC Explorer click open the **User Administrator** and add the rights from No. 18 to No. 28 according to the following list:



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
	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	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
	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	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
	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	007
C_VALVE	R1/R2/Stop	05: Process controlling	Processcontrolling_backup	all
	Process parameter	22: Object Parameters		all
	Reset Operation	23: System Operations		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	007

Object type	Operation	Right/ default right	Instance specific settings possible via Attribute	Project Code
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/Local/Auto/ Single/QuickStop	05: Process controlling	Processcontrolling_backup	all
	Auto	05: Process controlling	Processcontrolling_backup	006
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		all
C_ANNUN8	Process Parameters	22: Object Parameters		all
	Simulation	24: Interlocking Signals		all
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
	All Buttons	05: Process controlling		all

Selection of area specific rights

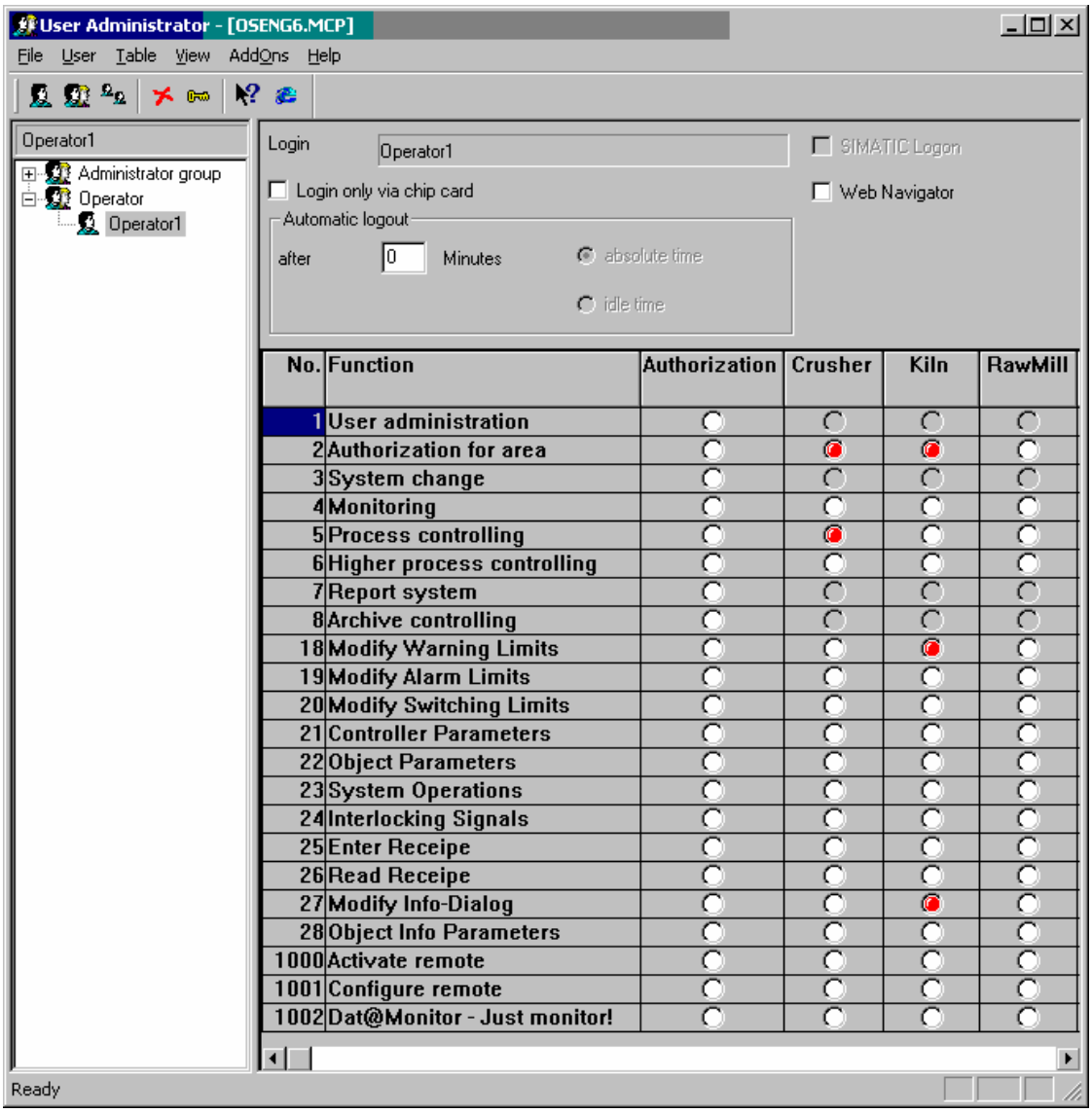
After the definition of the additional user rights for CEMAT you must define the user groups and the users and in order to specify the rights for each user.

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

With the right 2 “Authorization for area” it is defined which area can be viewed.

With the right 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 „Operator1“ is allowed to:

open, watch	→ **Crusher, Kiln**
operate	→ **Crusher**

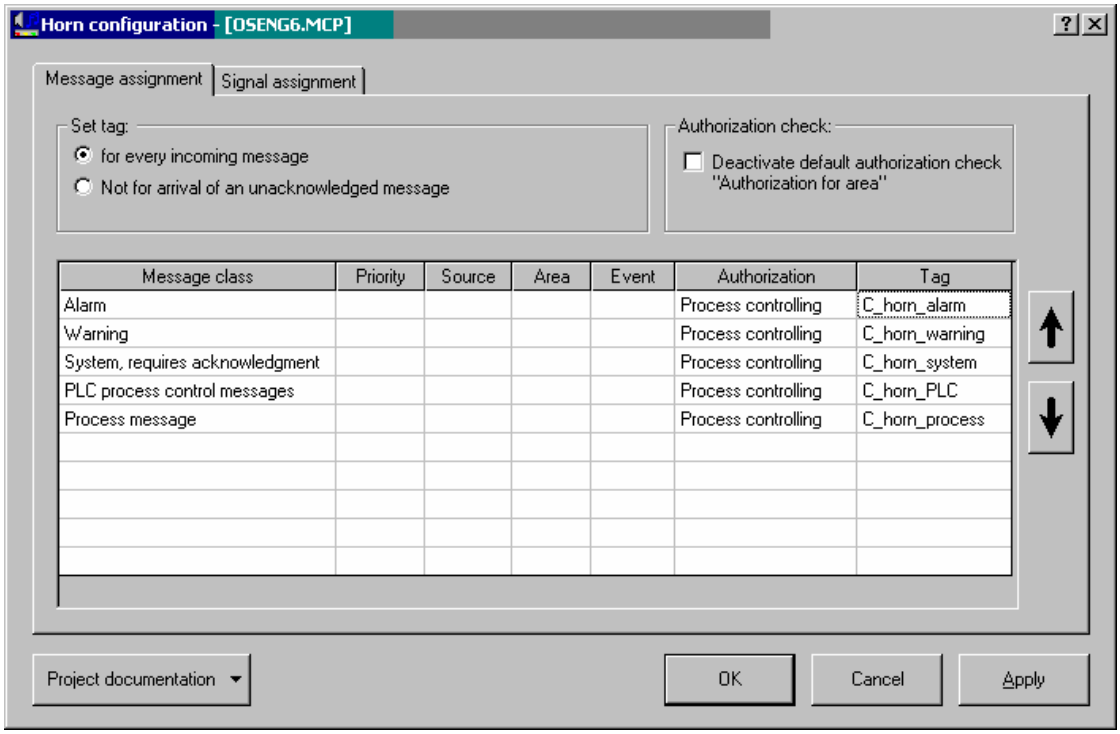
The area **RawMill** is disabled for Operator1.

Messages are only shown for areas which are enabled for operation. Operator1 will get only the messages from the Crusher area and he can acknowledge only these messages.

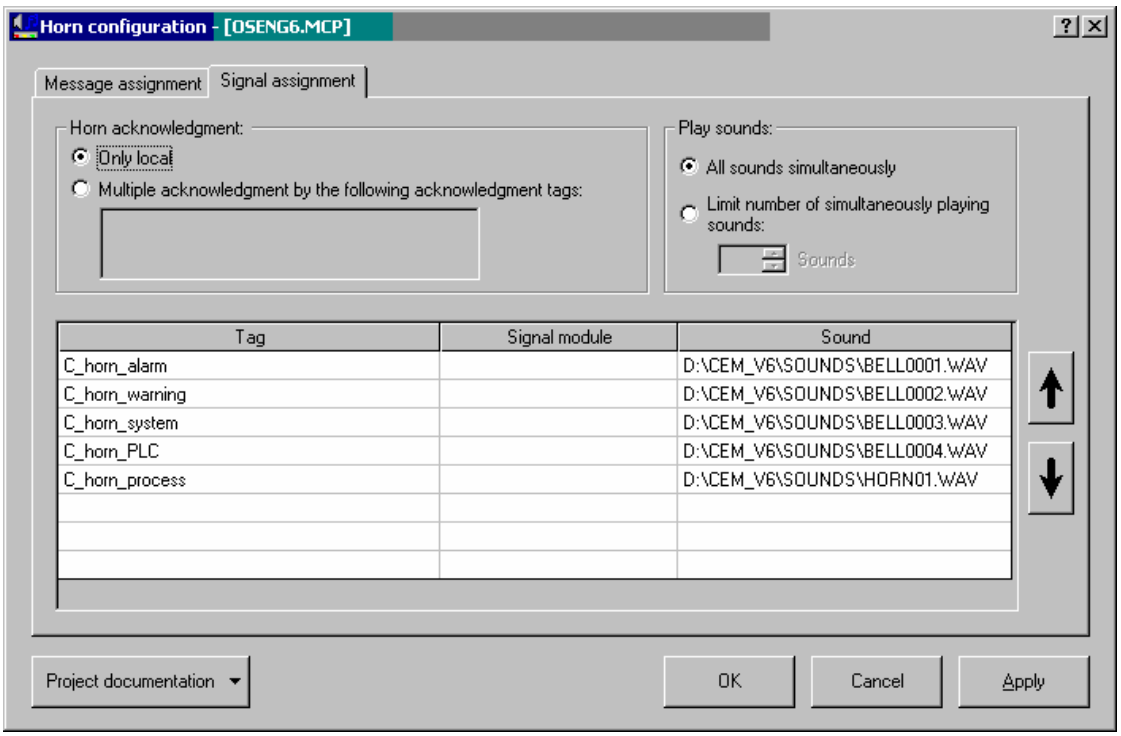
Horn configuration

From CEMAT V6.1 you can define a separate Sound for each Message class. The horn must be configured as follows:

1. First define in tab “Message assignments” all message classes which should create a sound. These are Alarms, Warnings, System messages which require acknowledgement, PLC Process control message and Process messages. If you use other message classes with alarms in your project you have to define these as well.
2. Assign the Authorization „Process controlling“ (with this the Authorization is enabled per Area).
3. Per message class, define an internal variable with computer local actualisation. Insert the variable name. With “Apply” the variable is automatically created in the tag management.
4. The definition can also be carried out per Area.



5. In tab “Signal assignment” you can define for each variable an individual sound. A sample of different sounds you find in directory D:\CEM_V6\Sounds.



CEMAT specific preparations

Before you start with the configuration of the OS you have to copy the CEMAT Functions in to your Project. These are Global Script Actions, Project Functions, Standard Pictures and Bitmaps. Please proceed as follows:

- Close the WinCC Explorer

- Copy the Global Scripts from 'D:\CEM_V6\WinCC\Library' into the PCS7 Project ...\wincproj\<OS>\Library'.

- Copy from "D:\CEM_V6\WinCC\GraCS' all Objects into the PCS7 Project ...'wincproj<OS>\GraCS'
  The following standard pictures contain adaptations for CEMAT. The existing pictures must be overwritten by the pictures from the CEM_V6 directory.

@Overview1.pdl	Overview Range
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@CSIGQuit.PDL	Acknowledge fault function
@HornQuit.PDL	Acknowledge horn function

- Copy file @PCS7Typicals.cfg from ’D:\CEM_V6\WinCC\WScripts’ into the PCS7 Project '<OS>\WScripts'. (File @PCS7Typicals.cfg contains the list of attributes which will not be overwritten with update block icons).

- Open the WinCC Project.

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

Preparation of the Template Pictures

With the CEMAT standard different template pictures are copied into the project.

The picture **@C_PCS7Typicals.pdl** and **@C_PCS7Typicals_V61.pdl** contain the default symbols for automatic generation of block icons from SIMATIC Manager.

- The picture **@C_PCS7Typicals.pdl** contains the block icons which were already used in CEMAT V6.0. This picture should be used as template if the project was already created with CEMAT V6.0 and shall be upgraded to CEMAT V6.1 with minimum effort.
- The block icons in picture **@C_PCS7Typicals_V61.pdl** have enhanced functionality (if an object creates an alarm and you open the picture via loop-in-alarm function the respective object gets marked).
- For the automatic generation of block icons PCS7 V6.1 uses the picture @PCS7Typicals.pdl or the pictures with this name plus an extension (e. g. @PCS7Typical1.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.pdl which contains the symbols you want to use in the project.

The picture **@C_Template.pdl** contains the default symbols for manual connection in WinCC (using wizard).

- The block icons in picture **@C_Template.pdl** correspond to the block icons which were used in CEMAT V6.0.
- The template picture @C_Template.pdl you also have to adapt and you have to save it under a different name (@Template1.pdl) in order not to overwrite the picture in case of an Update.

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

Caution: For some of the project versions of CEMAT, modified symbols are required. These examples you will find in the template pictures as well. 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.

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 Control Center 6.0*.

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 have to 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” or “Anwenderzyklus 1”. The default value of this is 2000ms. You have to adapt this value according to your requirement, e.g. 1000ms.

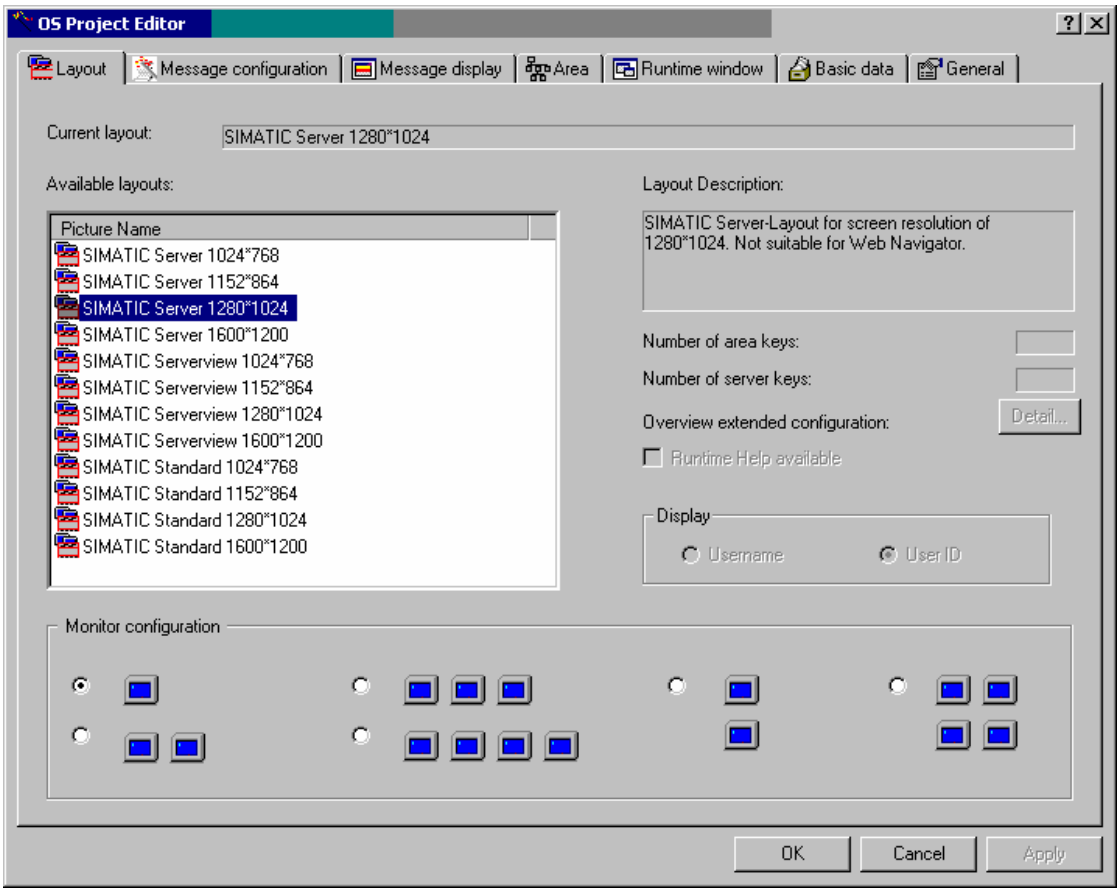
OS Project Editor

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 PCS7 Configuration Manual Operator Station. The following pages describe only the settings relevant for CEMAT.

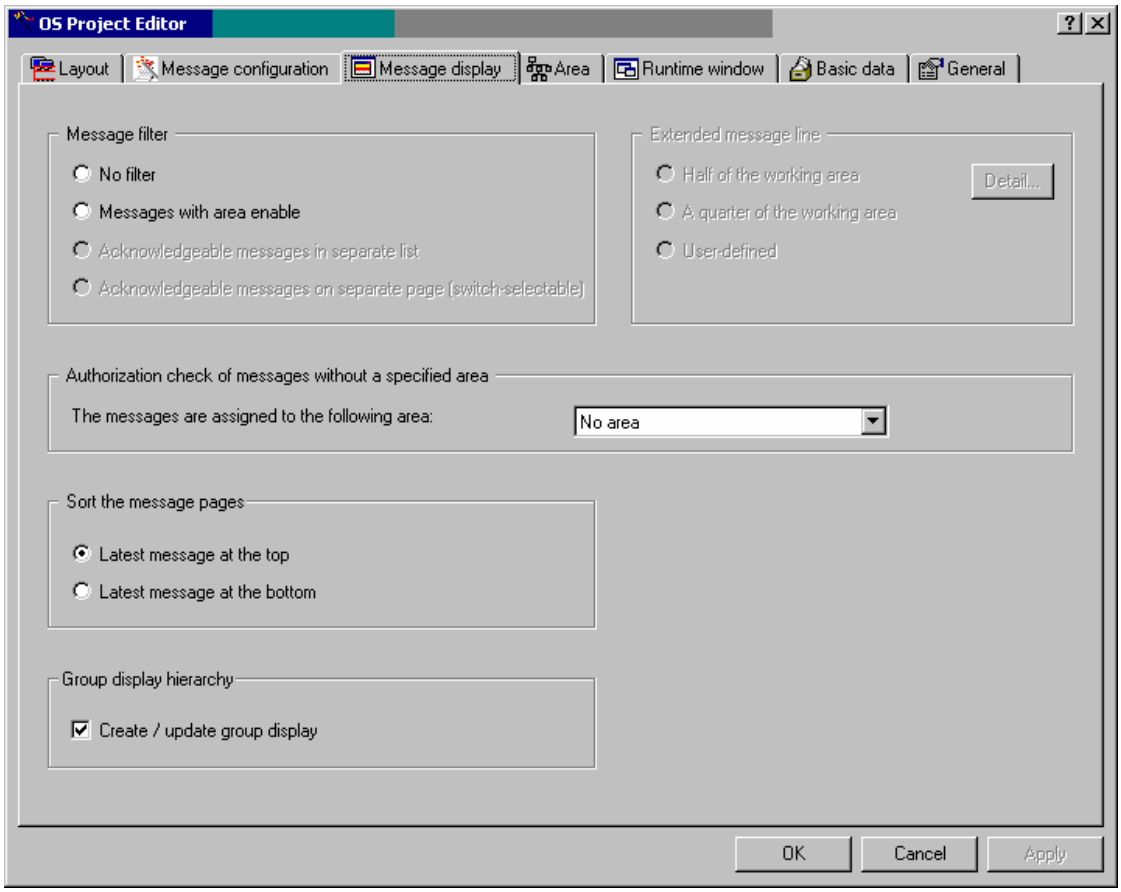
In Tab Layout choose the right screen resolution. It may be SIMATIC Server 1280*1024 for a Server. The server has no area keys.

Example for the Layout of a Server:



The standard screen resolution for CEMAT is 1280*1024. 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.

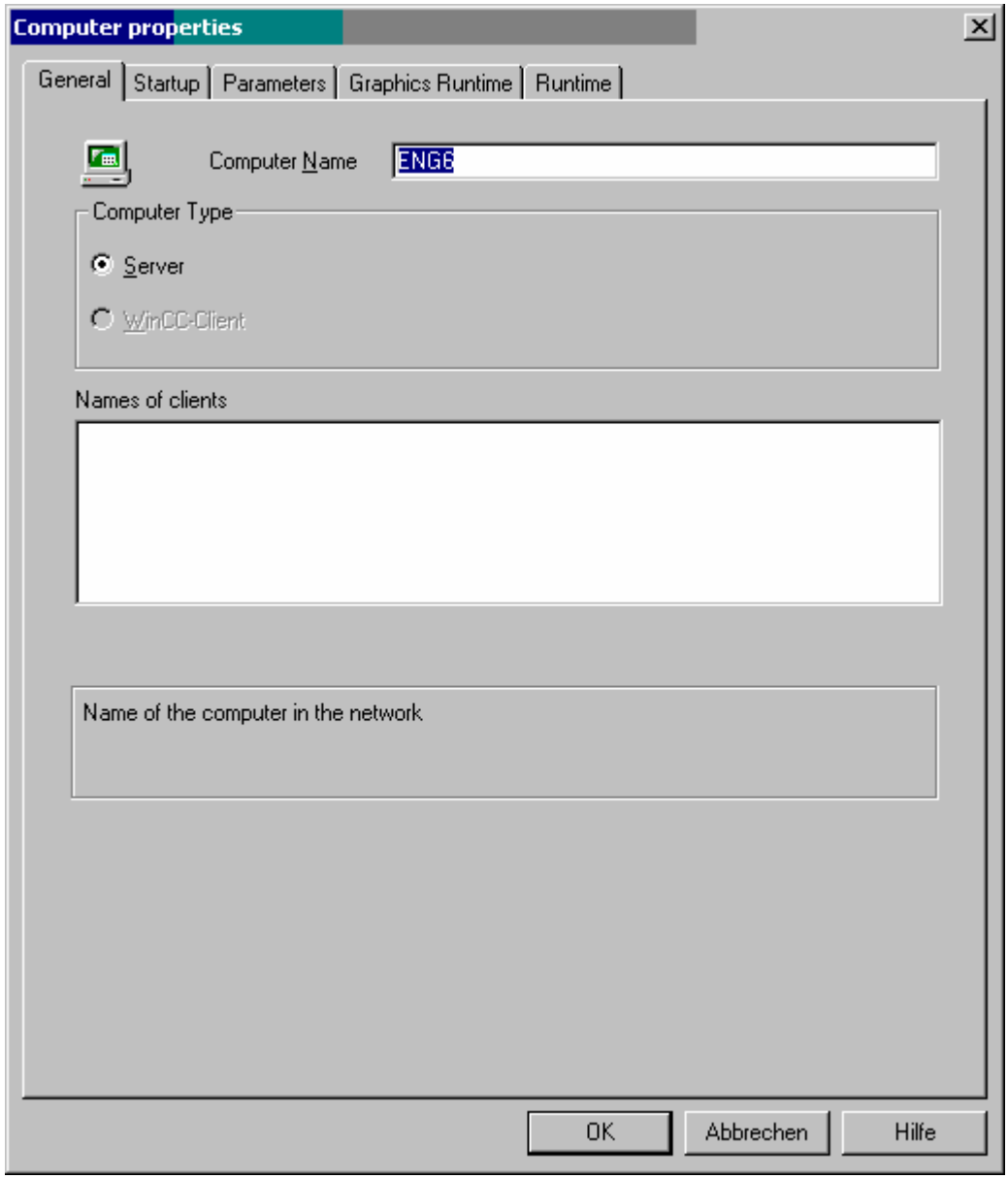
The rest of the settings in the OS Project Editor can remain as per default or can be adapted later.

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

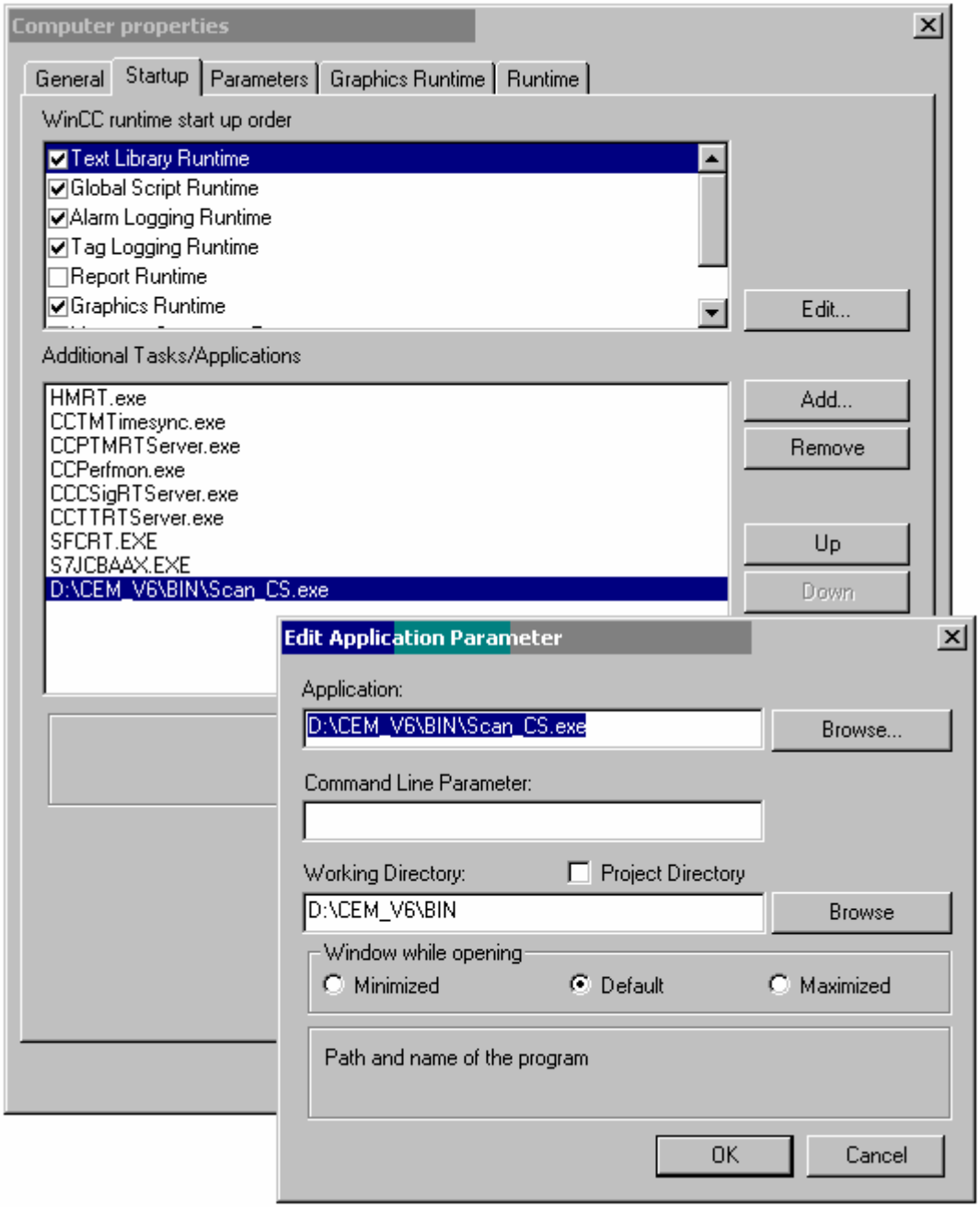


Caution: The Computer Name for all OS Projects is the name of the Engineering Station. During the download of the OS-Project to the individual PCs the name is automatically replaced by the Name of the PC Station.

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

Under Additional Tasks/Applications you have to add the Scan_CS.exe to the Startup list. The application is located in Directory D:\CEM_V6\BIN.

Scan_CS.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

Open the Tag Management and add the internal variables for CEMAT:

Add new Group “CEMAT”

Within this group add the following variables:

Name	Data Type	update
C_VIEW_TAG_A	Binary	Computer-local update
C_VIEW_TAG_D	Binary	Computer-local update
C_ServerName	Text tag 16bit	Computer-local update

If the message selection is carried out via alarm line (V6.0 principle), you additionally have to add the following variables. This is mainly used in case of migrated projects. If the message selection is carried out via user (V6.1) the following variables are not required:

Name	Data Type	update
C_AcknowledgeHorn	Unsigned 32-Bit Value	Project-wide update
C_<ComputerName>_AlarmSound	Singed 16-Bit Value	Computer-local update

Replace <ComputerName> by the station name, e. g. C_SERV61_AlarmSound.

Note: For the computer name use exact characters (Capital letters!). Otherwise the function will not work.

Horn Tags

From CEMAT V6.1 you can define a separate Sound for each Message class. The horn must be configured correspondingly. The horn configuration is carried out in the Client Project.

The following configuration in the tag management of the server 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 signalling 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.

Create a reset variable in the Group „Horn“

Name	Data Type	update
HornResetMC	binary	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!

General changes in the alarm logging

Select the „Alarm Logging“ in the WinCC Explorer and use right mouse button to open the alarm logging editor.

Click on the plus symbol in front of **Message blocks**

Select **System blocks**

- Use right mouse button to click on „Date“ and select properties
- Change the following settings:
  Date: Four-Digit Day.Month.Year

- Use right mouse button to click on „Time“ and select properties
  Time: H/M/S (deselect Milliseconds)

Select **User text block**

- Use right mouse button to click on „Origin“ and select properties
- Change the following settings:
  Change „Source“ to „Tag“ Length = 22 Flashing = off

- Use right mouse button to click on „Event“ and select properties
- Change the following settings:
  Length = 20 Flashing = off

- Use right mouse button to click on „Free 1“ and select properties
- Change the following settings:
  Change „Free 1“ to „Tag Comment“ Length = 42

- Use right mouse button to click on „Free 2“ and select properties
- Change the following settings:
  Change „Free 2“ to „FCL“ (Fault class) Length = 2 Flashing = off

Select **Process value blocks**

- Use right mouse button to click on „Process_Value_1“ and select properties
- Change the following settings:
  Change „Process_Value_1“ to „Old value“

- Use right mouse button to click on „Process_Value_2“ and select properties
- Change the following settings:
  Change „Process_Value_2“ to „New value“

Message classes and message types

Click on the plus symbol in front of **Message classes**

Change the settings for „Alarm“, „Warning“, „PLC Process Control System Messages“, „Process Message“, „Operator Input Messages“ according to the following table.

- Use right mouse button to click on „Alarm“ and select properties
  Go to folder „General Information“ and select „Properties“ button to change the colors.
  Go to folder „Acknowledgement“ to change the acknowledge parameters.

The colors and message types have to be changes as follows:

Message class	Message Type	Came in		Went out		Acknowledged				
		Text color	Back-ground color	Text color	Back-ground color	Text color	Back-ground color	Must be ackn.	Come in must be acknowl.	Flashing
Alarm	H L	white	red	black	green	black	pink	yes	yes	off
Warning	H L	black	yellow	black	green	black	light yellow	yes	yes	off
Process Message	Process	white	red	black	green	black	pink	yes	yes	off
Operator Input Message		black	turquoise					no	no	off

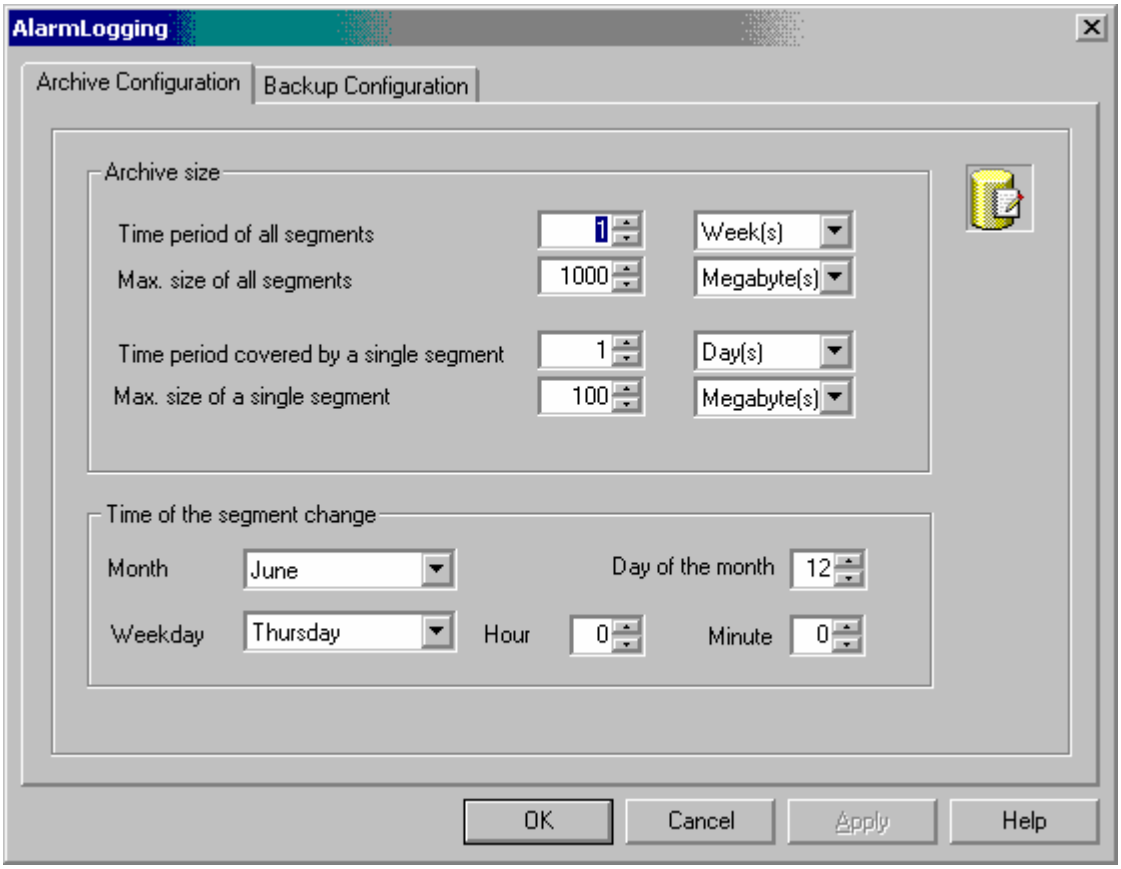
The acknowledge philosophy for the message classes „Alarm“, „Warning“ and „Process message“ has to be configured as follows:

- Acknowledge Came In: yes
- Acknowledge Went Out: no
- Flashing on: no
- Only initial value: no
- Messages without status Went Out: no
- Separate Acknowledge key: no
- By means of single acknowledge: yes

Settings for reload after power failure (short term archive)

Under “Archive Configuration” -> “Message archive” use right mouse button to make the following settings for “Reload after power failure”:

Under “Archive Configuration” -> “Message Archive” -> “Properties” you have to configure the archive according to the requirements of your project:



Exit the alarm logging editor and save the changes.

Tag logging

For each measure which needs to be archived, in the process value archive an archive tag has to be created.

How to create a process value archive and the archive tags is described in the engineering manual of PCS7 and not part of this description.

For CEMAT some additional rules must be followed:

- The Archive name for all measures, as per default, must be **ProcessValueArchive**. From CEMAT V6 SP3 the possibility exists to create multiple Archives with different names. This needs additional Engineering effort. See description under OS_Engineering chapter C_MEASUR.
- To enable the direct call of measured value archive from the faceplate of the measure, it is necessary that the name of the archive tag exactly corresponds to the name of the process tag.

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

At this step it is probably more convenient, only to prepare the process value archive (Size of the archive, archiving duration, default values for archiving cycles.) and to configure the archive later, because at this moment no measures exist and therefore no archive variables can be created.

User Archives

There are 2 user archives for CEMAT Functions.

The user archive **C_INFO** contains the object information (Info Database).

The user archive **C_DriveList** contains the Group-Object-Lists (list of objects associated to a route or group).

How to create the user archive for the Info Database

In directory D:\Cem_V6\ua you will find the default files for the Info Database.

File C_INFO.uap contains the structure of the Info Database.

Import the structure of the C_INFO and save:

1. Open the User-Archive
2. Import the structure of the Info Database from C_INFO.uap
3. Save the structure.

Through the import the Info Database is created with the following structure:

ID	integer	1
Obj_ID	double	2
C_TAG	varchar (20)	3
GL_Complex	varchar (40)	4
GL_Plant	varchar (40)	5
GL_Plant_zone	varchar (40)	6
GL_Equipment	varchar (40)	7
GL_LocationObject	varchar (40)	8
GL_Location_LS	varchar (40)	9
G_LoopA	varchar (40)	10
G_Infotext	varchar (250)	11
G_Sound_1	varchar (40)	12
G_Sound_2	varchar (40)	13
G_DRAW_POS	varchar (40)	14
G_Video_Name	varchar (40)	15
G_DOC_Name_1	varchar (40)	16
G_Spare	varchar (40)	17
G_Spare_1	varchar (40)	18
I_Building	varchar (40)	19
I_Location	varchar (40)	20
I_Area	varchar (40)	21
I_Cabinet	varchar (40)	22
I_Rack_Q	varchar (40)	23
I_Slot_Q	varchar (20)	24
I_Rack_I	varchar (40)	25
I_Slot_I	varchar (20)	26
M_Building	varchar (40)	27
M_Location	varchar (40)	28
M_Area	varchar (40)	29
M_Cabinet	varchar (40)	30
M_Feeder	varchar (40)	31
M_Fuse	varchar (20)	32
M_Bimetall	varchar (20)	33
H_Building	varchar (40)	34
H_Location	varchar (40)	35
H_Area	varchar (40)	36
H_Cabinet	varchar (40)	37
H_Rack	varchar (40)	38
H_PLC_Name	varchar (20)	39
H_Instanc_DB_NO	integer	40
LastUser	varchar (10)	41
LastAccess	timestamp	42

The Info Database must be filled by the user. The easiest way is to use an Excel Tool which is described in chapter OS Engineering.

Attention: You must create at least one line of runtime data into the Info Database. Select the input mode, in the first line enter in column C_TAG the string „Dummy“ and press Return.

How to create the user archive for the Group-Object-List

In directory D:\Cem_V6\ua you will find the default user archive for the Group-Object-List.

File DriveList.uap contains the structure of the Group-Object-List. The data for Group-Object-List are entered in runtime mode.

Through the import the Group-Object-List is created with the following structure:

GroupTAG	String	24
HasSubRoute	Number (integer)	
HierachyLevel	Number (integer)	
ObjectComment	String	50
ObjectTAG	String	24
ObjectType	String	15
OrderNo	Number (integer)	
ParentTAG	String	24

Import the structure of the C_DriveList and save:

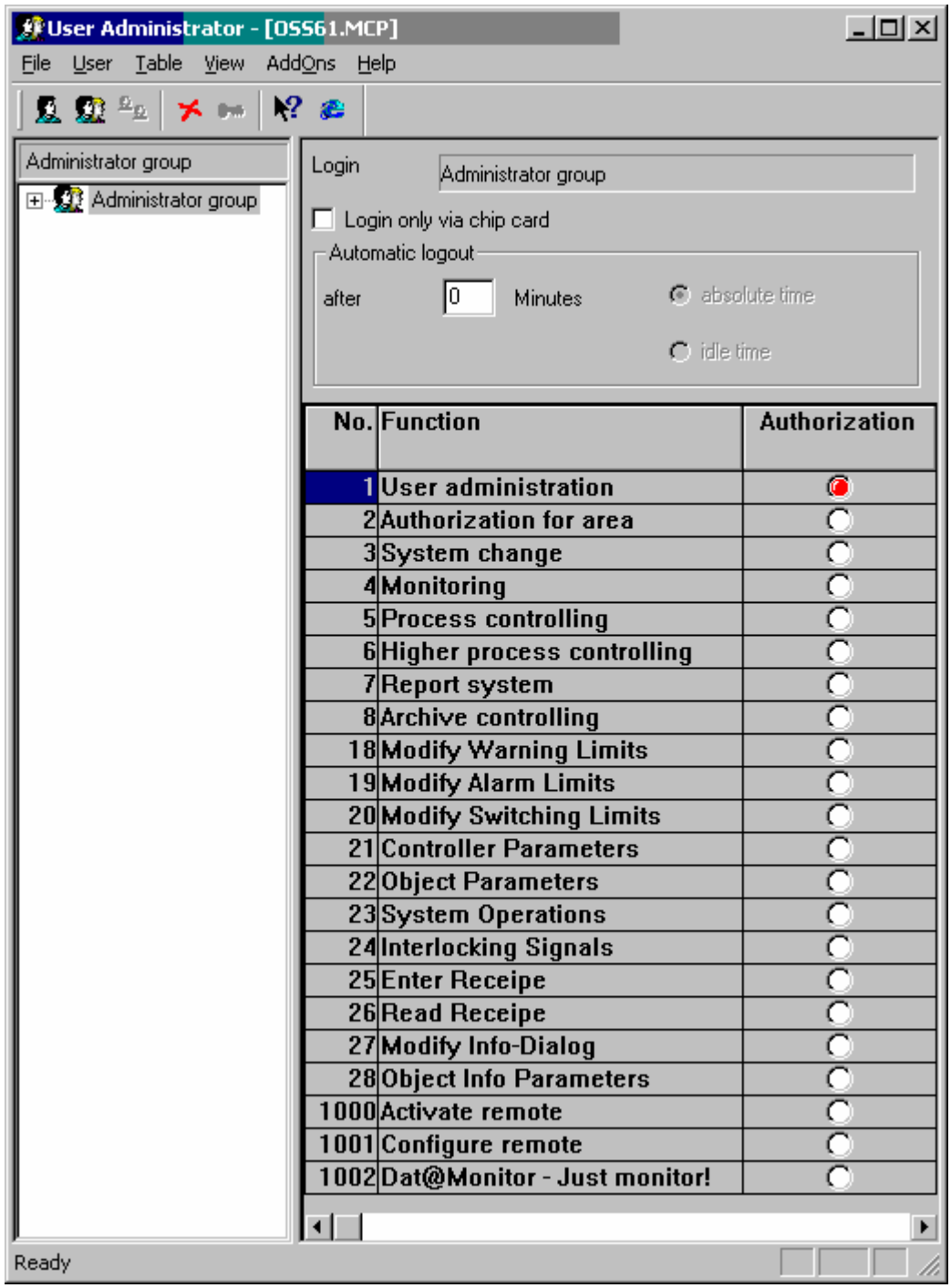
1. Open the User-Archive
2. Import the structure of the Info Database from C_INFO.uap
3. Save the structure.

You must not enter any data into the user archive Group-Object-List. The data will be entered during runtime.

Add User rights

In CEMAT Standard additional User rights must be added to the PCS7 User rights in order to define more detailed who is authorized for which action.

In the WinCC Explorer click open the **User Administrator** and add the rights from No. 18 to No. 28 according to the following list:



Under OS Configuration (Single-User System) you find a table, which shows how the user rights are assigned to the CEMAT Functions.

CEMAT specific preparations

Before you start with the configuration of the OS you have to copy the CEMAT Functions in to your Project. These are Global Script Actions, Project Functions, Standard Pictures and Bitmaps. Please proceed as follows:

- Close the WinCC Explorer

- Copy the Global Scripts from 'D:\CEM_V6\WinCC\Library' into the PCS7 Project ...\wincproj\<OS>\Library'.

- Copy from "D:\CEM_V6\WinCC\GraCS' all Objects into the PCS7 Project ...'wincproj<OS>\GraCS'
  The following 5 standard pictures contain adaptations for CEMAT. This adaptations are not relevant for a Server. **Don't overwrite the pictures!**

@Overview1.pdl	Overview Range
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@CSIGQuit.PDL	Acknowledge fault function
@HornQuit.PDL	Acknowledge horn function

- Copy file @PCS7Typicals.cfg from 'D:\CEM_V6\WinCC\WScripts' into the PCS7 Project '<OS>\WScripts'. (File @PCS7Typicals.cfg contains the list of attributes which will not be overwritten with update block icons).

- Open the WinCC Project.

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

Note: In case you have overwritten the standard pictures, you have to run the OS-Project-Editor again in order to get the original PDLs from PCS7 standard.

Preparation of the Template Pictures

With the CEMAT standard different template pictures are copied into the project.

The picture **@C_PCS7Typicals.pdl** and **@C_PCS7Typicals_V61.pdl** contain the default symbols for automatic generation of block icons from SIMATIC Manager.

- The picture **@C_PCS7Typicals.pdl** contains the block icons which were already used in CEMAT V6.0. This picture should be used as template if the project was already created with CEMAT V6.0 and shall be upgraded to CEMAT V6.1 with minimum effort.
- The block icons in picture **@C_PCS7Typicals_V61.pdl** have enhanced functionality (if an object creates an alarm and you open the picture via loop-in-alarm function the respective object gets marked).
- For the automatic generation of block icons PCS7 V6.1 uses the picture @PCS7Typicals.pdl or the pictures with this name plus an extension (e. g. @PCS7Typical1.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.pdl which contains the symbols you want to use in the project.

The picture **@C_Template.pdl** contains the default symbols for manual connection in WinCC (using wizard).

- The block icons in picture **@C_Template.pdl** correspond to the block icons which were used in CEMAT V6.0.
- The template picture @C_Template.pdl you also have to adapt and you have to save it under a different name (@Template1.pdl) in order not to overwrite the picture in case of an Update.

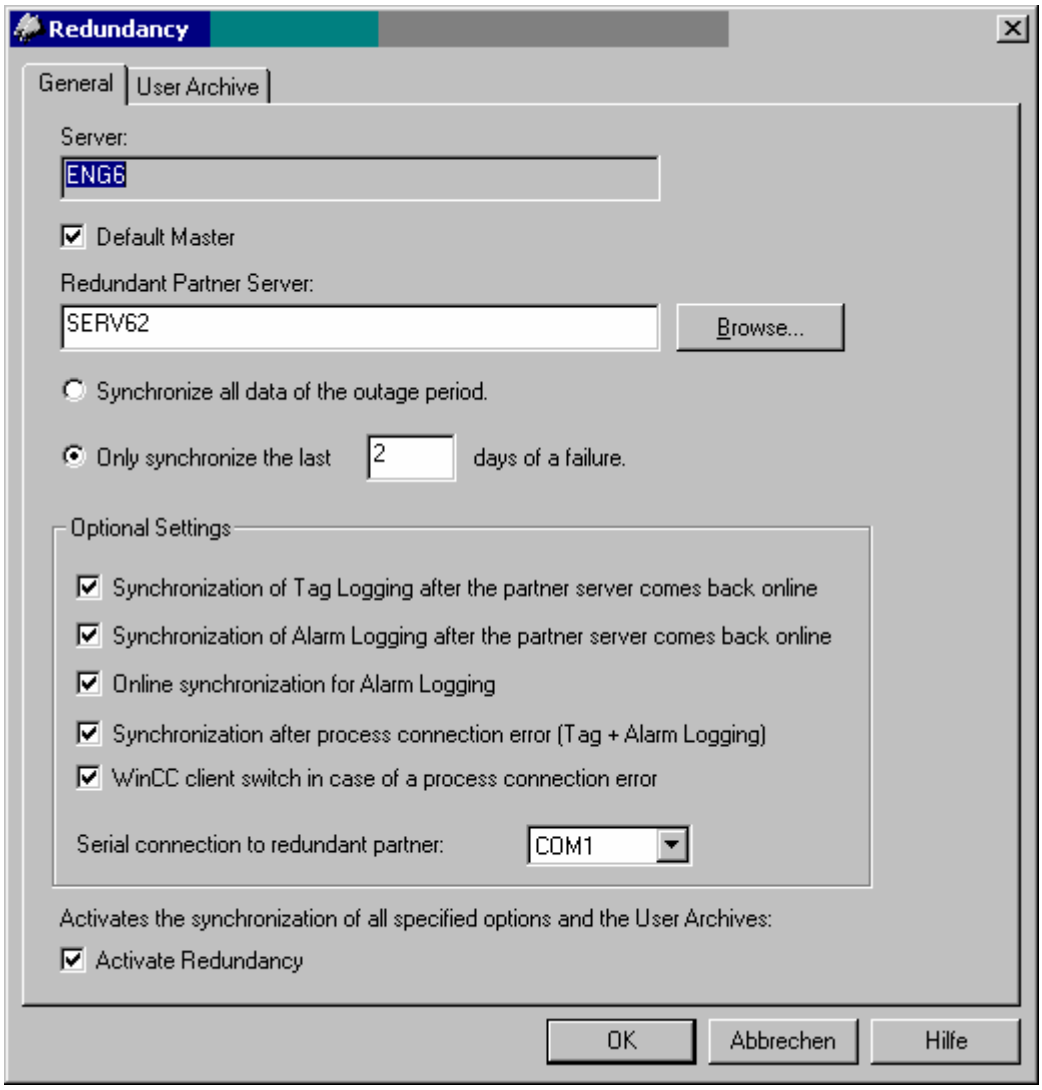
Further information regarding Template pictures you find in chapter OS Engineering.

Caution: For some of the project versions of CEMAT, modified symbols are required. These examples you will find in the template pictures as well. 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.

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.



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” or “Anwenderzyklus 1”. The default value of this is 2000ms. You have to adapt this value according to your requirement, e.g. 1000ms.

OS Project Editor

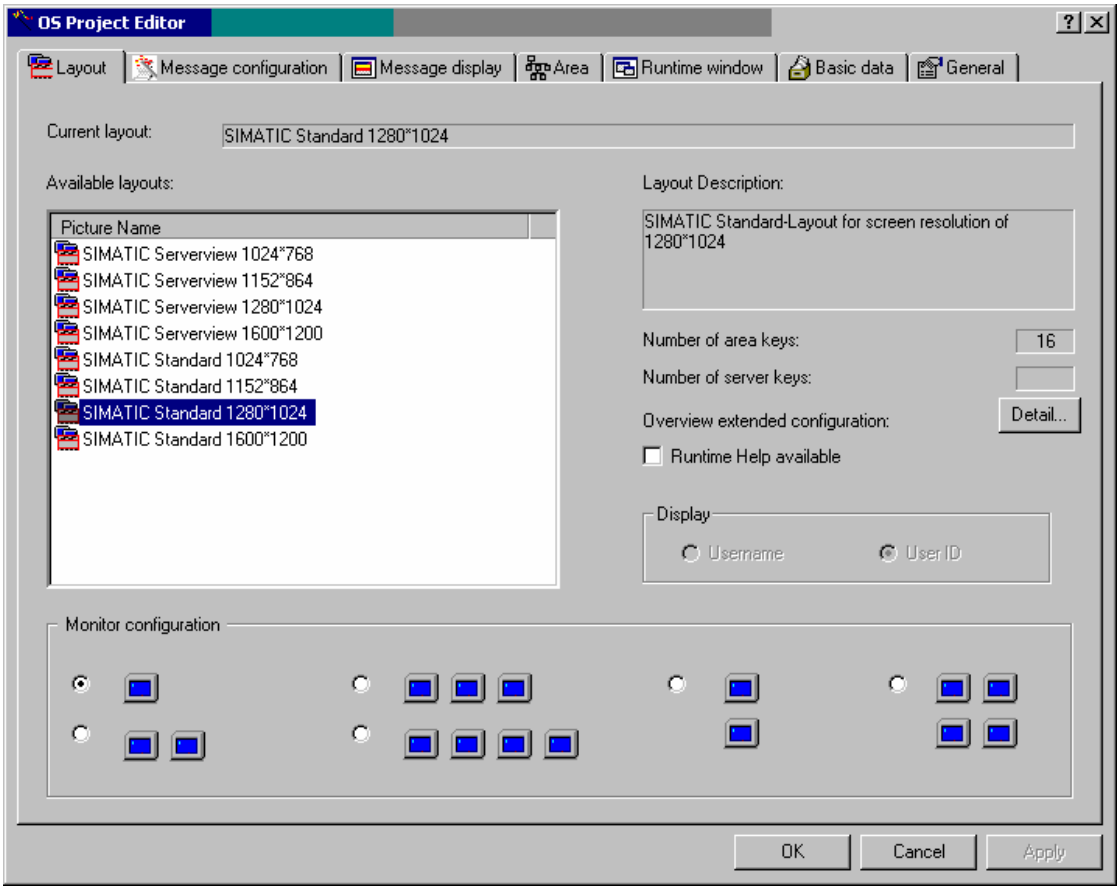
Settings for the user interface used by the plant operator for monitoring and controlling the plant during process operation In the OS Project Editor.

A detailed description to this item you find in the PCS7 Configuration Manual Operator Station. The following pages describe only the settings relevant for CEMAT.

In Folder Layout choose the right screen resolution. It may be SIMATIC Standard 1280*1024 or SIMATIC Serverview 1280*1024 if Selection for Server is required.

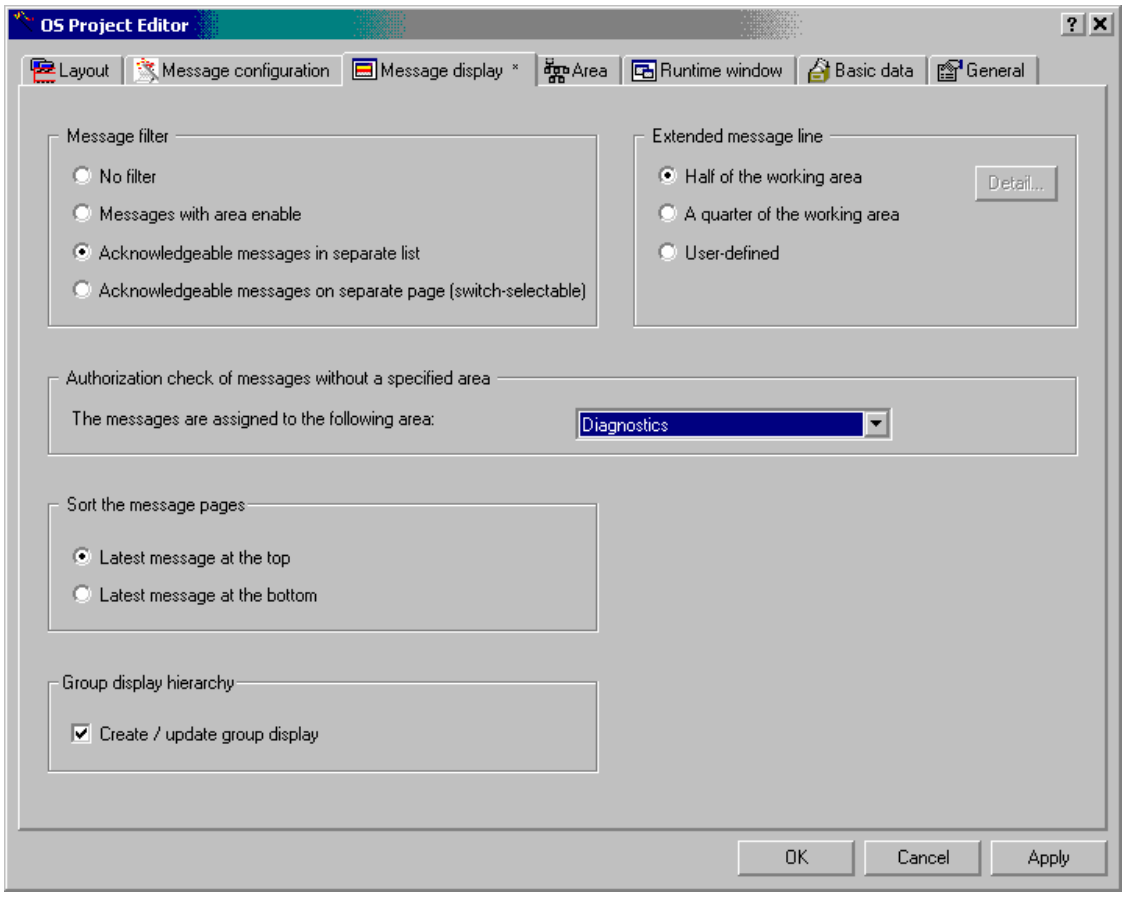
Also select the Number of horizontal and vertical area keys and the number of Servers under Detail (Picture Tree) and the Monitor Configuration of your PC.

Example for the Layout of an OS Client:



Concerning the OS Areas: Please keep in mind that the CEMAT message system can manage a maximum of 16 OS Areas at the moment!

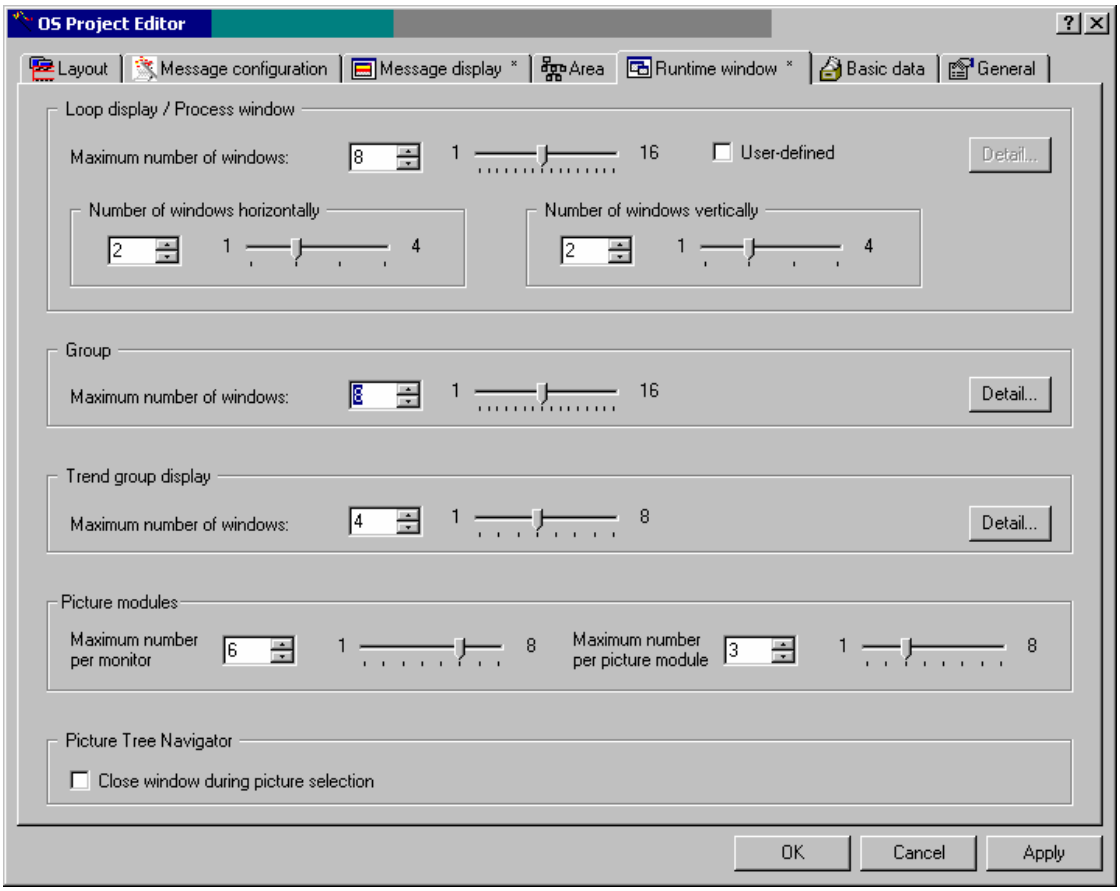
Message display for a Cliente Station:



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

The group display should be created and updated automatically.

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.

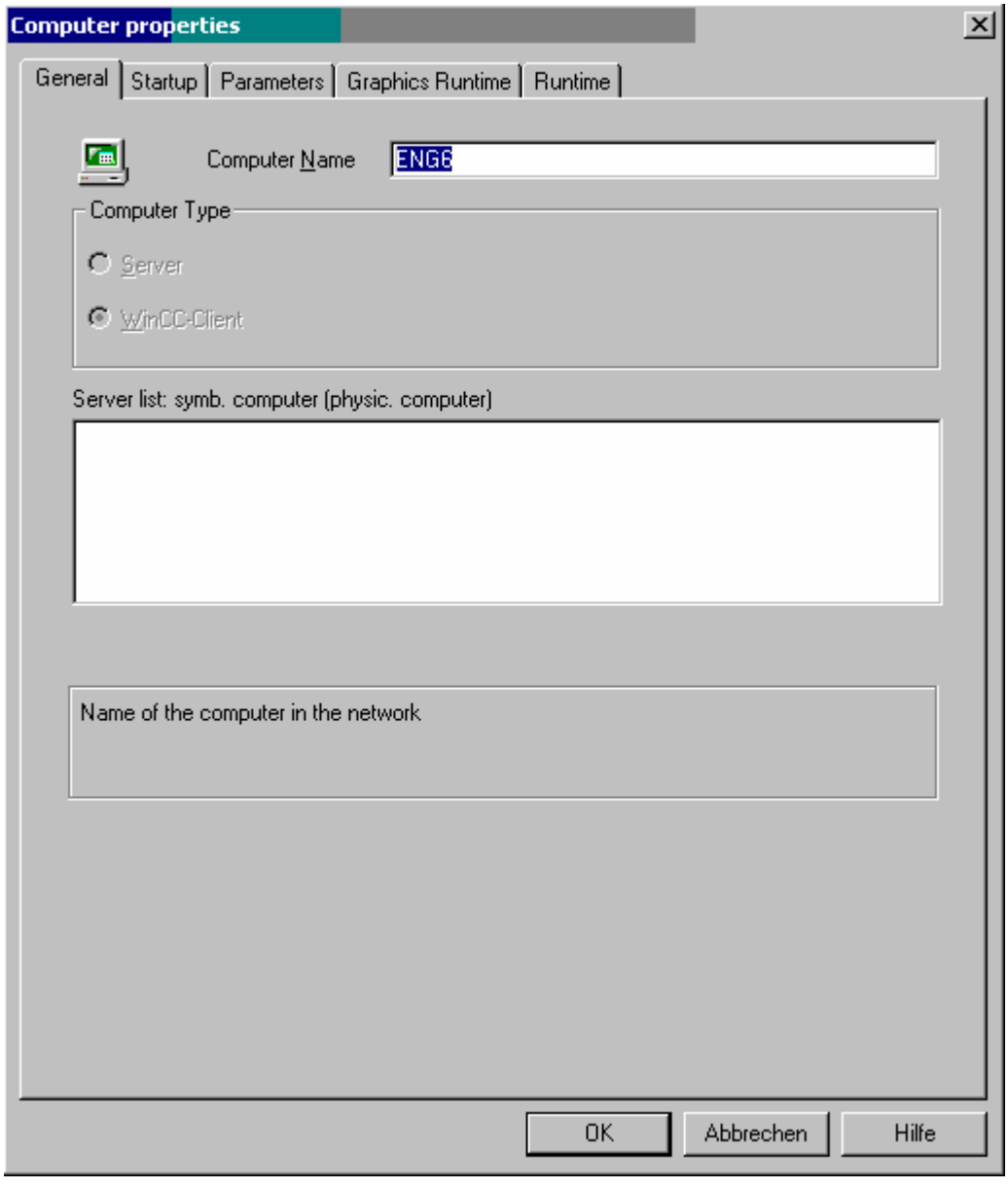
With “OK” all settings are carried out.

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).

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.

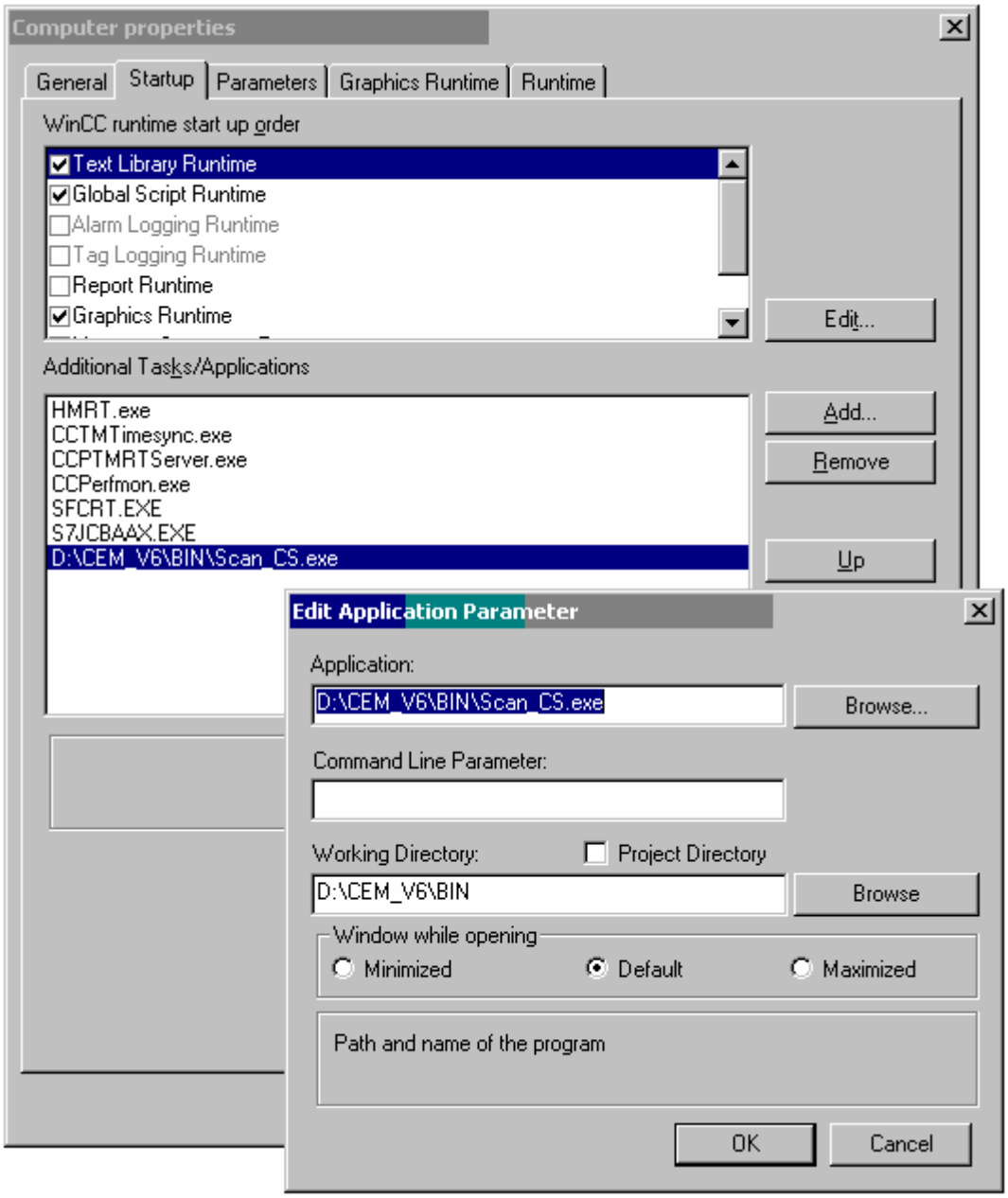
Folder General shows the Computer Name and the Computer Type



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

Under Additional Tasks/Applications you have to add the Scan_CS.exe to the Startup list. The application is located in Directory D:\CEM_V6\BIN.

Scan_CS.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

Open the Tag Management and add the internal variables for CEMAT:

Add new Group “CEMAT”

Within this group add the following variables:

Name	Data Type	update
C_VIEW_TAG_A	Binary	Computer-local update
C_VIEW_TAG_D	Binary	Computer-local update
C_ServerName	Text tag 16bit	Computer-local update

If the message selection is carried out via alarm line (V6.0 principle), you additionally have to add the following variables. This is mainly used in case of migrated projects. If the message selection is carried out via user (V6.1) the following variables are not required:

Name	Data Type	update
C_AcknowledgeHorn	Unsigned 32-Bit Value	Project-wide update
C_<ComputerName>_AlarmSound	Singed 16-Bit Value	Computer-local update

Replace <ComputerName> by the station name, e. g. C_CLIENT61_AlarmSound.

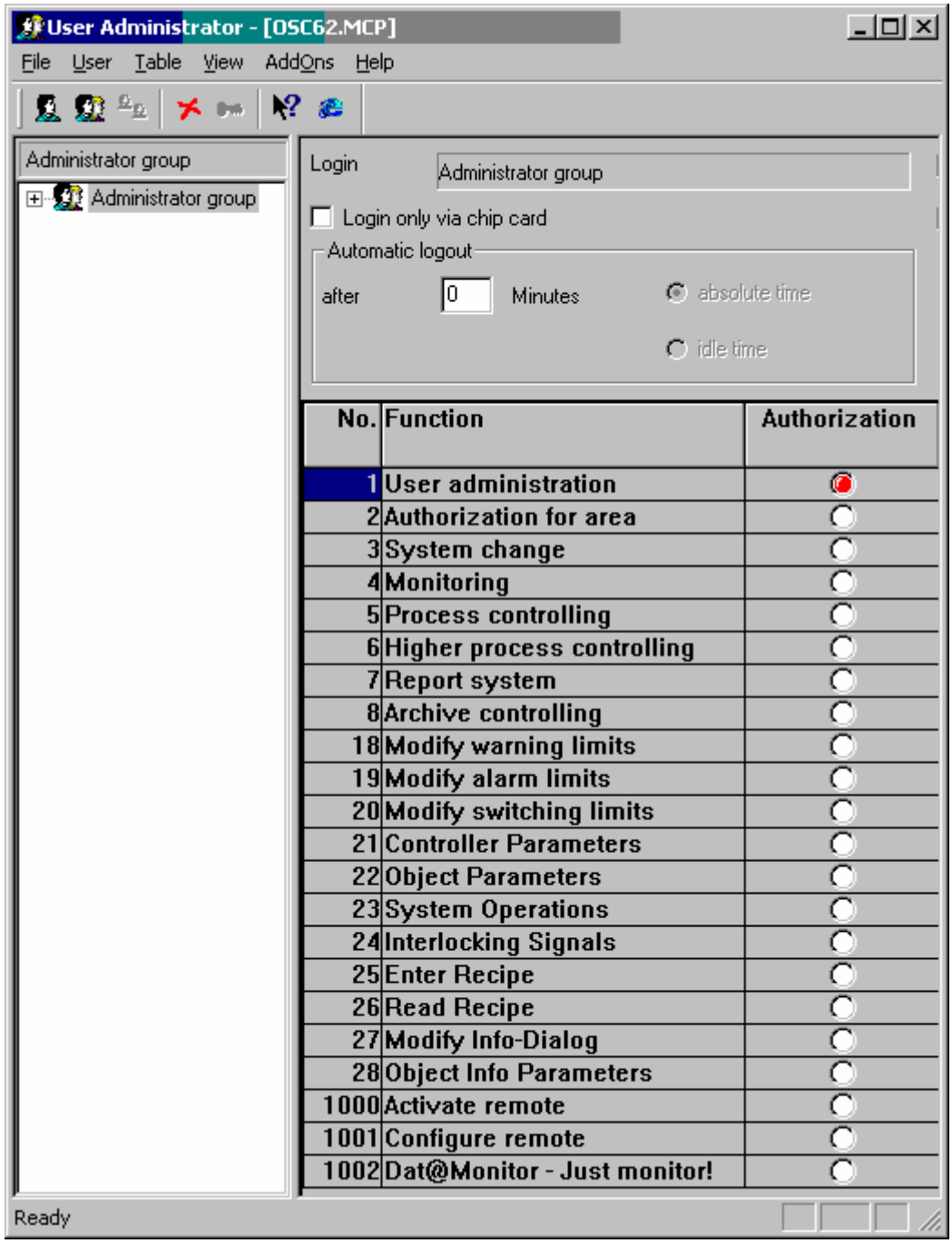
Note: For the computer name use exact characters (Capital letters!). Otherwise the function will not work.

Horn Tags

During the horn configuration additional tags are automatically created. You will find these variables in group “Horn”.

Add User rights

In the WinCC Explorer click open the **User Administrator** and add the rights from No. 18 to No. 28 according to the following list:



Under OS Configuration (Single-User System) you find a table, which shows how the user rights are assigned to the CEMAT Functions.

Later you have to define user groups and users to which you select specific authorisations. However the user rights related to specific areas can only be configured after the package is loaded.

CEMAT specific preparations

- Close the WinCC Explorer
- Copy the Global Scripts from 'D:\CEM_V6\WinCC\Library' into the PCS7 Project ...\wincproj\<OS>\Library'.
- Copy from ''D:\CEM_V6\WinCC\GraCS' all Objects into the PCS7 Project ...'wincproj<OS>\GraCS'
  The following standard pictures contain adaptations for CEMAT. The existing pictures must be overwritten by the pictures from the CEM_V6 directory.

@Overview1.pdl	Overview Range
@AlarmOneLine.pdl	Alarm line
@Buttons11.pdl	Button keys1
@CSIGQuit.PDL	Acknowledge fault function
@HornQuit.PDL	Acknowledge horn function

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

Generating and loading of the Server Data

To make the Server data available for the Client, a so-called package must be created for the Server-Project and loaded in the Client-Project.

In PCS7 V6 the package is automatically created and updated during OS Compile. It can also be created manually using SIMATIC Manager (under Options -> OS -> Generate Server Data.

The package name consists of PCS7-Project name and OS-Project name and it is located in the Server project under the computer name (in this case this is the name of the Engineering Station). There you find a folder “Packages”.

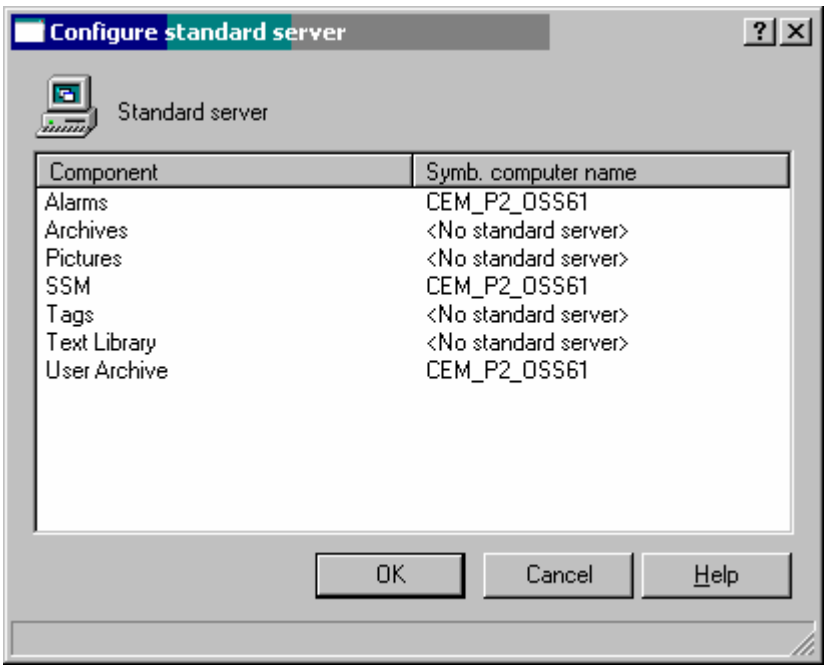
Loading of the Server Data in a Client Project

To load a Server data into a Client for the first time, open the Client-Project on the Engineering Station and use context menu of Server data and select Menu option „Loading“.

- In the "Open"-Dialog select the Server-Project and within that the computer name (Name if the Engineering Station). There you find a folder “Packages”.
- In the folder Packages select the package-file (.pck) and click on "open". The Serverdata will be copied into folder „packages“ and is available in the Client.

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

- Select the standard server for the components Alarms, SSM and User Archive.



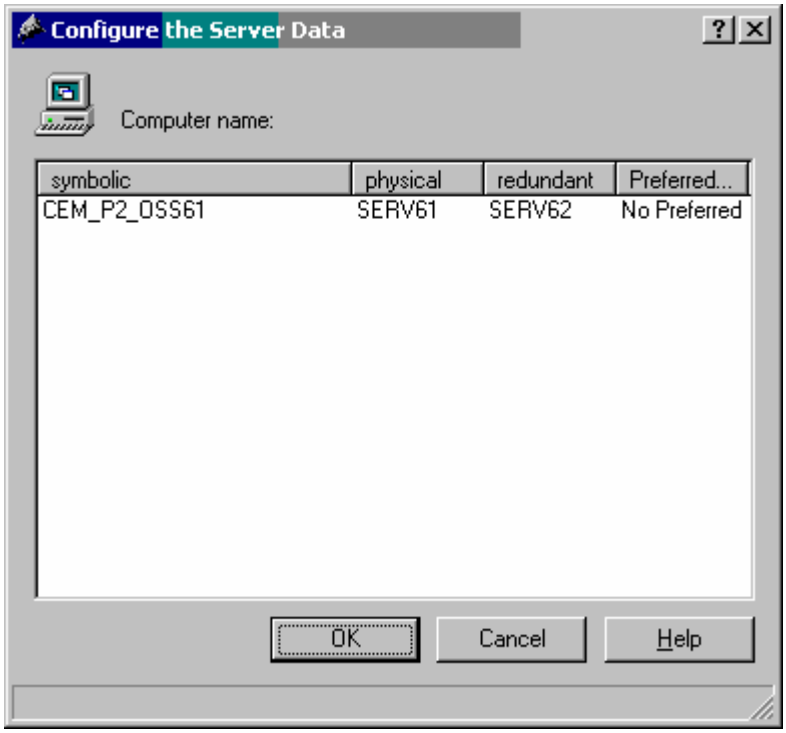
Once the Server data is loaded and configured the package gets automatically updated during compilation of the OS-Data.

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.

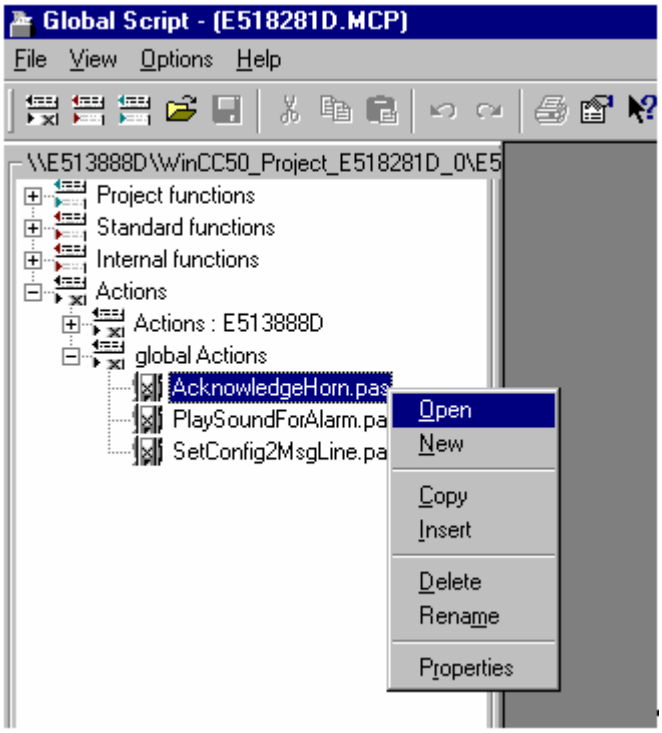
Trigger for Action AcknowledgeHorn

The following settings under Global Scripts are only necessary if the message selection is carried out via alarm line (V6.0 principle). This is mainly used in case of migrated projects. Only in this case the function AcknowledgeHorn.pas is existing.

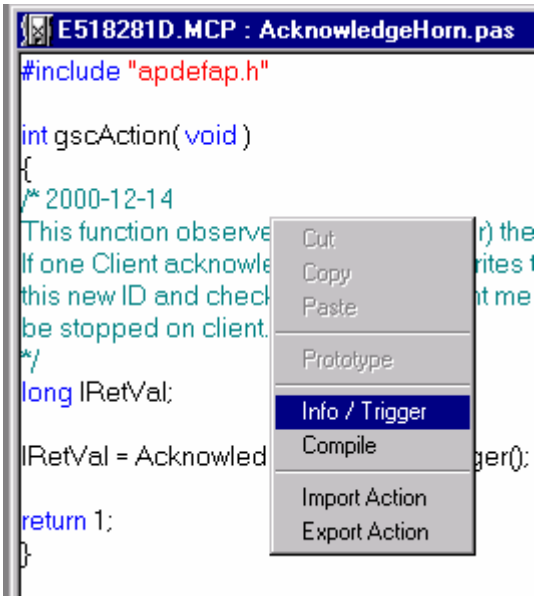
If the message selection is carried out via user (V6.1) the following settings are not required.

After loading the package(s) the Global Script Action ‘AcknowledgeHorn.pas‘ has to be adapted. Please follow the instructions:

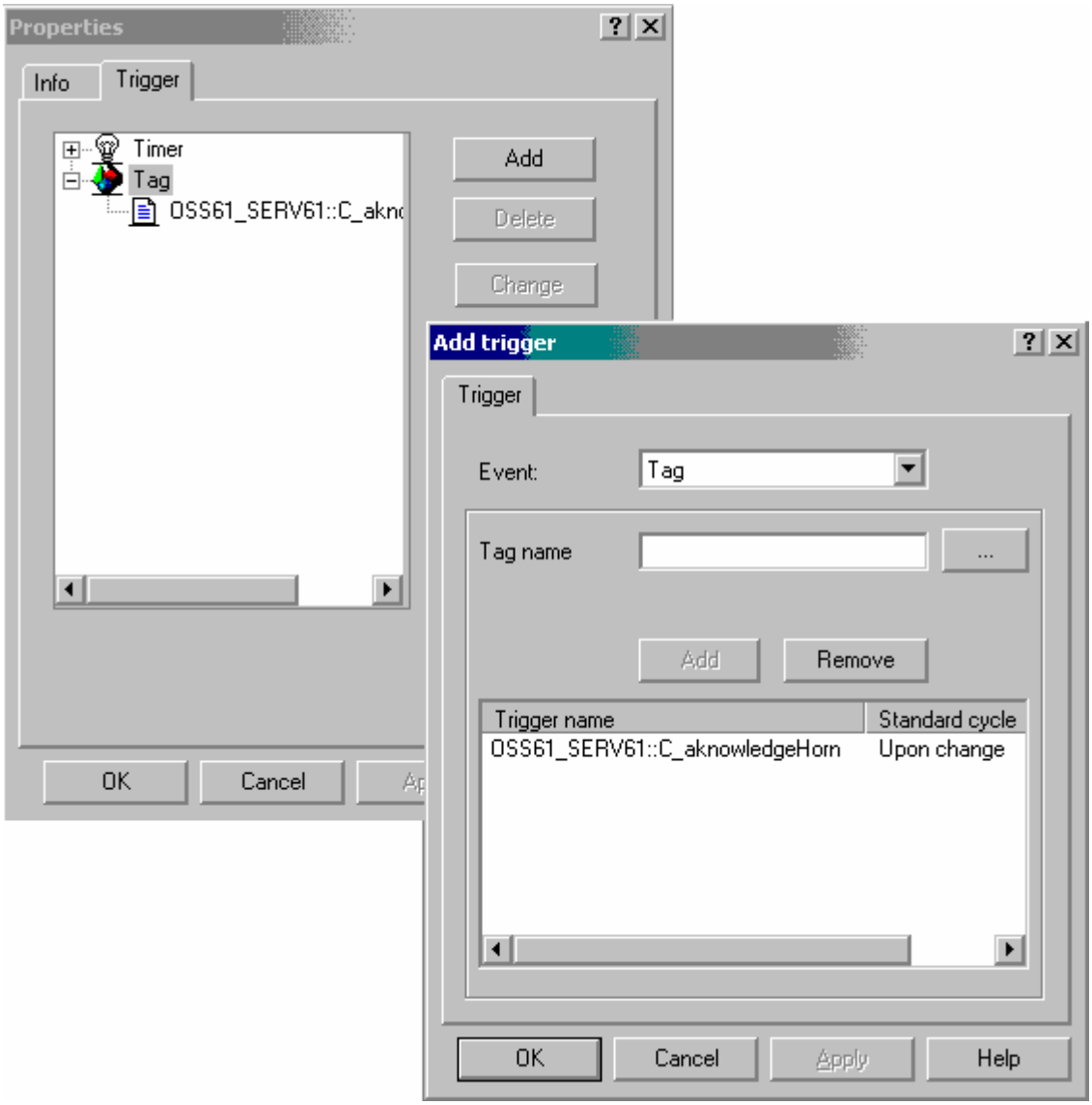
- Open the action ‘AcknowledgeHorn.pas‘.



For each Server install one new Trigger with cycle ‘upon change‘. Press right mouse button and select “Info / Trigger”



Select the variable C_AcknowledgeHorn from the Tag Management of the corresponding Servers.



Selection of area specific rights

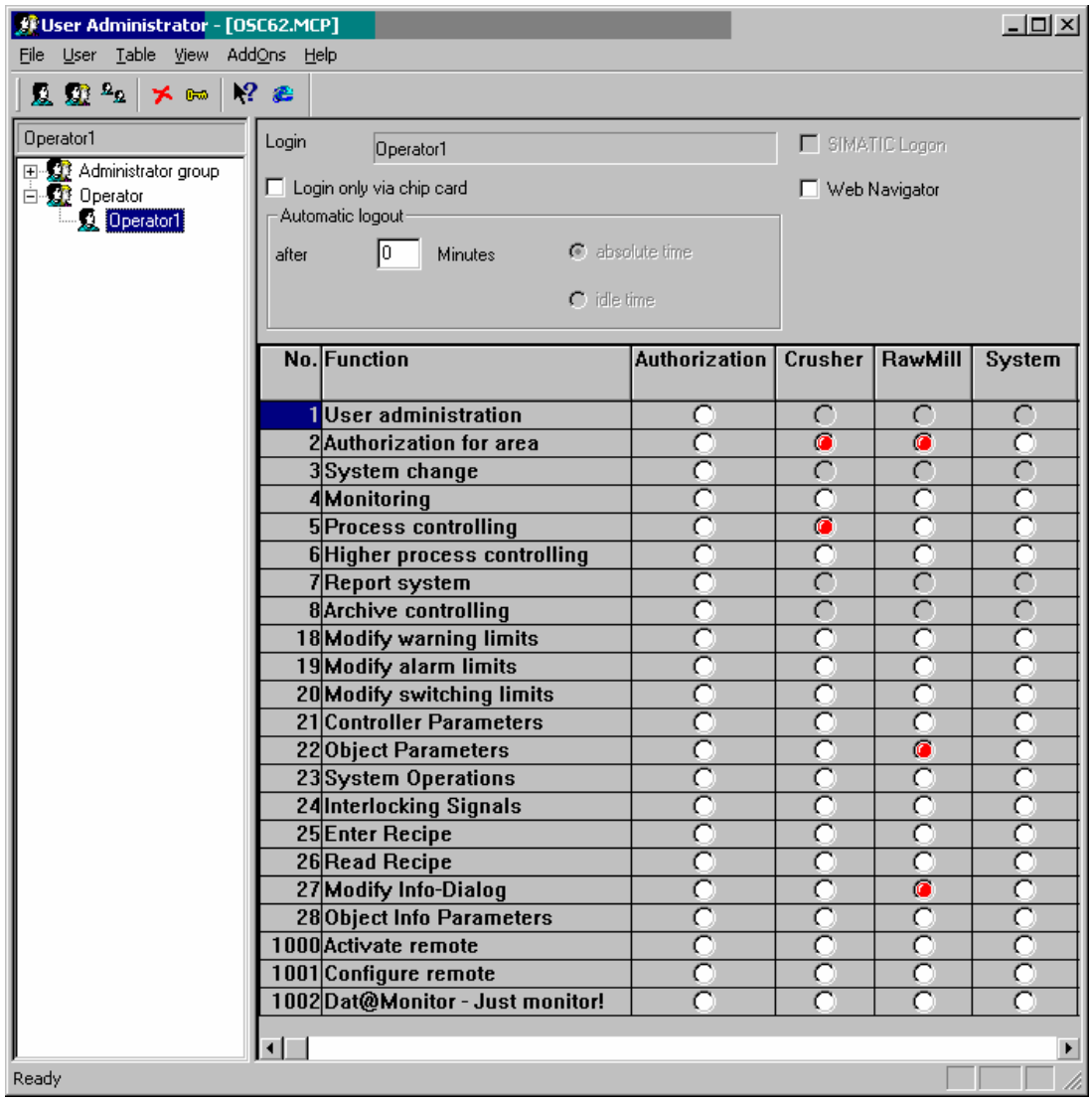
After the package is loaded the user groups and the users can be defined and their rights can be defined area specific.

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

With the right 2 “Authorization for area” it is defined which area can be viewed.

With the right 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 „Operator1“ is allowed to:

open, watch	→ **Crusher, RawMill**
operate	→ **Crusher**

The area **System** is disabled.

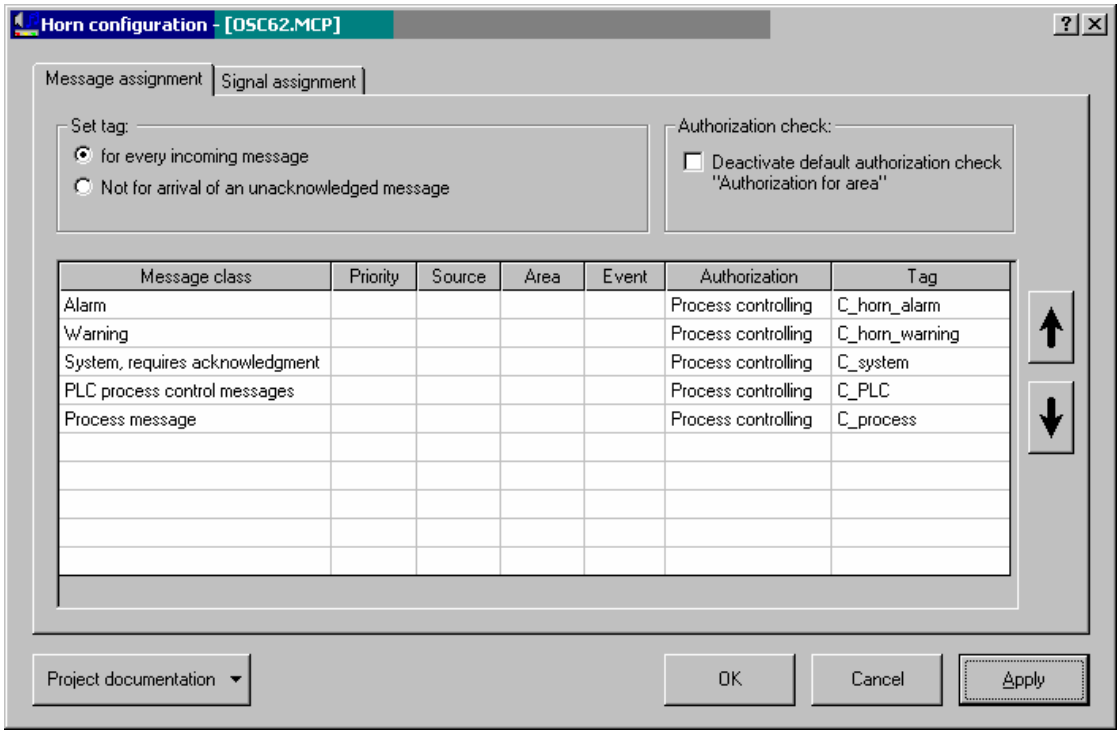
Only messages from areas which are enabled for operation

Messages are only shown for areas which are enabled for operation. Operator1 will get only the messages from the Crusher area and he can acknowledge only these messages.

Horn configuration

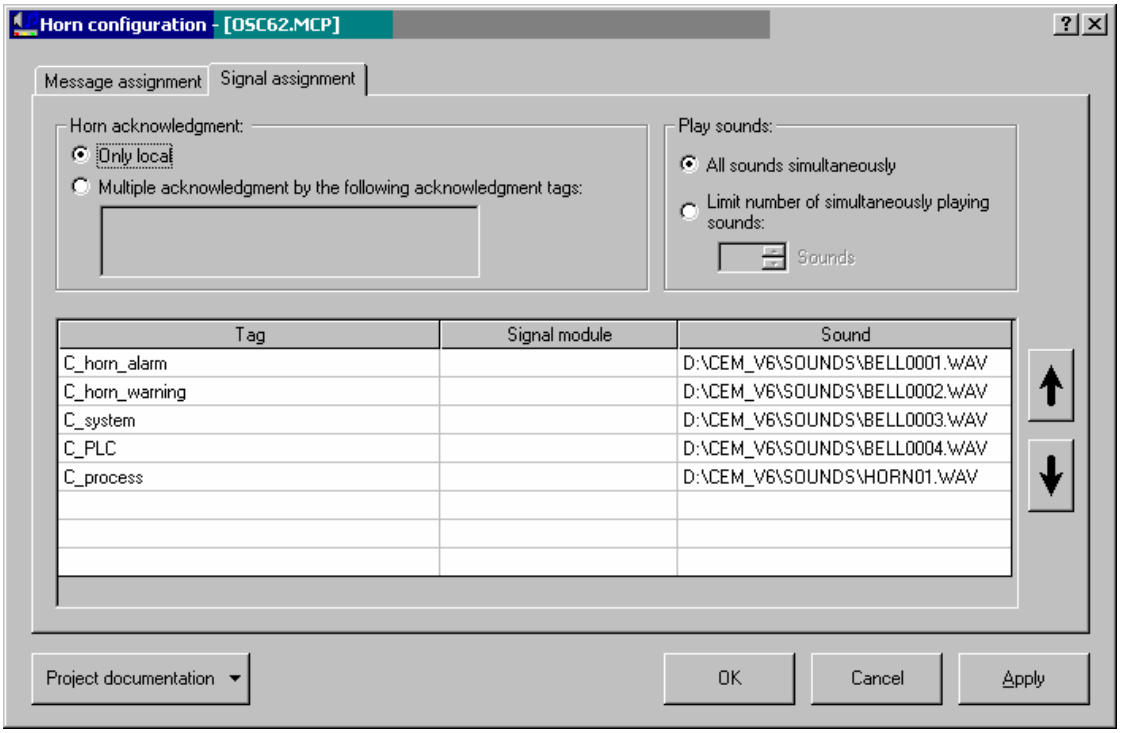
From CEMAT V6.1 you can define a separate Sound for each Message class. The horn must be configured as follows:

1. First define in tab “Message assignments” all message classes which should create a sound. These are Alarms, Warnings, System messages which require acknowledgement, PLC Process control message and Process messages. If you use other message classes with alarms in your project you have to define these as well.
2. Assign the Authorization „Process controlling“ (with this the Authorization is enabled per Area).
3. Per message class, define an internal variable with computer local actualisation. Insert the variable name. With “Apply” the variable is automatically created in the tag management.
4. The definition can also be carried out per Area.



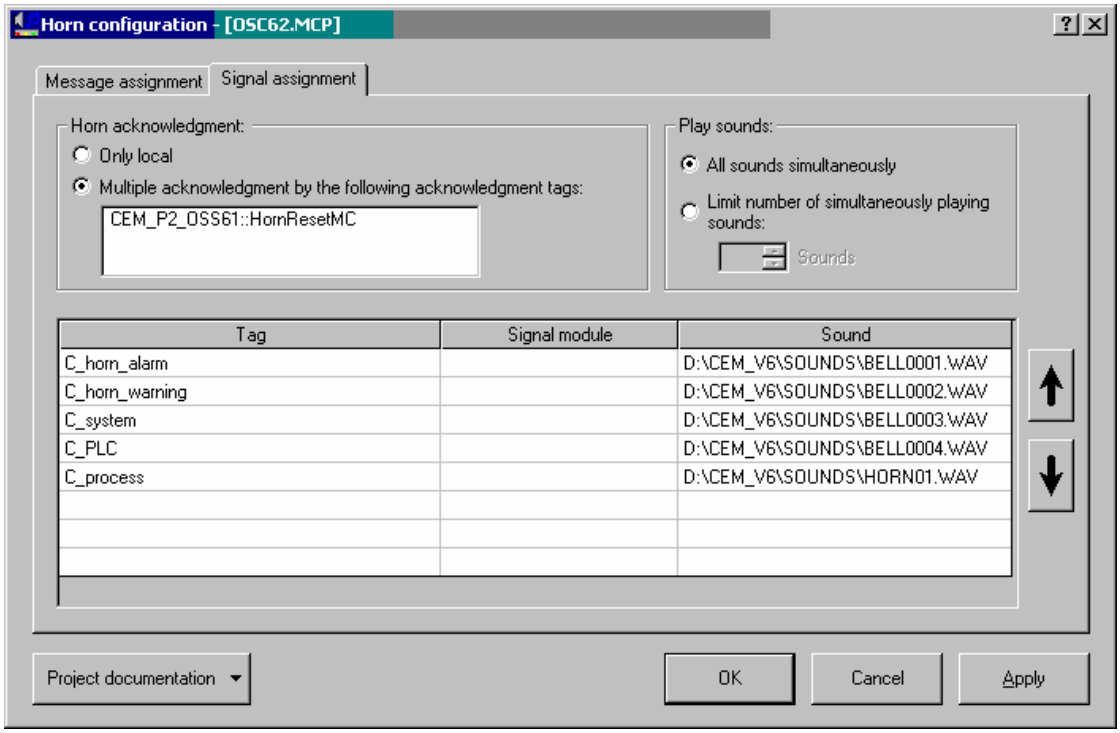
5. In tab “Signal assignment” you can define for each variable an individual sound. A sample of different sounds you find in directory D:\CEM_V6\Sounds.
   The following configuration depends on whether you want to acknowledge the Horn on each Client independently or you want to acknowledge the Horn for different Clients at the same time.

   For Stand-alone WinCC Clients the Horn acknowledgement is “Only local”. Here you only have to select a sound file for each tag.



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.

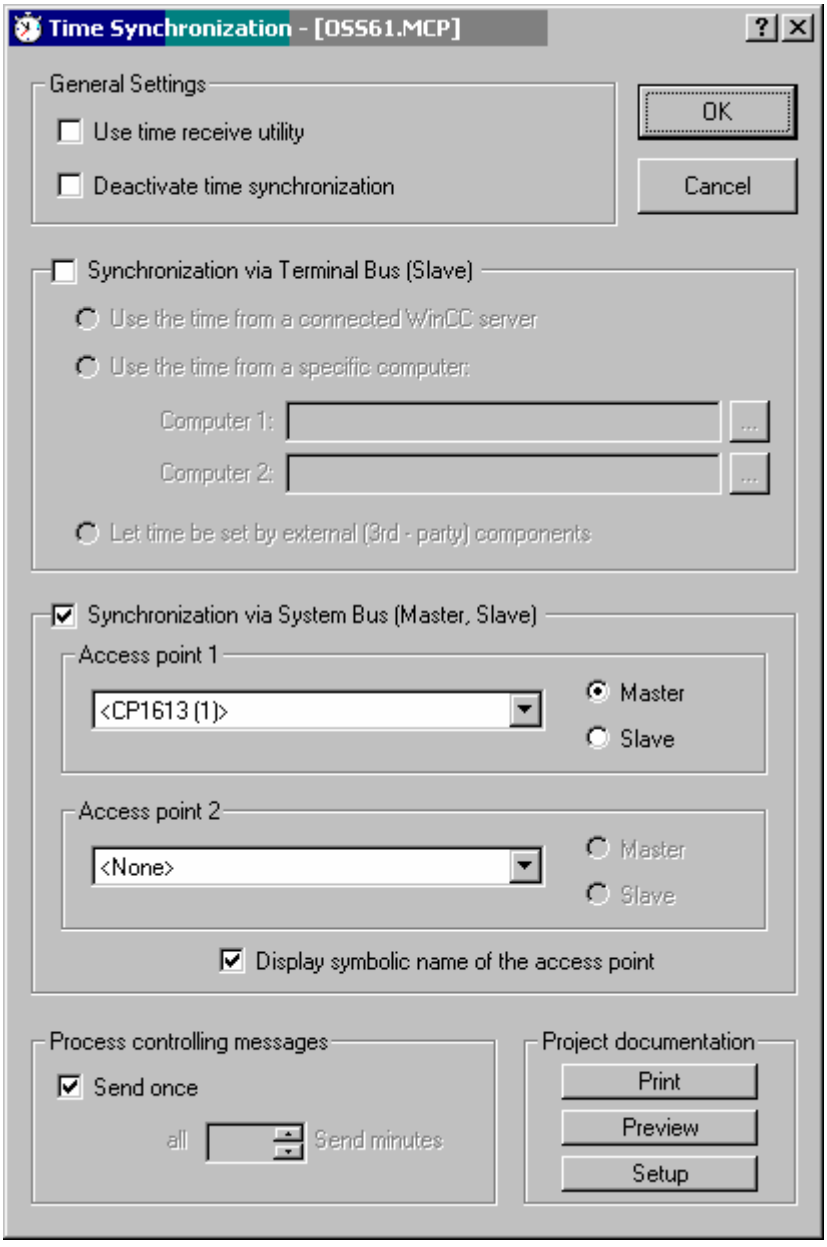
In this case select the Horn acknowledgement option „Multiple acknowledgement by the following acknowledgement tags“ and after that select the reset variable from the internal variables of the Server.
Select a sound file for each tag.



Time Synchronization

Time Synchronization Server

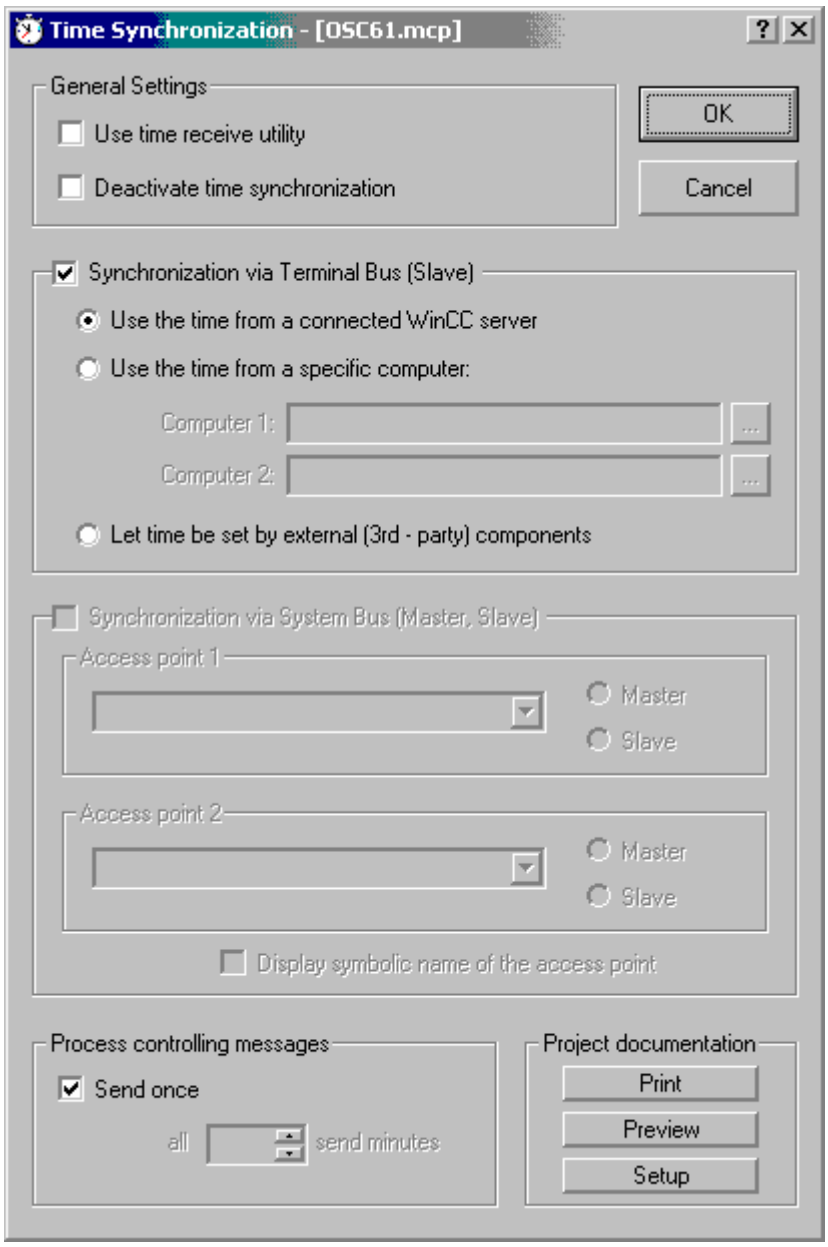
The following picture shows the settings for the time synchronization. In this example the Server PCs (Primary and Standby) will be the time Master:



Note: If the Engineering Station has no CP1613 the access point can not directly be selected. In this case select the option “Display symbolic name of the access point” and select the CP1613 from the list.

Time Synchronization OS Client

The OS Client PC gets the time via Terminal Bus from the Time Master. In this example the Time Master is the Server:



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.

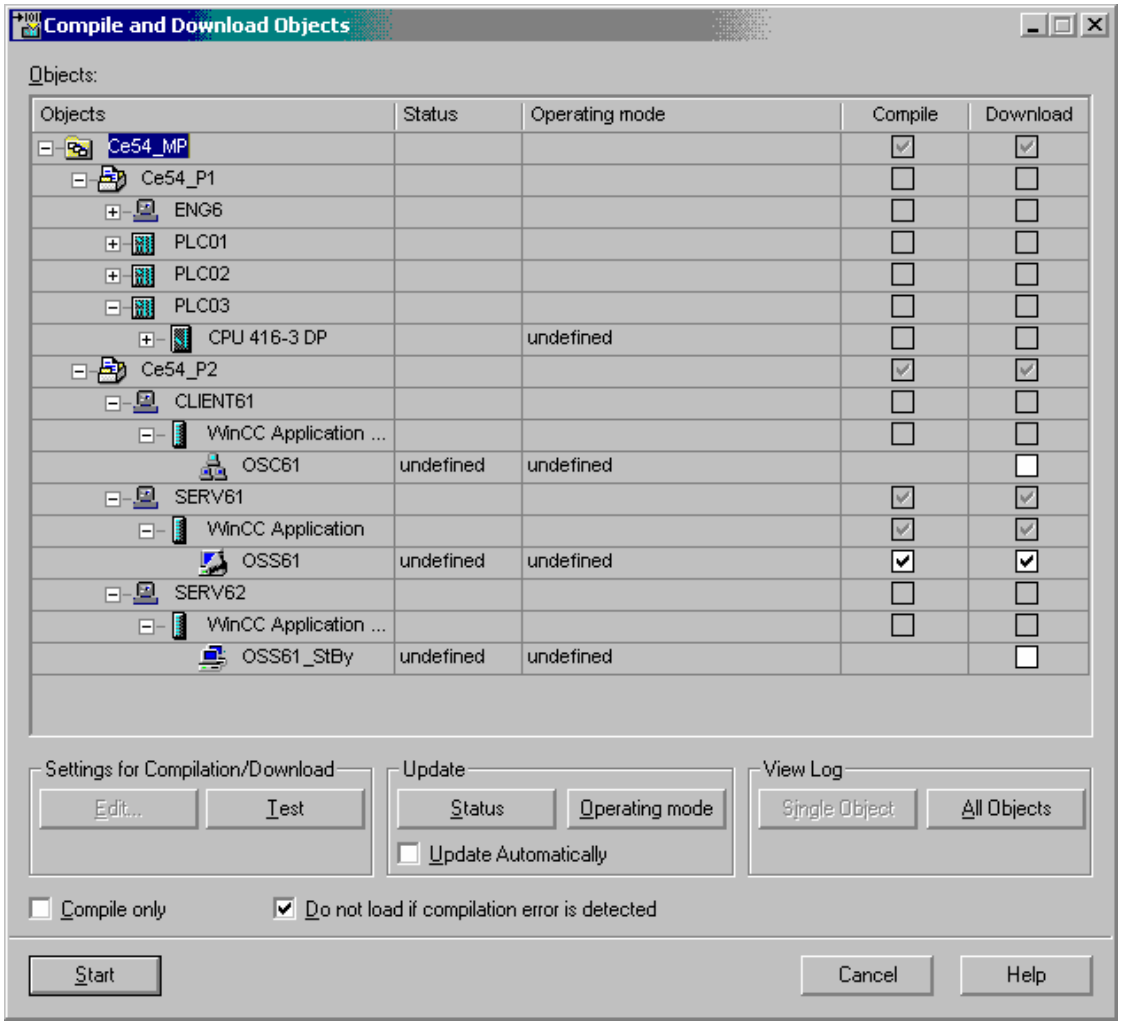
Important: The modifications in the OS-Projects for the Server, Standby-Server and Clients are never performed directly on the OS Stations. Any modification has to be carried out on the Engineering station and then being downloaded are different OS Stations.

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:

1. Compile the PLC-Program (from CFC)
2. Compile the OS Project(s)
   The compilation 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.
3. Now you can carry out the Project-Download from the Engineering Station to the Server Station. The Download is carried out from the SIMATIC Manager by selecting the OS-Project and pressing Download Button.
4. Wait for the “Download successfully completed” – Message before you continue.
5. Now you can carry out the Project Download from the Engineering Station to the Standby-Server Station. (Same procedure as for the Main Server).
6. After the “Download successfully completed” – Message the Project Download is finished!
7. For the download into the Single User Station use the same procedure.
8. If it is necessary to update the Clients as well (may not always be required), use the same procedure for each Client.

"Make" Dialog: Compile and Download Objects:



The Picture shows the required Selection to update the Servers (Primary and Standby Server) and the Client.

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 actualise 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 primary 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 Standby-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.

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. CEM_P2_OSS61

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 chose 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

Ausgabe : 28.01.2004
D:\CEM_V6\DOCU\English\Engineering\04_Assignments.doc

Assignments FB, FC, DB

FB

000 499	PCS7
500 999	ILS
1000 1199	CEMAT
1200 2047	User

FC

001 400	PCS7
500 999	ILS
1000 1199	CEMAT
1200 1399	User
1400 2047 *	CFC-Tasks

DB

001 399	PCS7
400 599	User
600 999	CEMAT
1000 4095	CFC-Instances

UDT

001 999	ILS
1000 1499	CEMAT
1500 4095	User

Memory, Timer, Counter

Memory	
0000.0 0000.7	PCS7
0001.0 0099.7 *	ILS
0100.0 0199.7	CEMAT
0200.0 2047.7	User

The biggest memory address depends on the selected CPU Type.

ILS and 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

Ausgabe: 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\05_Engineering_Examples.doc

General

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

In part 2 we show you the correct connections for the basic functions.

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

In the function diagrams of this description, the following letter codes are used to define the CEMAT object types:

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

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 (MAU) should then be used for the further use in the program.

Start/Operation Interlock of the Group

To permit on status call-up the interlock setting to be displayed, such interlock signals (also internal flags) should be maintained on 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, such as, transport into various destination silos.

Advantage:

- reduced configuration effort
- reduced program run-time

The classical route module is particularly suitable for long transport routes that use many drives. If route modules are used, an individual status call can be made for each route and so permit this diagnostic function to be invoked for each section.

The question whether a selection module, a route module or various groups should be used must be decided individually.
This section merely describes the various possibilities.

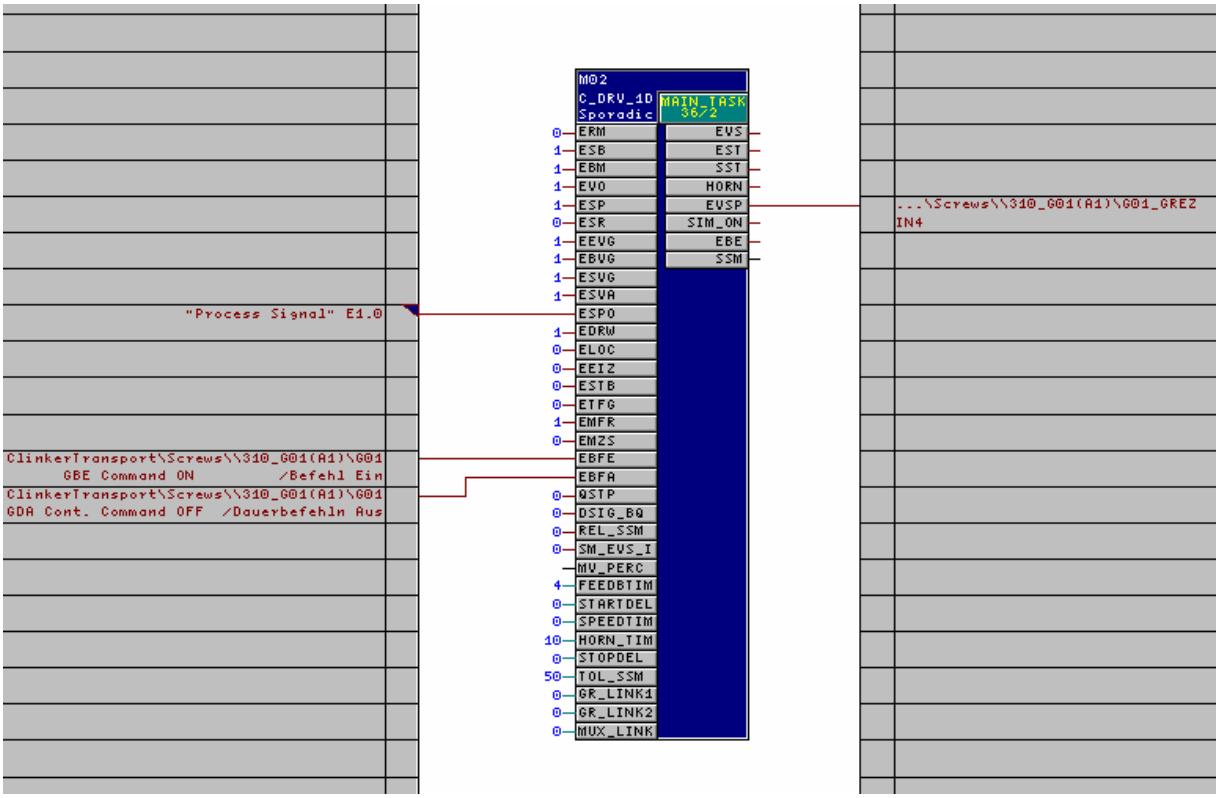
Examples

Unidirectional Drive, Sporadic Operation

Sporadically running drives are activated with the start signal GBE of the Group or WBE of the route and will be started and stopped via interface ESPO. 0-Signal on interface stops the drive without resetting the command memory EKS. The drive remains activated and restarts automatically with 1-Signal on this interface.

To stop the motor completely 1-Signal at EBFA or 0-Signal at EBVG is required. If the motor is stopped by a fault, it must be restarted through the associated group. This interface does not work in local mode.

For the feedback to the group signal EVSP is used. EVSP has 1-Signal as long as the drive is activated.



Applications:

Heating probe
Lubrication equipment
Water pump

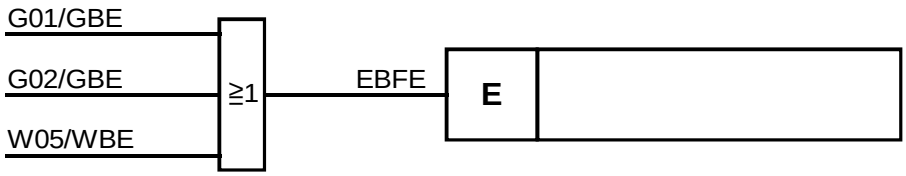
The same function exists for the bi-directional drive and for the valve.

Start Unidirectional Drive

In automatic operation, the ON command from the group or route starts the unidirectional drives.

Only these ON signals, but no interlock conditions, are connected to EBFE.

Example:

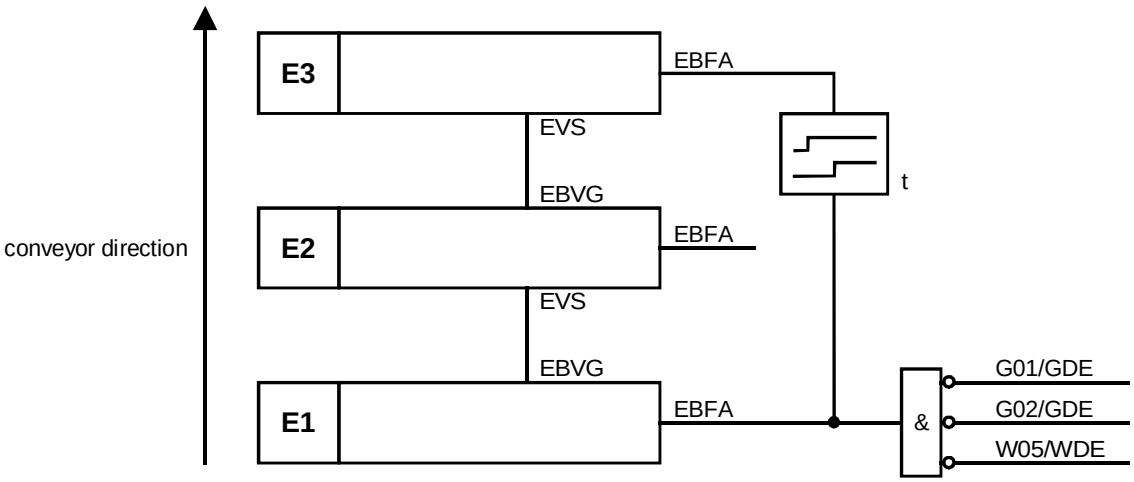


Note: A continuous signal must never be switched on the EBFE. After the acknowledgment of a fault condition, the start command must not be set immediately to “1” again.
Refer to Unidirectional Drive, Sporadic Operation.

Stop Unidirectional Drive

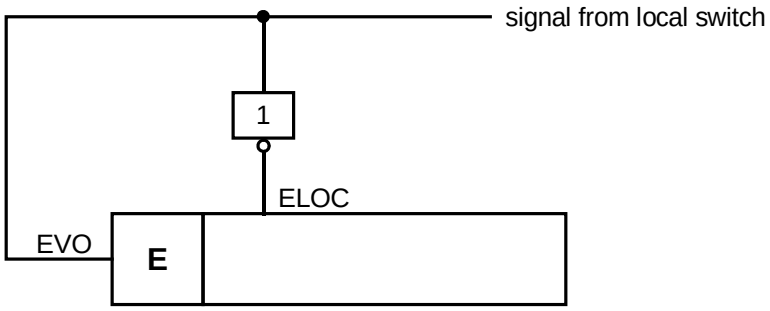
In automatic operation, drives are stopped with the negated ON commands from groups or routes, or with operation interlock.

Example:



Local Unidirectional Drive

It is possible to have a classical local operation, i.e., the hardware circuit starts the drive (contactor control). The programmable controller in this case only receives the signal from the local switch that is switched to the EVO parameter of the C_DRV_1D. This signal must still be switched negated on the ELOC interface.



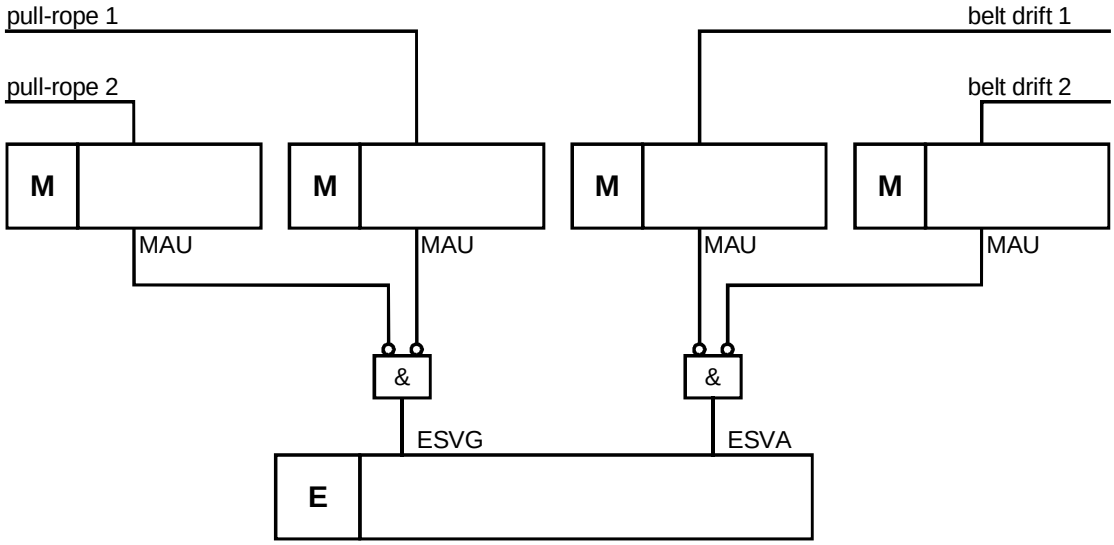
The local control realized in the programmable controller is used for the software solution.

The Start and Stop signals (of the local switch) are switched to the C_DRV_1D parameters.

Connect the ELOC interface (local activation) with the GLO signal of the group.

For the protection interlock, ensure that those signals that must always be active (e.g. pull-rope) are switched on the ESVG.

The protection interlock that is not to be active in local operation (e.g. belt drift) is switched on the ESVA.



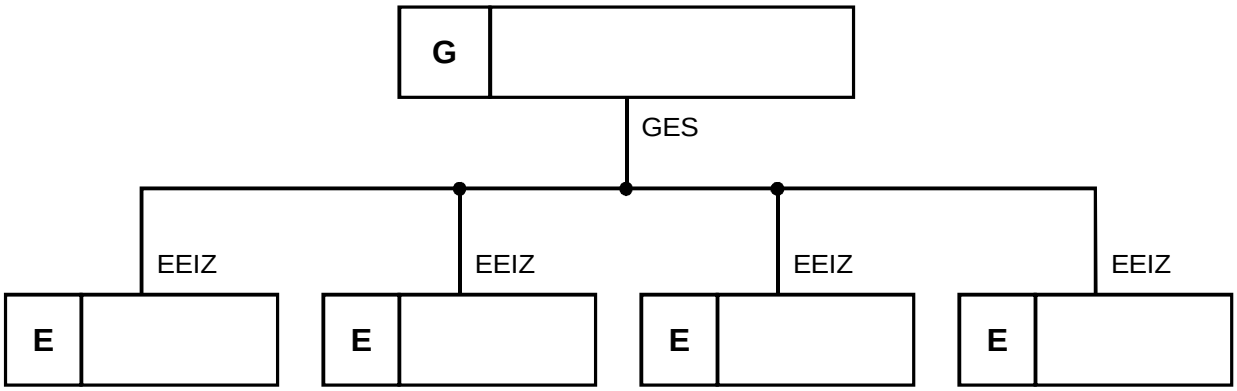
Single Start mode

In the Single Start mode, the individual drives can be started and stopped from the control panel (OS). It should be noted that all interlocks are effective. That is, the discharging drive must first run before a charging drive starts.

The Automatic and Single Start mode are formed by the group. The group must be switched to the appropriate mode.

We do not see any requirement for the single start mode in addition to the automatic operation and local operation.

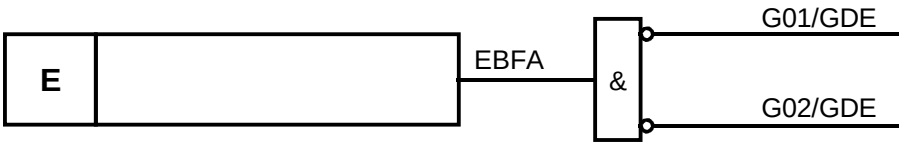
The GES signal of the group must be connected to interface ....EIZ of all drives (C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE).



The following notes must be observed to permit a smooth transition from “single” operational mode to “automatic”.

a) Take the GDA or WBA for **one** group or **one** route.

b) The following circuit should be used for multiple groups or routes.



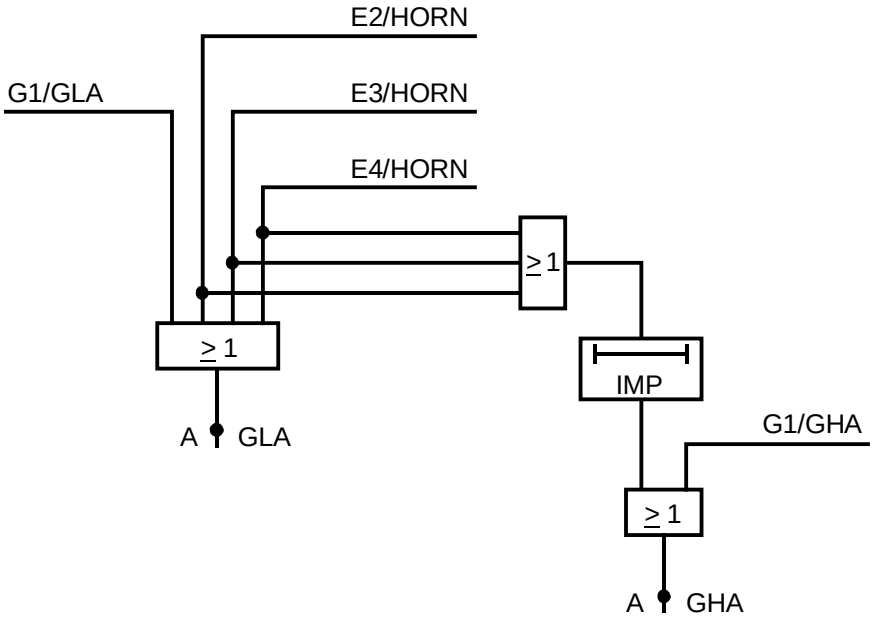
Start-up warning in the Single Start Mode

The group does not provide any start-up warning in the Single start mode.

After the individual start of a drive (C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE), the module starts the set wait time.
The drive is started only after this time expires.

The HORN output signal is “1” during this wait time and can be used to control the horn and lamp.

Example:



Calculated Values

Certain values are calculated during the plant configuration, e.g. total amounts from sub-amounts, temperature or pressure average values.

The values and measured values that are read using Analog Inputs are analog values for which a measuring value module must be programmed.

To improve the readability, the calculation should be performed before the invocation of the measuring value module.

The measuring value module permits a simple inclusion of the calculated values in the HMI system.

Limit Value Evaluation of Measured Values

The limit value signals are often used for start or operation interlocks of groups.

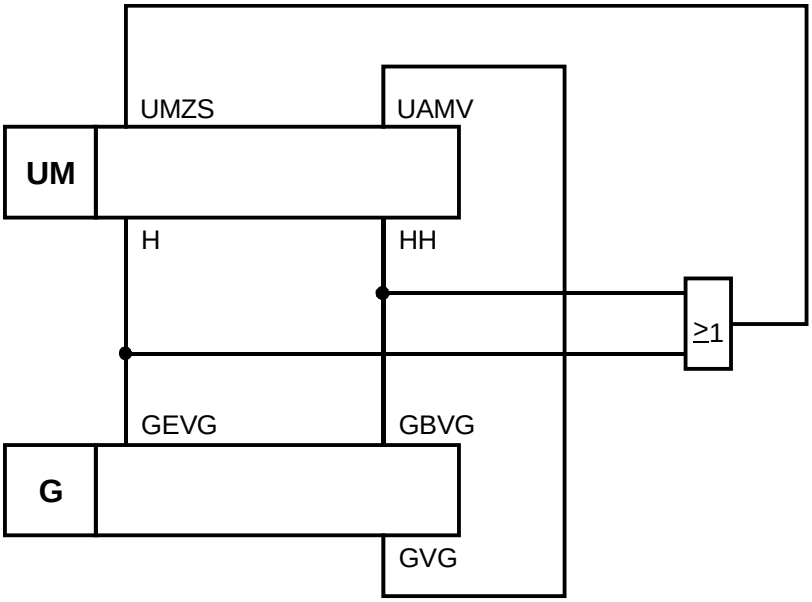
The following points should be observed:

- To prevent interlocks appearing on the group fault lamp, interconnect UMZS.
- If, for example, the alarm interlock is interconnected with GVG, the measuring value module provides alarm messages only for a running group.

The status call-up can at any time be used to request pending annunciations (also for a stationary group).

a) Example:

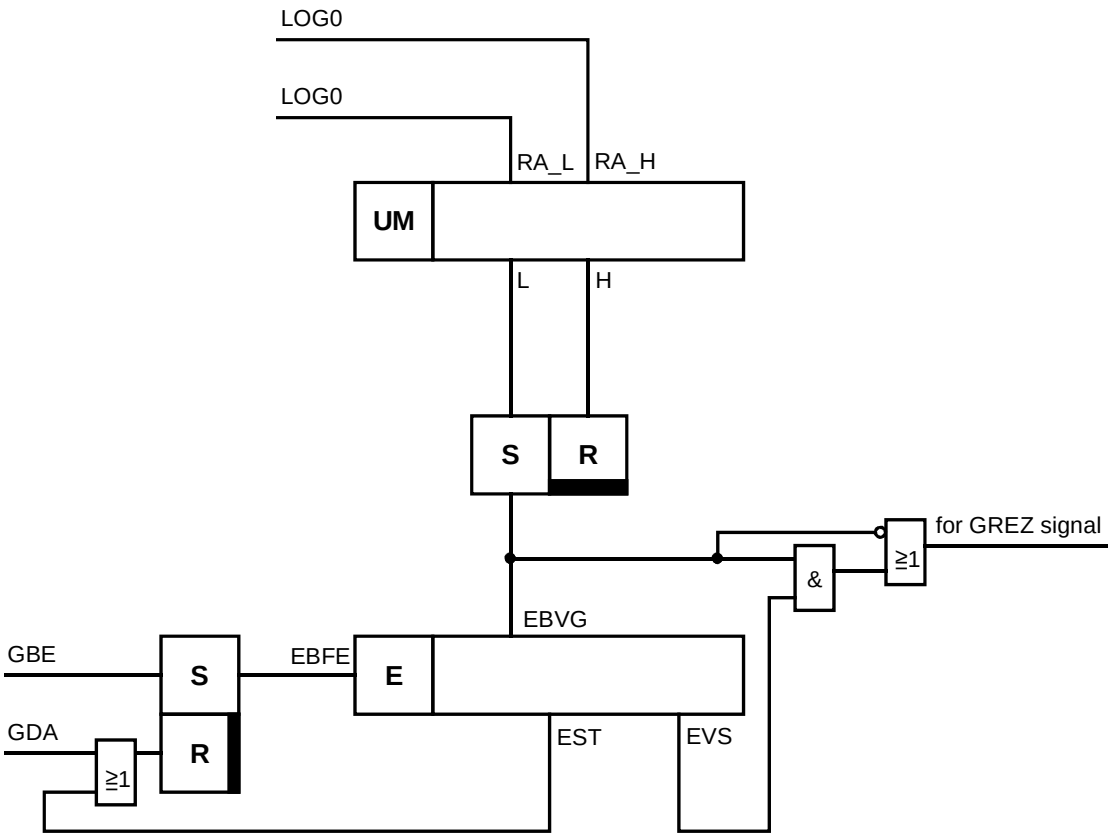
Silo full
Warning for 'H'
Switch-off for 'HH'



b) Example:

Pressure monitor

- Pump should switch on when the ‘L’ limit is fallen below and switch off again when the ‘H’ limit is exceeded.
- No annunciations should be created and the output of the group fault lamp should not be set if the limits 1 are violated.
- Annunciations should be created and the output of the group fault lamp should be set if the limits 2 are violated.



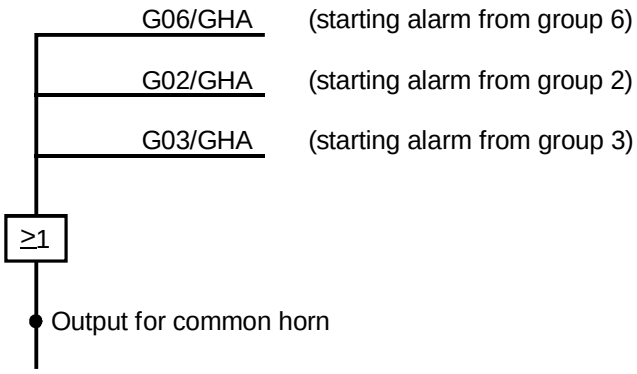
Group Start-up warning

If a machine group (e.g. cement transport) is started, then signal horns and lamps warn the personnel in this area that machines are starting and so increased caution should be exercised while working in this area.

Every group module provides a start-up warning during start.

The GHA or GLA output signal of the group must be switched to the horn or lamp outputs.

Example:

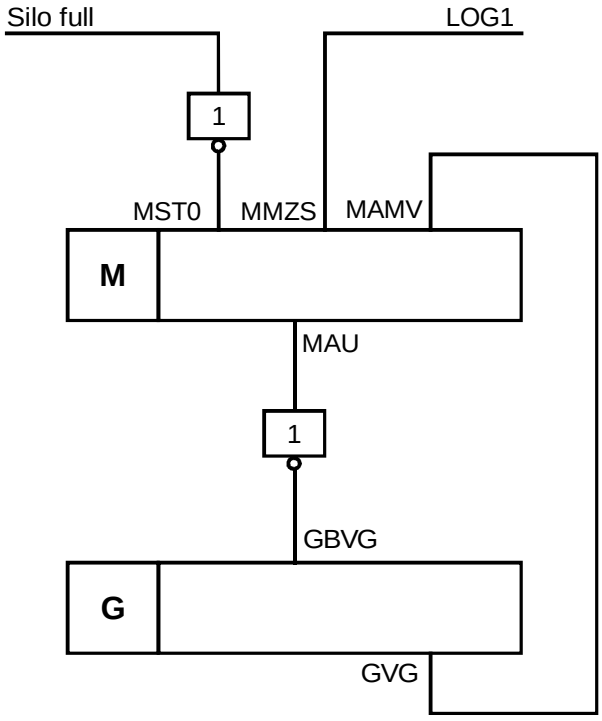


GBVG Group Operation Interlock

Process signals used to disconnect a group are placed on the operational interlock (e.g., silo full or requesting group unavailable).
The control panel operator must acknowledge such disconnections.

Note: To permit an alarm message to be created for the disconnection, this interlock must be maintained on an annunciation module.

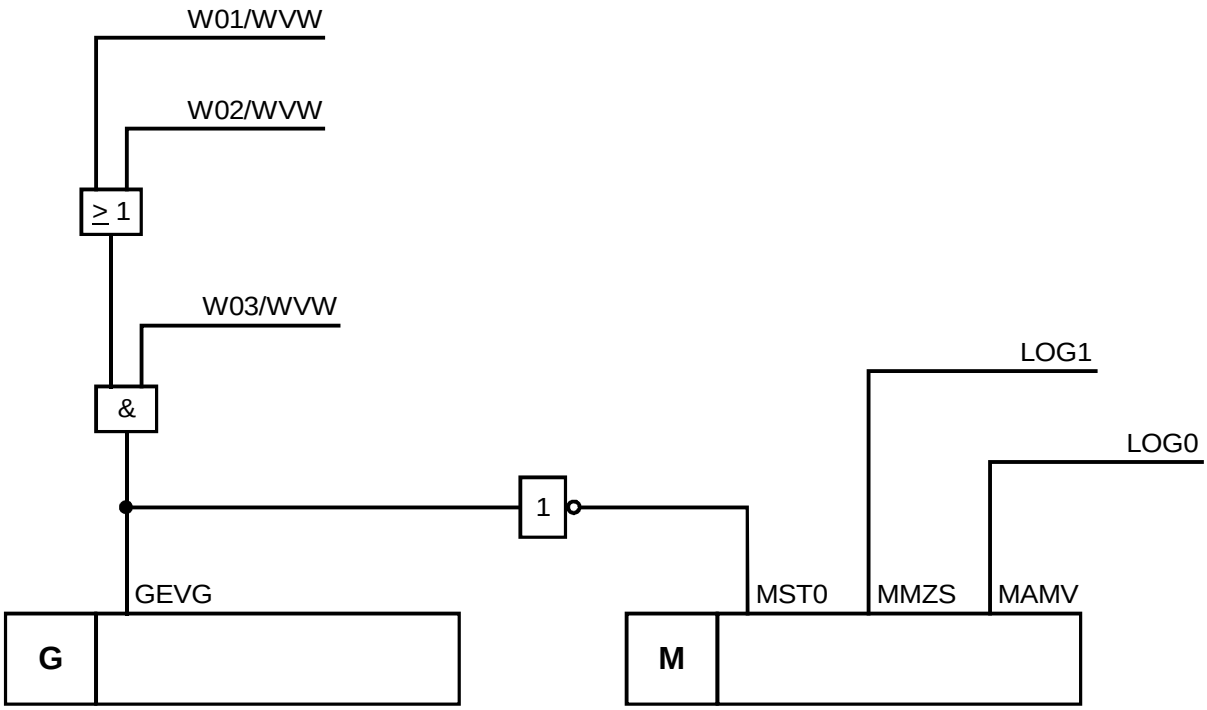
Example:



GEVG Group Start Interlock

The start interlock avoids a start-up warning being initiated for a non-operational group (e.g. route pre-selection must be made before the group start).

Example:

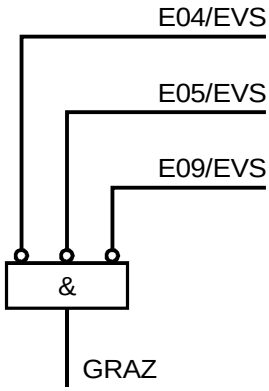


Note: To permit a start interlock to be reported for a “status call-up”, start interlocks should be maintained on an annunciation module.

GRAZ Group Feedback Off

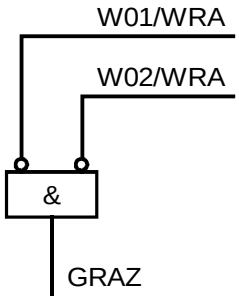
A signal that uniquely specifies that the group is stationary (negated feedback of the group drives).

a) Group without routes



b) Group with routes

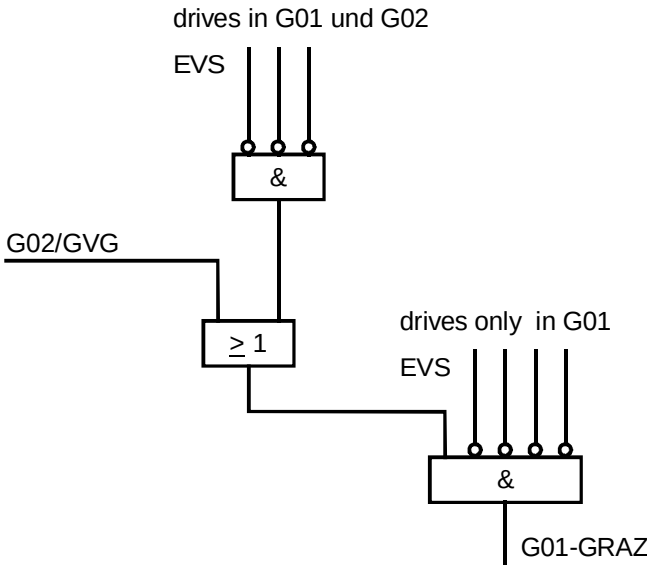
WRA=0 means: Route is not selected or route has been selected but all associated drives are off.



c) Drives in several groups or routes

Drives that belong to several groups/routes must be interlocked for the group feedback with the GVG or WVW signal of the other group/route.

Example:



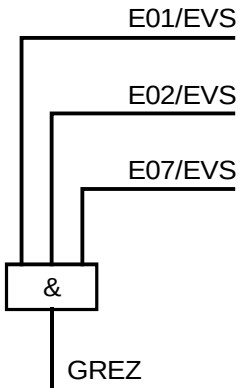
GREZ Group Feedback On

This signal indicates that the group is operating completely.

a) Group without route

- Feedback of the last drive in the connection sequence,
- if all drives are started in parallel: each the feedbacks of the last drives.

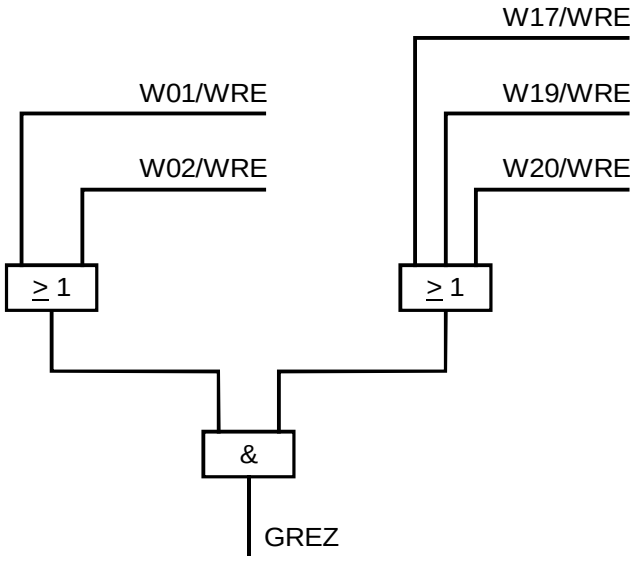
Example:



b) Group with routes

WRE=1 means: Route has been selected and is operating

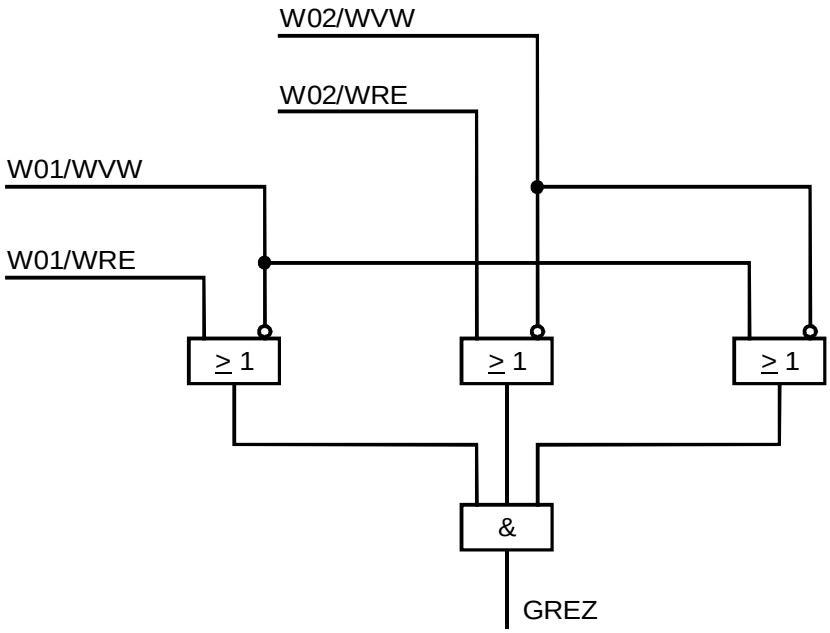
Example:



c) Group with routes (subsequently startable)

The simple allocation of the route feedback to the group feedback does not suffice if additional routes or drives are to be subsequently started for an operational group. It must be ensured that the GREZ is set to zero on selection of the “new” route, because only then can the group be restarted.

Example:



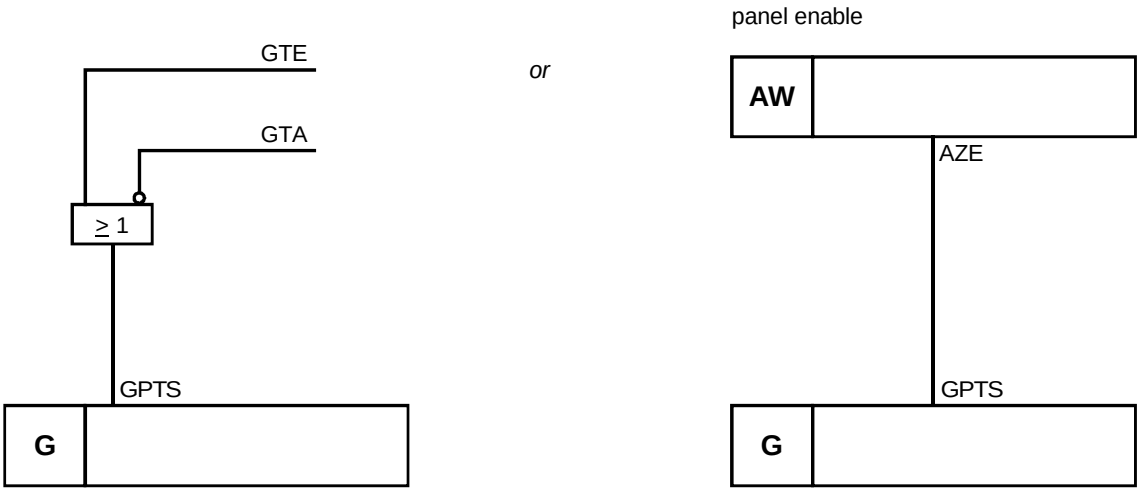
Group Start/Stop

It is possible to start or stop a group:

- from the screen (GPTS=0)
- with the classical panel keys (GPTS=1)

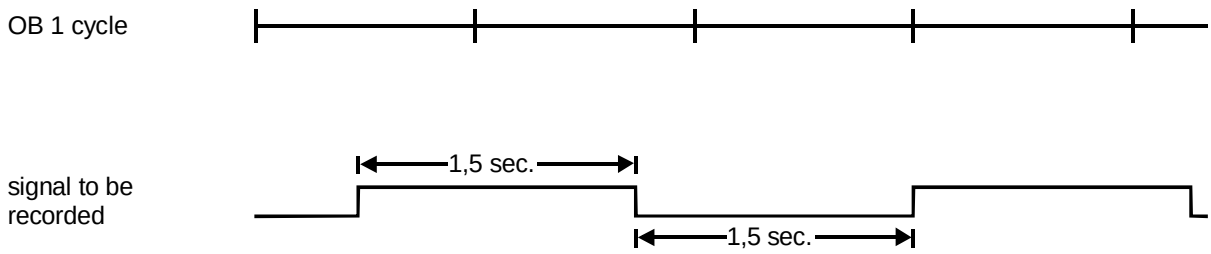
Note: The supplied state is: operation from the screen
Interconnect GPTS interface with LOG1 if panel keys are to be used for test purposes (simulator box)!

A parallel operation may be required in limited circumstances.
The following connections can be used for this purpose.



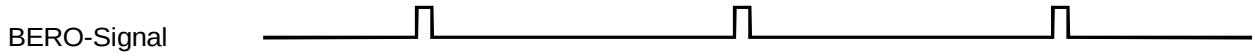
Impulse Processing

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 1.5 seconds.



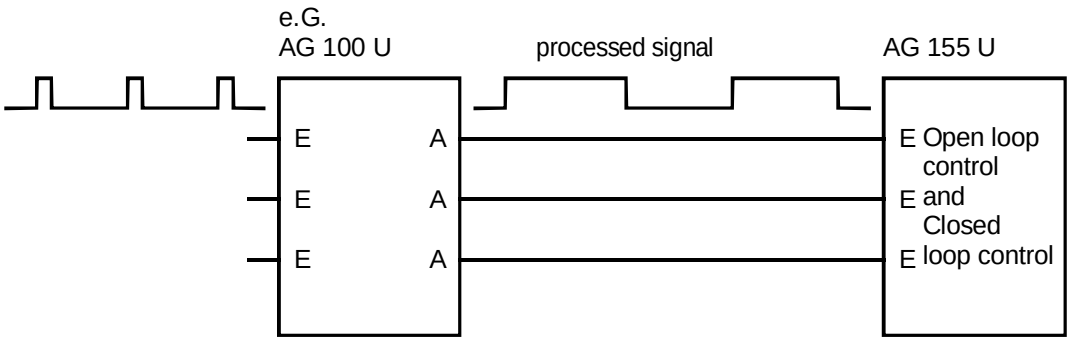
If the signals are recorded in the 100ms-OB cycle (time-controlled alarm processing), the pulse times and the pause times of the signal must both be longer than 150 ms.

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.

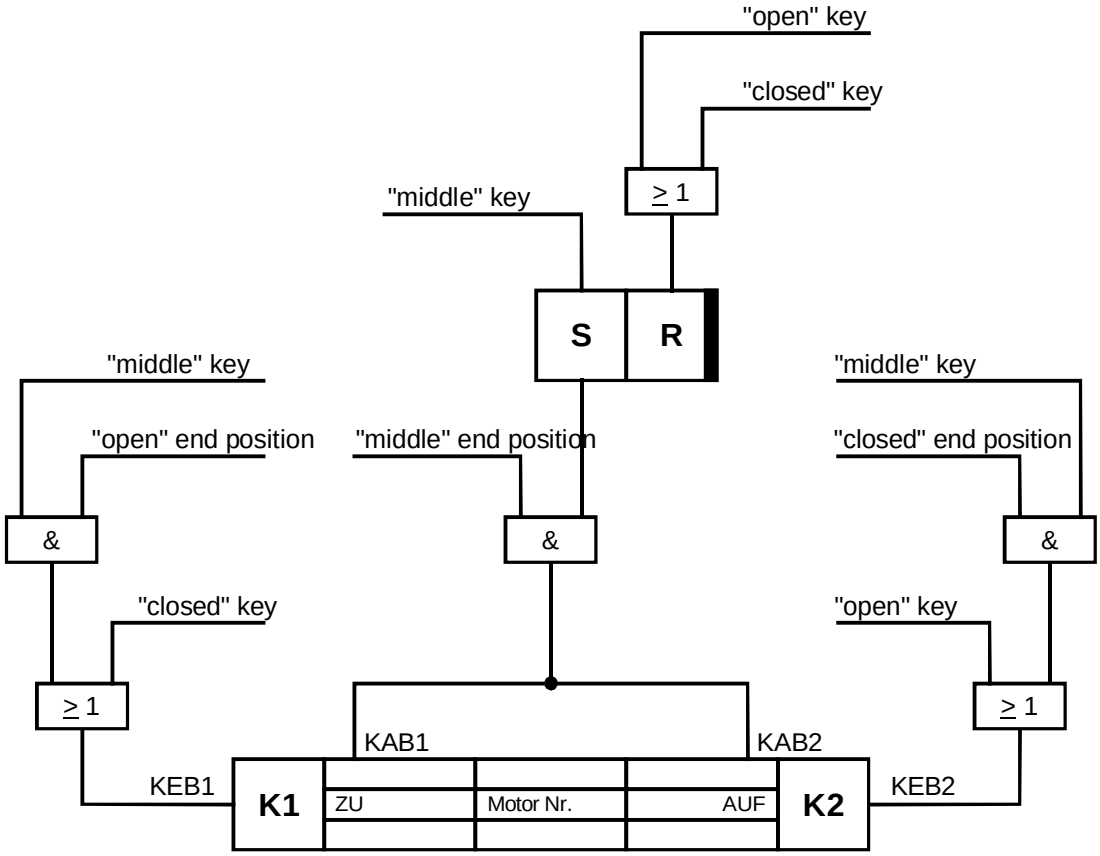


Such signals can occur in silo pilots or speed monitors.

Damper with Middle Position

The damper module normally monitors two end positions. End position 1 = closed, end position 2 = open. In exceptional circumstances, there are also dampers with three end positions (open-middle-closed) used in the cement plants.

The following example provides a possible solution for the connections for a damper with middle position.

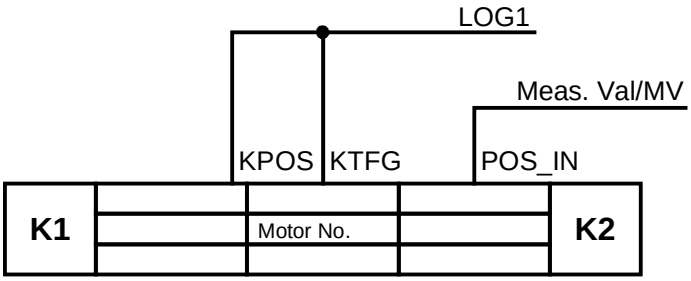


Damper Positioner

Dampers are sometimes used as portioning organ and so must be controlled to a specific position. An exact positioning is achieved by controlling the damper in positioning operation. A measured value for the damper position is required.

The release of the positioning operation is achieved by applying a 1-signal at the KPOS and KTFG interface. The output ‘MV’ of the associated measured value must be transferred to the interface POS_IN.

The actuator run time, the set point limits, the minimum impulse length and hysteresis must be specified as process parameters for the damper.

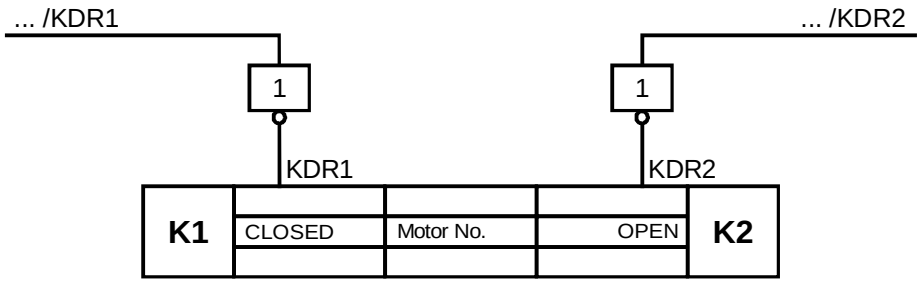


This connection permits the operator to enter the numerical positioning value from the keyboard.

If the positioning value is to be specified from a primary controller, then a 1-signal must be applied at the KWEE interface. The controller output of the associated controller must be connected to the interface KWEX of the damper. For the 1-signal at the KWEE interface, the damper module reads the external set point.

Damper with Torque switch

Should an operating damper be stopped on activation of the torque switch, these signals must be applied to the KDR1/KDR2 interface. The “torque failure” message will be output if the damper operates using torque.



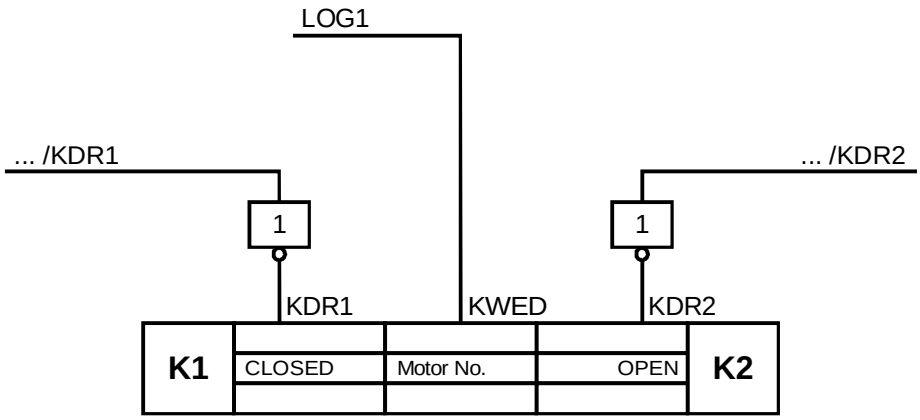
Wagging

The damper is returned to the initial position when the torque switch activates. There is then a new start attempt made in the chosen direction. This process is called wagging.

The number of start attempts (wagging actions) can be set as process parameter WAGG_NO.

If the selected end position even after the set end position is not achieved after the specified number of start attempts, the damper reports “mechanical failure”.

Damper connection when the activation of the torque is to result in wagging:



Ventilator Damper

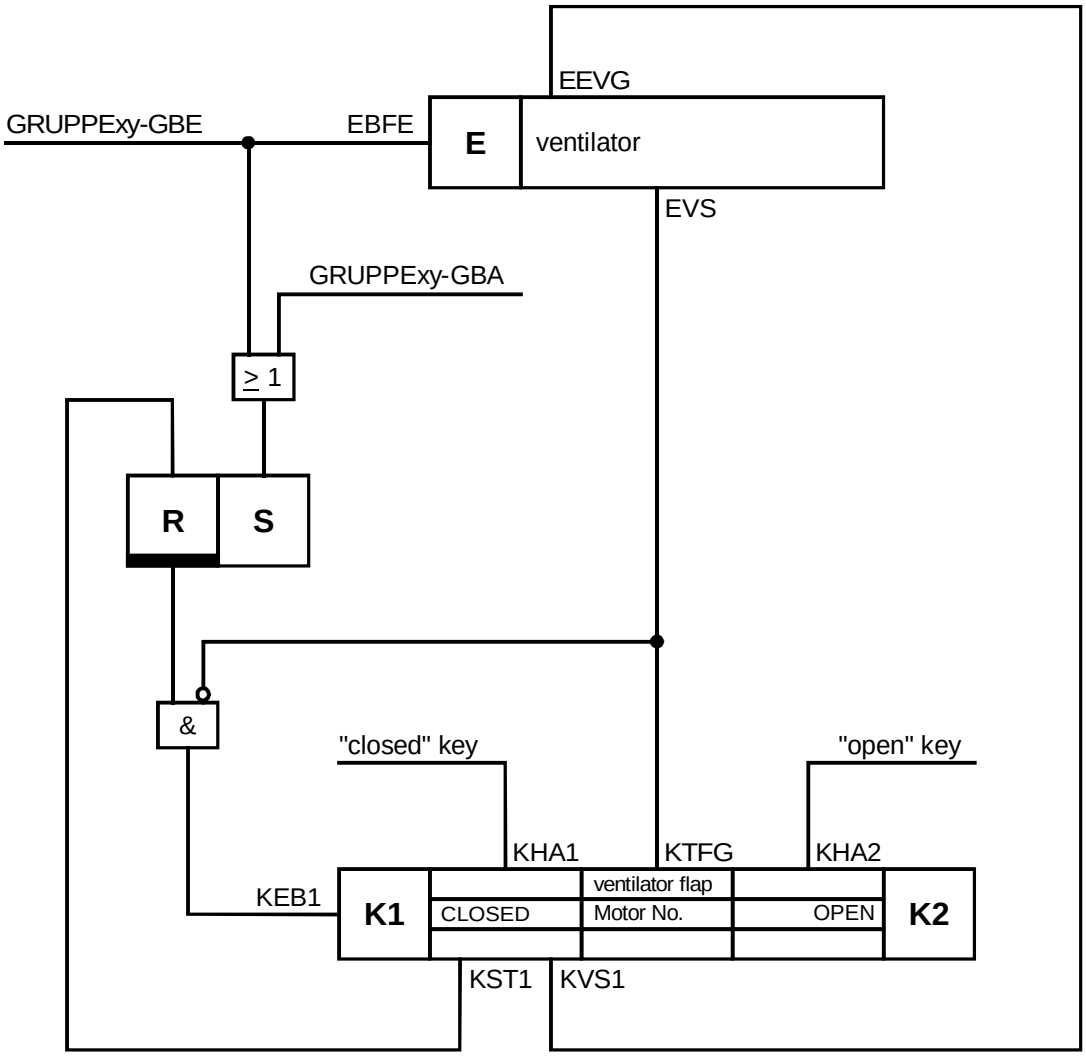
Because larger ventilators may start only when the damper is closed, it is desirable to close the associated damper when the group is stopped.

The following function diagram shows how this problem can be solved. Even after a possible failure of the damper, a renewed stop command or group start can close the damper.

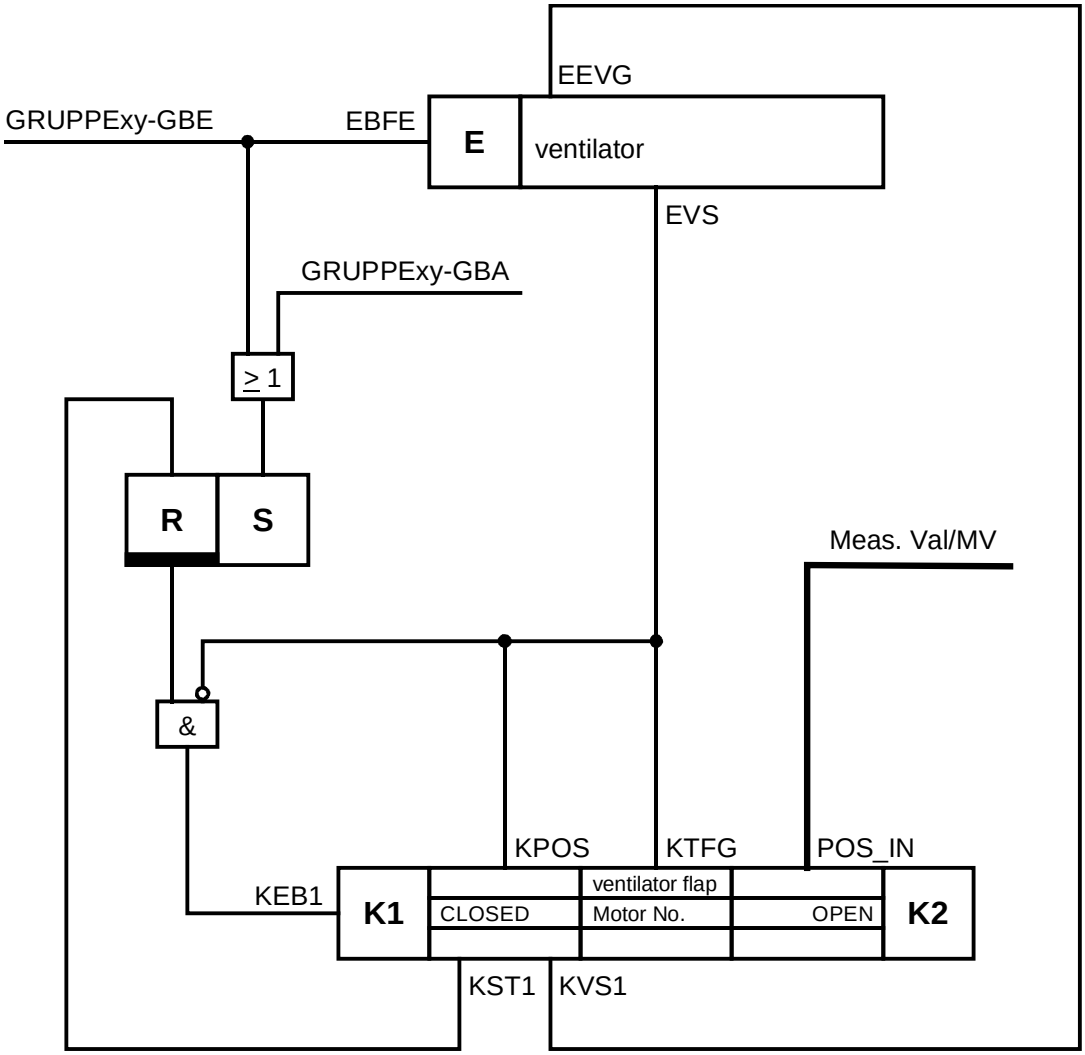
In Example a), the positioning of the damper is realized with the inching interface of the damper module. This solution is suitable for a conventional panel with keys and lamps.

In Example b), a positioner is used for the same task. This permits the numerical input of the position value and the exact positioning to the required value. The positioner is realized using the damper positioner function.

a) Interconnection of ventilator and damper during the positioning using panel keys:



b) Interconnection of ventilator and damper positioner



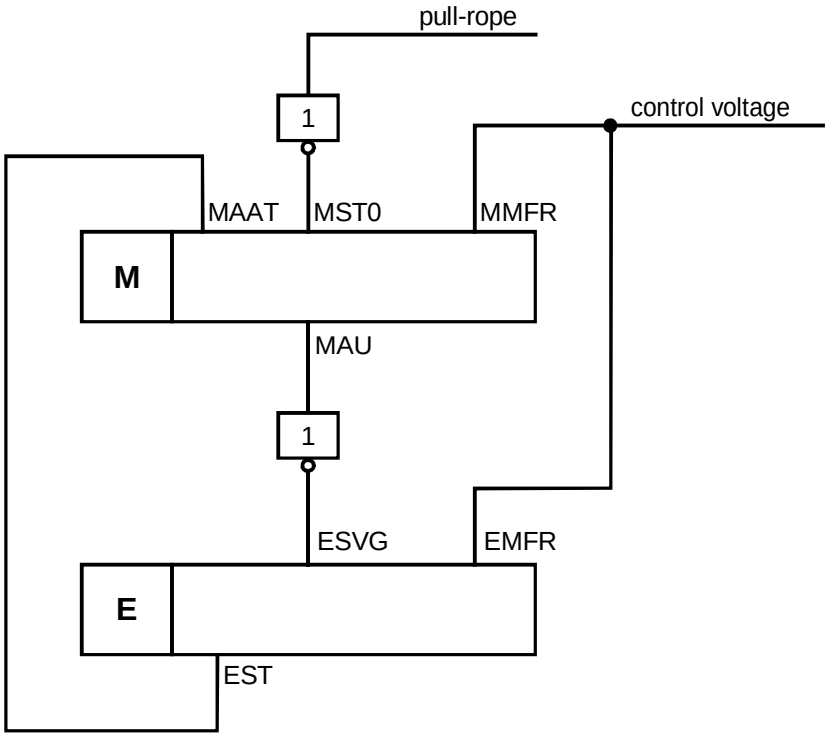
Annunciation Modules

There are two distinct uses for annunciation modules.

a) Drive failures

Annunciations of drive failures that the drive itself cannot report. These are all signals that are switched to the protective logic (belt drift, pull-ropes, bearing temperature, etc.)

- The message should appear together with the disconnection of the drive and the annunciation module output (connect alarm activation and use the annunciation module output for the disconnection).
- Interconnect the annunciation release from drive and annunciation module with the same control voltage.



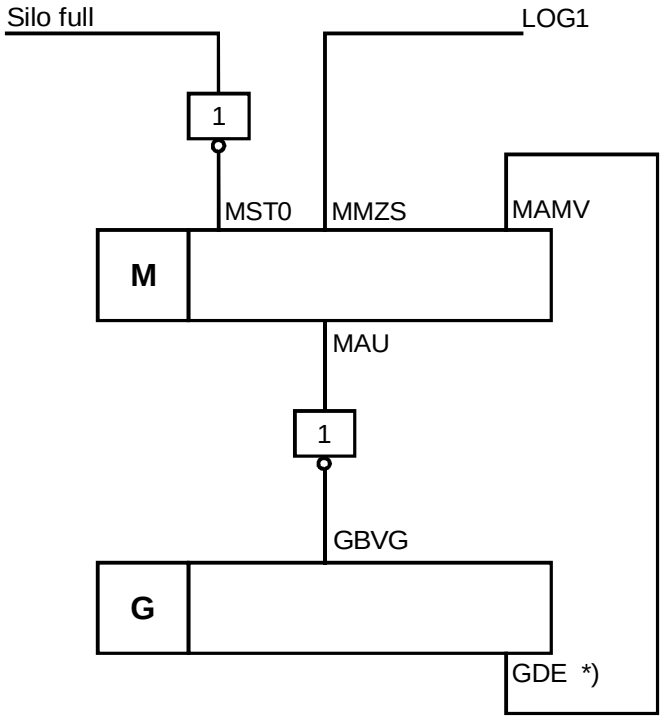
b) Process signal annunciations (interlocks)

Report process signals, such as silos level and other interlocks.

Please observe the following points:

- Interlocks should not be transmitted to the group as a “fault”, i.e. they should not be indicated at the group summarizing fault indication and the group start GBE should not be interrupted in case of an alarm, caused by this signal.
  This is achieved by connecting a LOG1 signal to interface MMZS.
- To ensure annunciation and interlocks occur concurrently, use the annunciation module **output signal** for the interlock.
- If the alarm interlock is interconnected with, for example, GVG, the annunciation module provides an alarm message only for a running group.

The status call-up can at any time be used to request pending messages (also for a stationary group).



*) Refer to “Alarm Interlock”

Annunciations, Alarm Interlocks

It is guaranteed for the drives modules that they produce a fault message only when they are “active”, that is

a) if the drive was running and detects a fault,

b) if the drive is faulty and is switched on, and

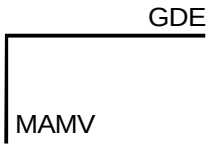
c) if the drive is in stand-by mode and has a failure.

The appropriate connection provided for the annunciation modules (C_ANNUNC, C_MEASUR) must itself ensure that an alarm is produced only when the associated plant section is active and the message is plausible. For drive failure messages, this is guaranteed by the interconnection of the alarm activation (MAAT).

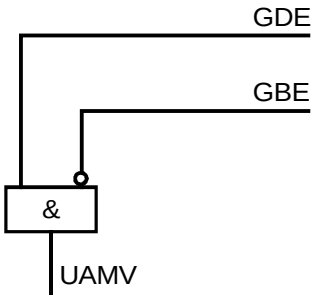
For the annunciation of special signals and limit value annunciations, the MAMV/UAMV alarm interlock must be so interconnected that this annunciation is plausible. Depending on the technological function, the alarm interlock must be interlocked accordingly. Criteria for the plausibility of a message can be: group feedbacks, drive running signal, group states, etc.

Examples:

a) A fault should produce an alarm only when the group has been started but not stopped.



b) A limit value violation should be reported only when the group has been run up completely.



End Limit Switches (Limit switch 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.

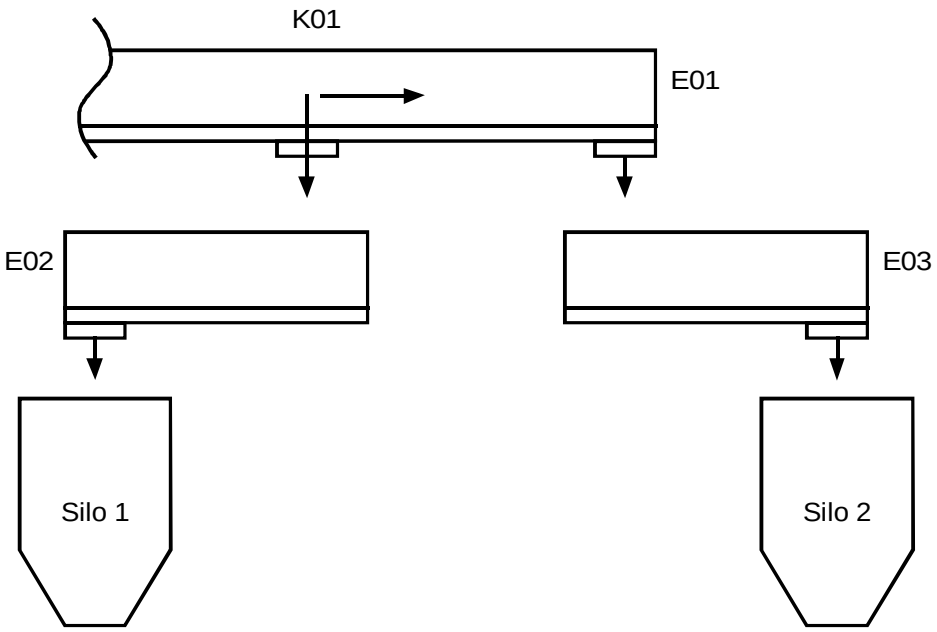
Star-Delta Starting; Slipring Rotor with Starter Motor

For drives with star-delta starting or slipring rotor with starter motor, the “1” state can be achieved at much later time after applying the ERM signal (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.

Route change without interrupt



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 slide 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:

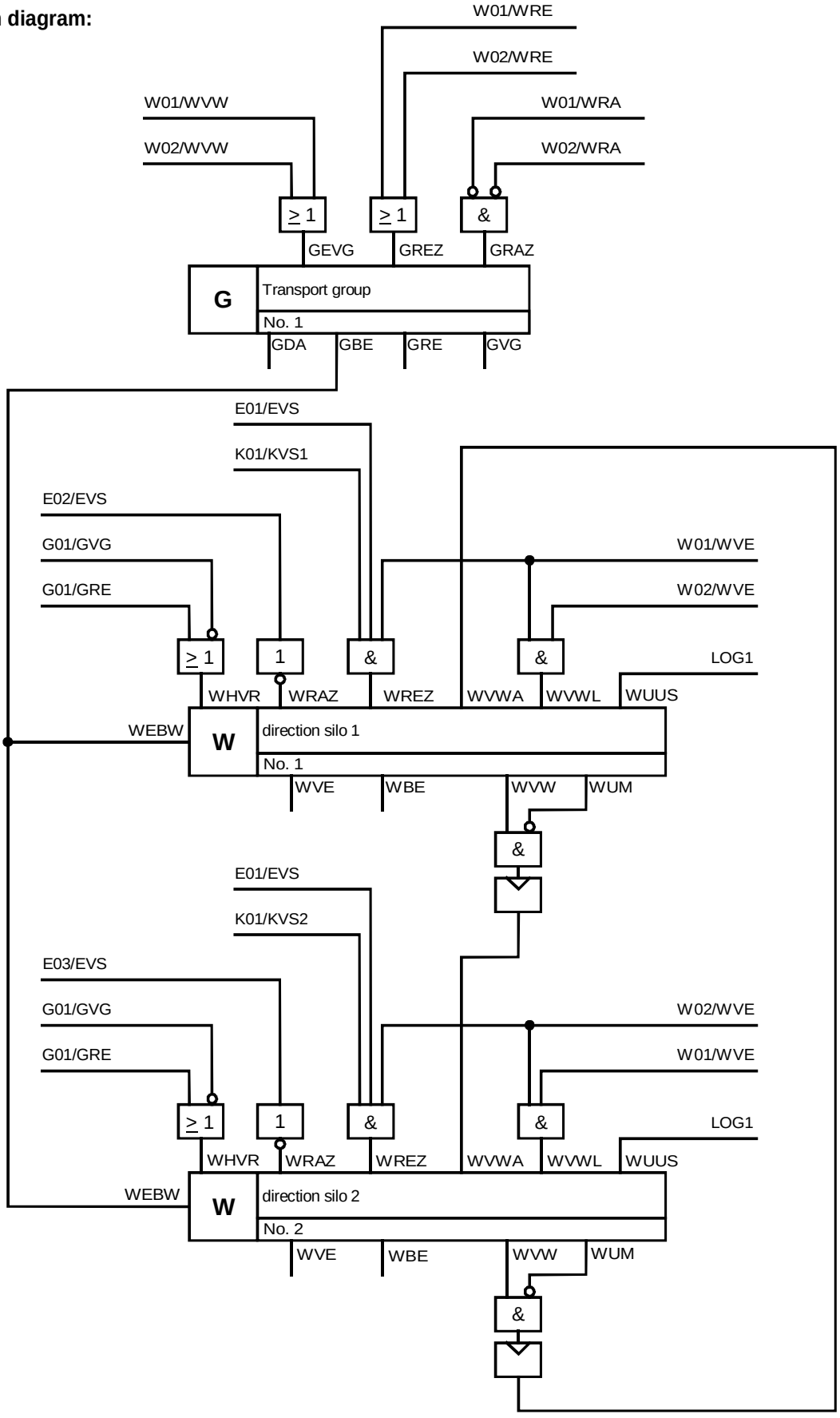
The air slides E02 and E03 and the damper position K1 (direction silo 1) belong to route 1.
The air slides E01 and E03 and the damper position K2 (direction silo 2) belong to route 2.

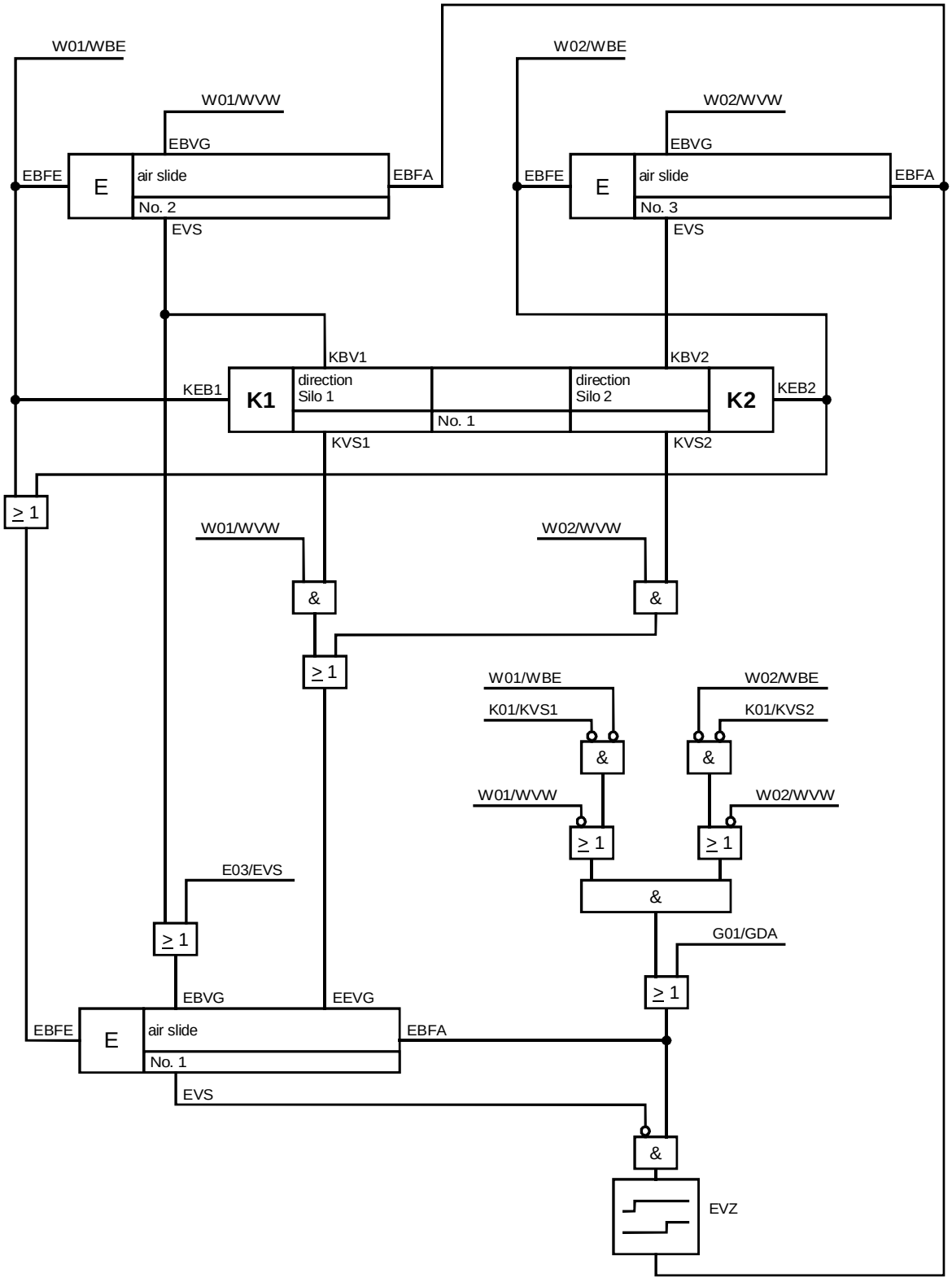
Operation:

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 important.

Function diagram:





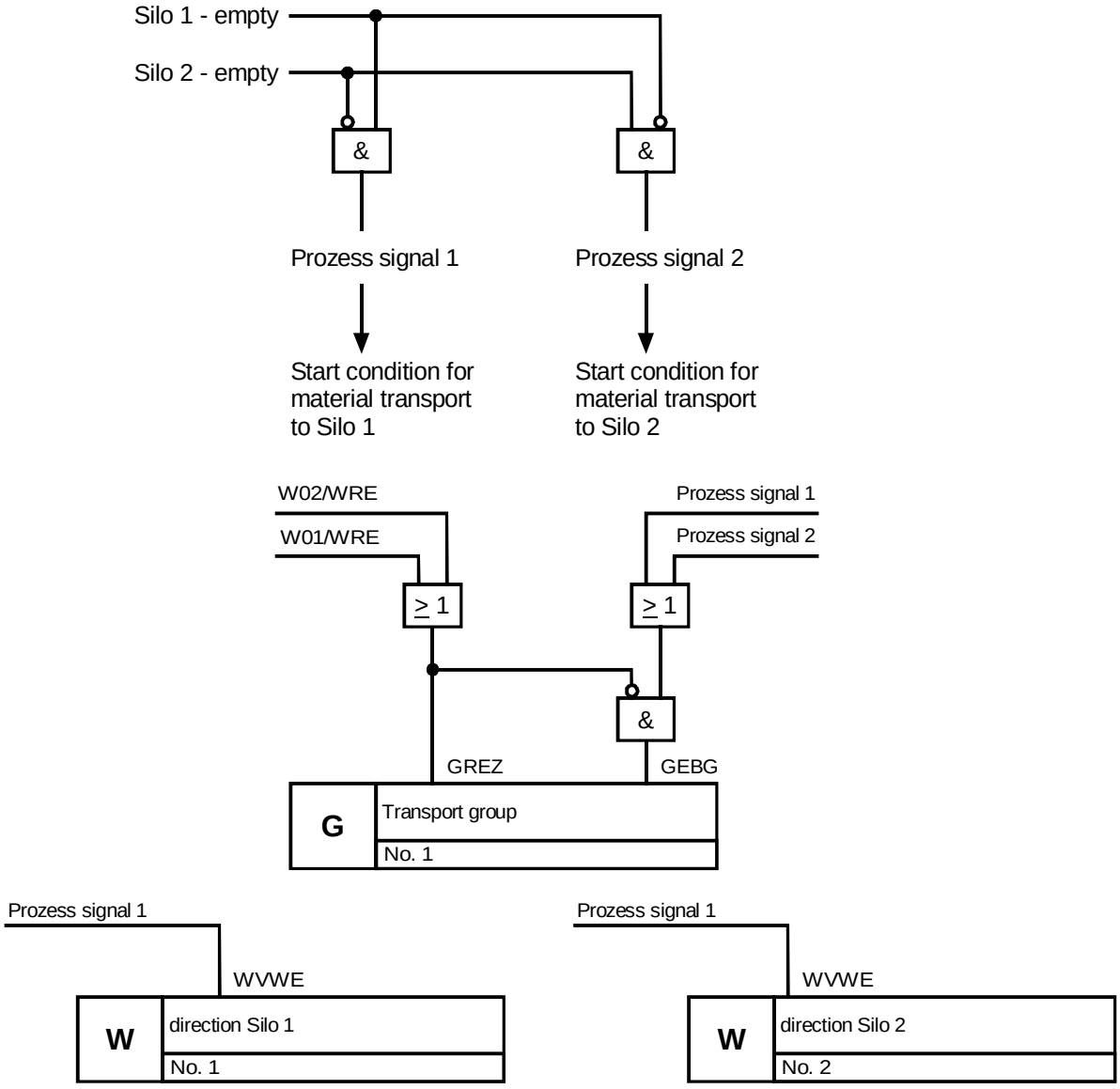
The connection example becomes clearer by considering the individual functions that must satisfy a change of routes without interrupt.

- 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

Here 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

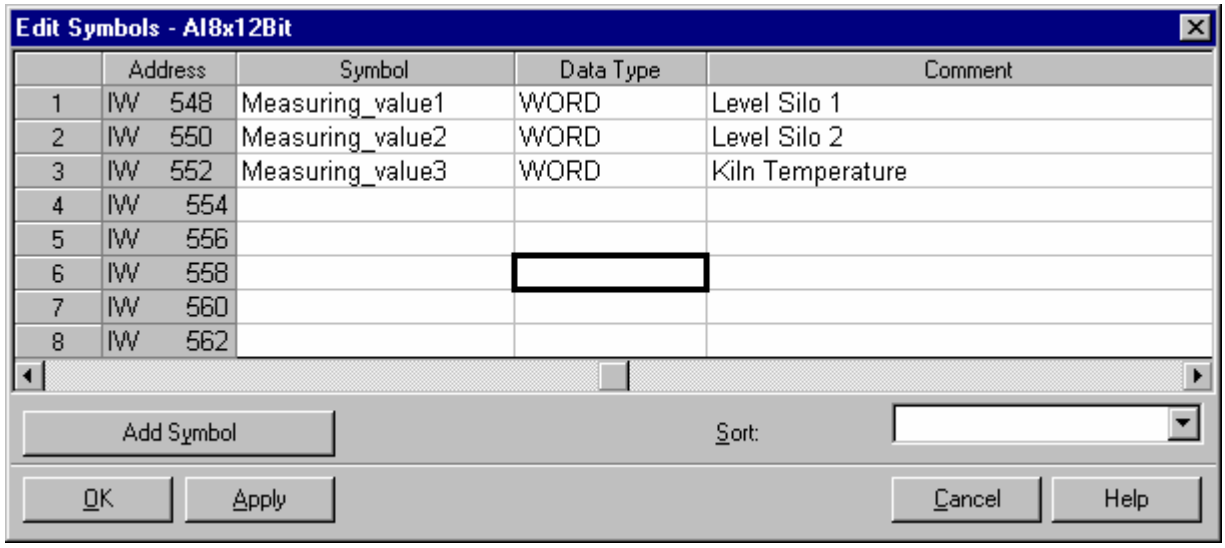
PLC Engineering

Content

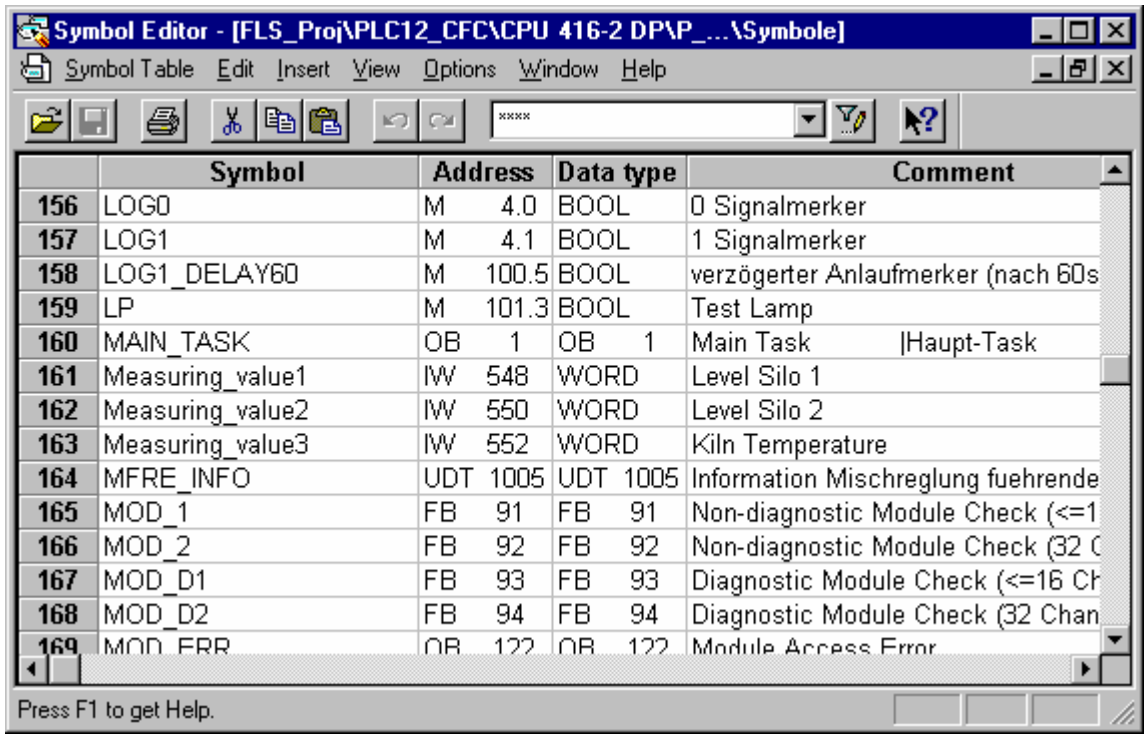
Ausgabe : 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\06_PLC_Engineering.DOC

Signal list (Symbols)

The hardware signals can be entered directly in the hardware configuration.



They will be added to the Symbols list and can furtheron be used in CFC.



There is also the possibility to import the hardware signal list from an excel file.

Creating the PLC Program with CFC

CEMAT Functions

Drive functions

C_DRV_1D	Unidirectional drive
C_DRV_2D	Bi-directional
C_DAMPER	Damper
C_VALVE	Valve
C_SIMO_A	Adapter for SIMOCODE

Annunciation functions

C_ANNUNC	Annunciation module
C_ANNUN8	Annunciation module with 8 Alarms
C_MEASUR	Measuring value

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
CTRL_S	Step Controller
RATIO_P	Ratio Controller

Caution: CEMAT V6 has no special block for analogue output function. You have to use the PCS7 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 function

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

The detailed description of the CEMAT Functions you find in the object description an in the only help.

All other blocks from the PCS7 standard library can be used as well.

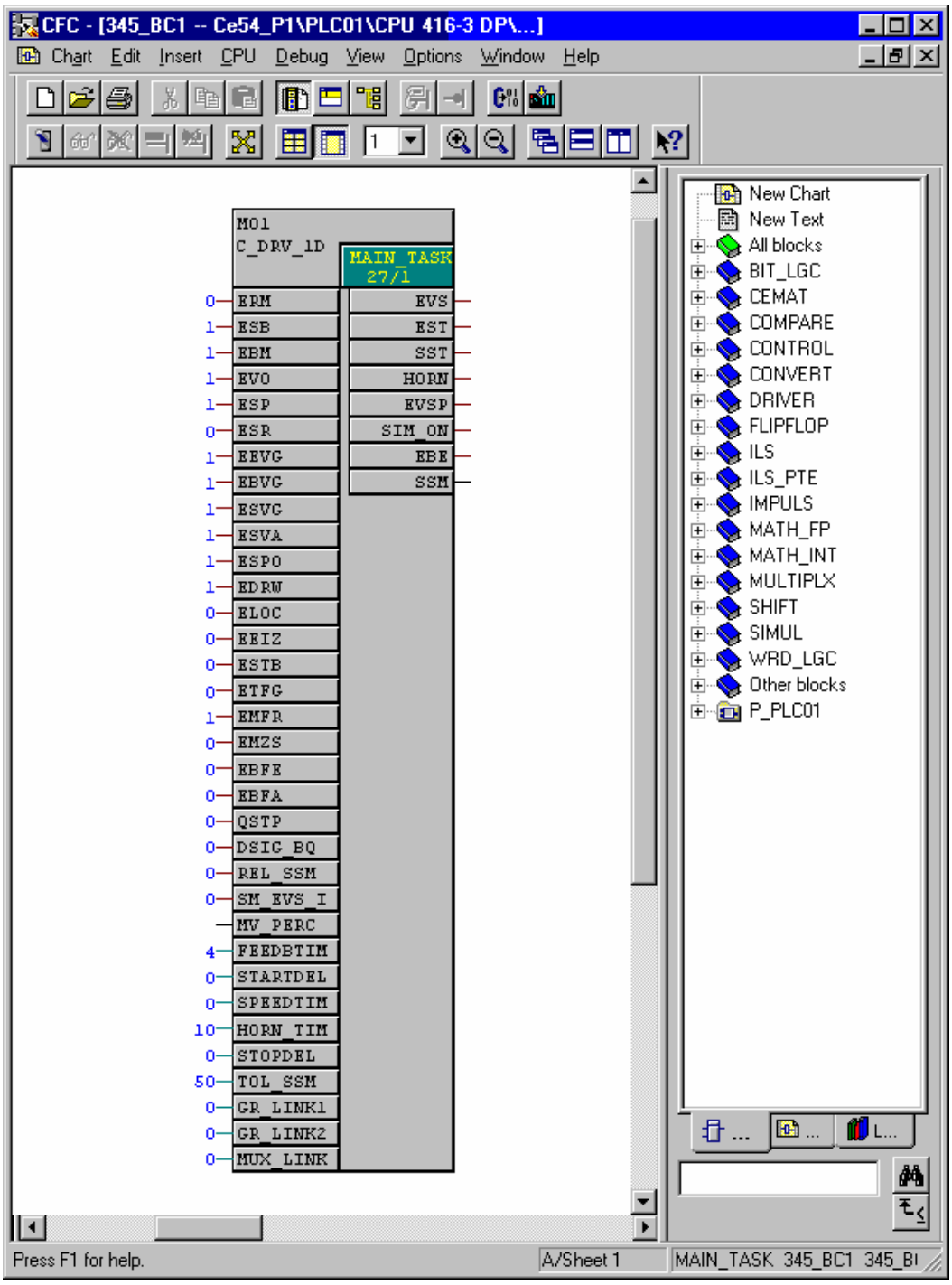
CFC Programming

The program logic is written in the Continuous Function Charts (CFC). To insert new Charts use Plant view or Process object view. Later the charts can be edited from all views of the SIMATIC Manager.

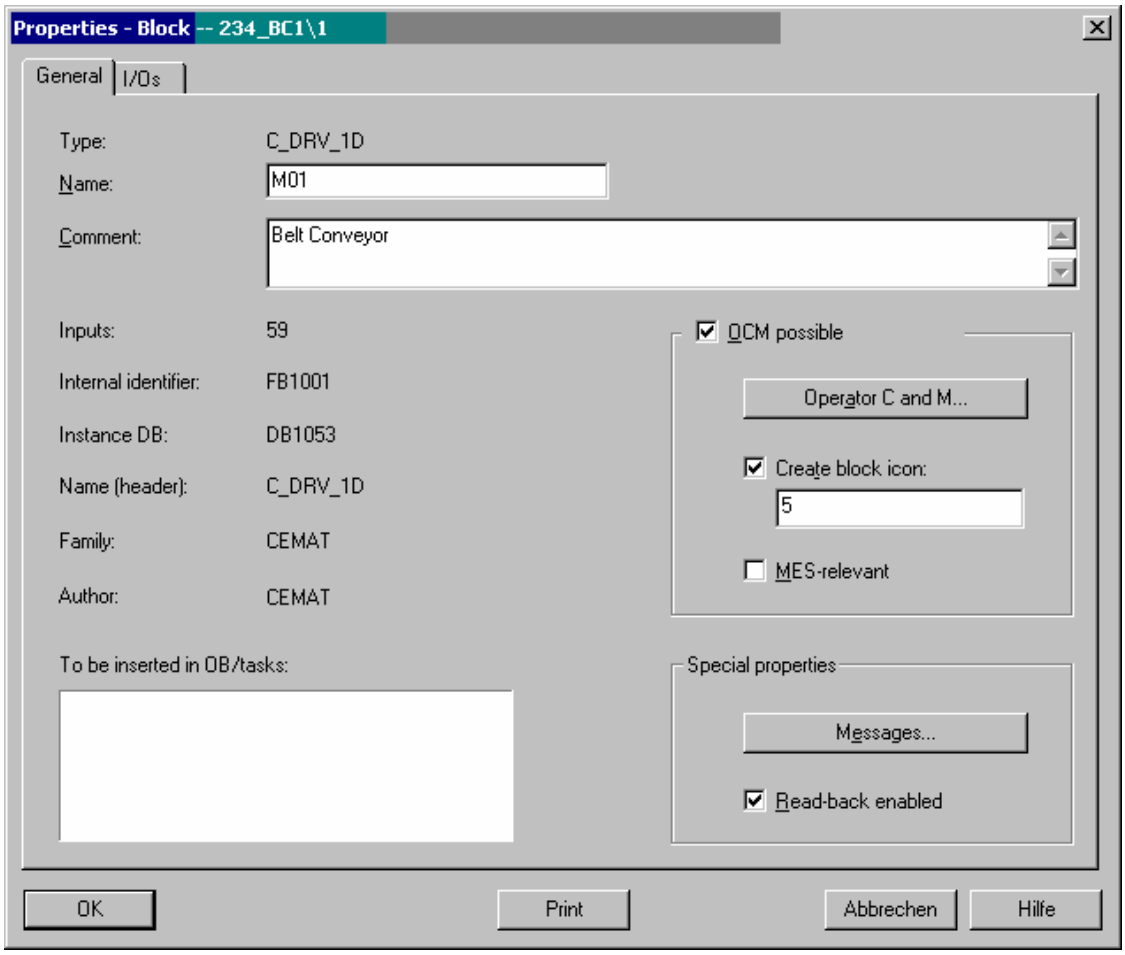
After adding a new chart you have to rename it according to your tagging system. The chart name is the first part of the tagname. In this example the chart name is 234_BC1.

Open the chart and select a block from the catalog. Drag and drop it to the chart.

The CEMAT Standard functions are located in Folder CEMAT. The controller functions you will find in Folder CONTROL.



Double-click on the block in order to open the properties dialog.



Carry out the following settings in the property window:

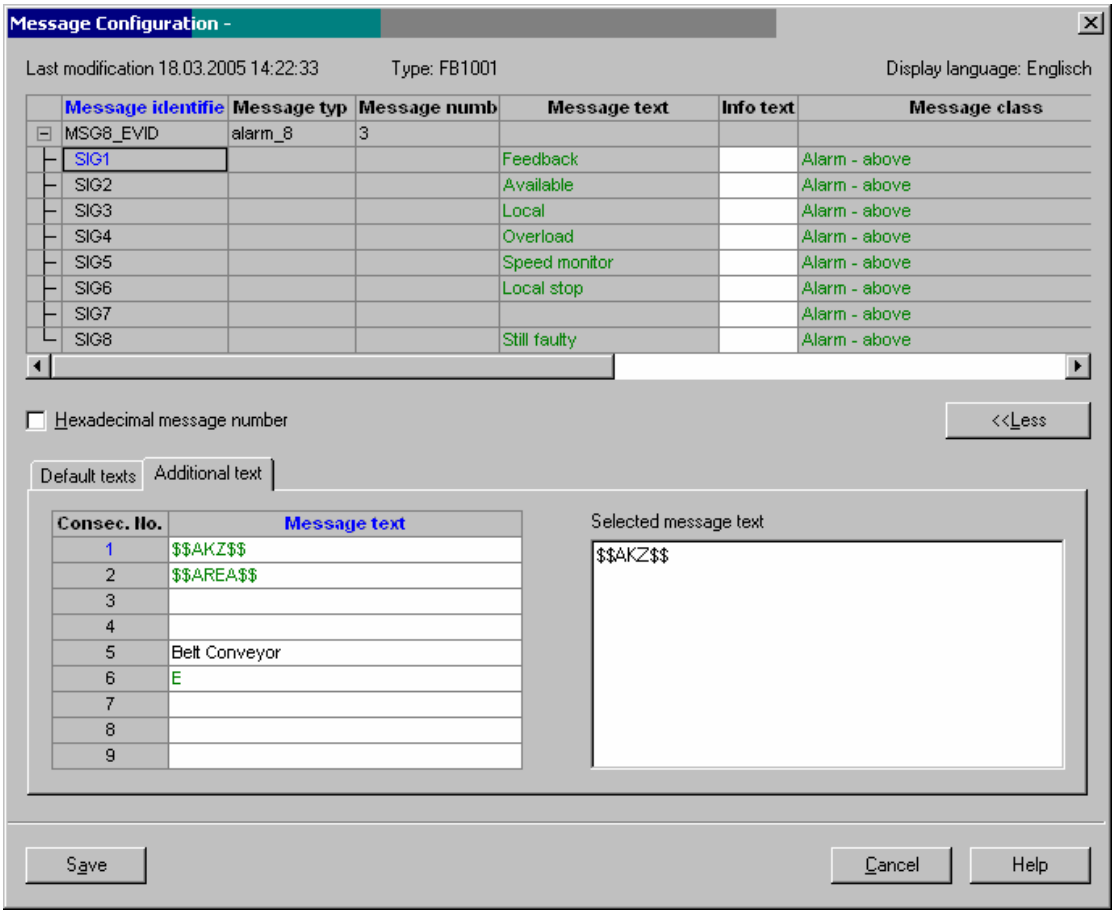
Name:
The name (instance name, e.g. M01) is the second part of the tagname of the motor. If the Hierarchy folder name is not part of the tagname (project setting), the tagname of the motor consists of the chart name and the instance name. In the above mentioned example this would be 345_BC1/M01.
Make 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 e. g. in the faceplate of the motor.
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.

Messages...
Each drive block uses an Alarm8 function, whose signals are already predefined in the CEMAT standard. To show the block comment also in the alarm line, the block comment must be available for each signal under Special properties → Messages… in tab “Additional Texts”, Consec. No. 5.



You can copy the block comment and paste into each message in row 5.
However an easier way to enter the block comments, is using the process object view. There the tag comment must be in tab Messages under “Free Text 1”.

After that you can link and parameterize the module in the CFC according to it’s function.

Make sure that the block is called at the right position in the runtime sequence. (Refer to object description).

Definition of the run sequence

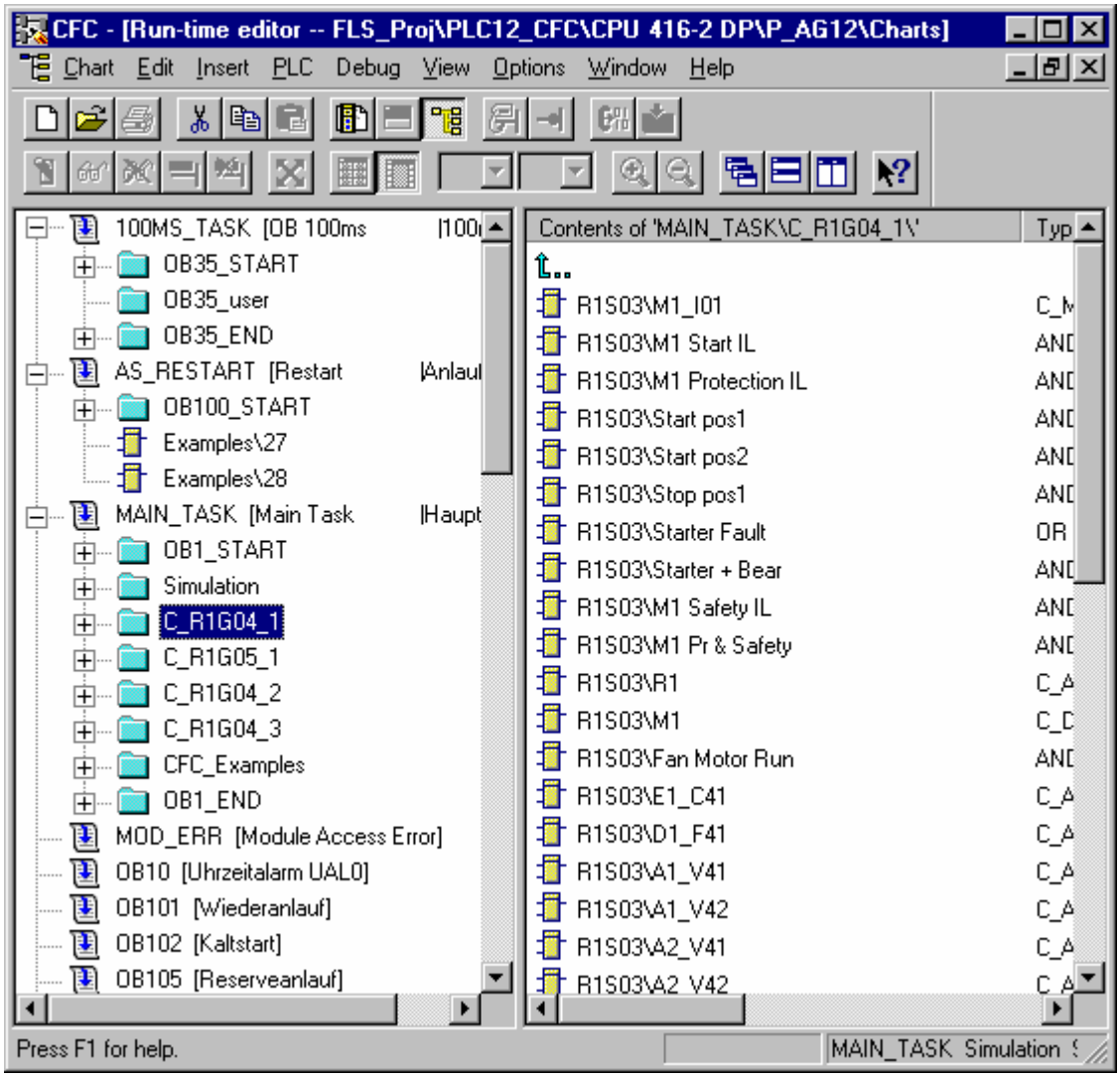
Most of the CEMAT objects have to be called in OB1 (MAIN_TASK). The only exceptions are C_SPEEDM, C_SPCNT and C_COUNT. These types have to be called in OB35 (100MS_TASK).

Remember the sequence:

1. evtl. MUX
2. Annunciations and Drives
3. Corresponding Routes
4. 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.



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.

Positioner function

The following interfaces of the 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		
KWEE	External setpoint ON	BOOL	0	I	U		
KSNF	Setpoint tracking	BOOL	1	I	U		
POS_LZ	Live Zero for position	BOOL	0	I	U		
KWEX	External setpoint	REAL	0.000000e +000	I	U		
POS_IN	Position value 0-100	STRUCT		I	U		
W_OS	Sollwert of OS (KWCO)	REAL	0.000000e +000	I	U	+	
KWUG	Setpoint lower limit	REAL	0.000000e +000	I	U	+	
KWOG	Setpoint upper limit	REAL	1.000000e +002	I	U	+	
SCB	Scale beginning	REAL	0.000000e +000	I	U	+	
SCE	Scale end	REAL	1.000000e +002	i	U	+	
UNIT	Unit	STRING [8]	‚%‘	I	U	+	
TMIN	Min. pulse length	REAL	5.000000e -001	I	U	+	
TM	Actuator run-time	REAL	3.000000e +001	I	U	+	
AN	Switch on of the dead zone	REAL	1.000000e +000	I	U	+	
AB	Switch off of the dead zone	REAL	1.000000e +000	I	U	+	
X_POS_OS	Damper position display	REAL	0.000000e +000	O	U	+	
KPO	Positioner ON	BOOL	0	O	U		

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 PLC. If Fault Acknowledgement, Lamp Test or Release Functions should work only for specific CEMAT objects, the interfaces at C_PUSHB can not 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**

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**

With a positive Edge on QT all dynamic faults in the PLC are acknowledged.

QT_H Acknowledge Horn **Basic State: 0-Signal**

With a positive Edge on QT_H the horn gets switched off.

LP Lamp Test **Basic State: 0-Signal**

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**

After this time the horn is switched off.

Output Parameters

HORN Horn

Output for an acoustical alarm in the control room.

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

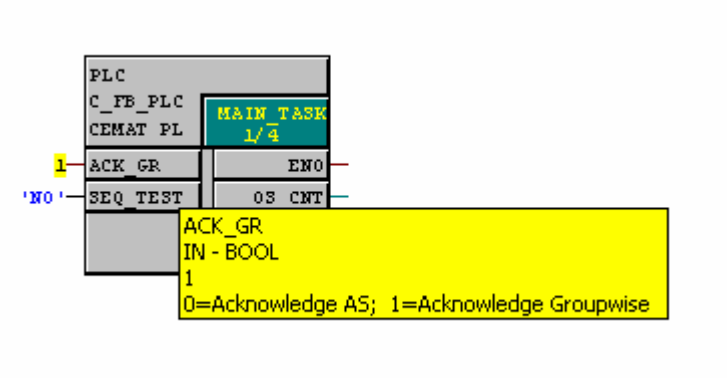
Acknowledgement mode

In CEMAT V6.1 two acknowledgement modes exist: The fault acknowledgment can be carried out either for the complete PLC or group-wise (refer to Reference Manual, System).

By default the fault acknowledgement is carried out per PLC. If the fault acknowledgement shall be carried out per group, this requires a modification of the settings in the system chart and some additional programming for each object.

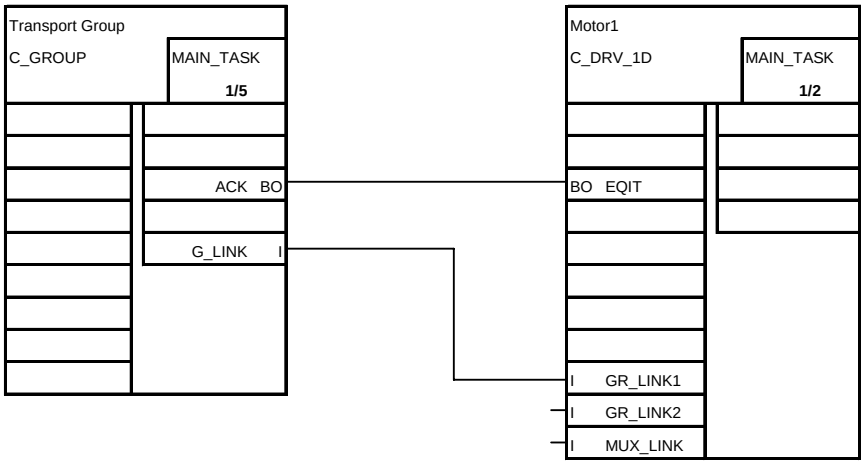
Group-wise acknowledgement

In the System chart at block C_FB_PLC you have to change the setting for parameter AKC_GR at block to 1-Signal.



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.

Example for a Motor:



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 PLC.

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 PLC.

To leave the sequence test mode, enter string ‘NO’ on input parameter SEQ_TEST and restart the PLC.

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.

Using driver blocks with CEMAT

From CEMAT V6 the driver concept is supported by the CEMAT blocks. It is still not absolutely necessary to use the module drivers. The inputs and outputs of the CEMAT blocks can still directly be connected to the Periphery (Process Image).

The following description is only valid for projects where driver blocks are used.

Driver functions and drive blocks

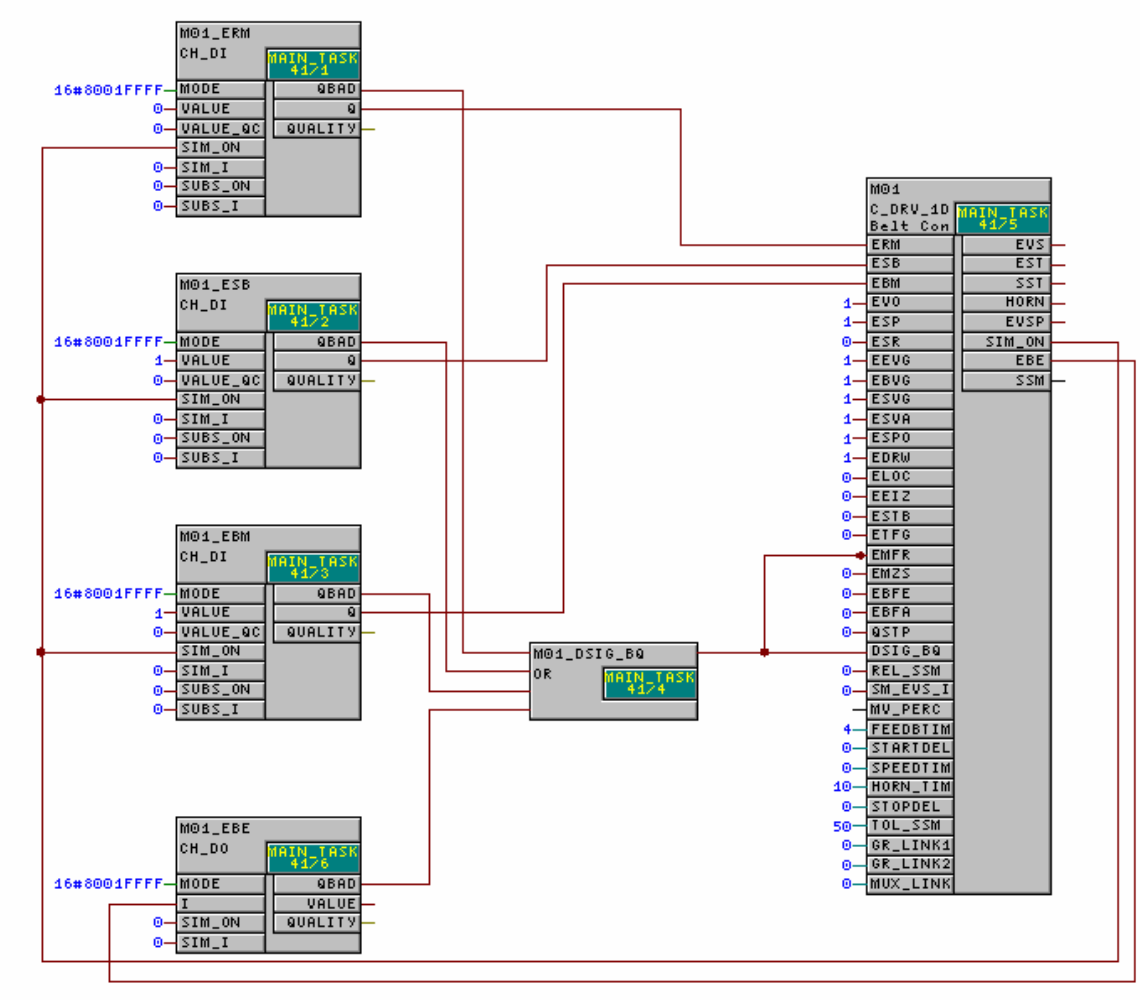
The parameterization of the driver bocks can exclusively be carried out in CFC, at the driver block itself. Only in the CFC the simulation value can be set and the behavior in case of module fault (substitution value or last valid value) can be parameterized.

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.

Furthermore it is possible to show the status “Module fault” in the block icon of the motor and in the faceplate. To enable this, the output QBAD of the driver block has to be connected with an OR function to interface DSIG_BQ of the motor.

Caution: The Display of the Module status in the block icon of the motor requires special attributes of the block icons. See OS Engineering.

Example:



Note: In example shown above, the Bad Quality Signal was connected to Annunciation Release EMFR to prevent alarms in case of a module failure. Please take in mind that for Annunciation Release additional signals may be included.

Driver functions and Annuciation 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 annunciation block to the input SIM_ON of the Driver block, the simulation can be switched on/off online (from the Diagnostic picture of the Annunciation block). Changing to sequence test mode all annunciation blocks are automatically switched Simulation.

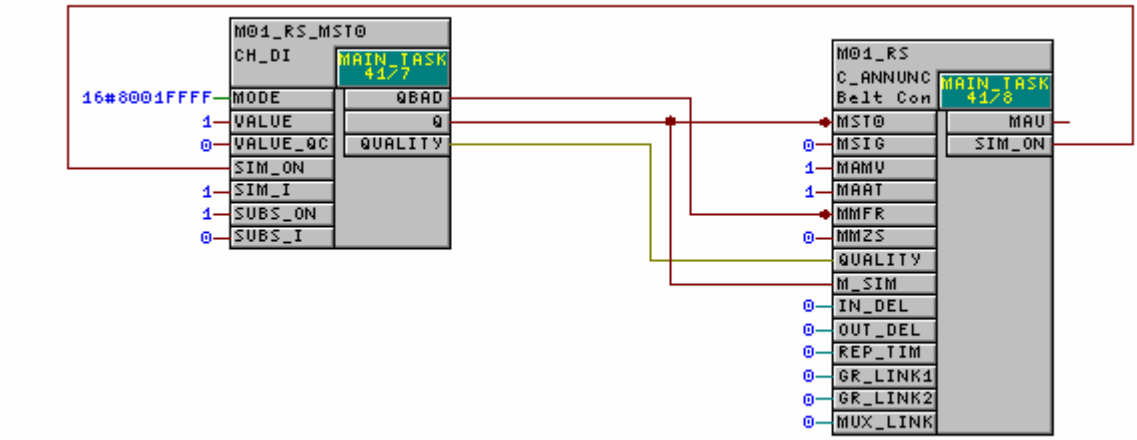
To display the simulation value in the Diagnostic picture, the output Q of the driver block has to be connected to input M_SIM of the Annunciation block.

Note: The display of the Simulation value in the diagnostic picture will be actualized only if simulation mode is activated.

Connecting output QUALITY of the driver block with input QUALITY of the annunciation block, the status “Bad Quality” and “Simulation” are shown in the block icon of the annunciation module and in the faceplate.

Caution: The Display of the Module status in the block icon of the annunciation block requires special attributes of the block icons. See OS Engineering.

Example:



Note: In example shown above, the Bad Quality Signal was connected to Annunciation Release MMFR to prevent alarms in case of a module failure. Please take in mind that for Annunciation Release additional signals may be included.

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.

The output SIM_ON is only an indication which shows that the measure is in simulation mode. Connection to the driver block is not required.

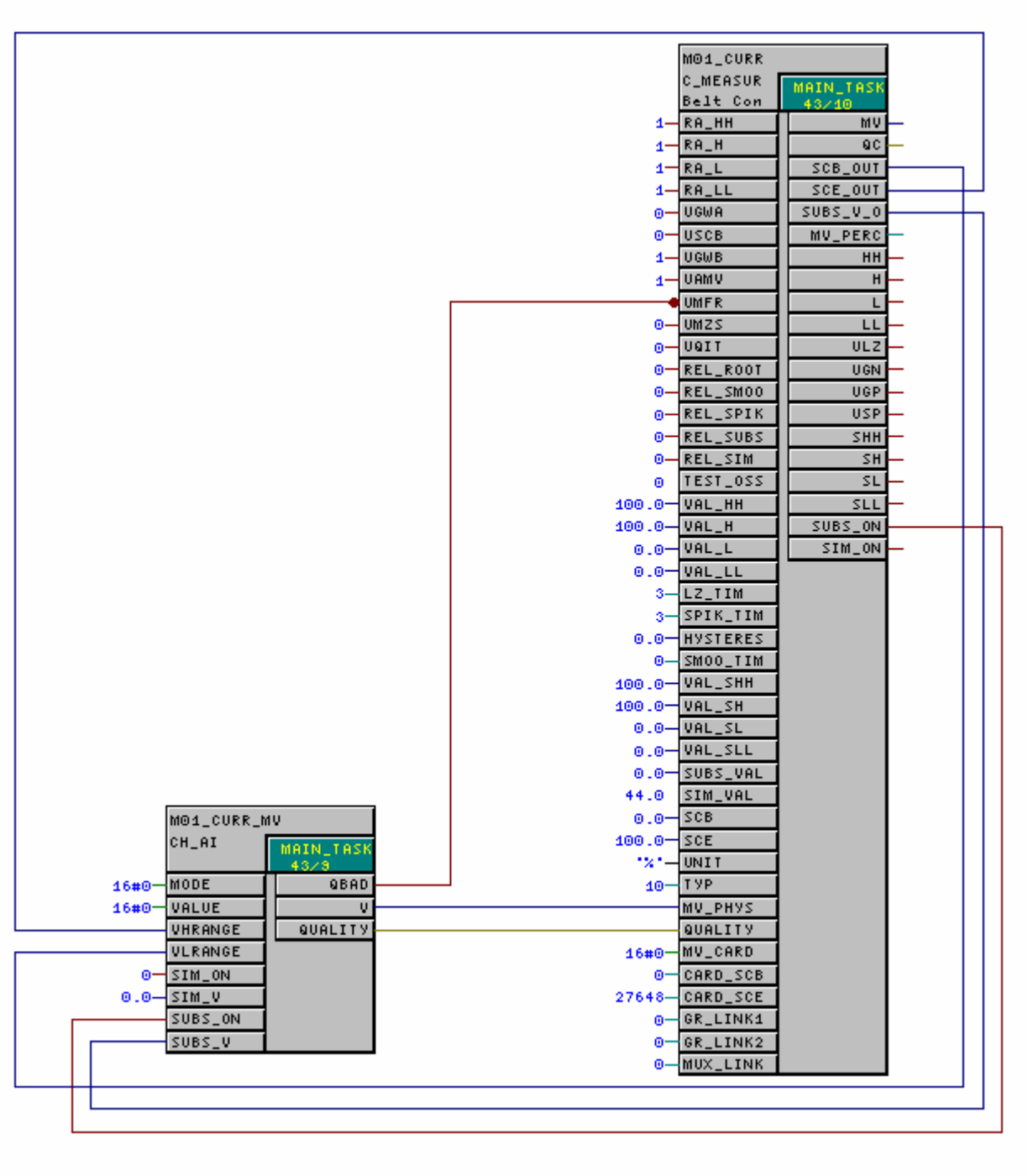
The output SUBS_ON of the Measuring value has to be connected to input SUBS_ON of the driver block. This enables the change between Substitution value and Last valid value.

The output SUBS_V_O of the Measuring value has to be connected to input SUBS_V of the driver block. The driver will get the simulation value from 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!!!

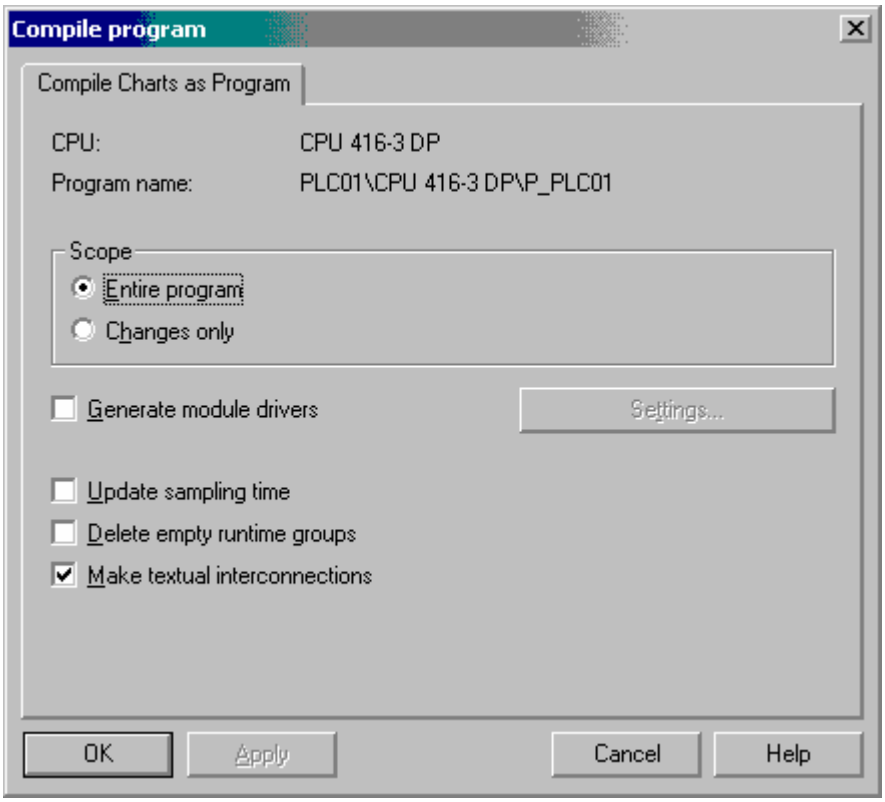
Example:



Note: In example shown above, the Bad Quality Signal was connected to Annunciation Release UMFR to prevent alarms in case of a module failure. Please take in mind that for Annunciation Release additional signals may be included.

Compile CFC

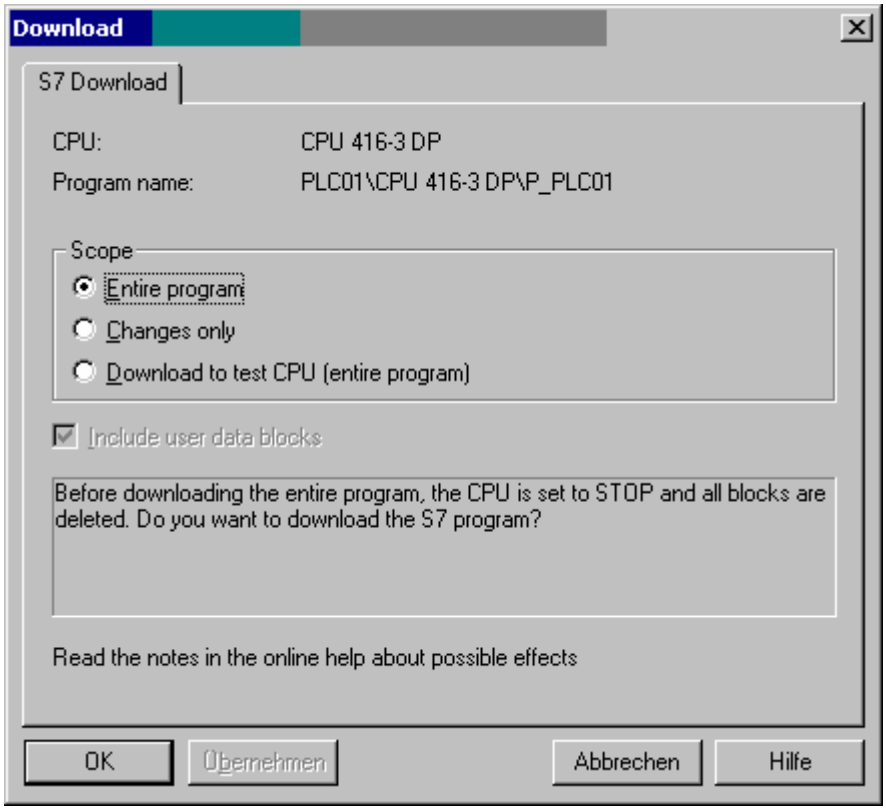
Before you can download the program into the PLC it must be compiled. It will be transferred into a machine language (SCL). During compilation the program will be checked for errors and warnings. You can compile the complete program or only the changes.



Download to the PLC

First the hardware definitions and the network configuration has to be downloaded into the PLC (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 a PLC stop.

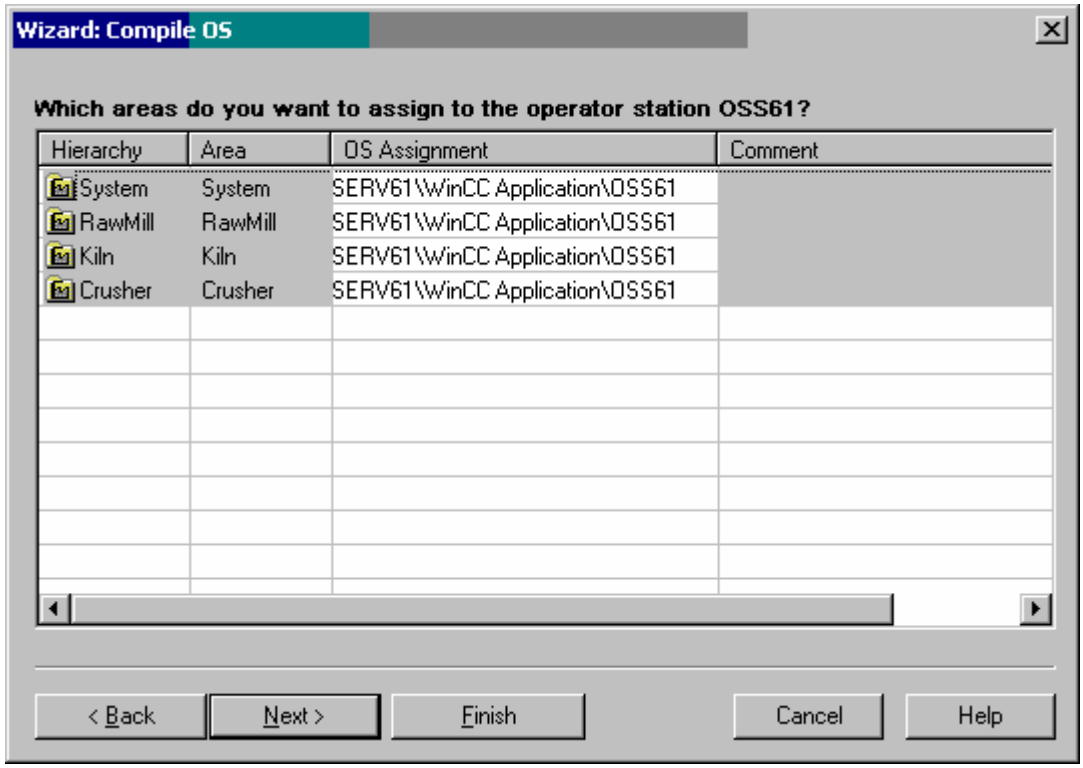


OS Compile

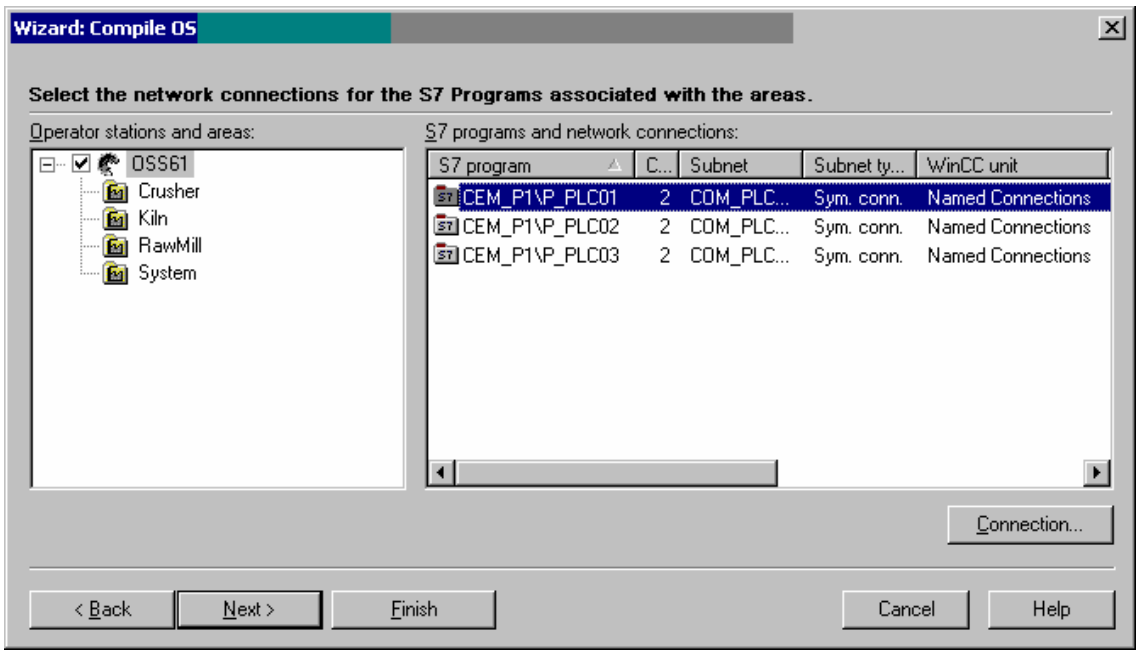
In order to make the object variables available in the tag management of WinCC the OS Compile must be carried out. This function you can find in the SIMATIC Manager under Options -> OS -> Compile.

In the step ‘Compile OS’, according to the selected scope of the compilation the Picture Tree and the block icons can be generated/updated together with the variable transfer.

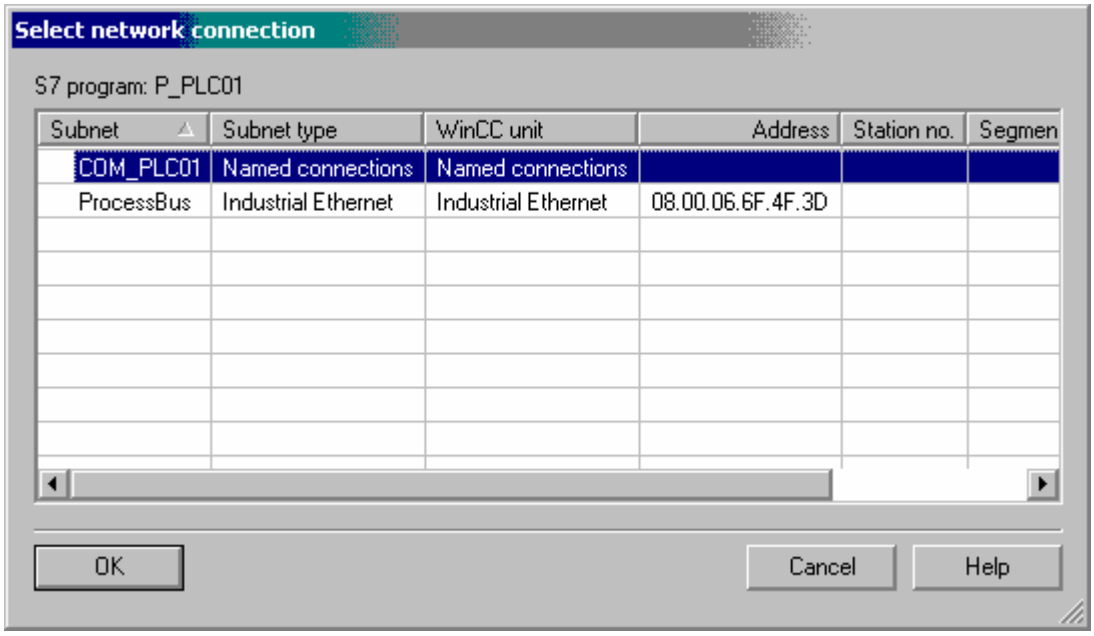
In the following window you see the assignment of the areas to the operation station.:



In the following window you can select the network connections for the S7-Programs associated with the areas.



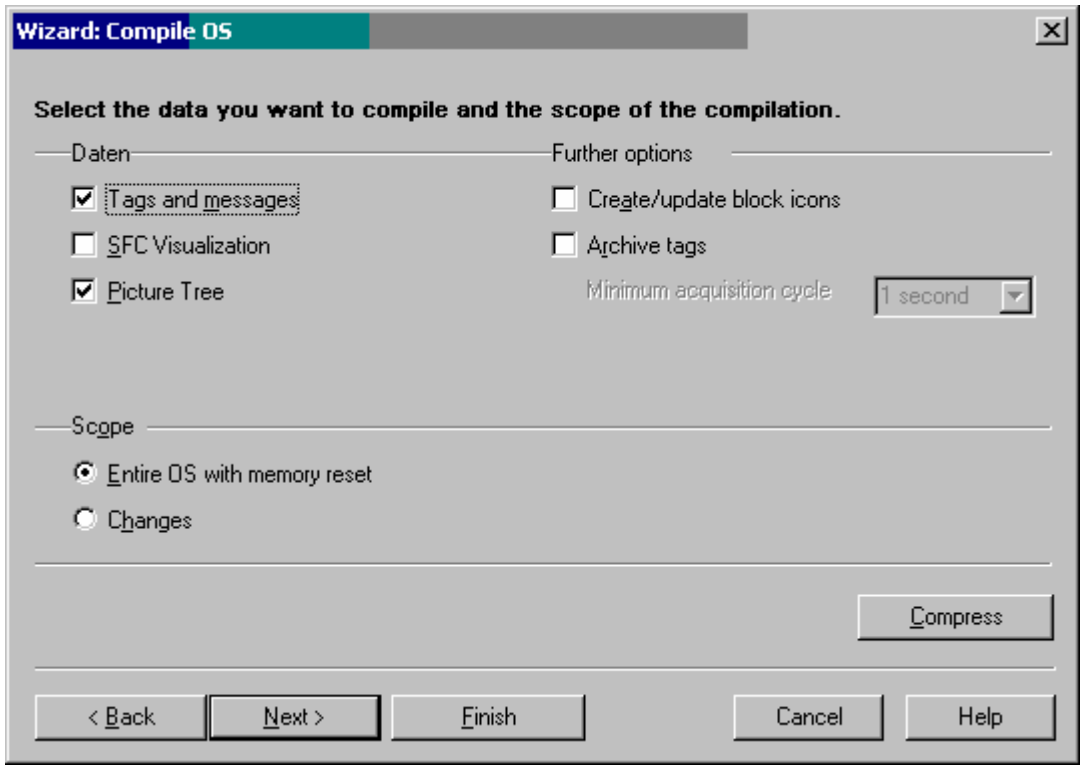
Select each PLC and press “Connection”



Select the network connection you want to use for the communication to WinCC.
In PCS7 V6 for a real PLC always use ‘Named connections’. In case of PLCSIM you have to use MPI.

Finally select the transfer data and the scope of compilation.

Don’t forget to deselect the picture tree, if you created the picture tree manually in the OS-Project. Otherwise it is overwritten!



Now press “Finish” to start the transfer.

PLC-PLC Coupling to older CEMAT Versions

Inhalt

Version : 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\07_PLC-PLC-Coupling.doc

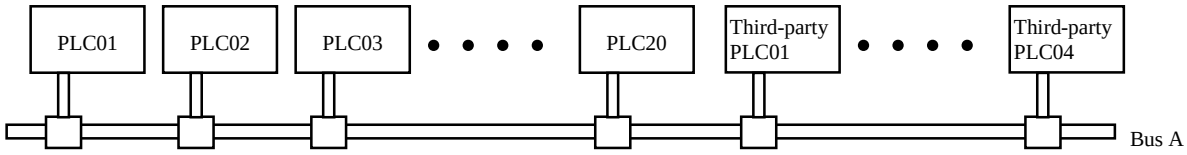
General

Advice: In the following description the expression „AS“ is used for automation system. Only in the TSAP and in some tables the expression „AG“ is used for automation system. This is necessary to be compatible with older CEMAT versions (V3, V4).
This PLC PLC communication should exclusively be used for communication to AS with older CEMAT versions (V3, V4), because the connection type has to be a ISO transport connection. For the communication between V6 PLCs, we recommend “S7 connections” and the standard communication blocks from the PCS7 V6 library.

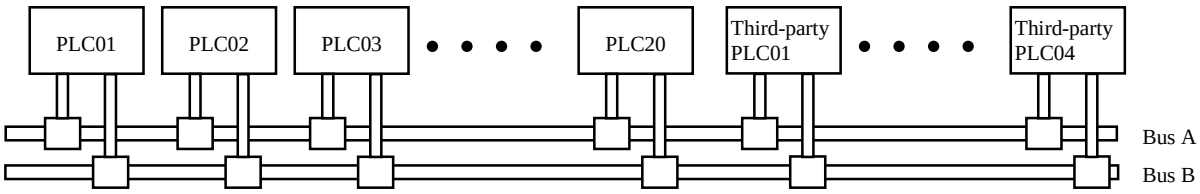
The PLC-PLC Coupling CEMAT-V6 (PC S7) enables the transmission of user data blocks (DB599 >= DB => DB 400) between SMR-PLCs. The communication is performed via Industrial Ethernet which can be designed as a double bus. The user configures the job parameters in CFC with the blocks C_PLC_SEND = FB1052 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.

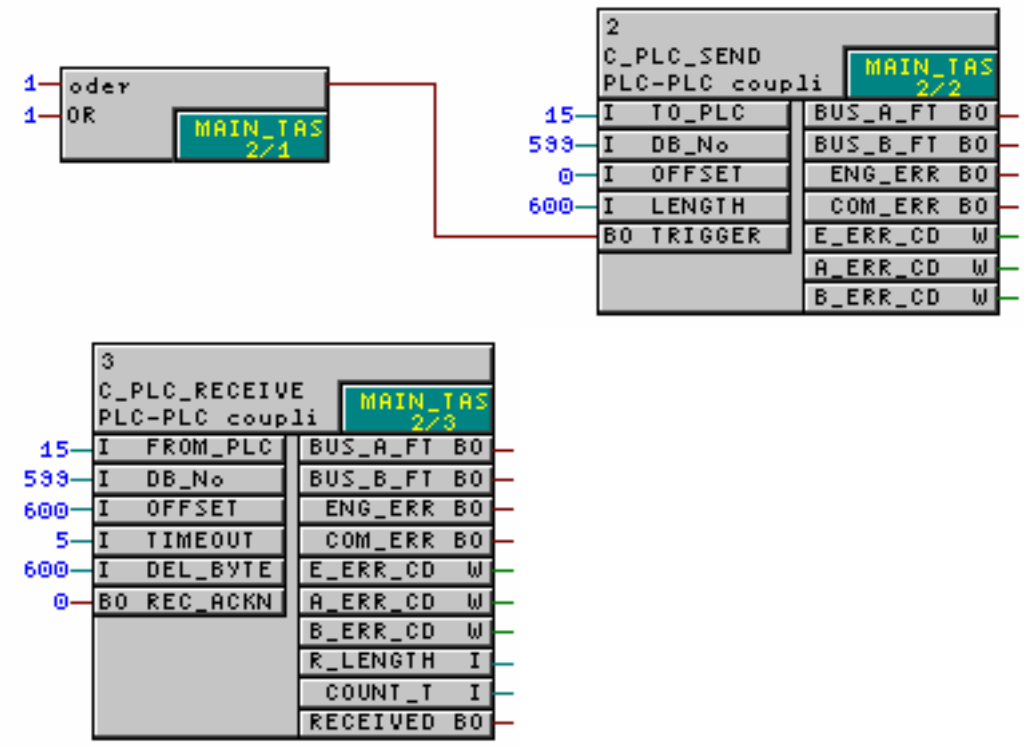
1. Parameterize the send parameters on the function block C_PLC_SEND = FB1052.
2. Parameterize the receive parameters in the corresponding PLC on the function block C_PLC_ RECEIVE = FB1053.
3. Programming the logic for initiation (SET TRIGGER).
4. Compile the CFC chart new and transfer the changes to 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

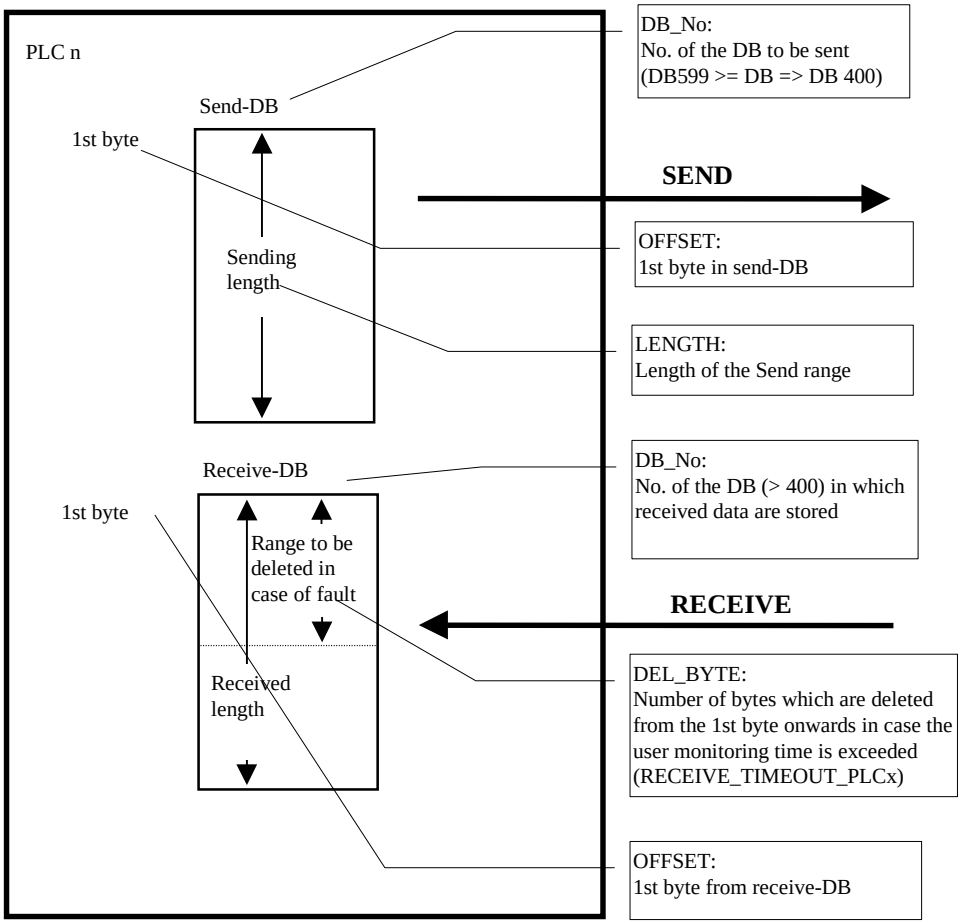
All settings are made in the own PLC. There are no settings (like DB no.) for the partner PLC. The send parameters are made in the sending PLC. The receive parameters are made in the receiving PLC.

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).



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

Parameterization Error-Codes

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!

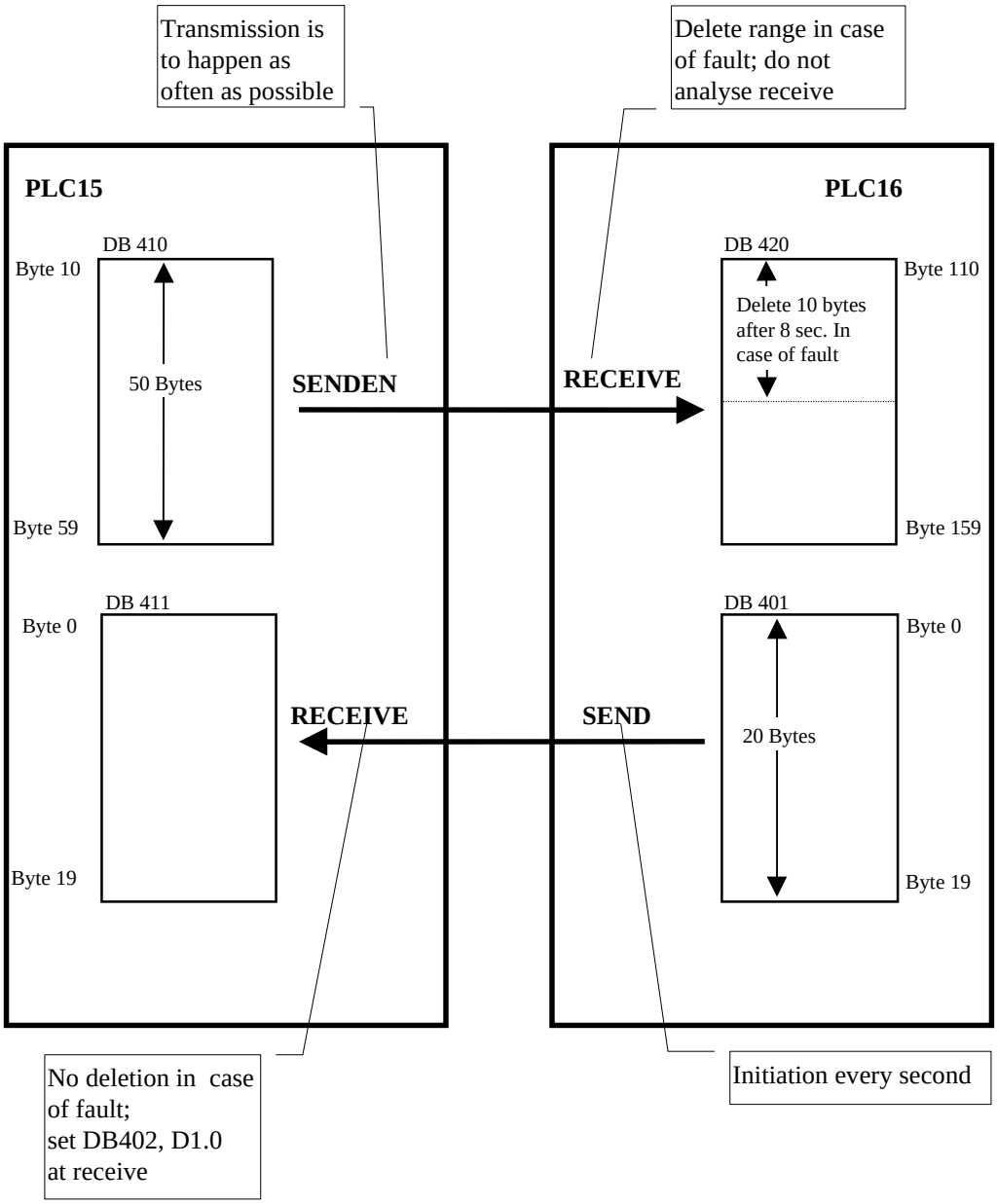
Status codes:

"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.

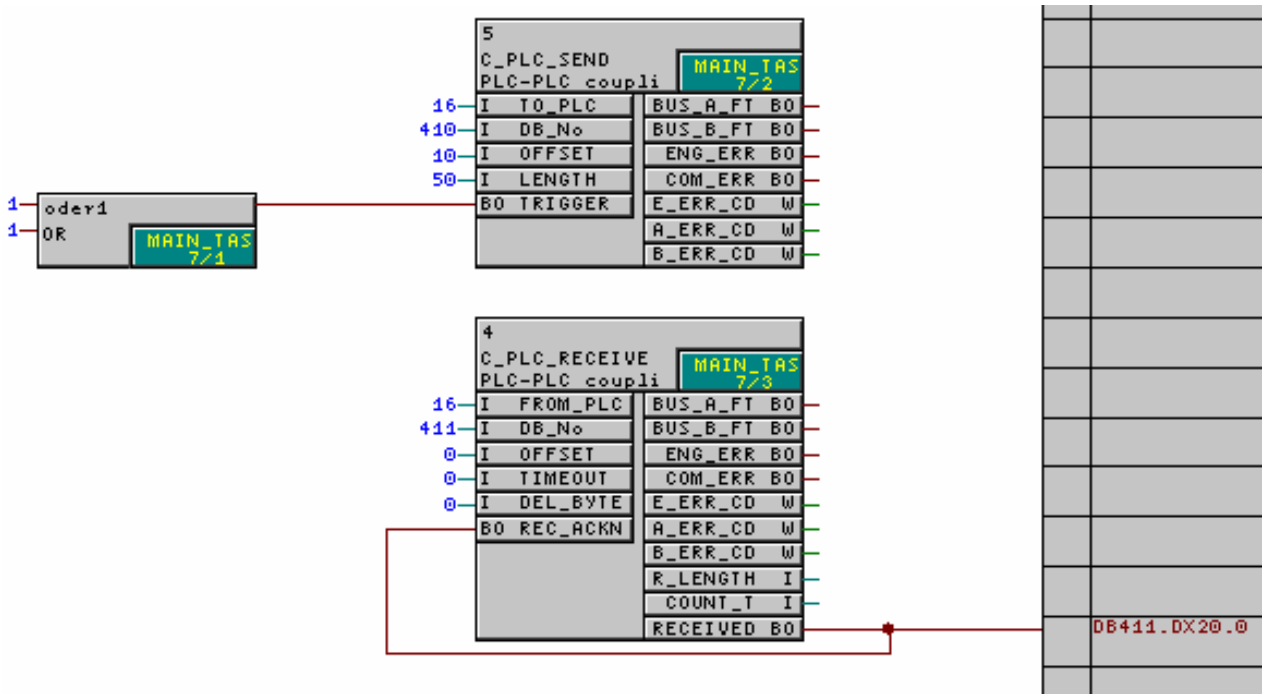
Configuration Example

Exemplary Description

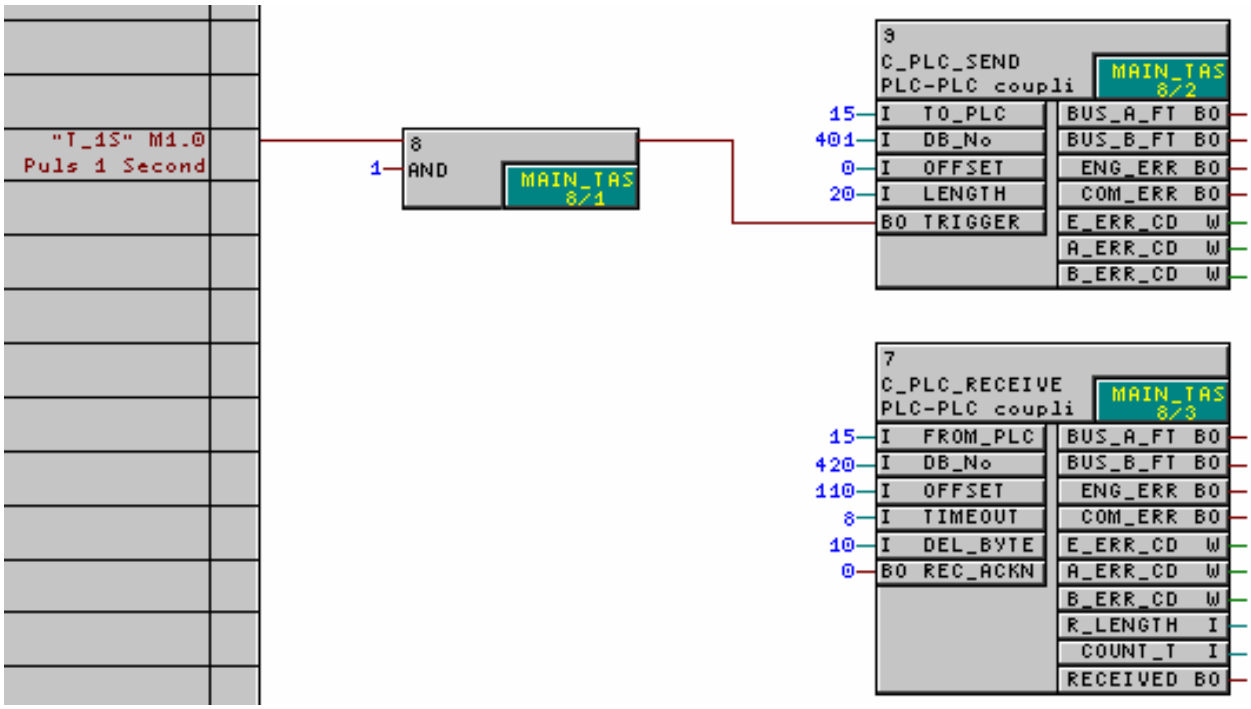


CFC chart for example

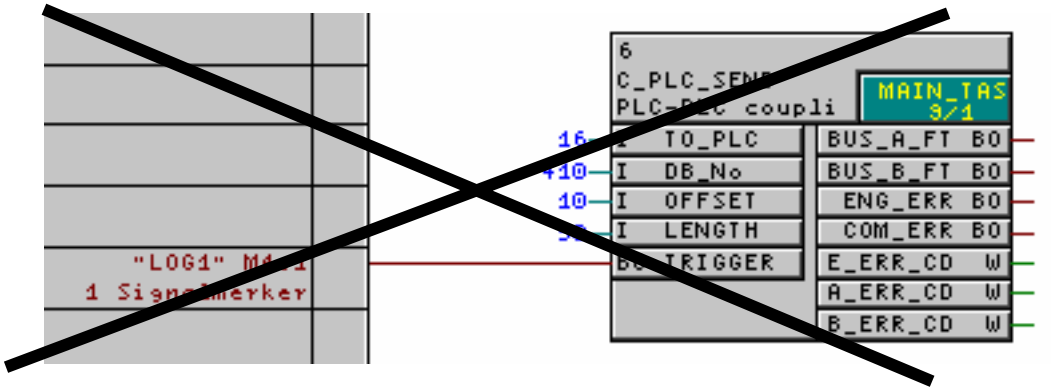
PLC 15:



PLC 16:

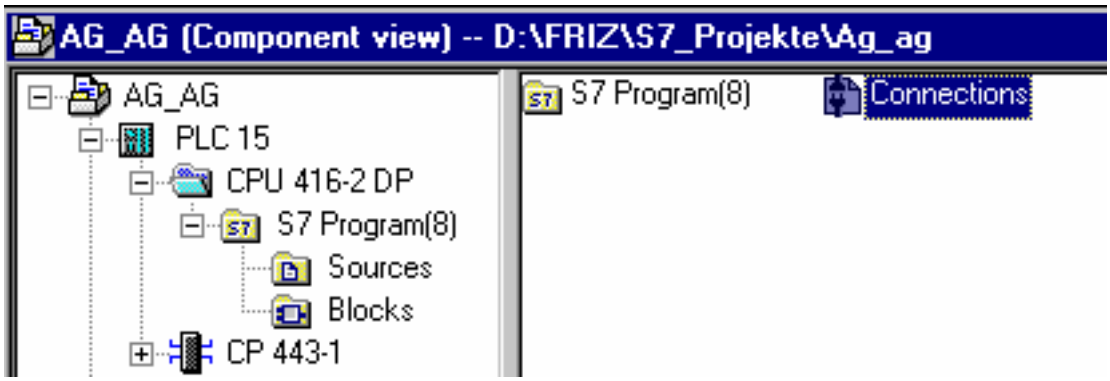


Please be careful during connecting the TRIGGER bit on the SEND block. Please do not use the LOG1 signal. The LOG1 signal would be reseted after a succesful SEND job.

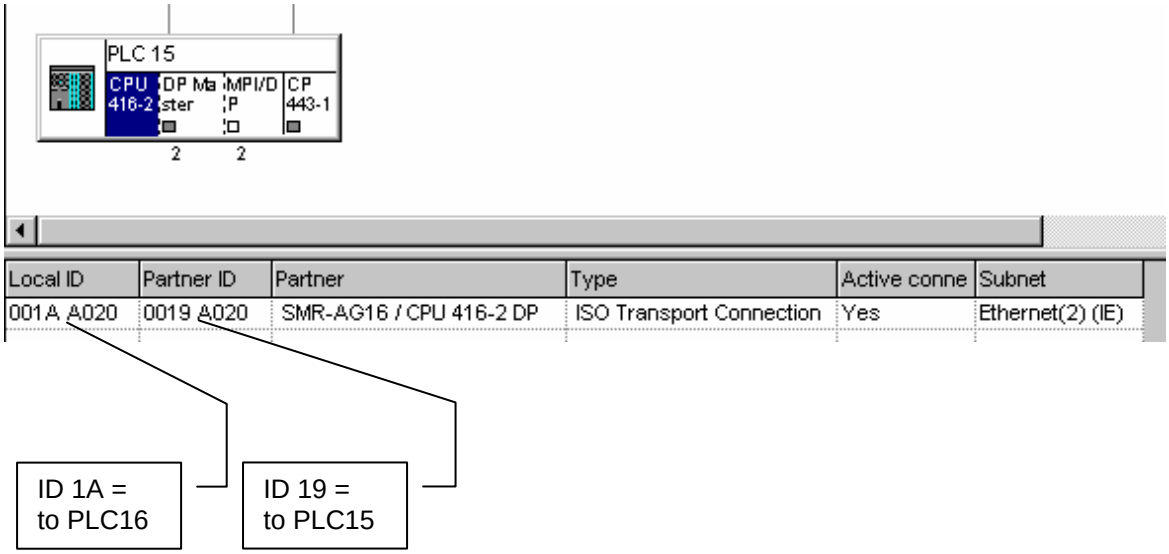


Project work in the network settings

Select in the SIMATIC Manager the CPU container and open NETPRO with double click on the connections.

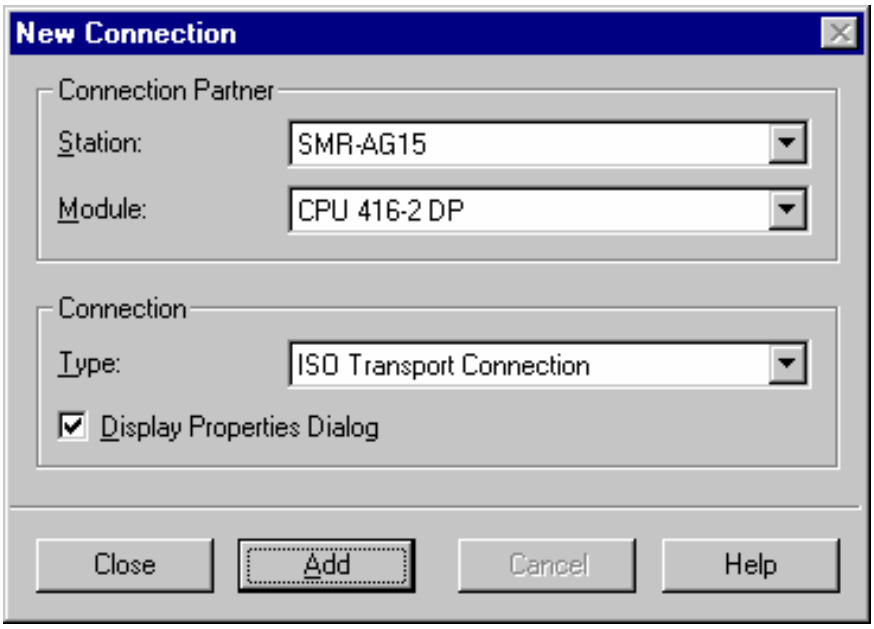


Select the CPU -> you see the connection table:



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.

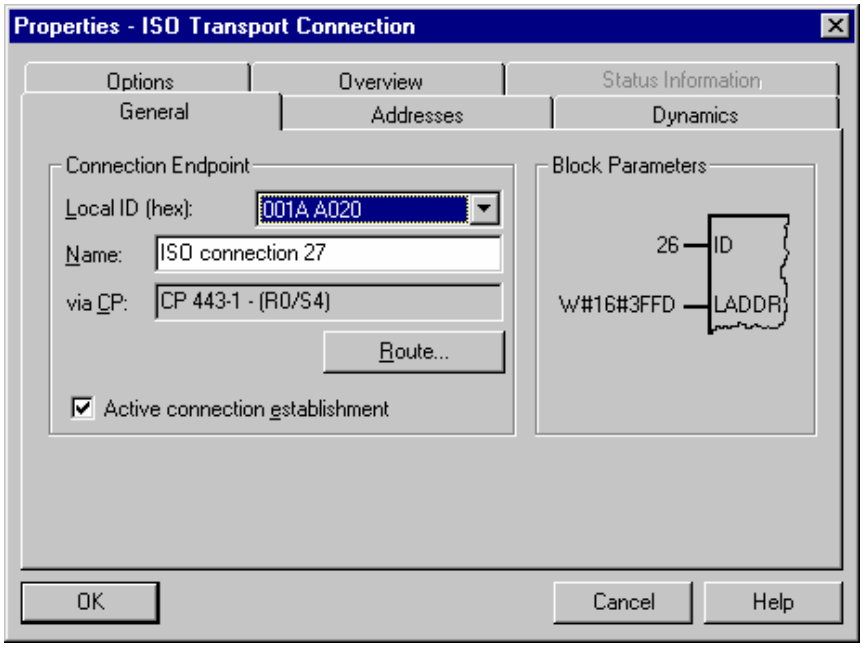
Connection type

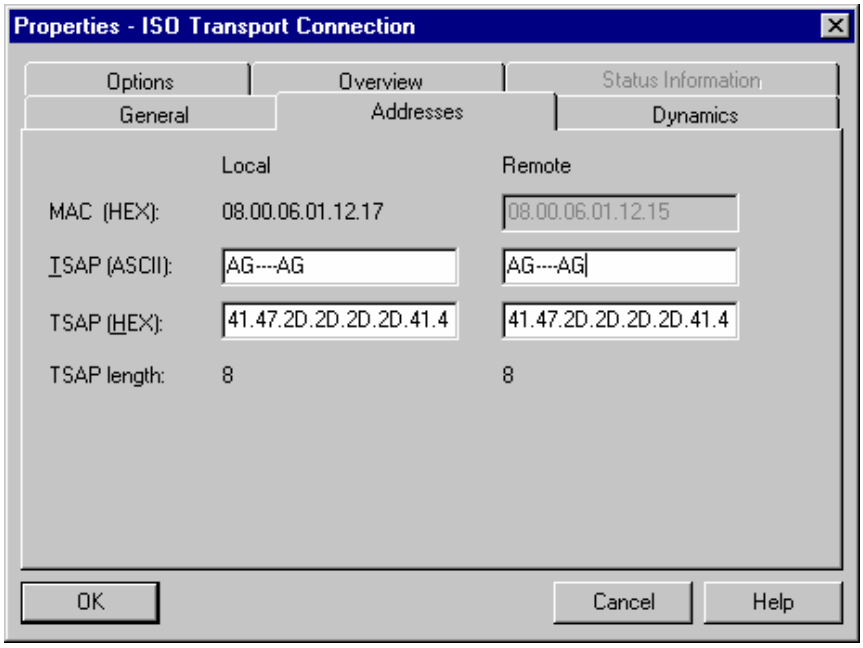
The connetion type is *ISO-Transport-Connection*

(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

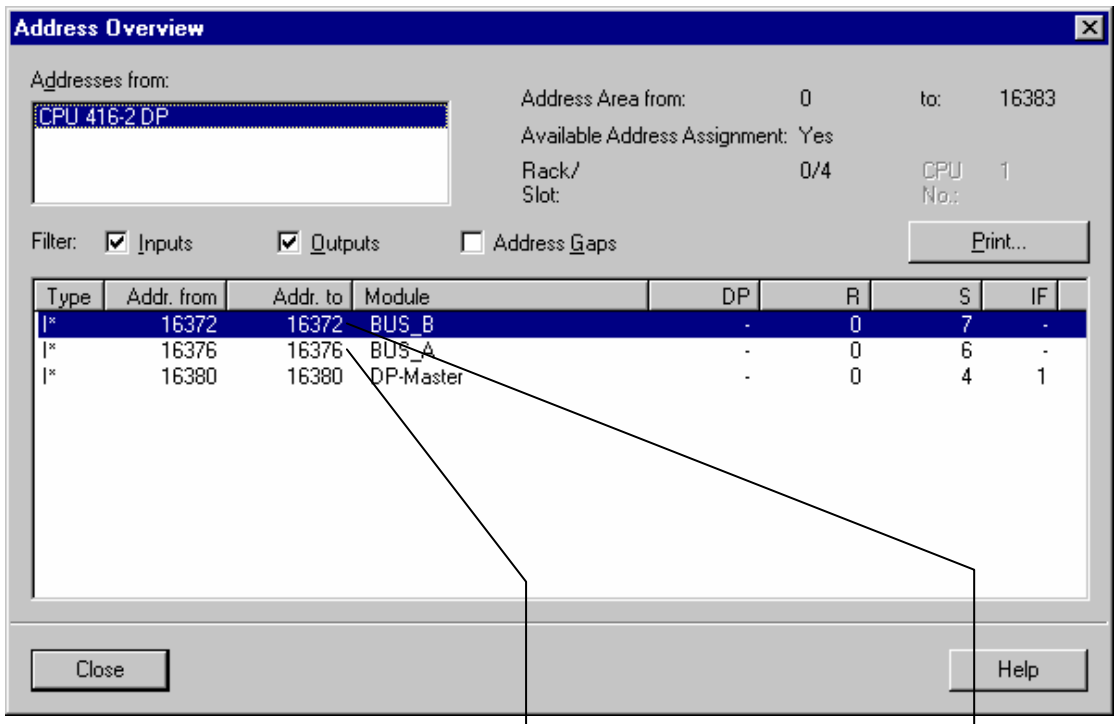
For PLC-PLC-Connections is defined: The communication partner with the smaller PLC no is **activ** for the connection setup.

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.

(0) UR2

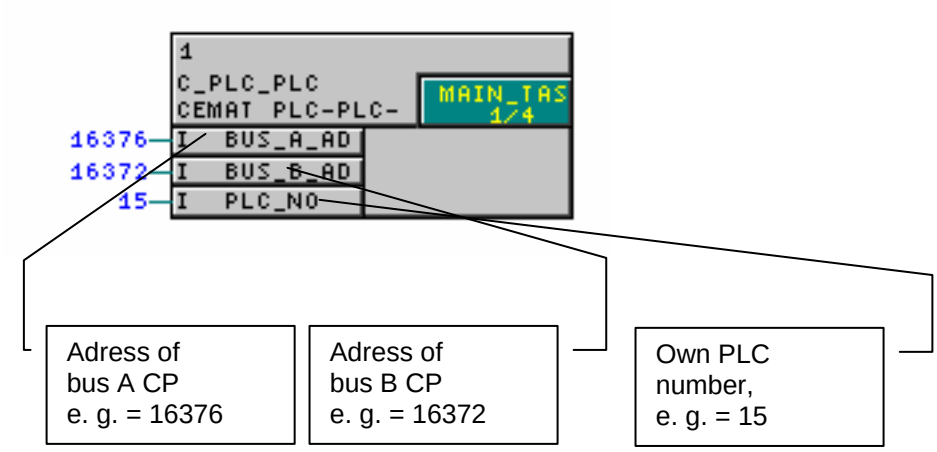
Slot	Module	Order number	MPI address	I address	Q address	Comment
1	PS 407 20A	6ES7 407-0RA00-0AA0				
4	**CPU 416-2 DP**	**6ES7 416-2XL01-0AB0**	**2**			
X3	*DP-Master*			*16380**		
6	BUS_A	6GK7 443-1EX02-0XE0		16376		
7	BUS_B	6GK7 443-1EX02-0XE0		16372		
8						
9						

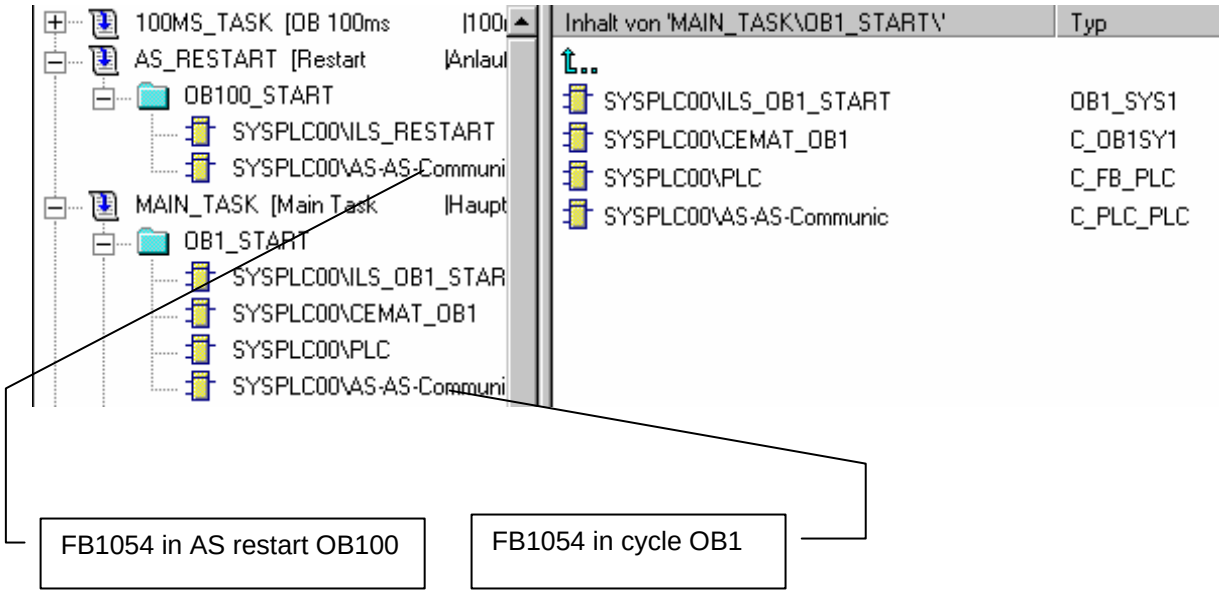


16376 =
e. g. adress of
Bus A CP

16372 =
e. g. adress of
Bus B CP

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.

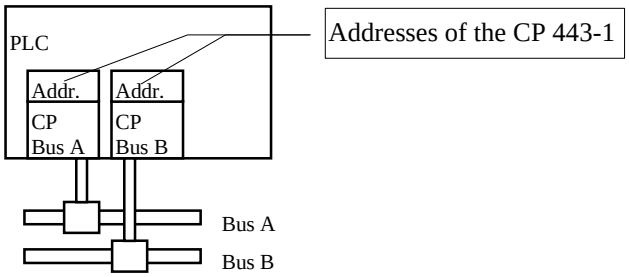
Remedies in Case of Faults

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

Ausgabe : 01.04.2005
D:\CEM_V6\DOCU\English\Engineering\08_OS_Engineering.doc

Generation of Process Pictures

All process pictures which should be available in the picture tree (for direct access from Overview Range) 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) have to be created in the component view of the SIMATIC Manager (directly assign to the OS-Project).

For a screen resolution of 1280x1024 the picture size has to be 1280x825.

Templates

During the Installation procedure, three Template-Pictures have been copied into your project, which contain predefined symbols (block icons) for CEMAT Objects. The block icons in these pictures are examples and have to be adapted according to the requirements of the project.

Note: 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.pdl** contains the default symbols for automatic generation of block icons from SIMATIC Manager. In principal these are the same symbols as they were used in CEMAT V6.0. This picture should be used in case the PCS7 Project was already created with CEMAT V6.0 and shall be upgraded with minimum effort.

The picture **@C_PCS7Typicals_V61.pdl** contains as well as picture @C_PCS7Typicals.pdl 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 PCS7 V6.1. If you start with a new project with CEMAT V6.1, you should use the templates from this picture.

During the automatic generation of block icons PCS7 V6 is looking for a template picture with the name @PCS7Typicals.pdl. From PCS7 V6.1 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.pdl** or **@C_PCS7Typicals_V61.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 unnecessary long because always the complete picture is scanned!)

In the following description you can find the attributes, which are important for CEMAT.

The picture **@C_Template.pdl** contains the default symbols for manual connection in WinCC (using wizard). It corresponds to the template picture @C_PCS7Typicals.pdl for the automatic generation.
A similar picture for the @C_PCS7Typicals_V61.pdl is not available. You have to create you own template picture.

You should also adapt the template picture @C_Template.pdl and save it under a different name (@Template1.pdl) in order not to overwrite the picture in case of an Update.

Note: The Update Cycle for all symbols in the default PDLs is set to “User Cycle 1”.
You have to adapt the value for the “Update Cycle 1” in the WinCC Explorer in the general settings of your project.

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 pictures @C_PCS7Typicals.pdl and @C_PCS7Typicals_V61.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.

The function for the automatic generation of block icons is available in the Plant View or Process Object view of the SIMATIC Manager under Options -> Plant Hierarchy -> Create/Update Block Icons or as a Further Option during OS Compile Function.

After the automatic generation the symbols are located in the upper left part 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, Window position when opening the faceplate.

In order not 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 was already copied into the OS-Project folder under WScripts.

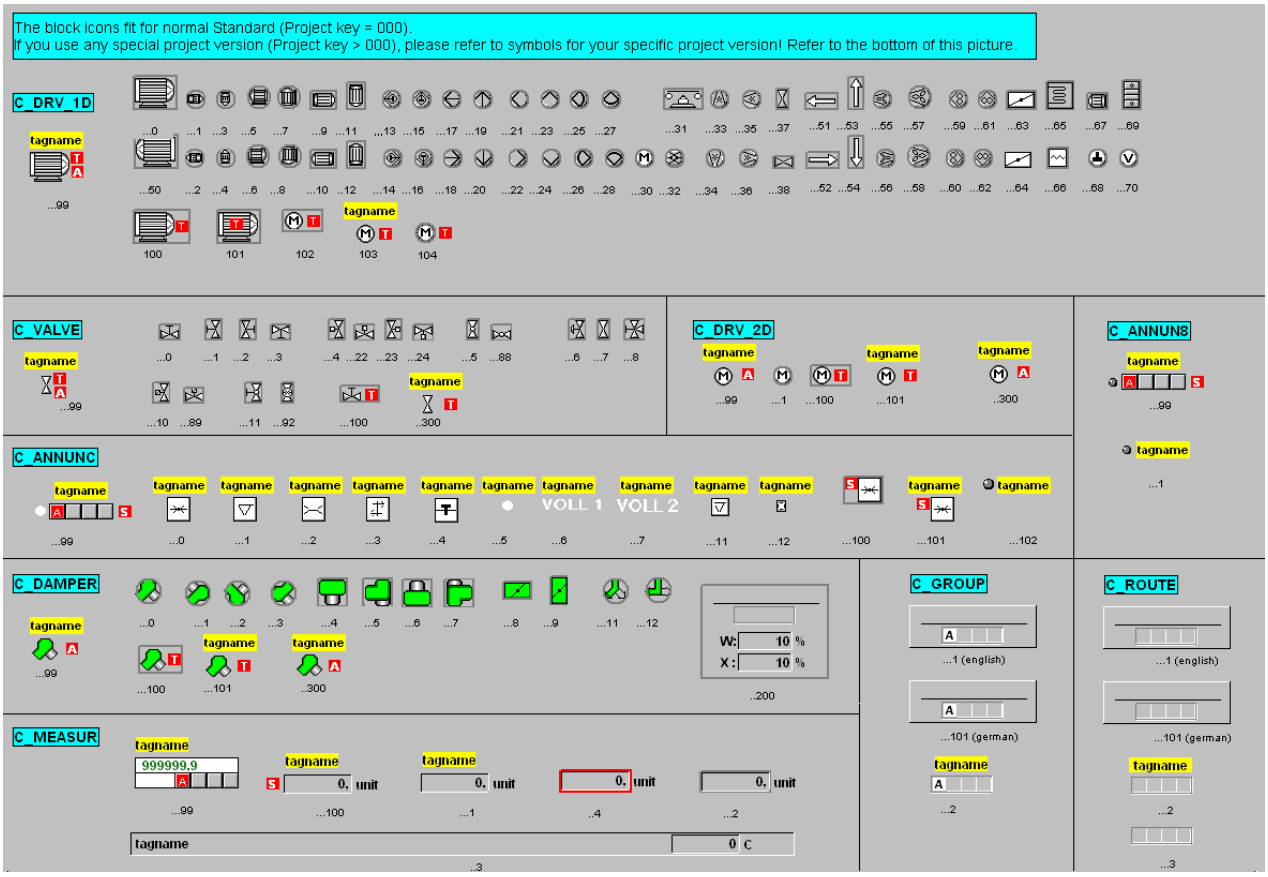
Manual symbol generation

Open picture @Template1.pdl and copy the required symbol into your process picture. Use wizard or manually link the symbol to the variables of your object. After that you may modify and adapt the symbol properties.

After that you can adapt the symbol properties (e.g. TooltipText, Formats, Window Position when opening the faceplate).

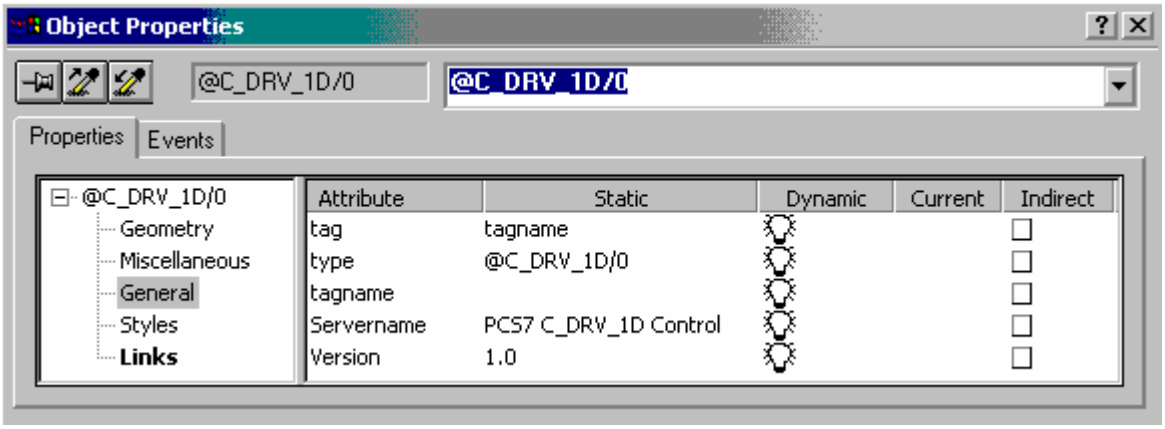
@C_PCS7Typicals_V61.pdl

The template picture @C_PCS7Typicals_V61.pdl contains the block icons for automatic generation of CEMAT Objects. The symbols include the functionality of PCS7 V6.1.



Caution: The Symbols from CEMAT V5 can not be used in CEMAT V6 any more!

Block Icon properties in @C_PCS7Typicals_V61.pdl

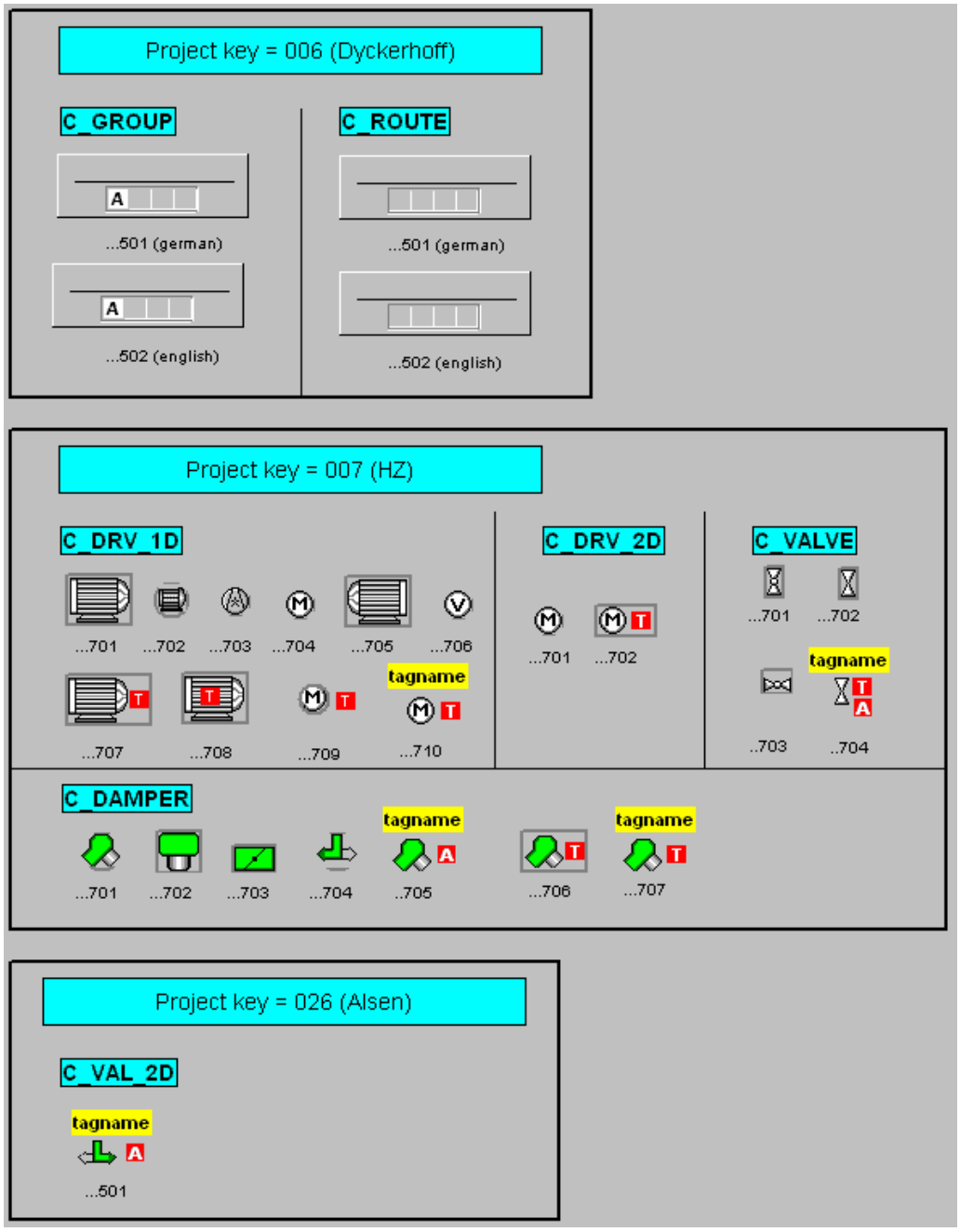


Property type contains the @block type and the index (e. g. @C_DRV_1D/0 for index = 0).

For some of the project versions of CEMAT, modified symbols are required. These symbols you find at the bottom of the template pictures.

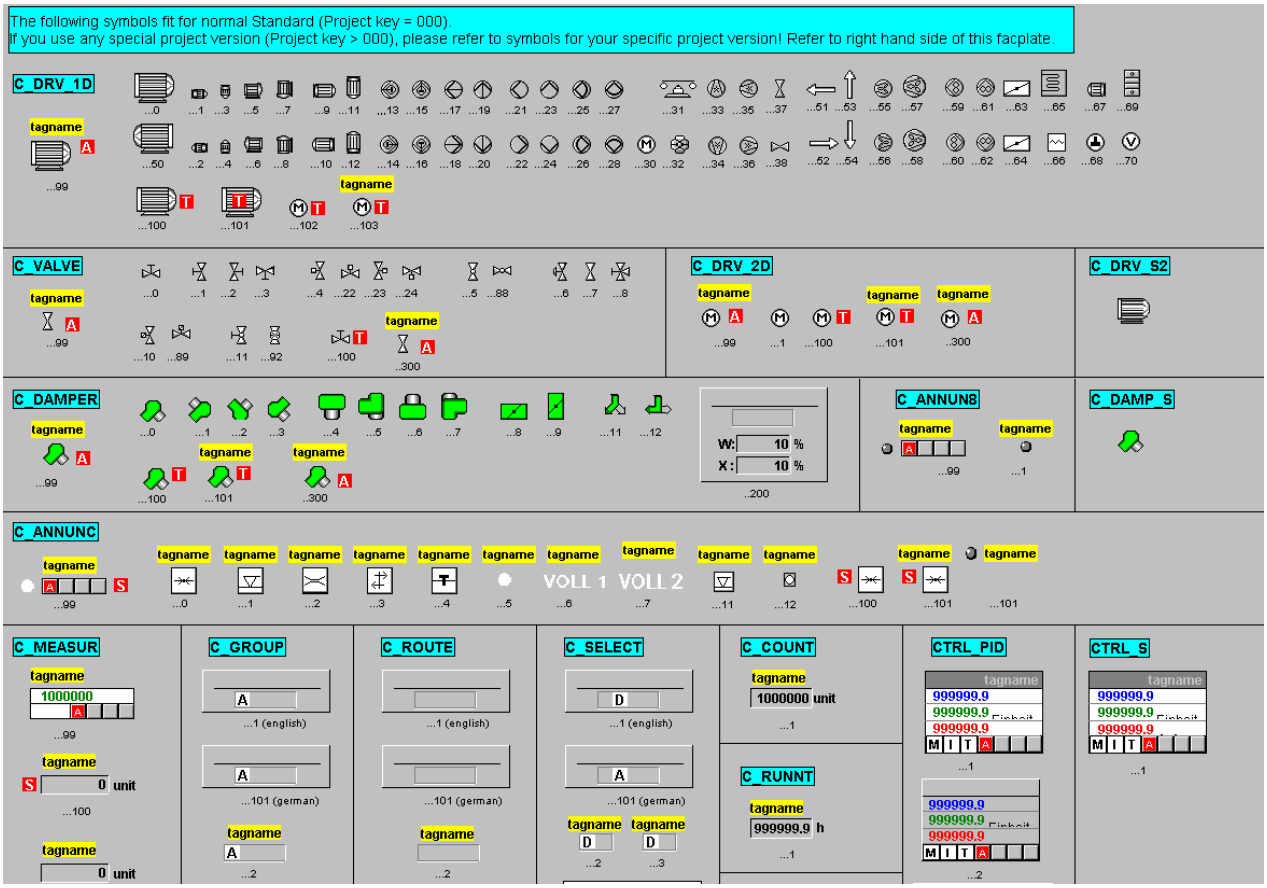
If you use a project version (Project key > 000) please check if special symbols are available! In this case, if you use the normal symbols, some functions are not guaranteed.

Example for Project Symbols:



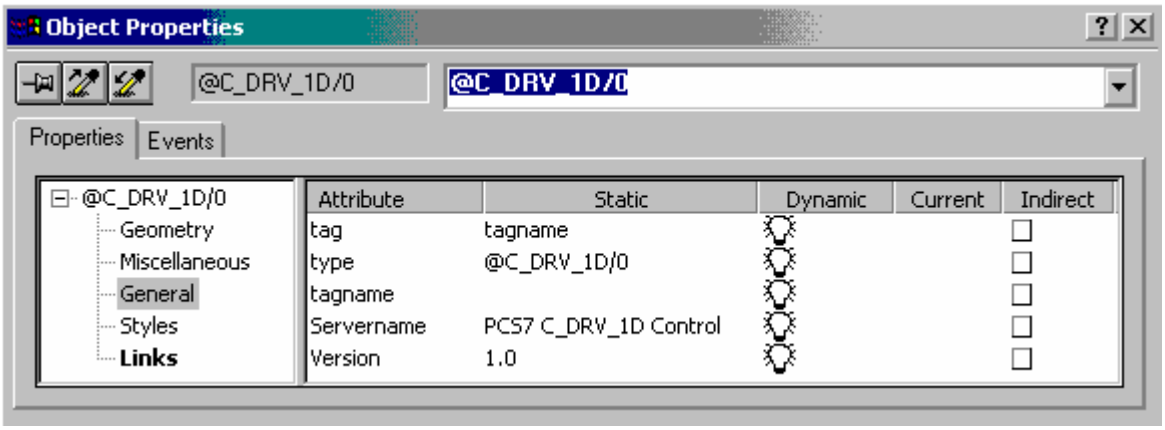
@C_PCS7Typicals.pdl

The template picture @C_PCS7Typicals.pdl contains all block icons for automatic generation of CEMAT Objects, which have been used already in CEMAT V6.0.



Caution: The Symbols from CEMAT V5 can not be used in CEMAT V6 any more!

Block Icon properties in @C_PCS7Typicals.pdl

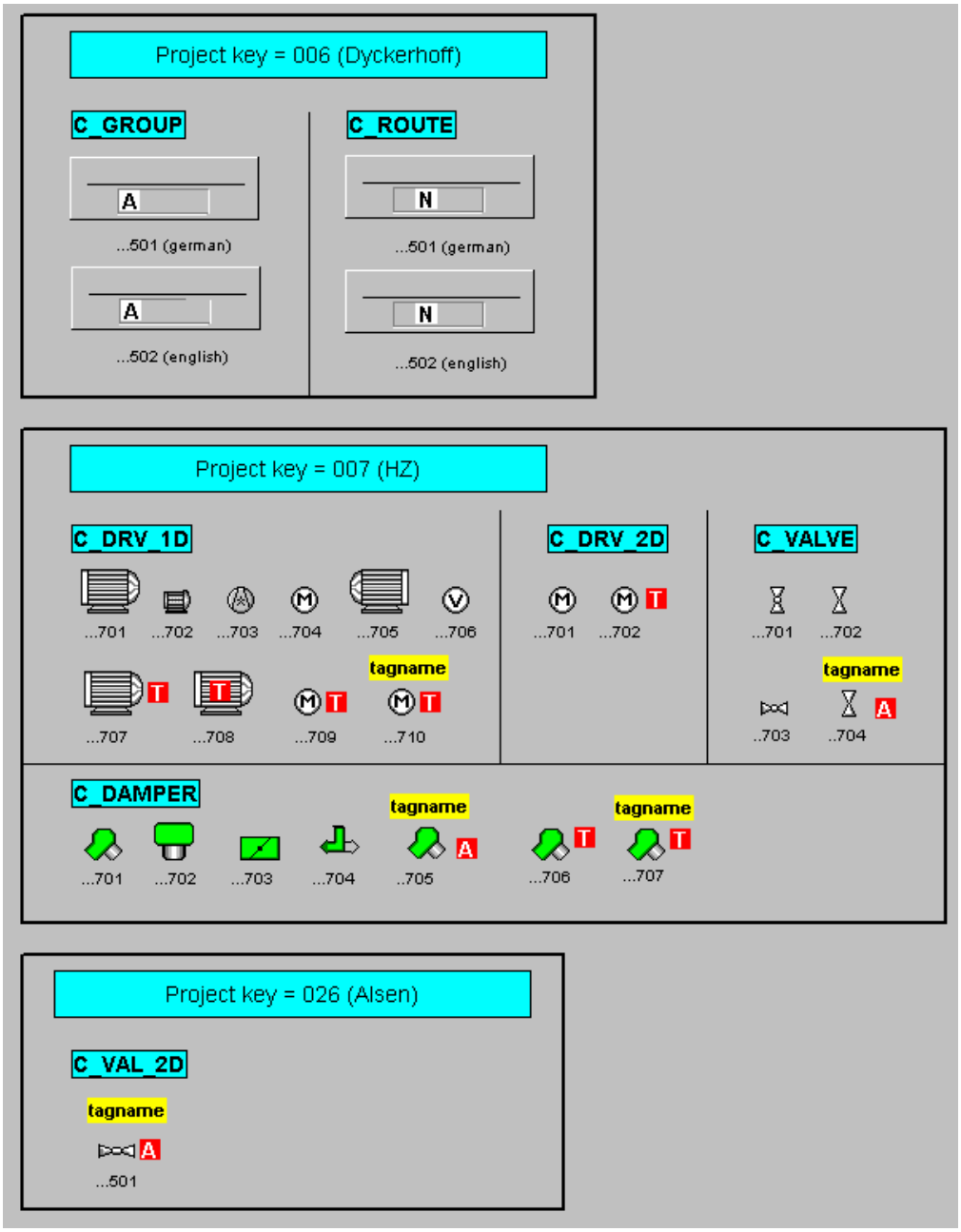


Property type contains the @block type and the index (e. g. @C_DRV_1D/0 for index = 0).

For some of the project versions of CEMAT, modified symbols are required. These symbols you find at the bottom of the template pictures.

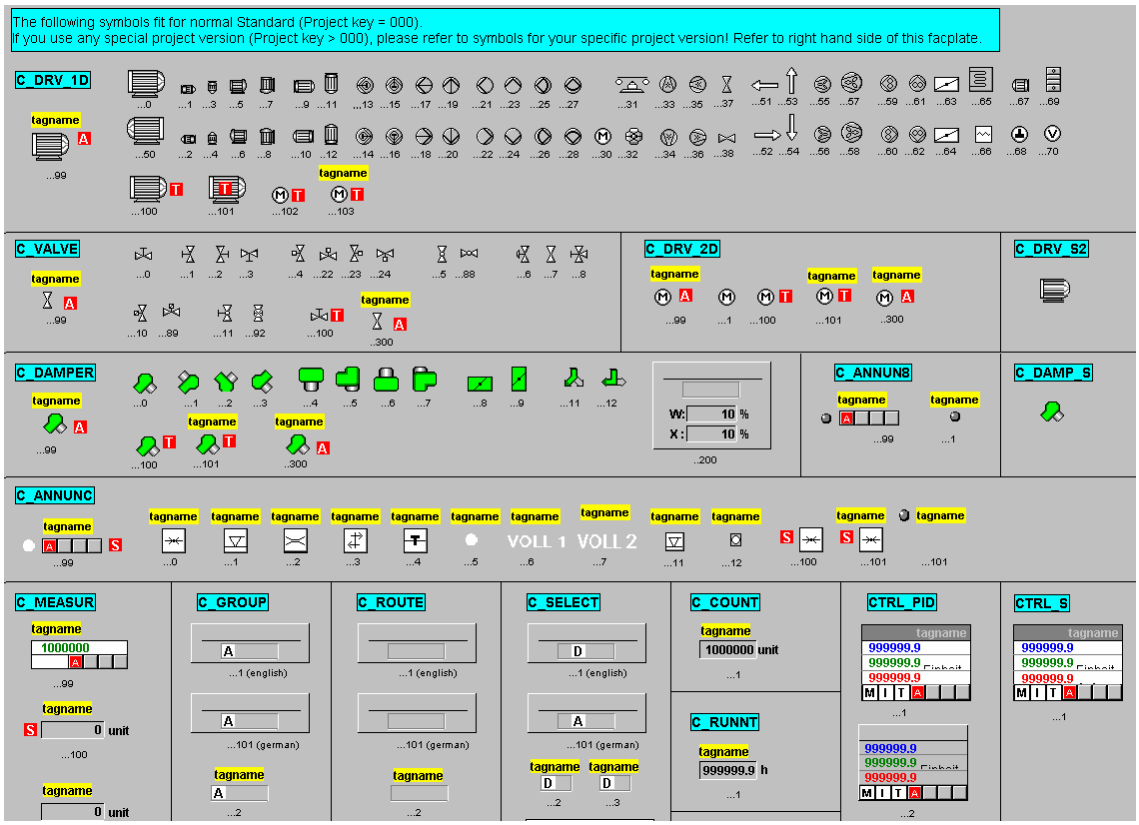
If you use a project version (Project key > 000) please check if special symbols are available! In this case, if you use the normal symbols, some functions are not guaranteed.

Example for Project Symbols:

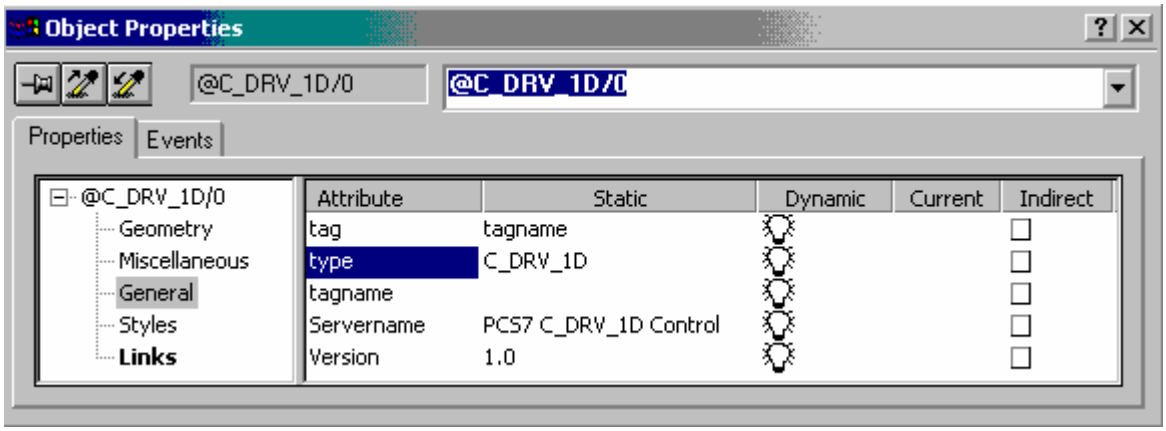


@C_Template.pdl

The template picture @C_Template.pdl contains all block icons for manual connection to variables. The symbols are exactly the same as in @C_PCS7Typicals.pdl, only property type is different.



Block Icon properties in @C_Template.pdl



Property type contains the block type (e. g. C_DRV_1D).

Caution: This property type is different in @C_PCS7Typicals.pdl or @C_PCS7Typicals_V61.pdl and in @C_Template.pdl.

Name definitions

Picture Names

See chapter preparations – Designations for the OS.

Object names

The object names are created in the Engineering of the CFC. In the OS only selection of existing objects is possible. New Objects can not be created in WinCC.

Properties of the Block Icons for CEMAT

Most of the CEMAT objects are available as Customized object (all objects, except TE_CTRL and TE_VSLCT). They can be generated automatically using template picture @PCS7Typicals.pdl or linked manually using templates from @Template.pdl.

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 other attributes, except Tooltip-Text are standard and don't have to be modified.
  The Tooltip-Text is not automatically generated and must be entered manually.
- In Property **Styles** the appearance of the or the faceplate can be modified. The possible settings you will find in the description of the object types below.
- In Property **Links** the symbol is connected to a number 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.

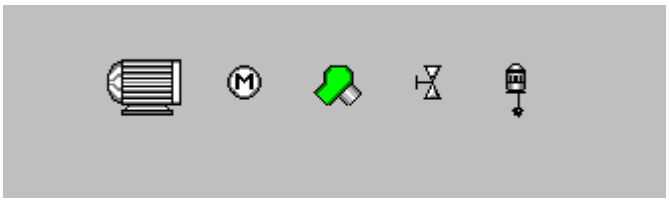
Index of the block icons

The symbols are sorted based on the functionality and numbered accordingly.

from 0 - 98	simple state announcement
99	State announcement with alarm symbol and test symbol
100 - 199	State announcement with "test mode" and "bad Quality" announcement symbol and partly with TAG indication
200 - 299	Special representations
300 - 399	like 100.th Styles with additional properties to label the key in the Faceplates variably. (C_DRV_2D, C_VALVE, C_DAMPER)
>500	Project standards

C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE and C_SILOP

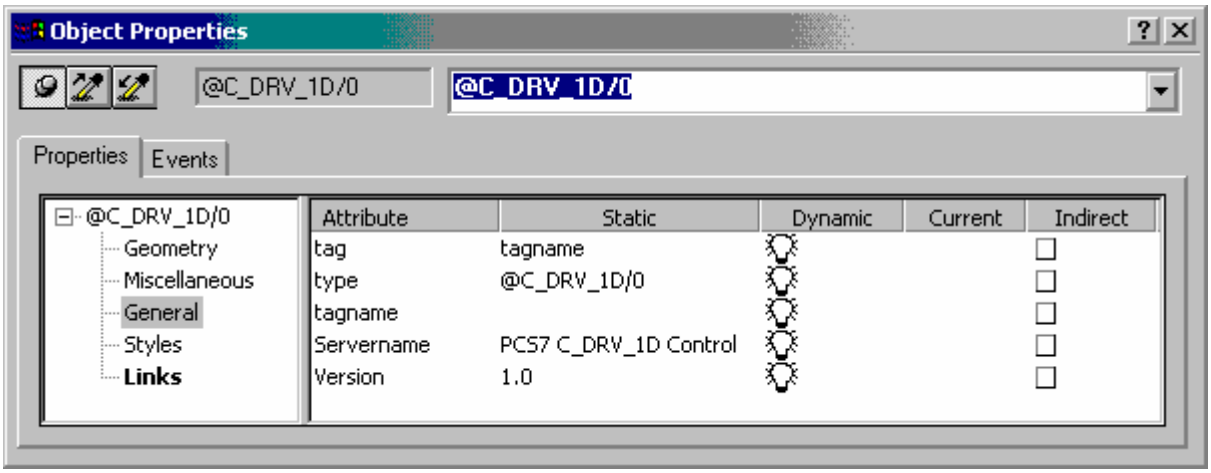
In the CEMAT template pictures, for most of the block icons of C_DRV_1D, C_DRV_2D, C_DAMPER, C_VALVE and C_SILOP only the drive symbol is visible. All further objects belonging to the customized object are hidden behind this symbol.



Each symbol has a unique Object Name, which consists of the object type (FB Name) and an index, e. g. @C_DRV_1D/0.

Property General

Property **General** of object type C_DRV_1D:



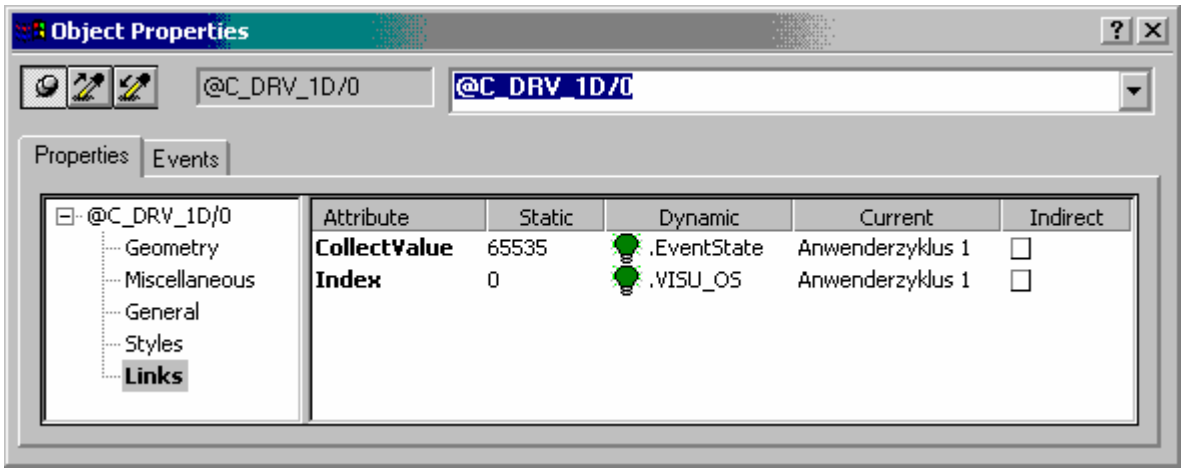
Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by C_DRV_1D.

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):

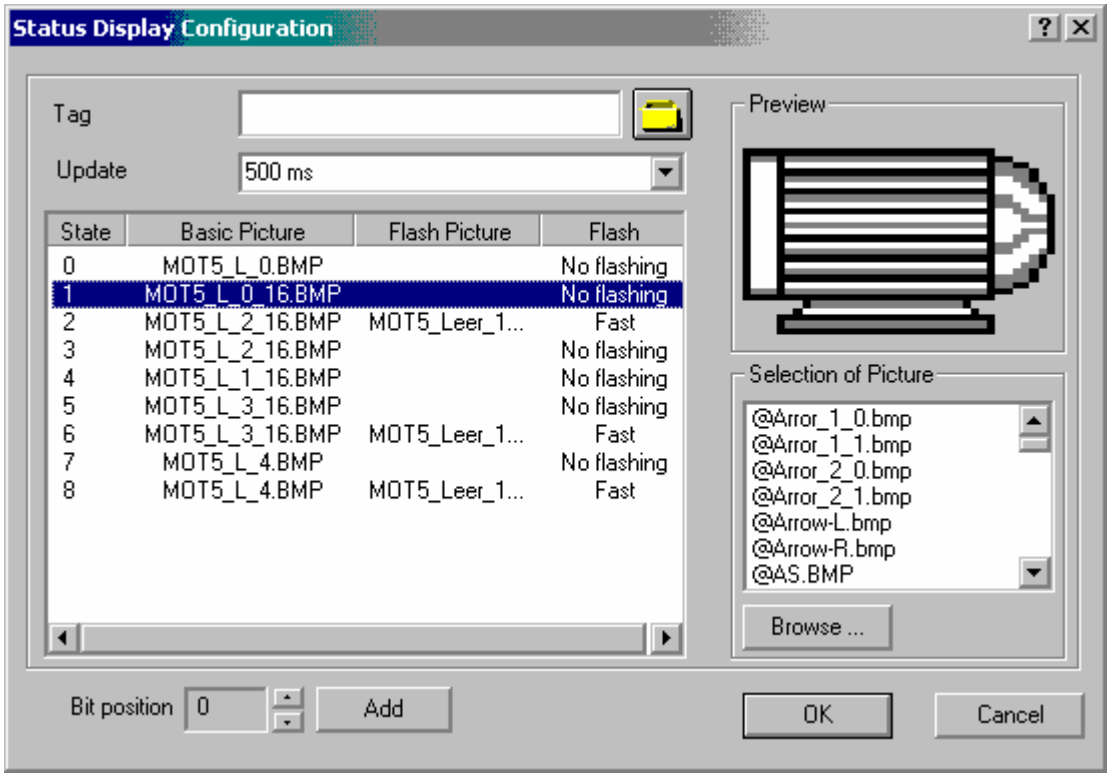
Index	to	.VISU_OS	to show the status of the object
CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)

Property **Links** of C_DRV_1D:



The 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:



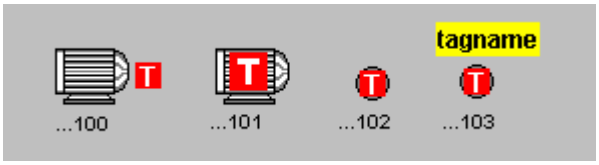
With an additional indication at the Block Icon the status for Sequence Test, Bad Quality and Bypass Speed Monitor (Simulation) can be shown.

Sequence Test T

Bad Quality B

Bypass Speed Monitor S

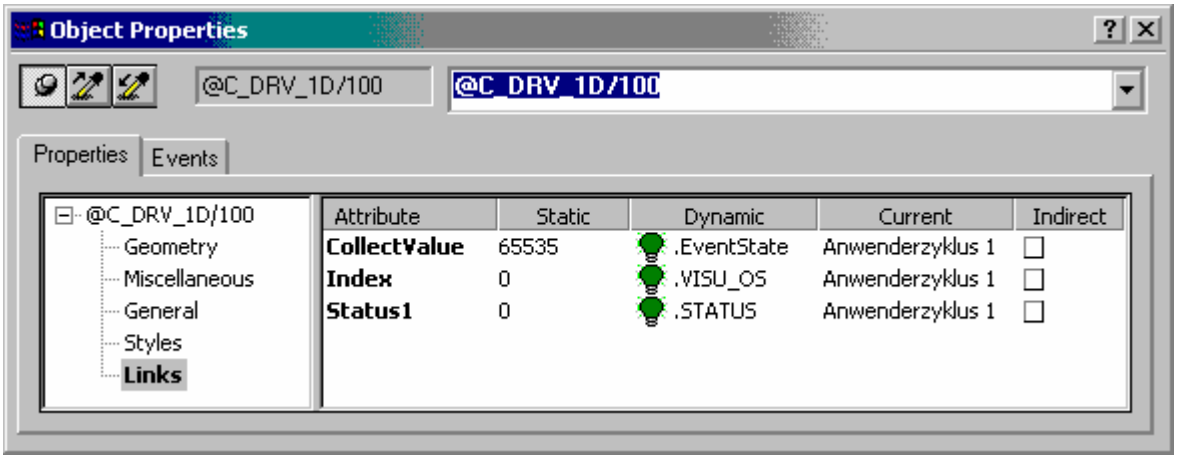
You will find a some examples in @C_PCS7Typicals.pdl. If this function is required, the other symbols have to be adapted in the same way.



An additional connection is required in Property Links (Connection will be generated automatically in case of automatic generation of Block Icons):

Status1 to .VISU_OS to show the status of the object

Property **Links** of C_DRV_1D with additional indication:

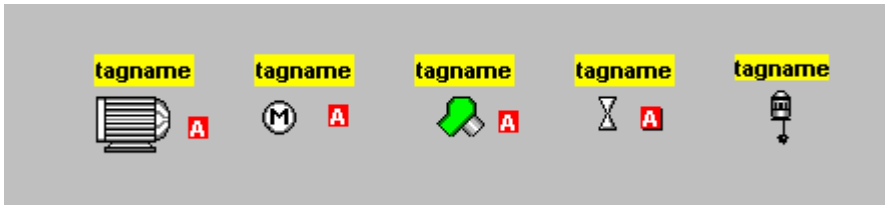


Property Styles

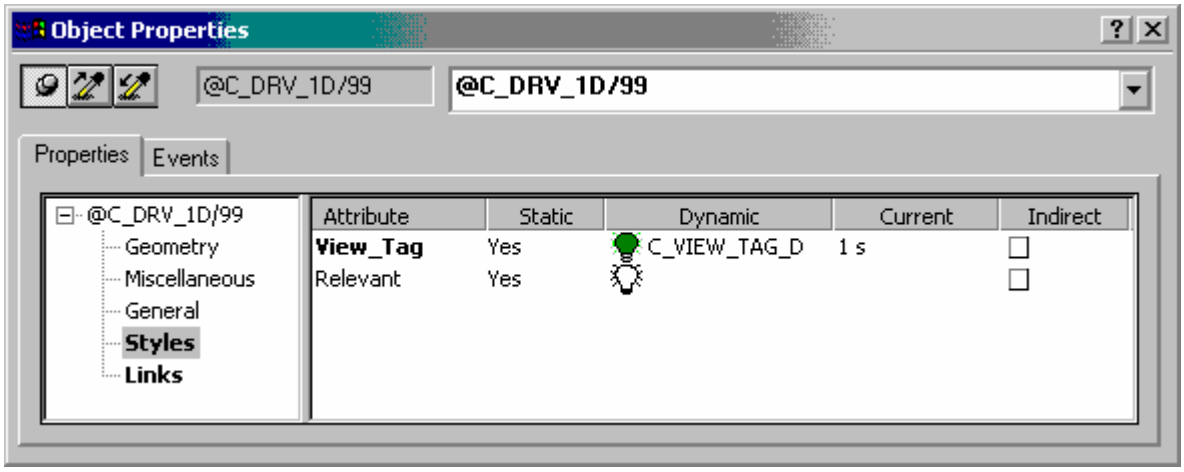
In Property **Styles** the appearance of the symbol or the faceplate can be modified.

It is possible to show the tagname and/or the Group Display together with the symbol. The template pictures contain only an example for this. If this tagname and/or group display is required, the other symbols have to be modified in the same way.

Examples:



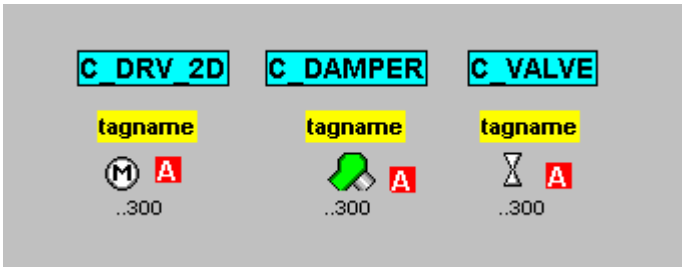
The display of the tagname can be switched on/off in the Runtime system, using the internal variable C_VIEW_TAG_D. To enable this, the attribute **View_Tag** of Property **Styles** has to be linked to the variable C_VIEW_TAG_D:



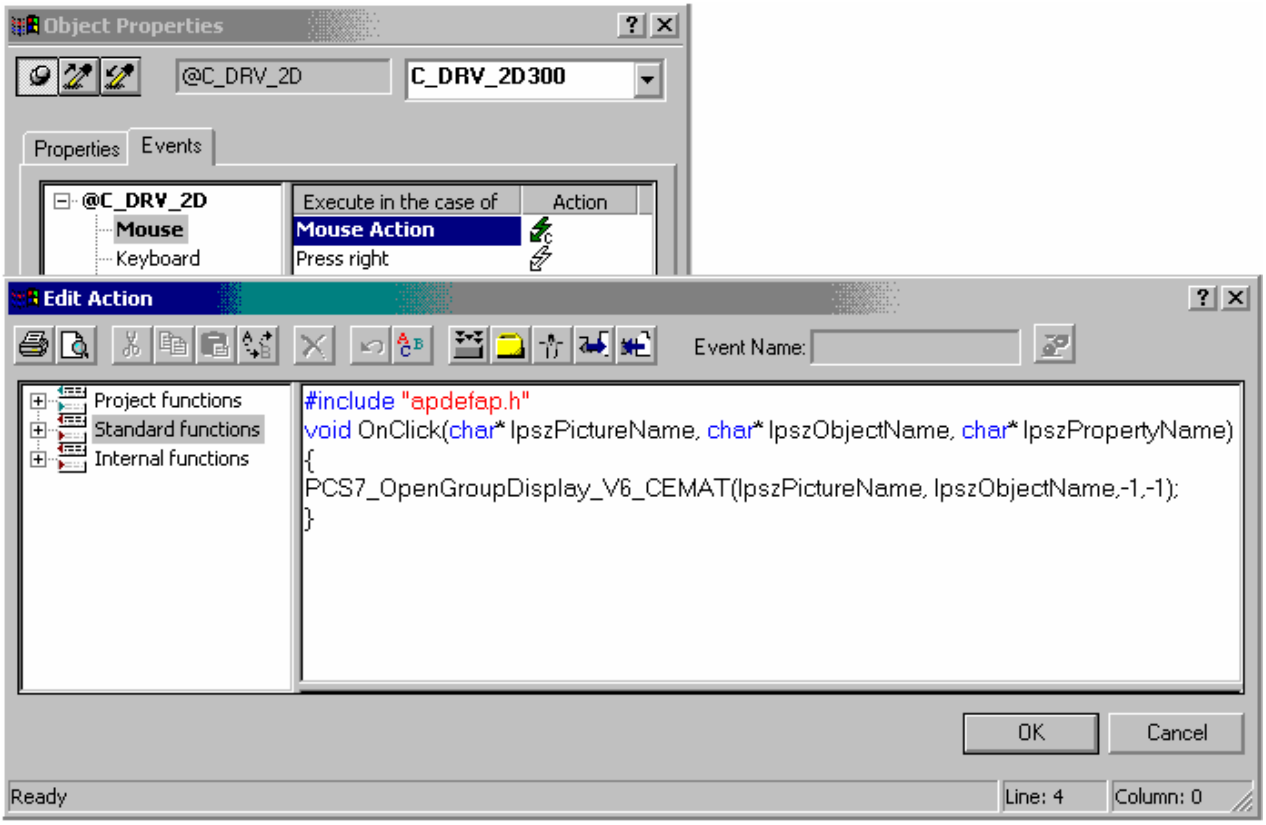
Additional Function “Button Text” for Bi-directional Drive, Damper and Valve

Via two additional IO-Fields **BText1** and **BText2** in the Customized Objects for C_DRV_2D, C_DAMPER and C_VALVE, the Button text of the faceplate can be modified.

An example for this can be found in the following symbols of the template pictures:

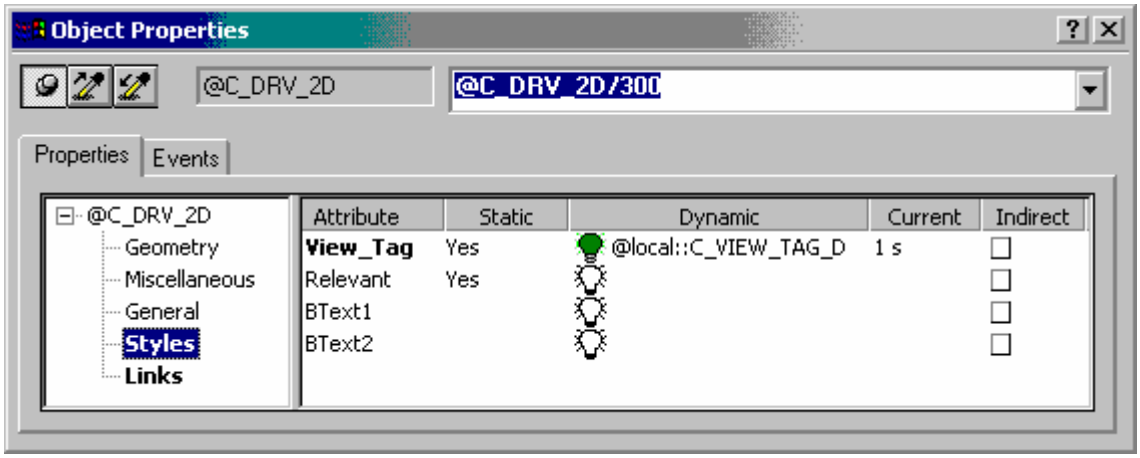


For the additional function Button text it is required to open the faceplate via function “PCS7_OpenGroupDisplay_V6_CEMAT”:



The definition of the button text is carried out vie Attributes **BText1** and **BText2** of Property **Styles**.

Property **Styles** for the Symbol @DRV_2D/300:

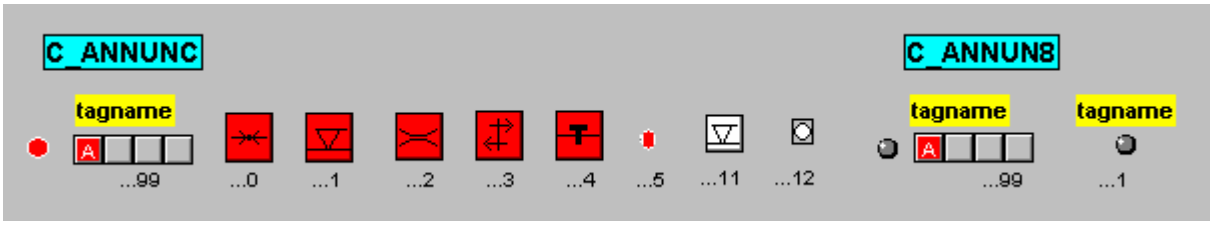


BText1 corresponds to the left button in the Faceplate (Dir.1)
BText2 corresponds to the right button in the Faceplate (Dir.2)

If BText1 and BText2 are empty, the default button text remains. If a text is entered under BText1 and BText2, the default button text will be replaced by this text.

C_ANNUNC and C_ANNUN8

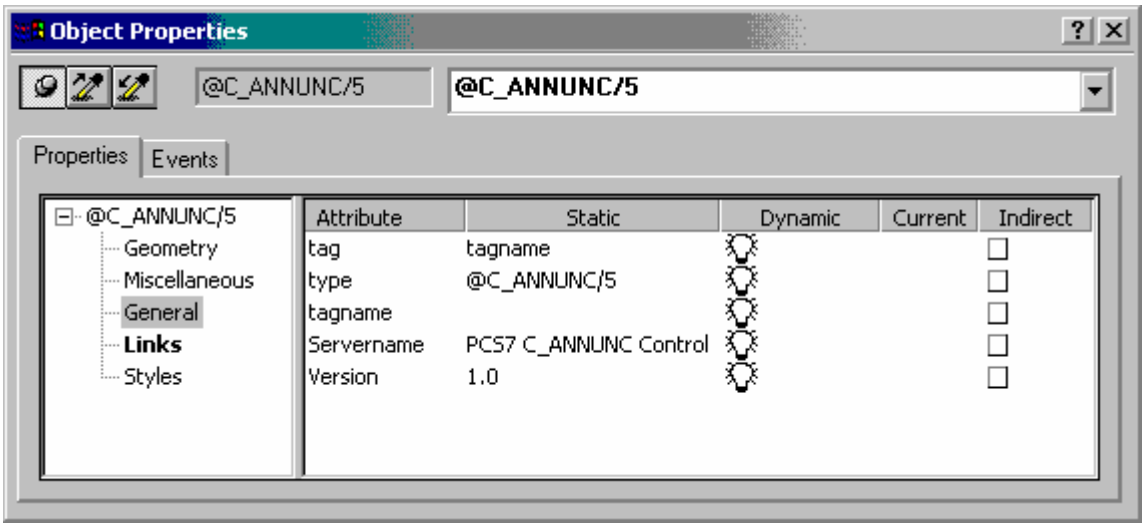
In the CEMAT template pictures, for most of the block icons of C_ANNUNC and C_ANNUN8 only the symbol itself (or the symbol and the tagname) is visible. All further objects belonging to the customized object are hidden behind this symbol.



Each symbol has a unique Object Name, which consists of the object type (FB Name) and an index, e. g. @C_ANNUNC/1

Property General

Property **General** of object type C_ANNUNC:



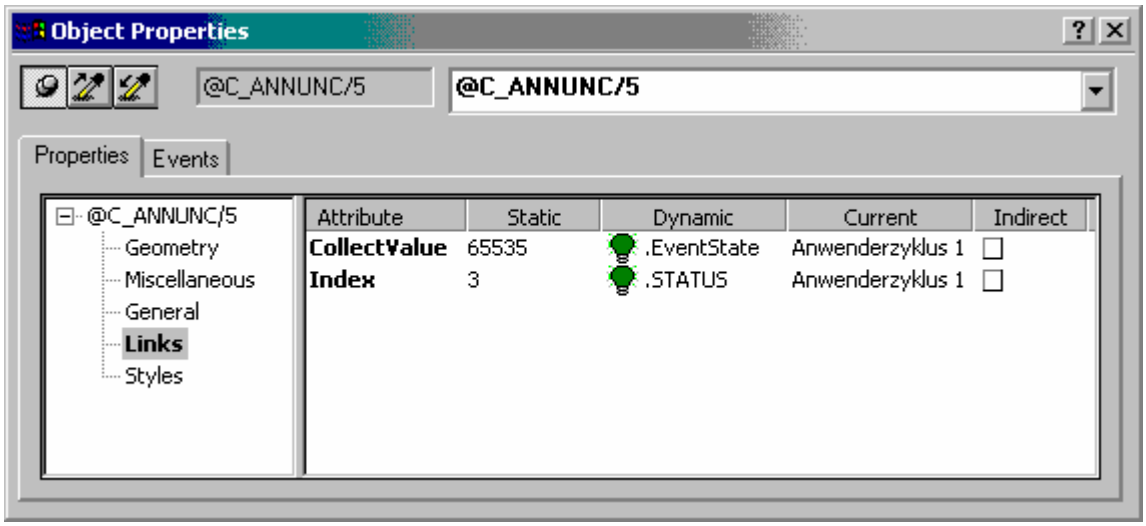
Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by C_ANNUNC.

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):

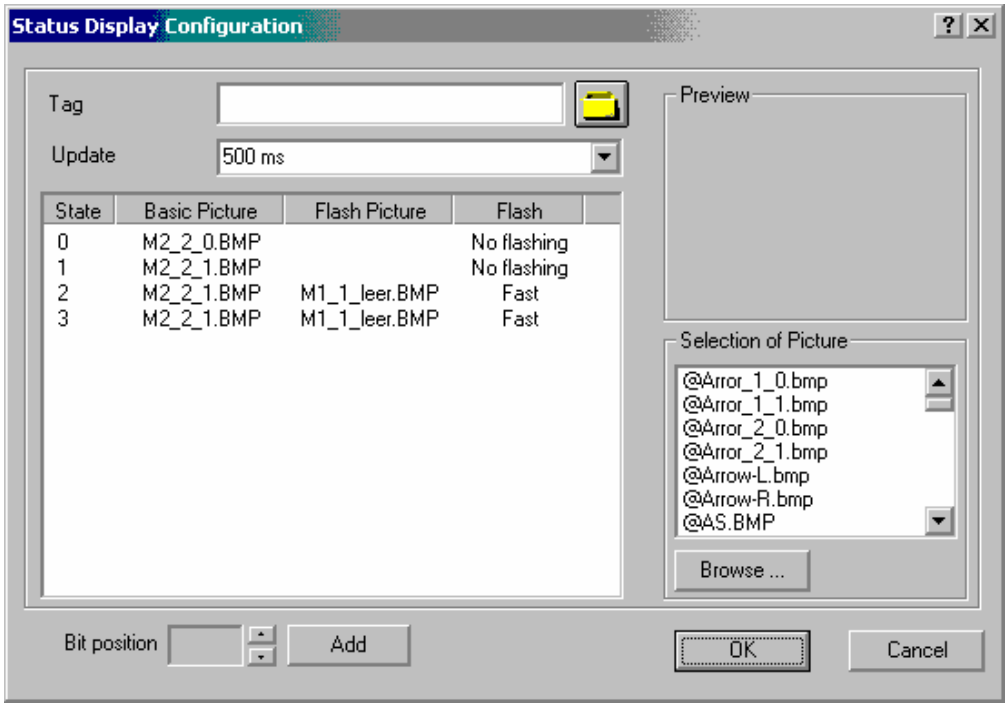
CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
Index	to	.STATUS	to show the status of the object

Property **Links** of C_ANNUNC:



For C_ANNUNC the Status Display function is used to define the representation of the block icon the different states. For C_ANNUN8 the extended Status display is used. The possible states of each object are documented in the object description of the corresponding object type (Variable STATUS).

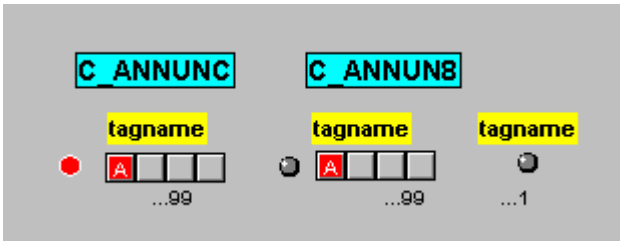
The following picture shows the states object type C_ANNUNC:



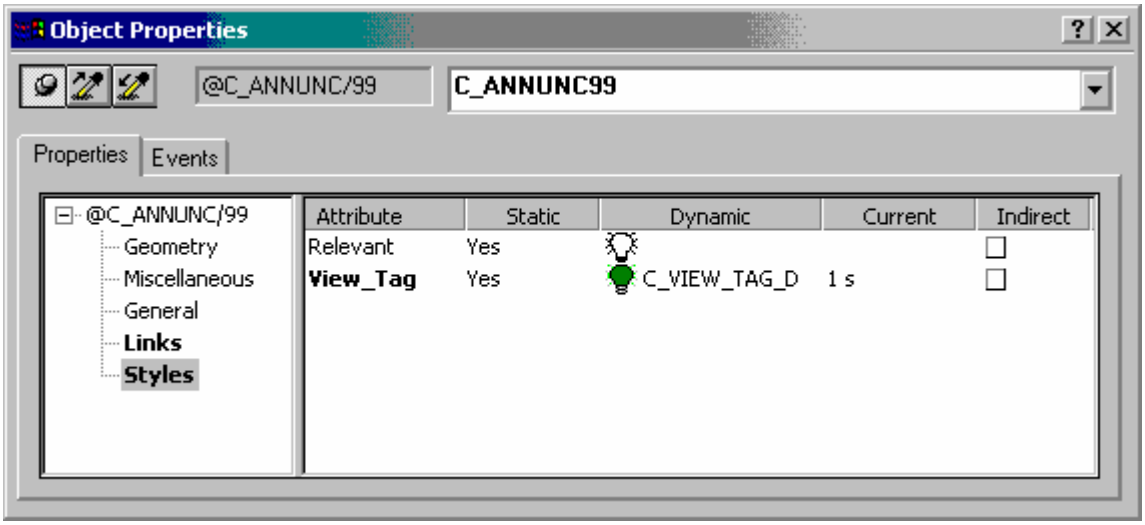
Property Styles

It is possible to show the tagname and/or the Group Display together with the symbol. The template pictures contain only some examples for this. If this tagname is required, the other symbols have to be modified in the same way.

Examples:



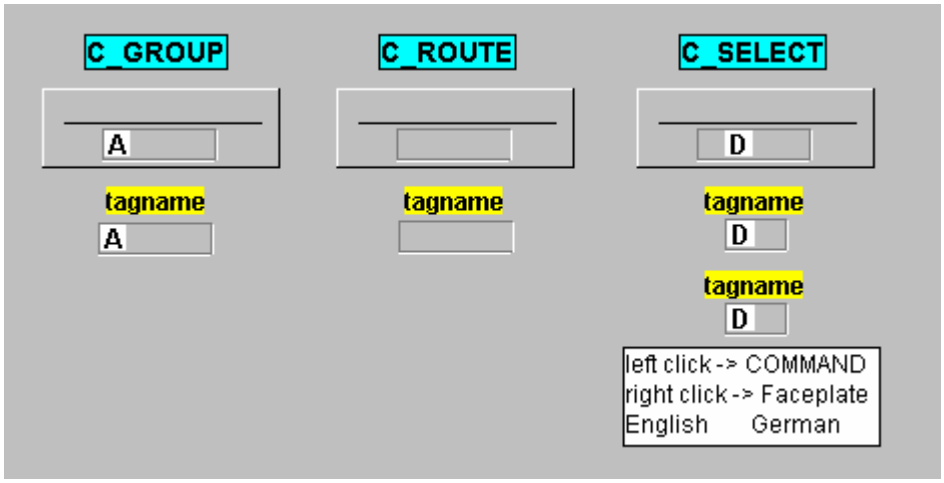
The display of the tagname can be switched on/off in the Runtime system, using the internal variable C_VIEW_TAG_D. To enable this, the attribute View_Tag of Property **Styles** has to be linked to the variable C_VIEW_TAG_D:



C_GROUP, C_ROUTE and C_SELECT

In the CEMAT template pictures, for most of the block icons of C_GROUP, C_ROUTE, and C_SELECT only the main symbol is visible. All further objects belonging to the customized object are hidden behind this symbol.

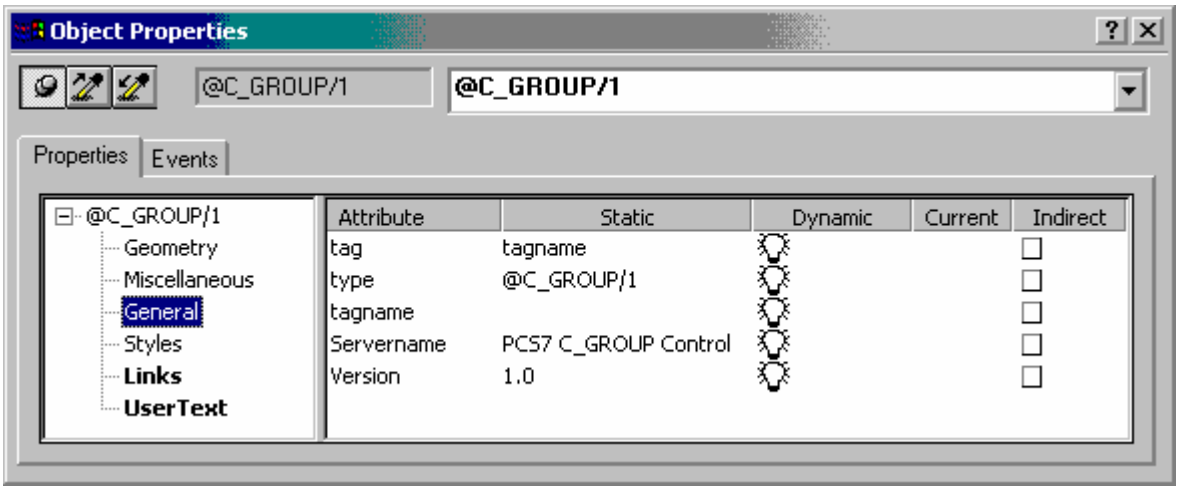
The following symbols for group, route and selection are available in the template pictures:



Each symbol has a unique Object Name, which consists of the object type (FB Name) and an index, e. g. @C_GROUP/1.

Property General

The following picture shows the **General** properties of object type C_GROUP:

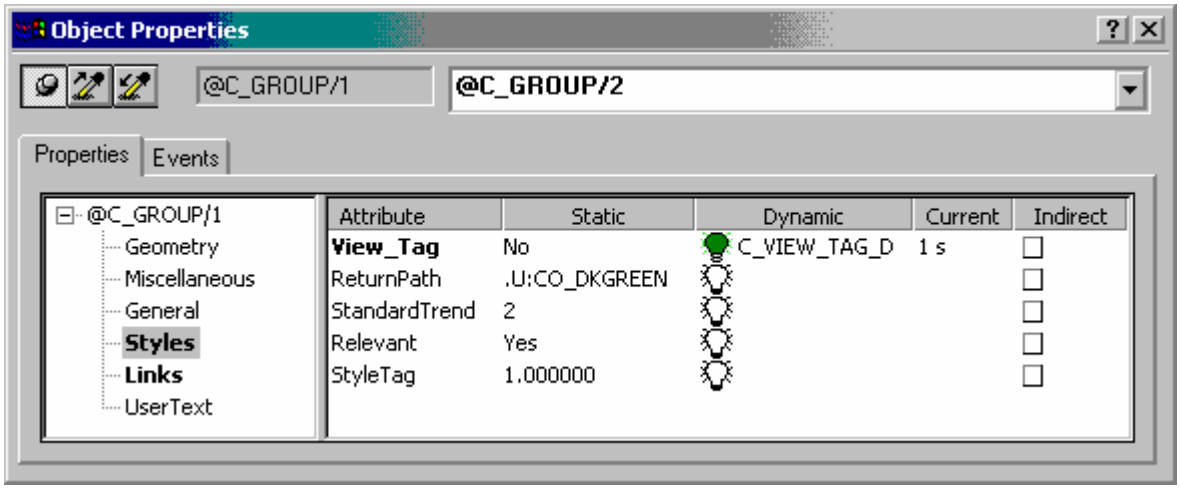


Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by C_GROUP.

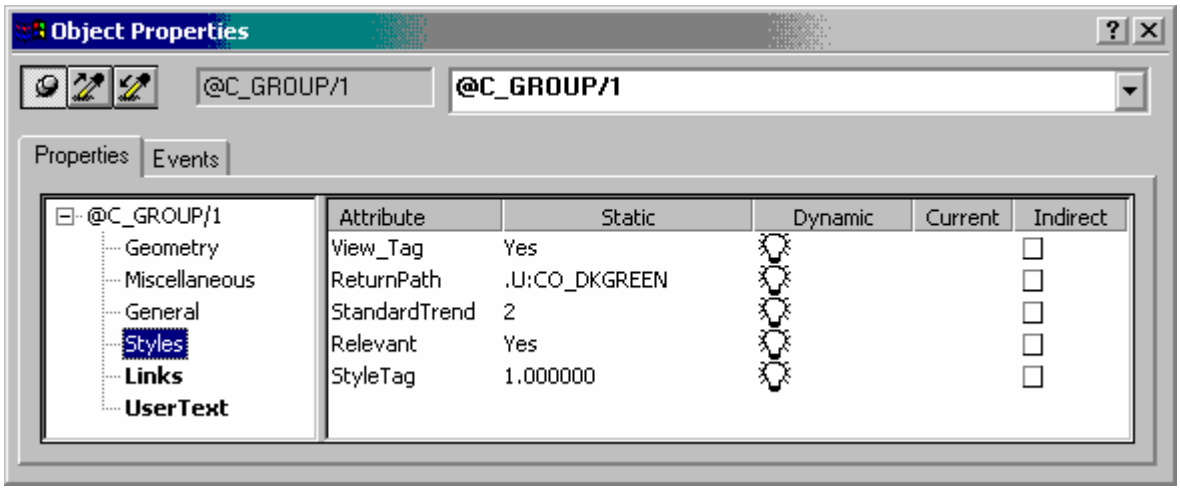
Property Style

In Property **Styles** the appearance of the symbol or the faceplate can be modified.

Attribute **View_Tag** is used if the tagname is displayed additionally to the symbol (only for the small symbols). Attribute View_Tag has to be connected to internal variable C_VIEW_TAG_D in order to switch the display of tagname on and off. See description of C_DRV_1D.



Attribute **StyleTag** is used the select the appearance of the faceplate. (to display or not to display the buttons).



Possible Styles of the group module:

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

Possible Styles of the route module:

0 – Select_Deselect (default)
1 – Buttons_Invisible

Possible Styles of the selection module:

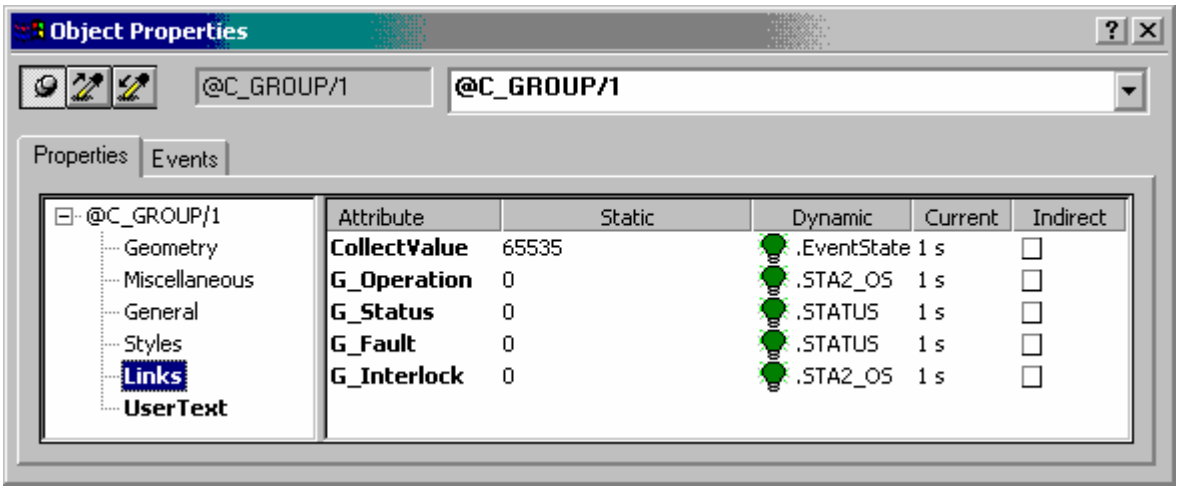
0 – Select_Deselect (default)
1 – Buttons_Invisible

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):

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
G_Operation	to	.STA2_OS	to show the Operation status (Auto/Single/Local)
G_Status	to	.STATUS	to show the Group Status (Stopped, running, ...)
G_Fault	to	.STATUS	to show the Fault Status
G_Interlock	to	.STA2_OS	to show the Interlock Status

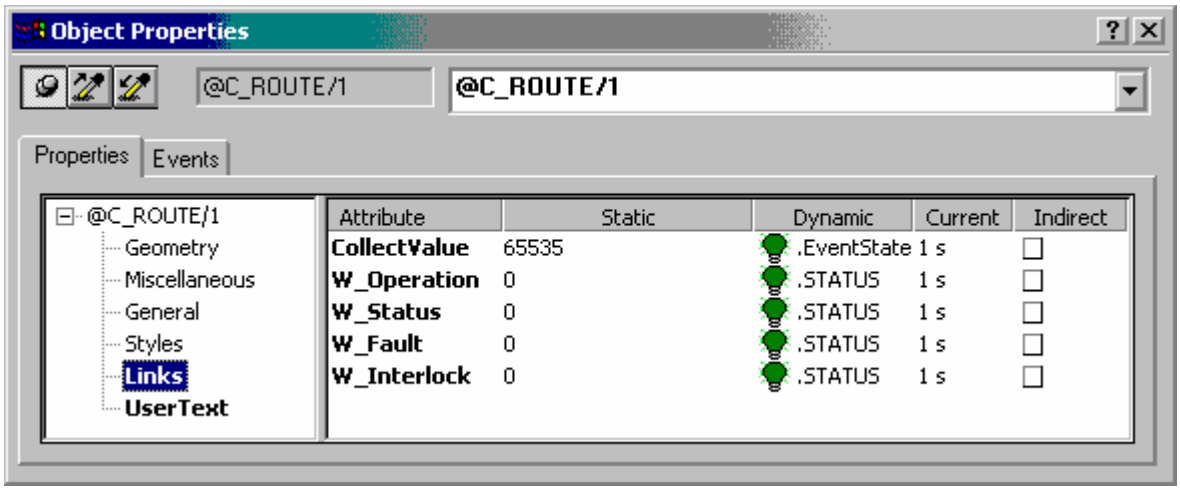
Property **Links** of C_GROUP:



For each route 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):

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
W_Operation	to	.STATUS	to show the “locked” status
W_Status	to	.STATUS	to show the route Status (selected, operating, ...)
W_Fault	to	.STATUS	to show the Fault Status
W_Interlock	to	.STATUS	to show the Interlock Status

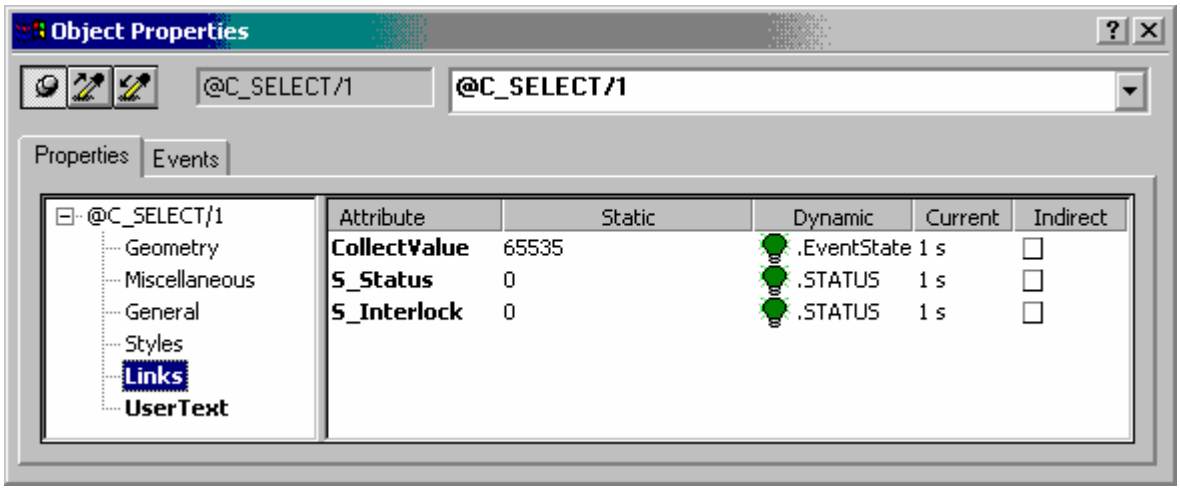
Property **Links** of C_ROUTE:



For each selection module 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):

CollectValue	to	.EventState	for Group Display in the overview Range (the group display is normally not shown at the object itself)
S_Status	to	.STATUS	to show the Selection Status
S_Interlock	to	.STATUS	to show the Interlock Status

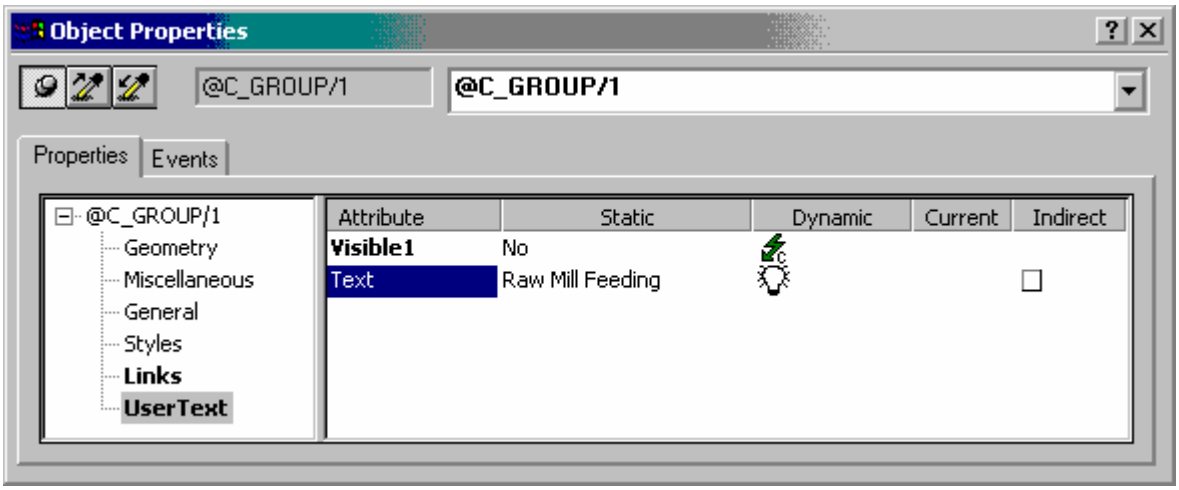
Property **Links** of C_SELECT:



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.

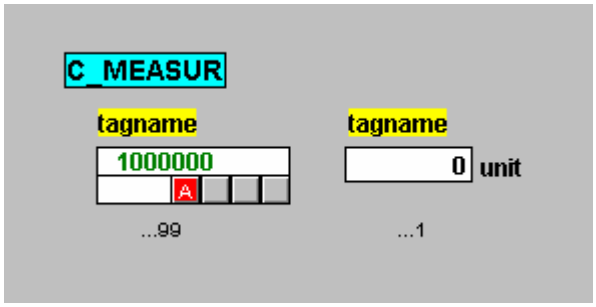
The text which is entered into Attribute **Text** of the **User Text** Property will be displayed in the symbol instead of the Tagname. If the field is empty the object tagname is displayed in Runtime mode.



C_MEASUR

In the CEMAT template pictures, for most of the block icons of C_MEASUR only the measure itself is visible. All further objects belonging to the customized object are hidden behind this symbol.

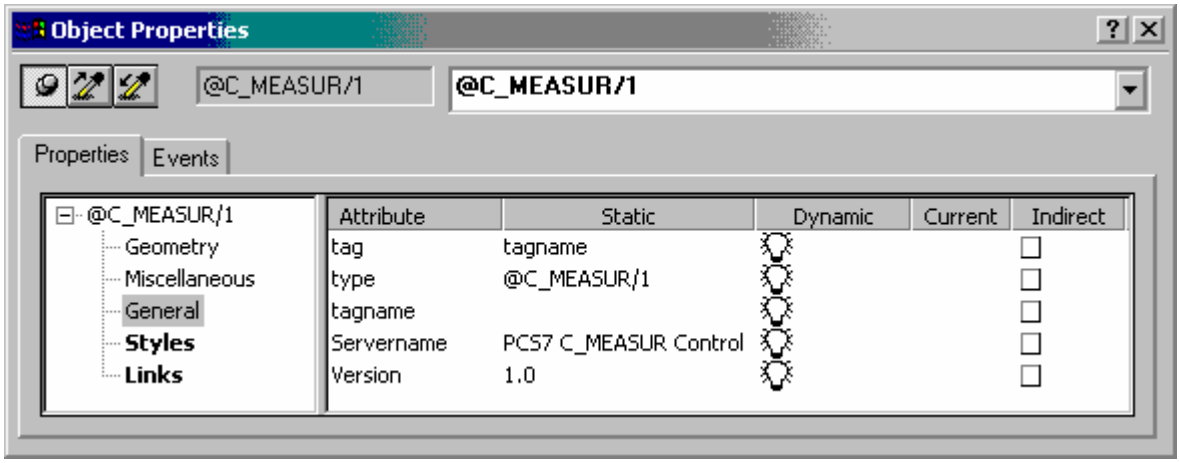
The following symbols for measures are available in the template pictures:



Each symbol has a unique Object Name, which consists of the object type (FB Name) and an index, e. g. @C_MEASUR/1.

Property General

Property **General** of object type C_MEASUR:



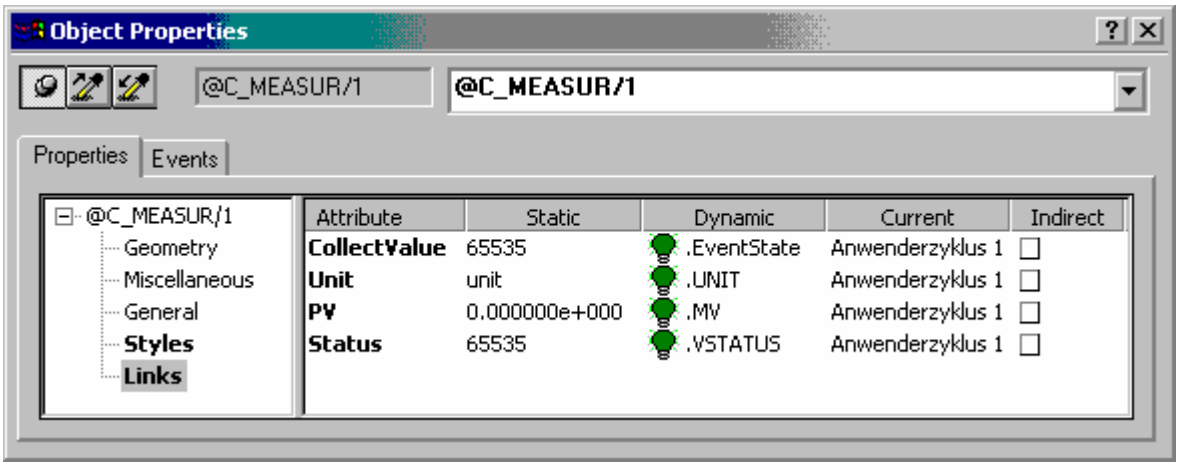
Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by C_MEASUR.

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):

CollectValue	to	.EventState	for Group Display in the overview Range
Unit	to	.UNIT	to show the Unit
PV	to	.MV	to show the process value
Status	to	.VSTATUS	to show the status of the object

Property **Links** of C_MEASUR:

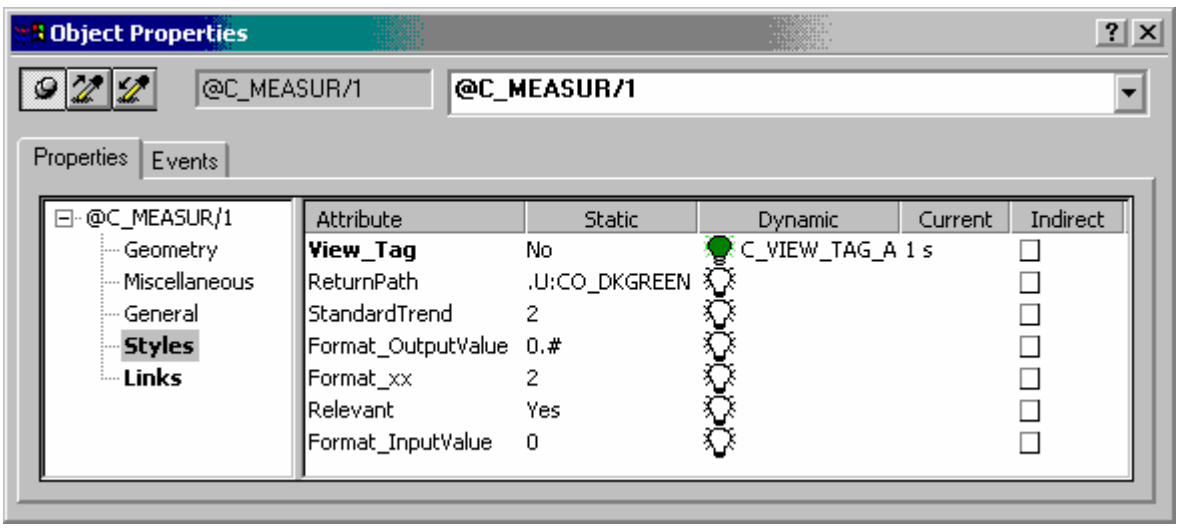


Property Styles

The display of the tagname can be switched on/off in the Runtime system, using the internal variable C_VIEW_TAG_A. To enable this, the attribute **View_Tag** of Property **Styles** has to be linked to the variable C_VIEW_TAG_A:

Attribut **Format_InputValue** has to be adapted according to the number of digits behind the comma (default value is 0).

0.0 one fixed decimal place
0.00 two fixed decimal places
0.0## one fixed decimal place and 2 variable digits if the value is not zero
These settings are taken on also into the Faceplate of the Objects.
Actual value and limit values will show the decimal places from the Style.



Including a Measured Value into a Process Value Archive

The faceplate of the measuring value shows the trend curve. If an archive curve is available for the measure, via button “Curve” the Archive Curve can be called.

In the Engineering for the Process Value Archive, certain rules have to be followed, in order to enable the call-up of the Archive curve.

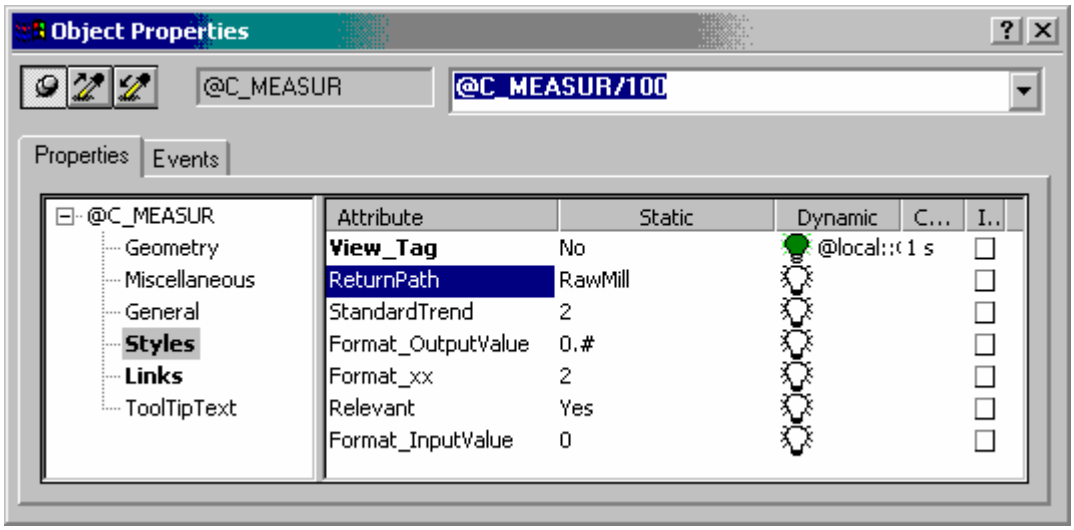
1. The archive tag name must be derived from the tag name of the measure. This implies the following:
   a) The object tag of the measure must not contain any special character, because the in the archive tag name, special characters are not allowed and will be removed by the system.
   b) The archive tag name, automatically given by the System must not be modified.
2. The name of the process value archive must be ProcessValueArchive. This means that all process values, which shall be shown in the Measuring Value Faceplate, are located in the same Archive.

   Restrictions for the Archive names up to CEMAT V6 SP2:
   The name of the process value archive must be **ProcessValueArchive**. This means that all process values, which shall be shown in the Measuring Value Faceplate, are located in the same Archive.

   Restrictions for the Archive names from CEMAT V6 SP3:
   From SP3 it is possible to use different process value archives with any name (without special characters) by entering an Archive name in the symbol properties of the measure symbol (under Styles, Attribute ‘ReturnPath’).

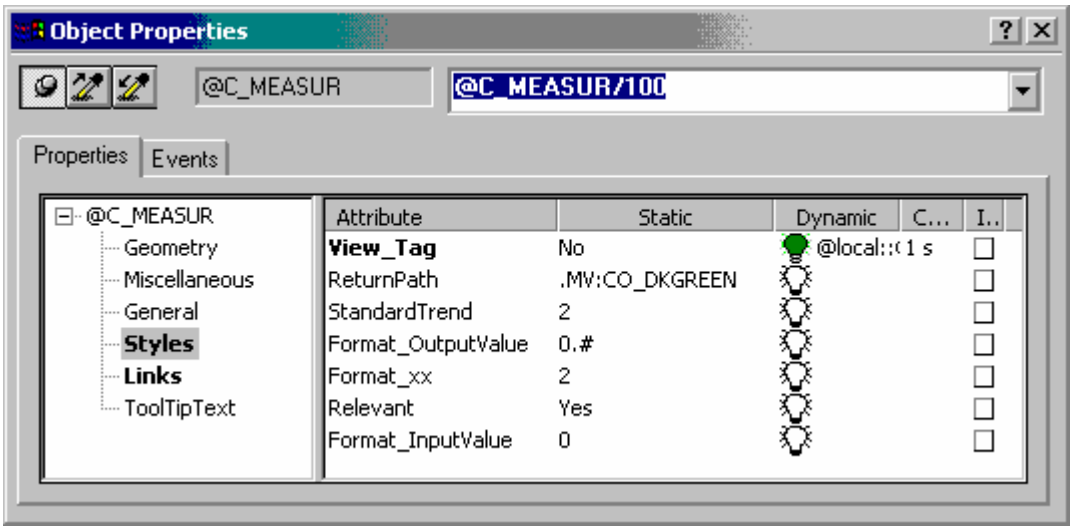
Exception: For motor currents which are displayed via „Analog“ button in the faceplate of the corresponding motor, the archive must be located in **ProcessValueArchive**.

If Attribute ‘Return Path’ contains an archive name, the archive variable will be searched in this Archive.



In this example the measuring values are stored in the Archive „RawMill“

- If the attribute "ReturnPath" contains the default value .MV:CO_DKGREEN, the archive variable will be searched in Archive **ProcessValueArchive**.



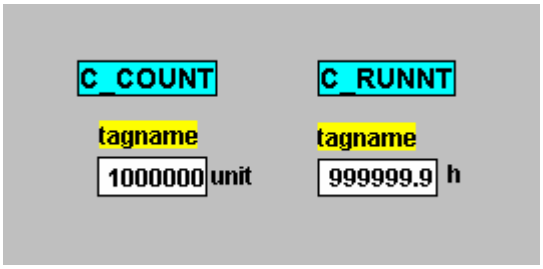
Default Settings: .MV:CO_:DKGREEN

3. The rest of the settings can be carried out individually.

C_COUNT and C_RUNNT

In the CEMAT template pictures, for most of the block icons of C_COUNT and C_RUNNT only the main symbol is visible. All further objects belonging to the customized object are hidden behind this symbol.

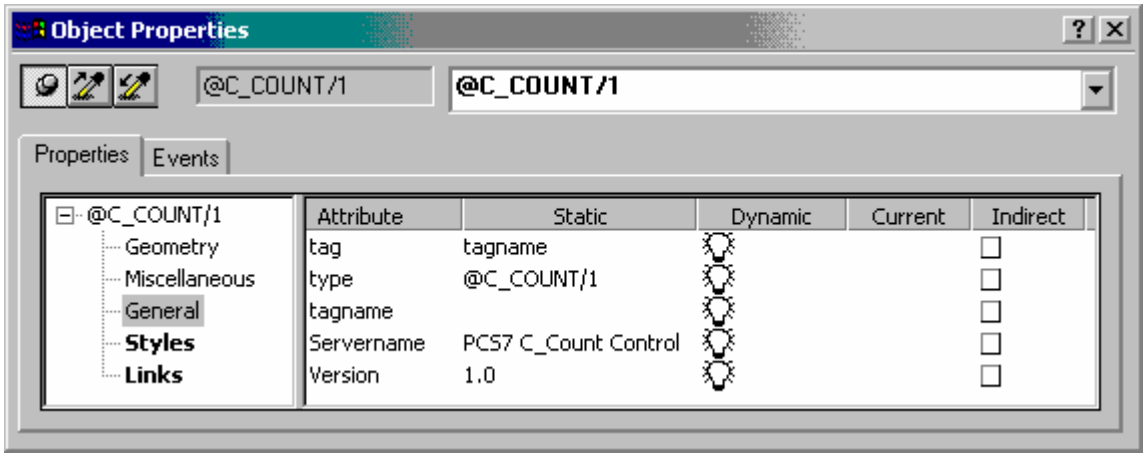
The following symbols are available in the template pictures:



Each symbol has a unique Object Name, which consists of the object type (FB Name) and an index, e. g. @C_COUNT/1.

Property General

Property **General** of object type C_COUNT:



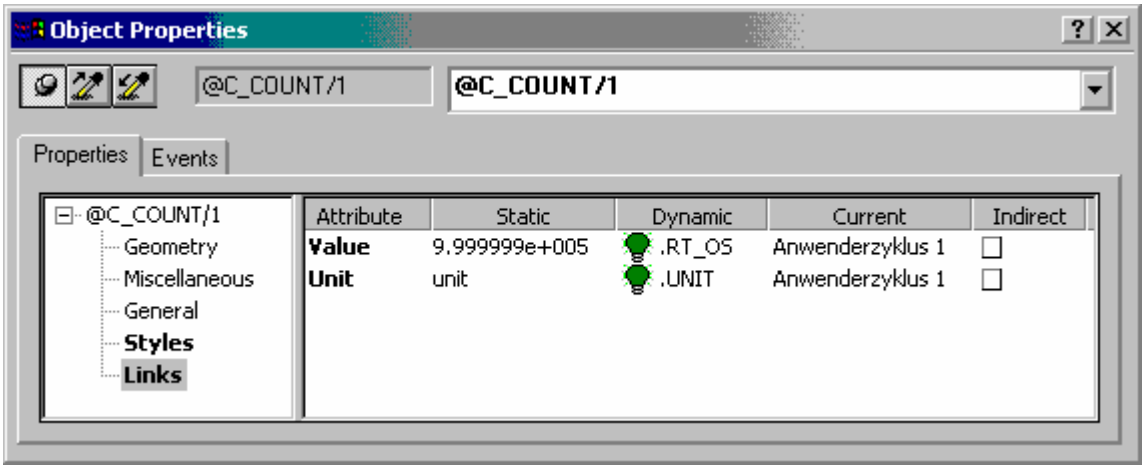
Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by C_COUNT.

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):

Value	to	.RT_OS	to show the counter value
Unit	to	.UNIT	to show the Unit

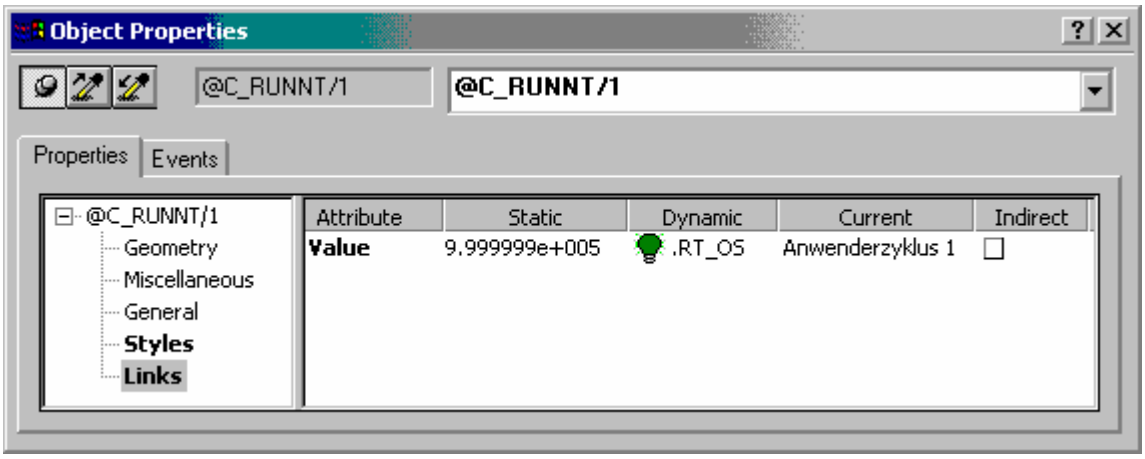
Property **Links** of C_COUNT:



For each Running Time in Property **Links** all Attributes must be connected as follows to the variables of the corresponding running time. (They will be connected automatically in case of automatic generation of Block Icons):

Value	to	.RT_OS	to show the running time value

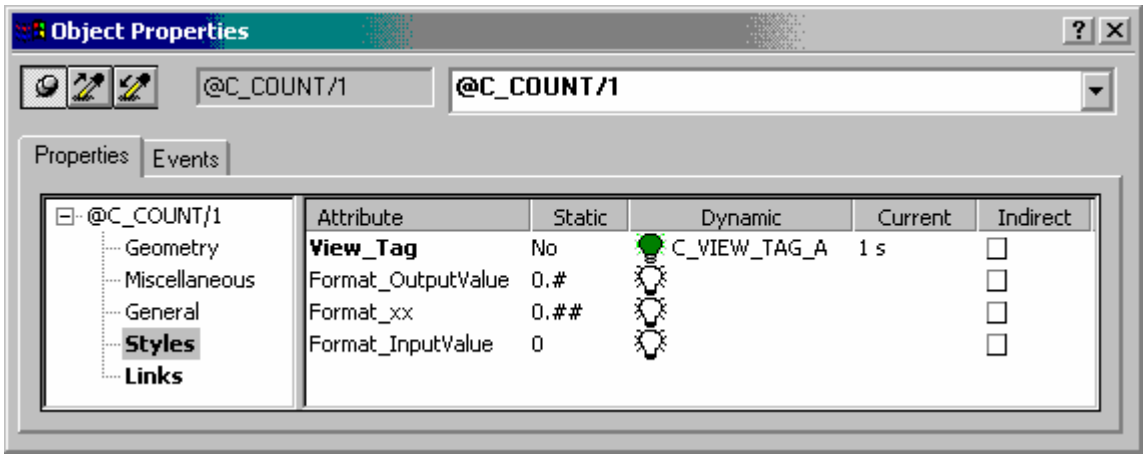
Property **Links** of C_RUNNT:



Property Styles

The display of the tagname can be switched on/off in the Runtime system, using the internal variable C_VIEW_TAG_A. To enable this, the attribute **View_Tag** of Property **Styles** has to be linked to the variable C_VIEW_TAG_A:

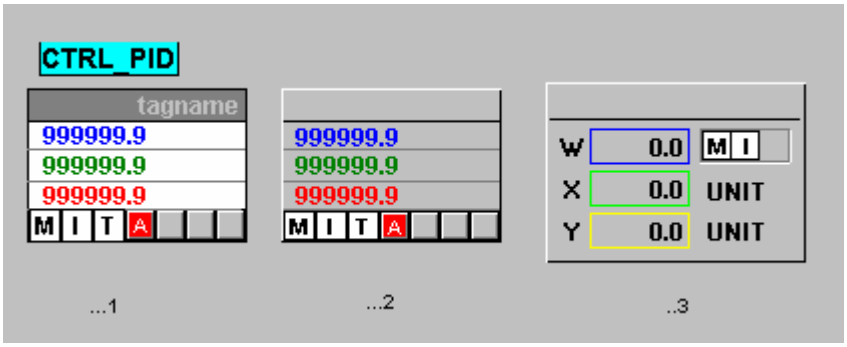
Attribut **Format_InputValue** has to be adapted according to the number of digits behind the comma (default value is 0).



CTRL_PID and CTRL_S

The controller functions CTRL_PID and CTRL_S are no CEMAT blocks and belong to the PCS7 Library V60. For further explanations see PCS7 Manuals.

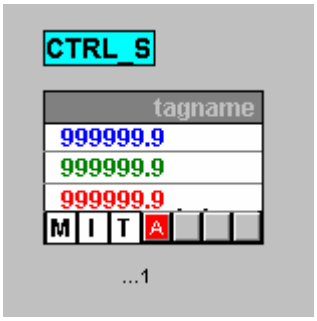
For the block CTRL_PID the following symbols are available in the CEMAT template pictures:



Symbol 1 and Symbol 2 were copied from the PCS7 Standard template pictures (with small modifications for colors in Symbol 2 . Click on the symbol opens the standard controller faceplate.

Symbol 3 was added for CEMAT. Click on the Symbol opens the CEMAT controller faceplate.

For the block CTRL_S the following symbol is available in the CEMAT template pictures:

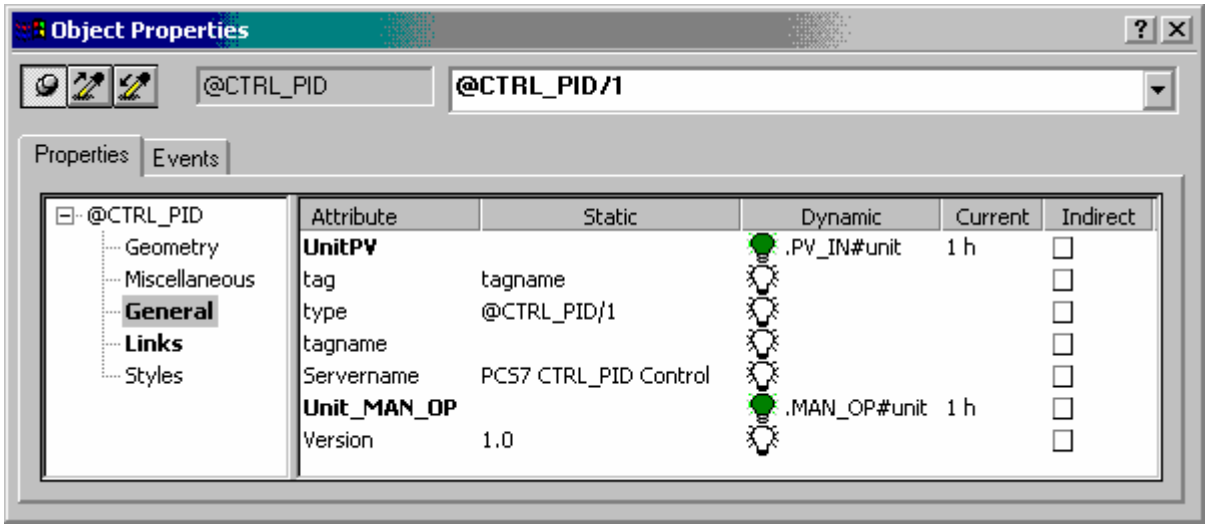


The Symbol was copied from the PCS7 Standard template pictures. Click on the symbol opens the standard controller faceplate.

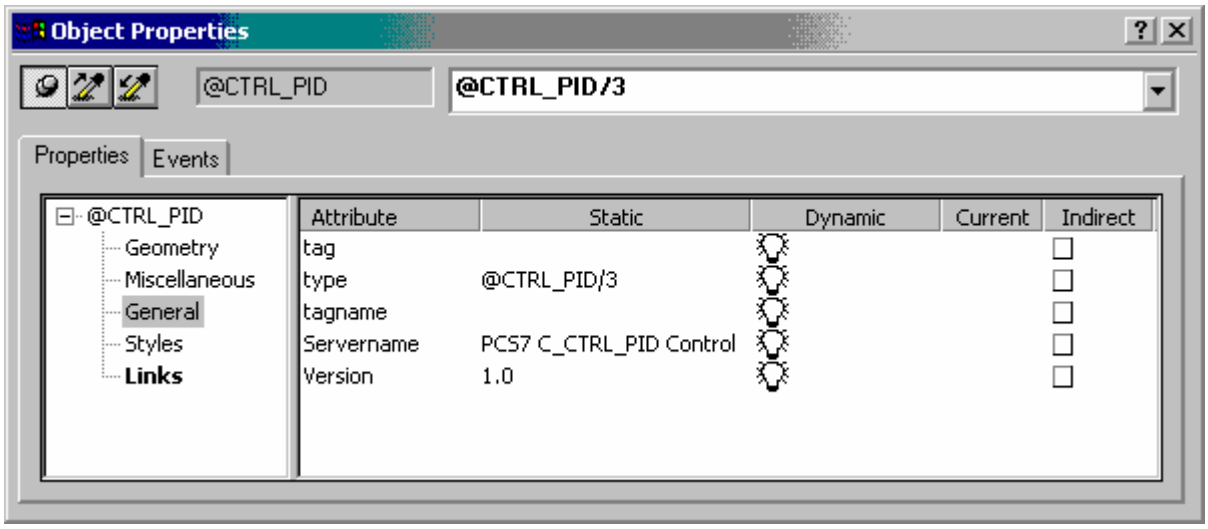
Each symbol has a unique Object Name, which consists of the object type (FB Name) and an index, e. g. @CTRL_PID/1.

Property General

Property **General** of object types CTRL_PID/1 und CTRL_PID/2:

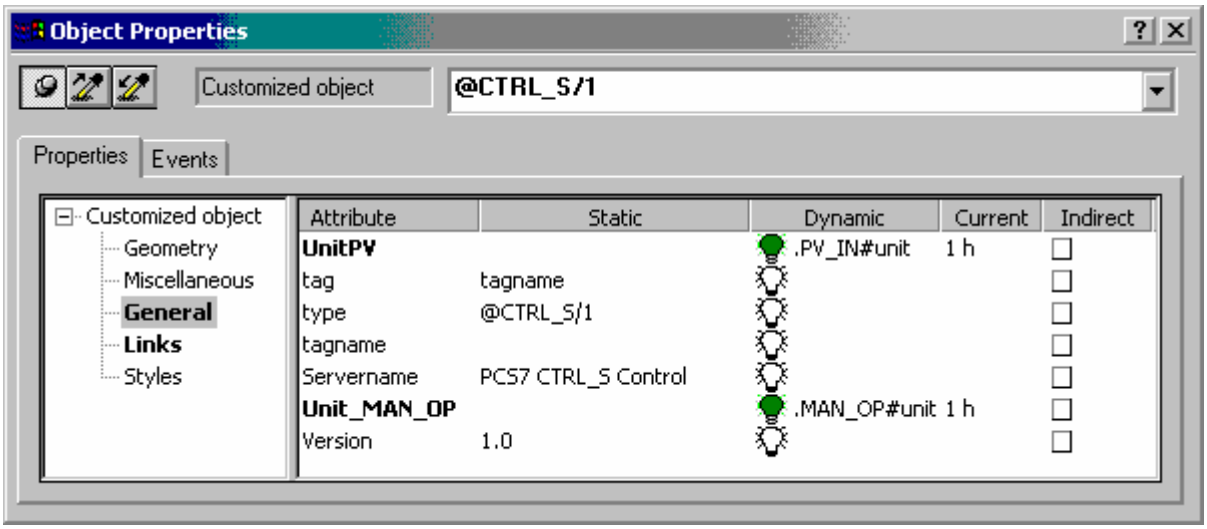


Property **General** of object type CTRL_PID/3



Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by CTRL_PID.

Property **General** of object type CTRL_S



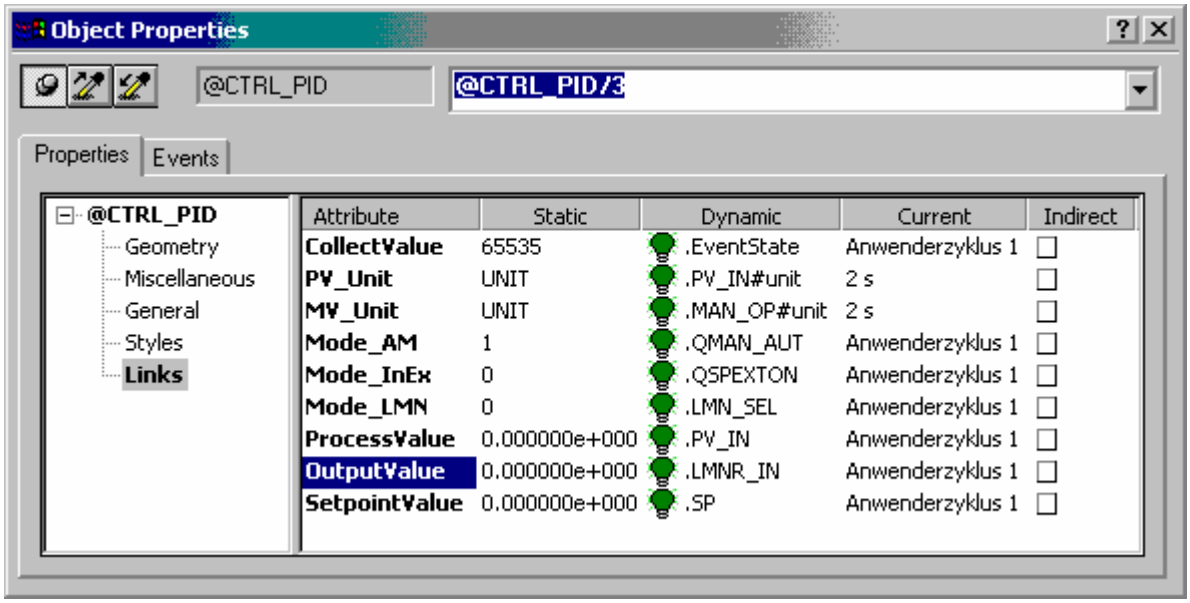
Note: The screenshot shows the properties in template picture @C_PCS7Typicals.pdl. In template picture @C_Template.pdl the attribute type has to be replaced by CTRL_S.

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):

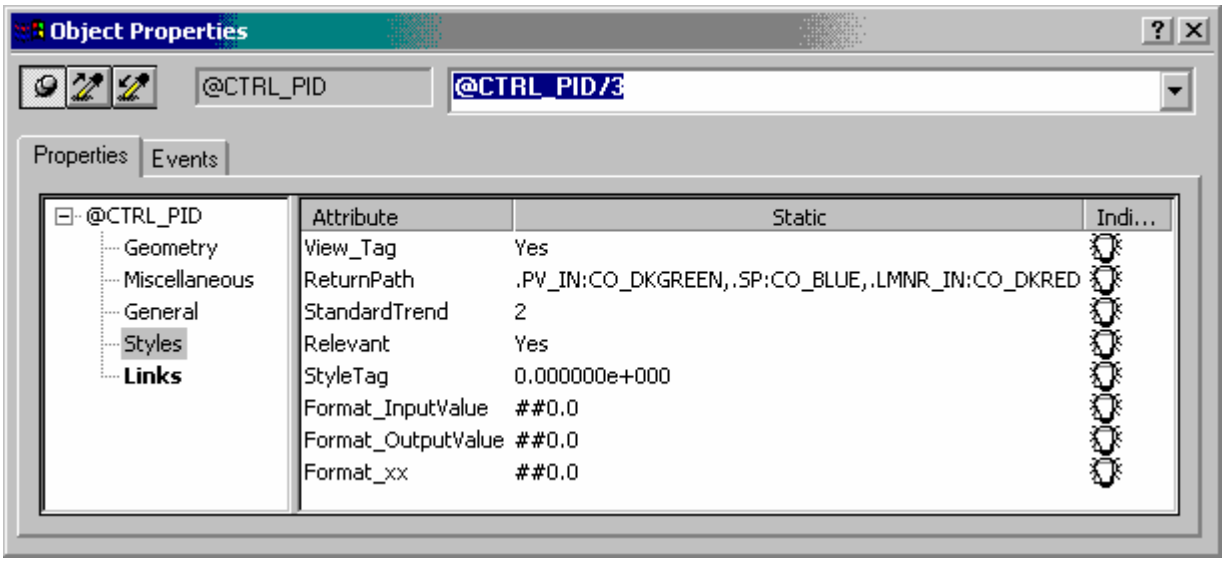
CollectValue	to	.EventState	to show the group display
PV_Unit	to	.PV_IN#unit	to show the Unit for the Actual value
MV_Unit	to	.MAN_OP#unit	to show the Unit for the Controller Output
Mode_AM	to	.QMAN_OUT	to show the Mode Automatik/Manuell
Mode_InEx	to	.QSPEXTON	to show the Mode Internal/External
Mode_LMN	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 Setpoint

Property **Links** for object type CTRL_PID:



Property Styles

Attribut **Format_InputValue, Format_OutputValue** and **Format_xx** have to be adapted according to the number of digits behind the comma (default value is one digit behind the comma).



Window Position of the Faceplates in V6.1

With the Template picture @C_PCS7Typicals_V61 of CEMAT V6.1 a new possibility for the positioning of the Faceplates exists.

The advantage of this function is that in case of a repetition of the generation of block icons the positions don't get overwritten.

The Position is defined in property **Miscellaneous**. There you find the attributes topPos, leftPos and defaultPos.

For the Positioning you have the following possibilities:

a) The Faceplate appears at the Curser-Position for

TopPos = 0
leftPos = 0
DefaultPos = no

b) The Faceplate appears at the set Position for

TopPos = Wert > 0
leftPos = Wert > 0
DefaultPos = no

c) The Faceplate appears at the Default Position

TopPos = irrelevant
leftPos = irrelevant
DefaultPos = yes

Window Position of the Faceplates (CEMAT V6.0 Method)

The new positioning function requires the new block icons (the block icons of CEMAT V6.0 did not have the attributes topPos, leftPos and defaultPos).

In case of an existing plant you may not want to generate the block icons again, using the new templates. In this case the CEMAT V6.0 Method for the positioning or the Faceplates can be used.

Also in CEMAT V6.0 from SP2 there was the possibility to define the window position for the Faceplates. To enable this, for opening faceplate the function „PCS7_OpenGroupDisplay_V6_CEMAT" must be used.

The window position is entered via parameter iX and iY. The following parameterisations are possible:

-1, -1	Faceplate will not be positioned (default) with this setting the attributes topPos, leftPos und defaultPos are evaluated!
<-1, <-1	(wrong entry) Faceplate will be positioned at 0,0
0, 0	Faceplate will be displayed at the upper left corner of the actual mouse position
>0, >0	If x or x are bigger than 0, the faceplate will be displayed at this coordinates.

Example:

```
PCS7_OpenGroupDisplay_V6_CEMAT(lpszPictureName, lpszObjectName, -1,-1);
```

The symbols delivered with the template pictures of CEMAT V6.0 SP2 already contain the new faceplate calls.

The new function PCS7_OpenGroupDisplay_V6_CEMAT replaces the function PCS7_OpenGroupDisplay_V6 as well as function PCS7_OpenGroupDisplay_V6_Ew.

Bitmaps

The bitmaps for the symbols are stored on all clients and servers in the standard directory (D:\CEM_V6\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.

The bitmaps for the Symbols (Drives, Valves) from Version 4.12 of CEMAT can be used. You find further models on the CD under CD:\Additional_Information\CEM_DRAFTS.

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 is not existing.

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.

Extraction 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	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
	auto/man. Interl./man. non Interl.	05: Process controlling	Processcontrolling_backup	006

The first four operation functions in the list are possible for all project versions:

1. The Operations **Start** and **Stop** 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 **Rest 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.

The following Operations are only available in project version 006:

5. The Operation **auto/manual Interlocked/manual non Interlocked** is assigned to authorization 5 „Process controlling“. The assignment is carries out via variable Processcontrolling_backup.

Please keep in mind that the operations **Start** and **Stop** and the Operations for **auto/manual Interlocked/manual non Interlocked** are always linked to the same right.

Message archive

The message system under PCS 7 is generated by the OS Project Editor. After running the OS Project Editor a few adaptations have to be performed (see Chapter „PCS7 Project“).

Tag logging

For each measure which needs to be archived, in the process value archive an archive tag has to be created.

How to create a process value archive and the archive tags is described in the engineering manual of PCS7 and not part of this description.

Follow the CEMAT rules:

- The Archive name for all measures, as per default, must be **ProcessValueArchive**. From CEMAT V6 SP3 the possibility exists to create multiple Archives with different names. This needs additional Engineering effort. See description under C_MEASUR in this manual.
- To enable the direct call of measured value archive from the faceplate of the measure, it is necessary that the name of the archive tag exactly corresponds to the name of the process tag.

Exception: For motor currents which are displayed via „Analog“ button in the faceplate of the corresponding motor, the archive must be located in **ProcessValueArchive**.

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

For each measure you want to archive, you have to create an archive variable. Select the proper archiving cycle according to the function of the measure:

- quick archiving for currents, pressures etc.
- slow archiving for temperatures etc.

User Archives

During the installation of the Project (chapter 3) two user archives have been created. Through the import of the files DriveList.uap and C_INFO.uap the data base structure was already created in the correct format.

The group instance list **C_DriveList** does not need any further engineering. The data will be provided online.

The User Archive **C_INFO** is empty and must be filled with data, in order to provide it for the Info Dialog of the Objects.

We recommend to create the Info-Data with Excel (CSV-File) and to import it afterwards into the user archive. An example for the CSV-File you find in directory D:\Cem_V6\ua. File C_INFO.csv contains one line with example data (only headers).

Structure of the Info data base

The structure of the Info data base is already explained in chapter 2. Subsequently you find some additional remarks regarding the particular data fields.:

- The content of fields ID, Obj_ID and C_TAG must be unique. ID and Obj_ID contain a consecutive number and C_TAG contains the tagname of the object.
- The rest of the fields are grouped by subject under which you will find the information later on in the Information Dialog. You can identify the subject on it's prefix.

  GL_ = General Locations
  G_ = Register Service
  I_ = Register Input/Output
  M_ = Register MCC Data
  H_ = AS Hardware / Software
- The fields LastUser and LastAccess can remain empty. They are used by the system itself.
- The most important entry in the Info Data base is the Picture name (for the Loop-in-Alarm-Function). All further entries are optional.
- By entering the file name for manuals and video files, these documents can be opened from the Info-Dialog by additional buttons.

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.

Note for the Generation of the Info data:

With CEMAT V6.1 an Excel Macro is available, which can be used to create the Info Data from the PCS7 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 archives C_INFO

This chapter describes the generation of the user archives C_INFO.

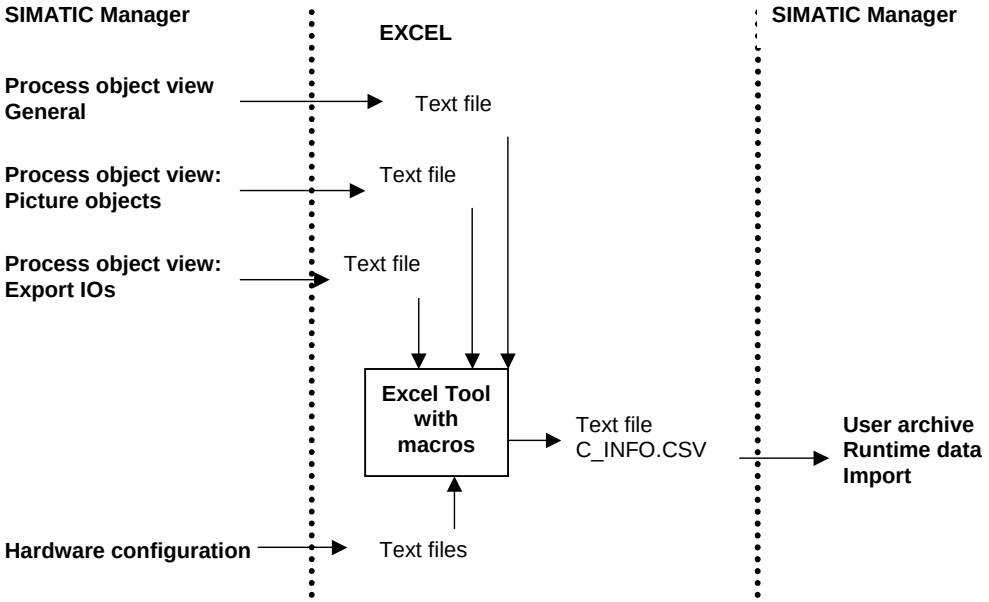
For the generation of the User Archive C_INFO from the process object view and the hardware configuration of the PLCs the following requirements must be met:

- The hardware configuration of each PLC must be exported as CFG file.
- The content of the process objects views „General“ and „Picture objects“ must be copied into text files.
- The I/Os of the process objects view must be exported into a CSV file.

The following columns are filled during this procedure:

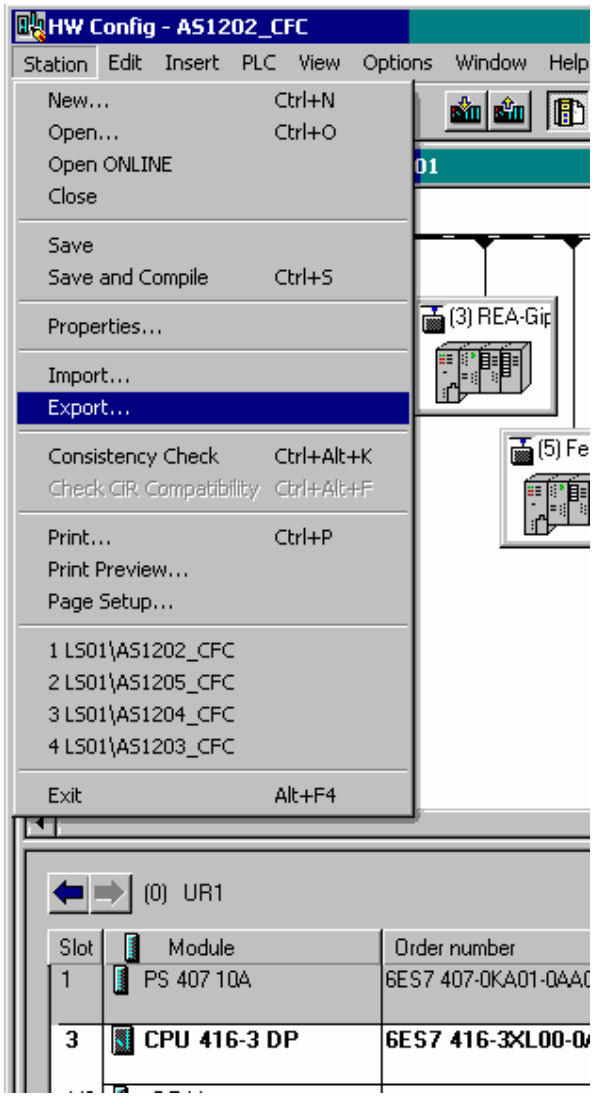
- ID -> will be numbered continuously.
- C_TAG = TAG
- GL_Complex = Complex = Manually input = identically for all objects.
- GL_Plant = Project = Manually input = identically for all objects.
- GL_Plant_zone = Plant zone = 1. Hierarchy folder.
- G_LoopA = Process mimic = from process object view „picture objects“.
- I_Rack_Q = DP address and subsystem number for output signal = from process object view I/Os and hardware configuration.
- I_Slot_Q = Slot for output signal = from process object view I/Os and hardware configuration.
- I_Rack_I = DP address and subsystem number for input signal = from process object view I/Os and hardware configuration.
- I_Slot_I = Slot for input signal = from process object view I/Os and hardware configuration.
- H_PLC_Name = PLC = from process object view “General” or hardware configuration.

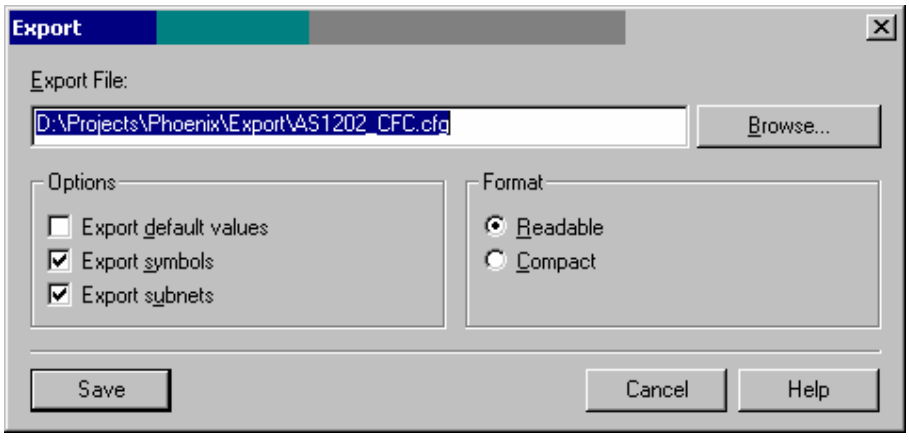
Overview



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“.





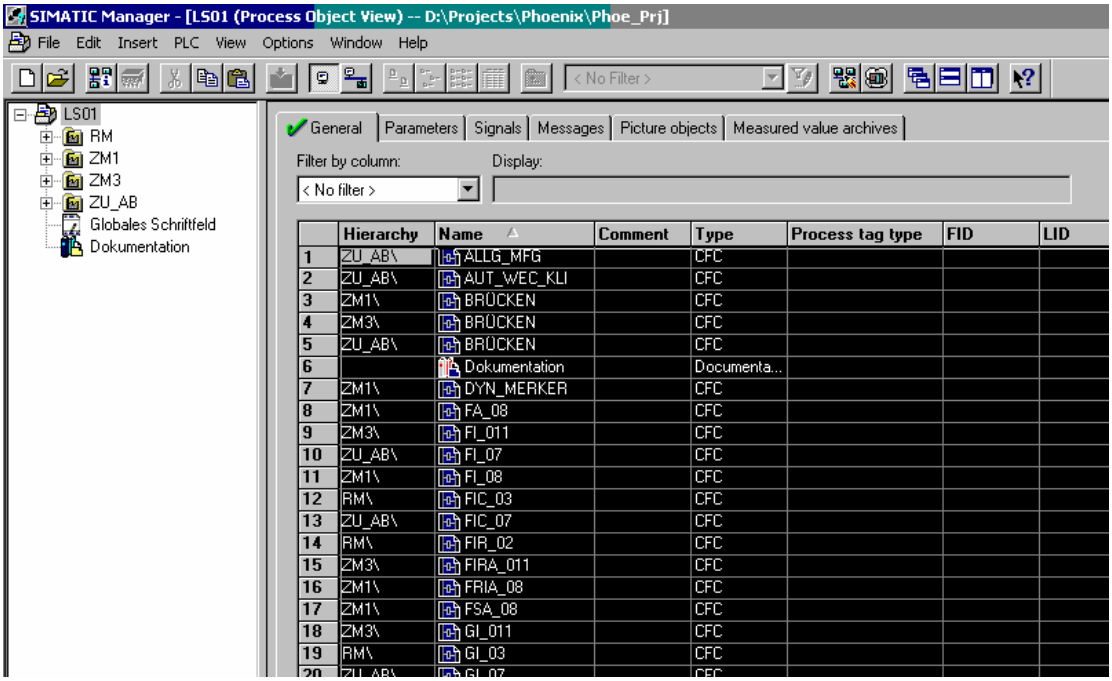
Repeat Step 1 to 3 for each PLC.

Export 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.

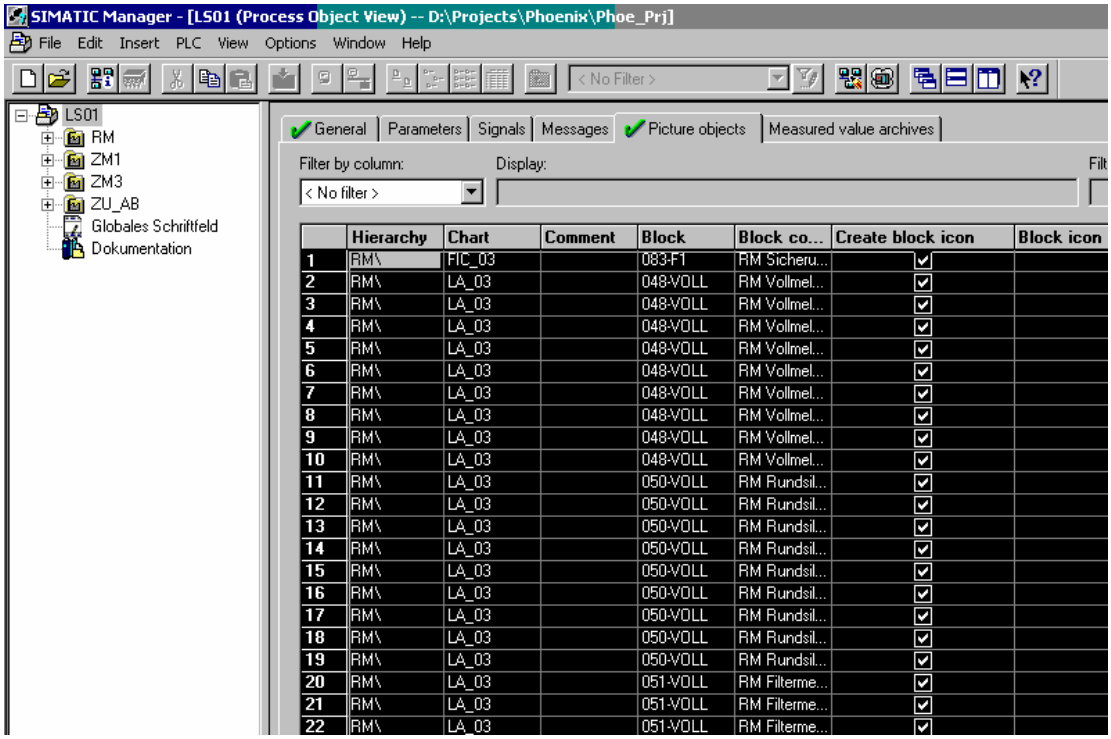
1. Select process object view.
2. Select the topmost hierarchy folder
   (from the project with the PLCs, not the multi project folder).
3. Refresh the view (F5).
4. Select in all lines all fields, except the line numbers.
5. Copy with CTRL-C all lines into the clipboard.
6. Start the text editor (notepad) and insert all lines with CTRL-V.
7. Save the text file, e.g. as GENERAL PO.txt.



Process object view „Picture objects“

Copy the process object view „picture objects“ as a text file, e.g. as PO picture.txt.

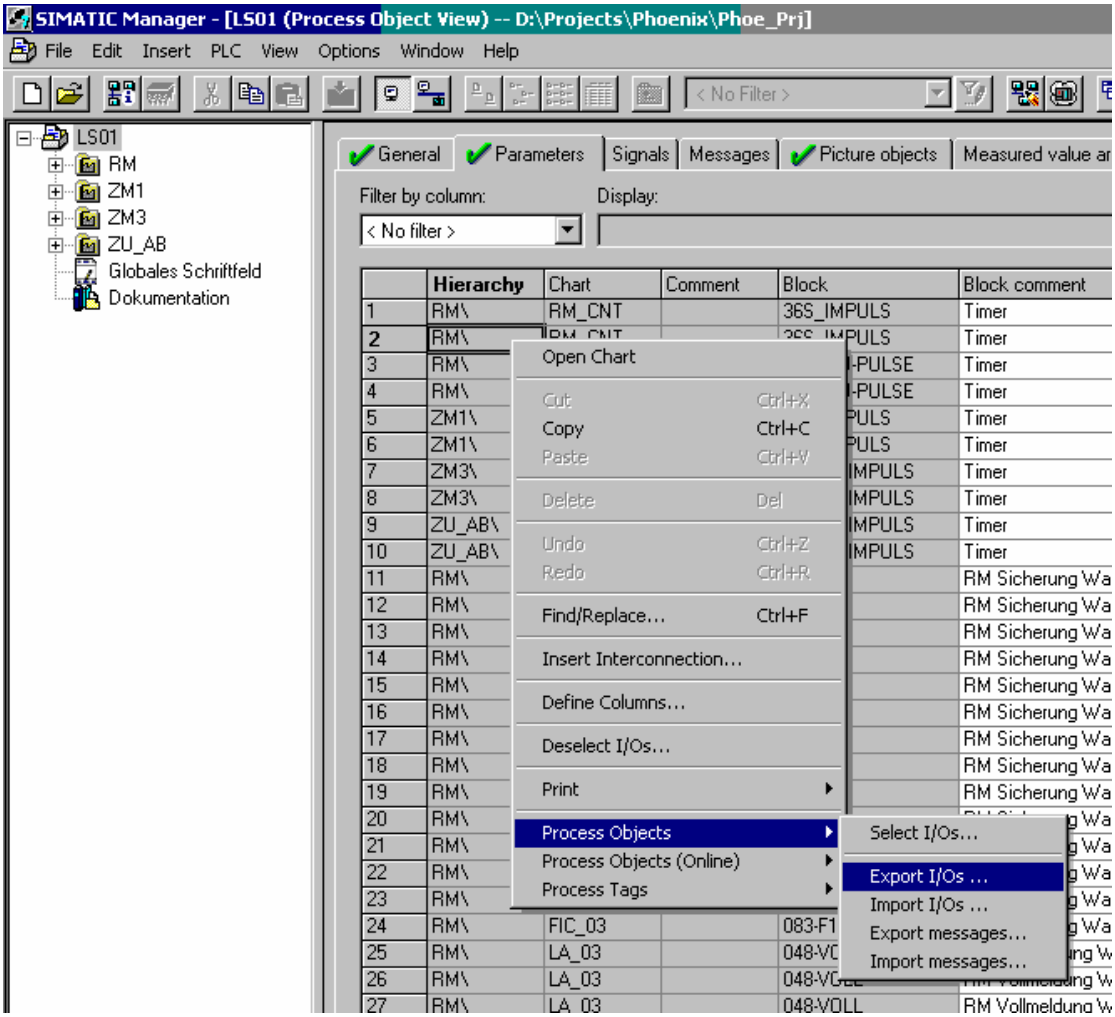
1. Select process object view “Picture objects”.
2. Select the topmost hierarchy folder
   (from the project with the PLCs, not the multi project folder).
3. Refresh the view (F5).
4. Select all lines.
5. Copy with CTRL-C all lines into the clipboard.
6. Start the text editor (notepad) and insert all lines with CTRL-V.
7. Save the text file, e.g. as PO picture.txt.



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 I/Os”.
5. Save the export file, e.g. as LS01_PS.csv.



Import into Excel

Open the file C_INFO.XLS 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 file with the I/O export from the object process view.

Names of the files with hardware config = export files with the hardware configuration 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.

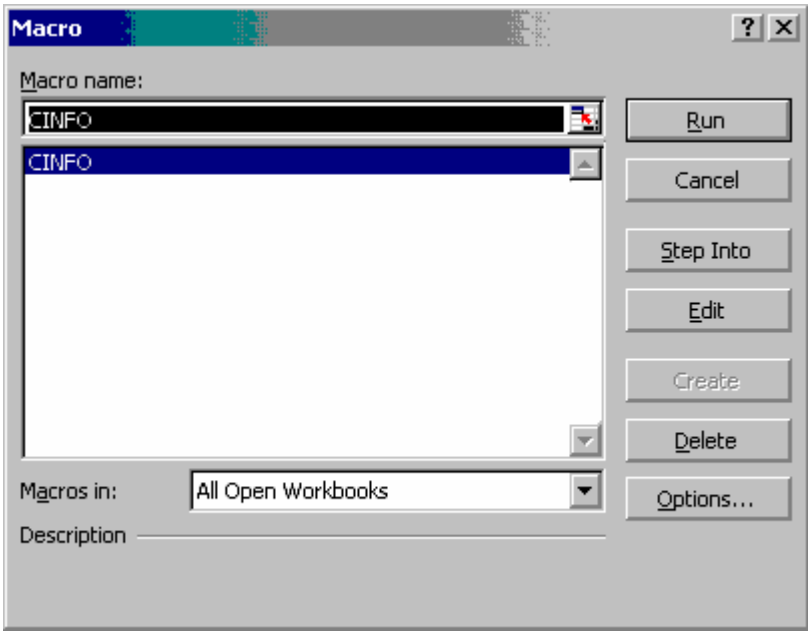
Project = same for all objects

PO picture objects = name of the text file with the process object view „picture objects“.

PO general	GENERAL PO.txt			
PO export files with parameter	LS01_PS.csv			
Files with hardware config	as1202_cfc.cfg	as1203_cfc.cfg	as1204_cfc.cfg	as1205_cfc.cfg
ID start number	1			
List separator	;			
Language	D			
Complex	Phoenix			
Project	Line 1			
PO picture objects	PO picture.txt			

Copy all export files into the same directory like the Excel file C_INFO.xls.

If all necessary files are present, you can start the Excel macro „CINFO“.



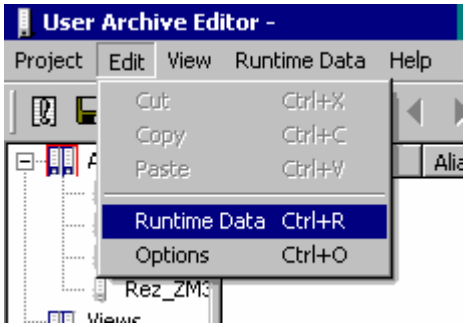
Wait until "Ready" appears again in the status bar of Excel. 

Check: The file „C_INFO.CSV“ should be created by the macro.

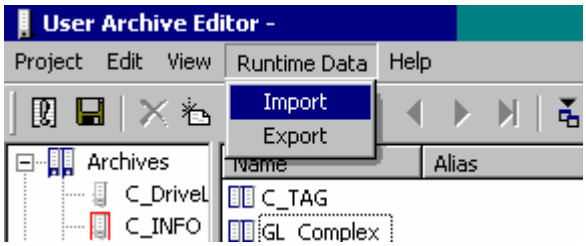
Import into the user archive

Import the runtime data into the user archive

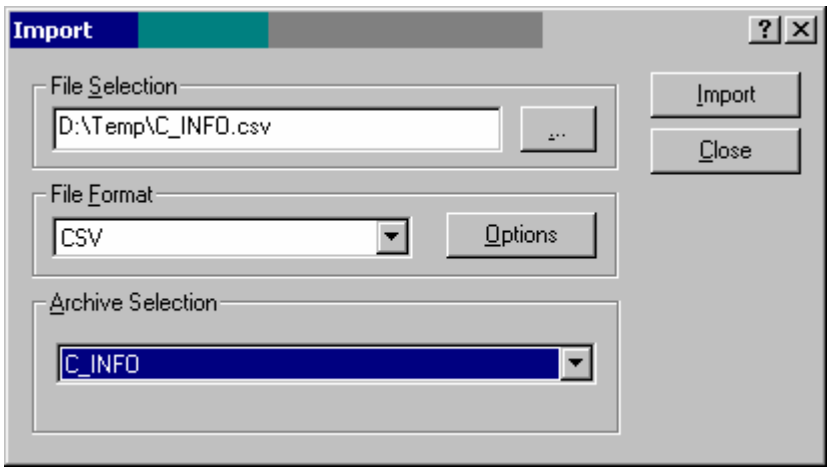
1. Open the User Archive and select C_INFO
2. Deselect in the User Archive Editor the menu item „Edit -> Runtime Data“.



3. Select in menu „Runtime Data -> Import“.



4. Select the file C_INFO.CSV, which was created by the Excel macro. Select the Archive „C_INFO“. If there are already runtime data in the archive, this data records must be deleted before the import. If you like to keep some data from the already existing records, please export the runtime data and copy them into the import file C_INFO.CSV.



If you import data records with an already existing ID, you will get an error message. The import will be then aborted.

Multimedia Interface

Representation

The representation is dependent functional-engaged and, therefore, on the respective application.

As an example application became the video - and maintinance manual Application.

All additional applications are dependent on the end customer, because are not known as for example MCC drawing or flow sheet are.

Engineering

Example: video and maintenance manual

Edit CFG- file

To use the user program neutral interfaces for multimedia applications the following changes are necessary.

Please open in the directory d:\CEM_V6\CONFIG the module specific to file, e.g., C_DRV_1D_007.cfg and edit the area as follows:

1. check the Video path in that the Player was installed

2. switch buttons visible by place the parameter xxxxxBUTTON = 1

```
[Programs]
VIDEO                   = C:\Program Files\Windows Media Player\mplayer2.exe
VIDEO BUTTON            = 1
SOUND OUTPUT BUTTON     = 0
LOCATION BUTTON         = 0
USER MANUAL BUTTON      = 1
AS DIAGRAM BUTTON       = 0
MCC DIAGRAM BUTTON      = 0
IO DIAGRAM BUTTON       = 0
```

User's files Name

The images for AC, IO, MCC are derived always from the TAG name.

Besides, is to be paid attention to follower:

- special characters in the TAG are to be substituted always by "_"
- File ending should be .jpg or .tif. (.bmp is not permitted)

The file names for manual, Map, Sound and video are freely eligible.

The file endings are suitable .pdf, .wav, .avi

Location of the multimedia files:

The multimedia directory must lie always on D:\CEM_V6. (it is created by the installation)

The user's files are stored in the structure shown below.

\Multimedia	
\AS	CAD Pictures of AS objects
\IO	CAD Pictures input/output of the objects
\Manual	Manual to the objects
\Map	Position plan
\MCC	CAD Pictures of the object from the MCC
\Sound	Sound files for info dialog
\Video	place of deposit of the video-files

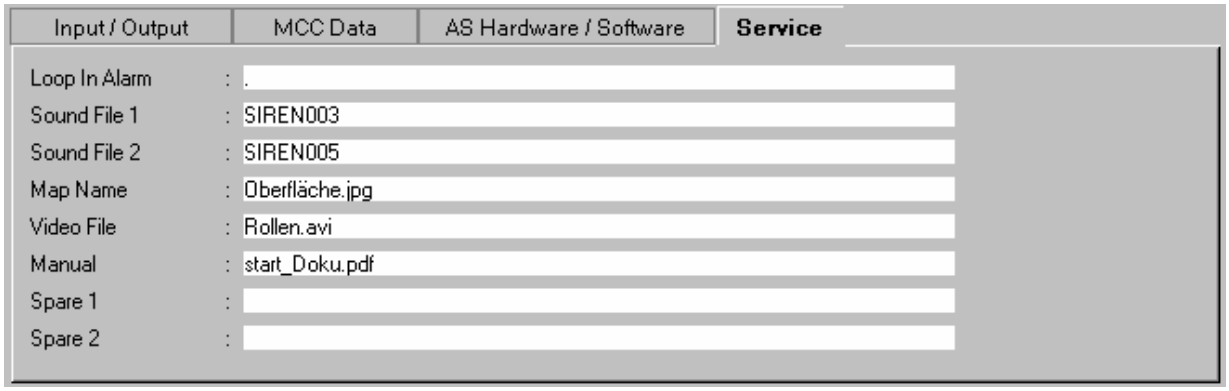
User's files inscribe:

The names for e.g. maintenance manual or video are to be put down for each object in the register service under manual or video without path.

e.g., start_Doku.pdf and Rollen.avi

Also the "Sound File 1" and "Map Name" can be used by entry of an available file name.

„ Sound File 2 "," Spare 1 "and" Spare 2 “ are present spare fields.



Note: The Map Name and the Manual are case sensitive!

WEB

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:

Help

The help keys in all Faceplates are locked.

Status

The group state call is not supported.

Objects

Group instance list is not supported.

Alarm line info

The **I** - button isn't available.

Alarm line LoopInAlarm

The **L** - button isn't available.

CEMAT Reporting system

The CEMAT reporting system isn't available.

The following functionality is limited:

The individual Faceplate positioning is switched off.

The announcement TAG.#comment in all Faceplates is only one-line .

The Hotkeys like **F1 - help** or **ESC** - picture close do not work, because this are intercepted of the Internet Explorer .

Project administration

Content

Ausgabe: 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\11_Project_Administration.doc

Distributed Engineering

A PCS 7 project can be edited by more than one user. At the same time, however, only one user is permitted per S7 program.

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

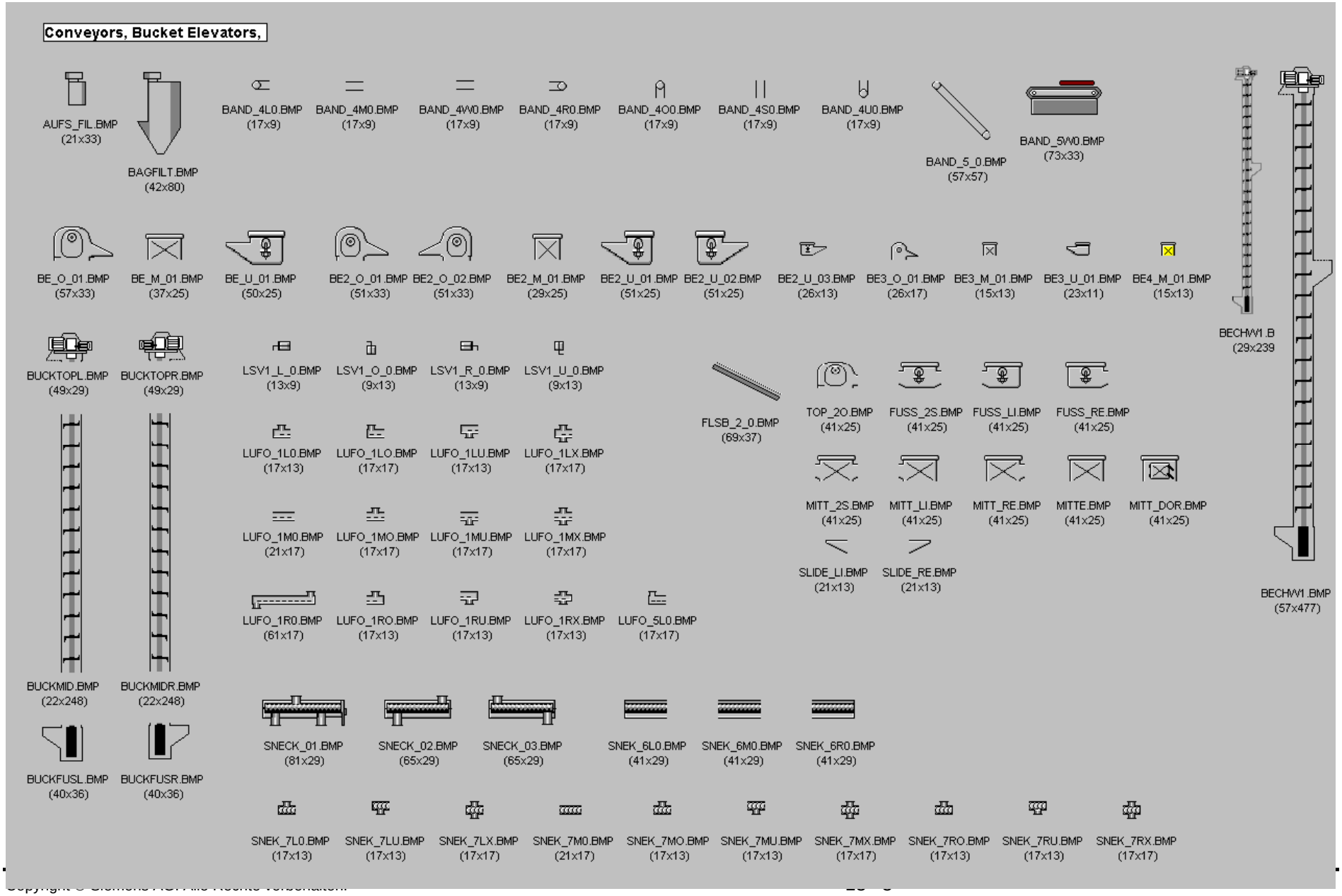
Save the PCS 7 V6 projects only via Archive Function within PCS7 (e.g. PKZip) and never directly under Windows!

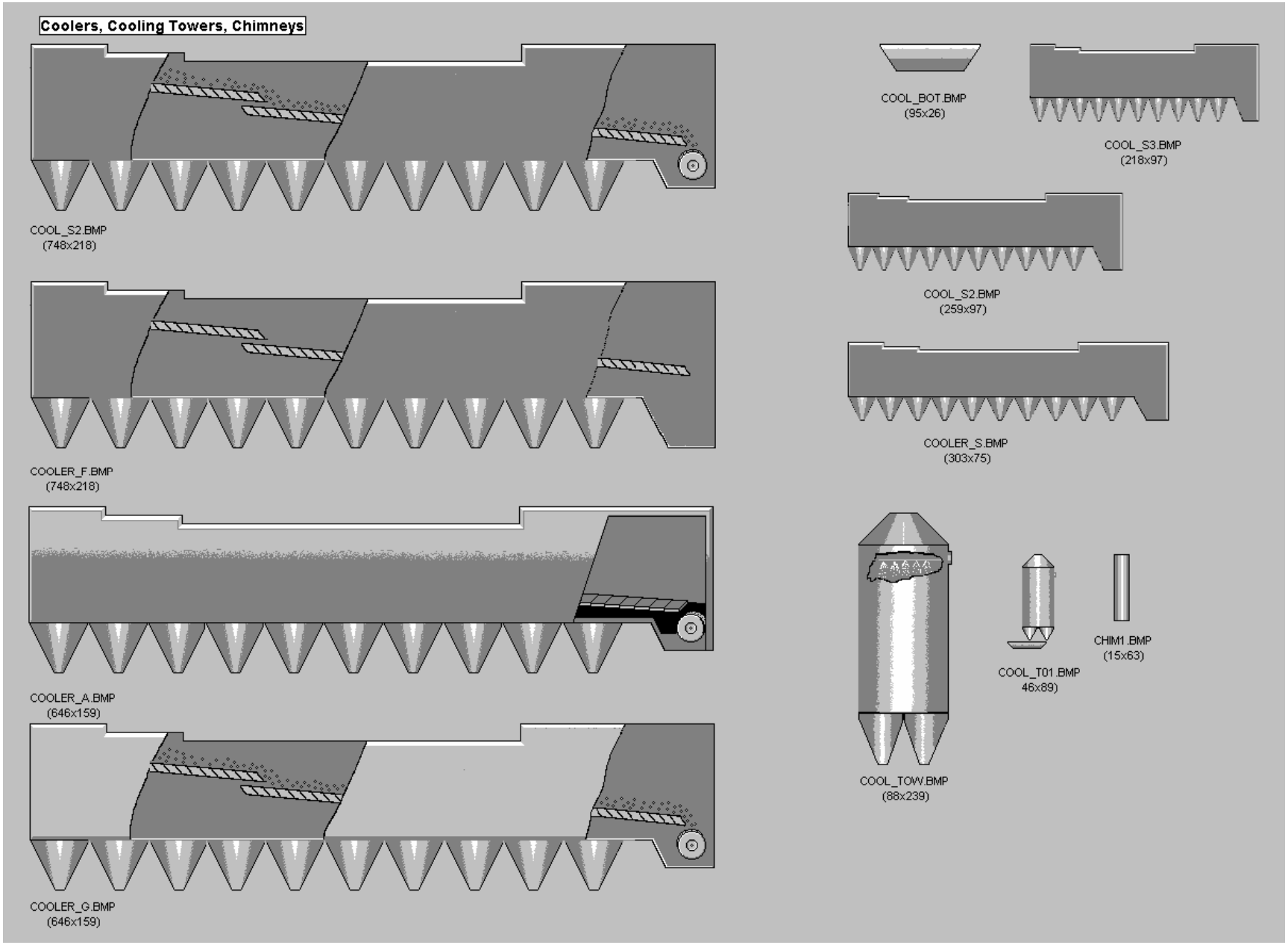
You should always save the PCS 7 project on a CD in the compressed format. (If you restore an uncompressed PCS7 Project from CD you have to remove the write protection afterwards.)

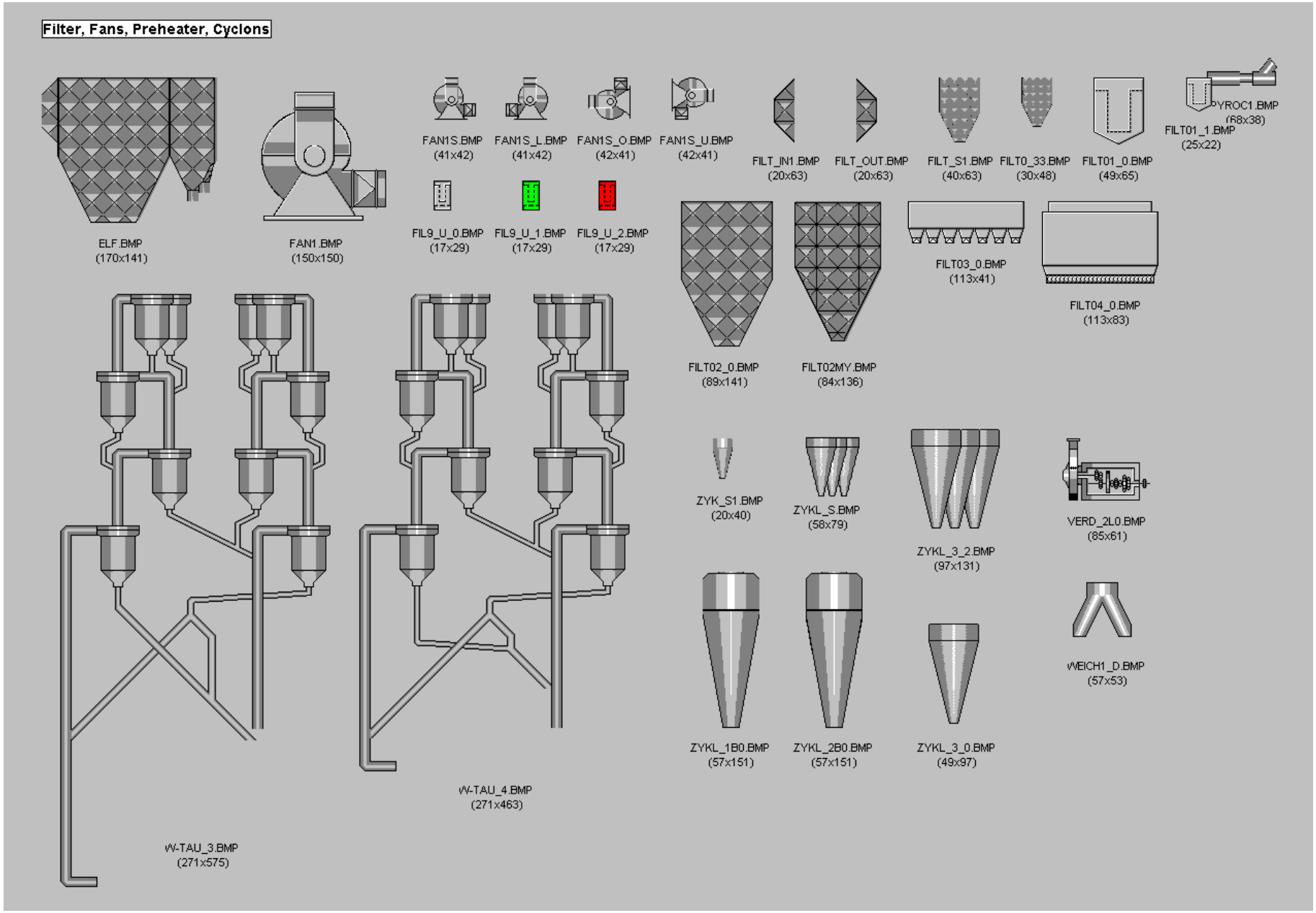
Graphic Templates

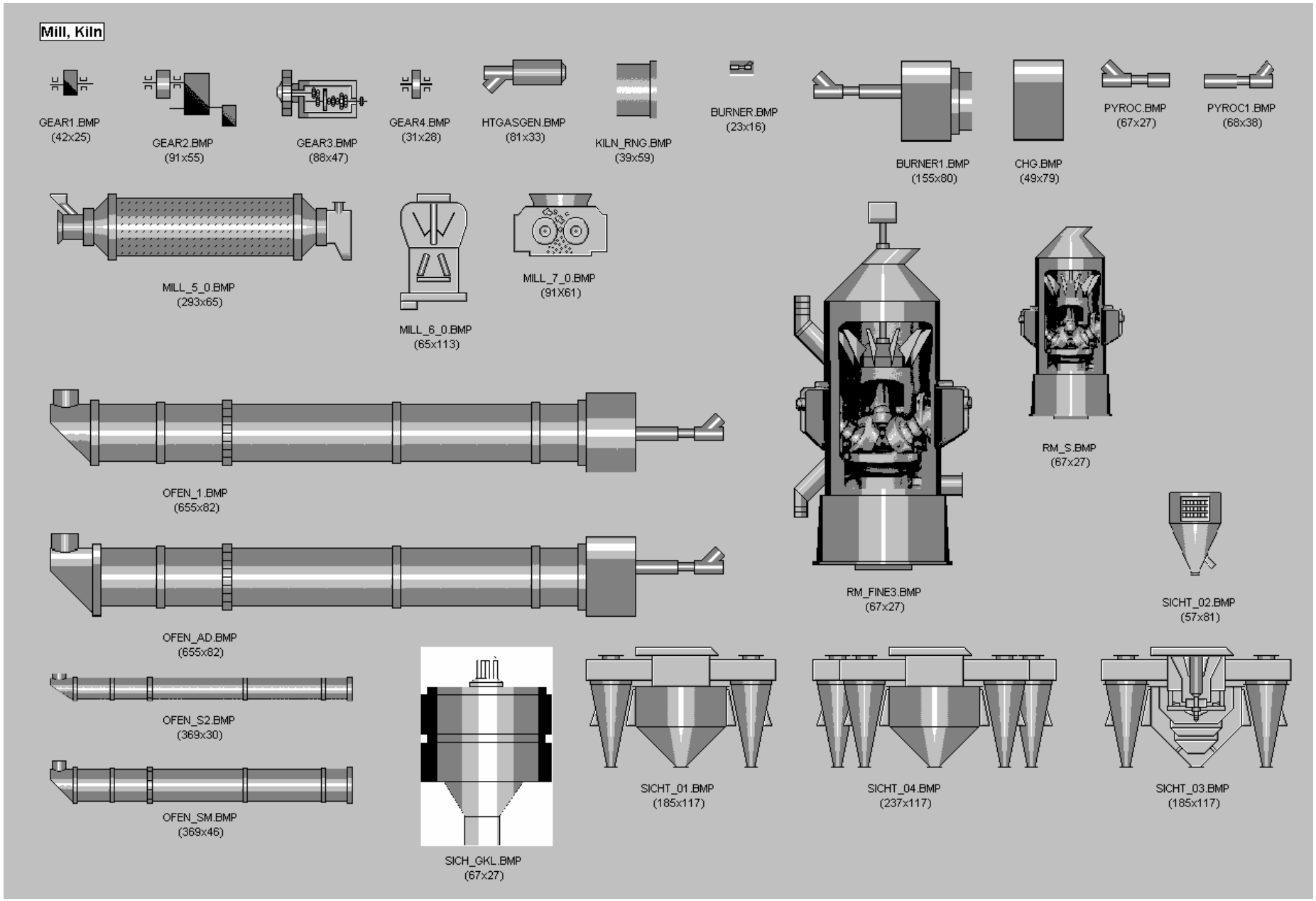
Content

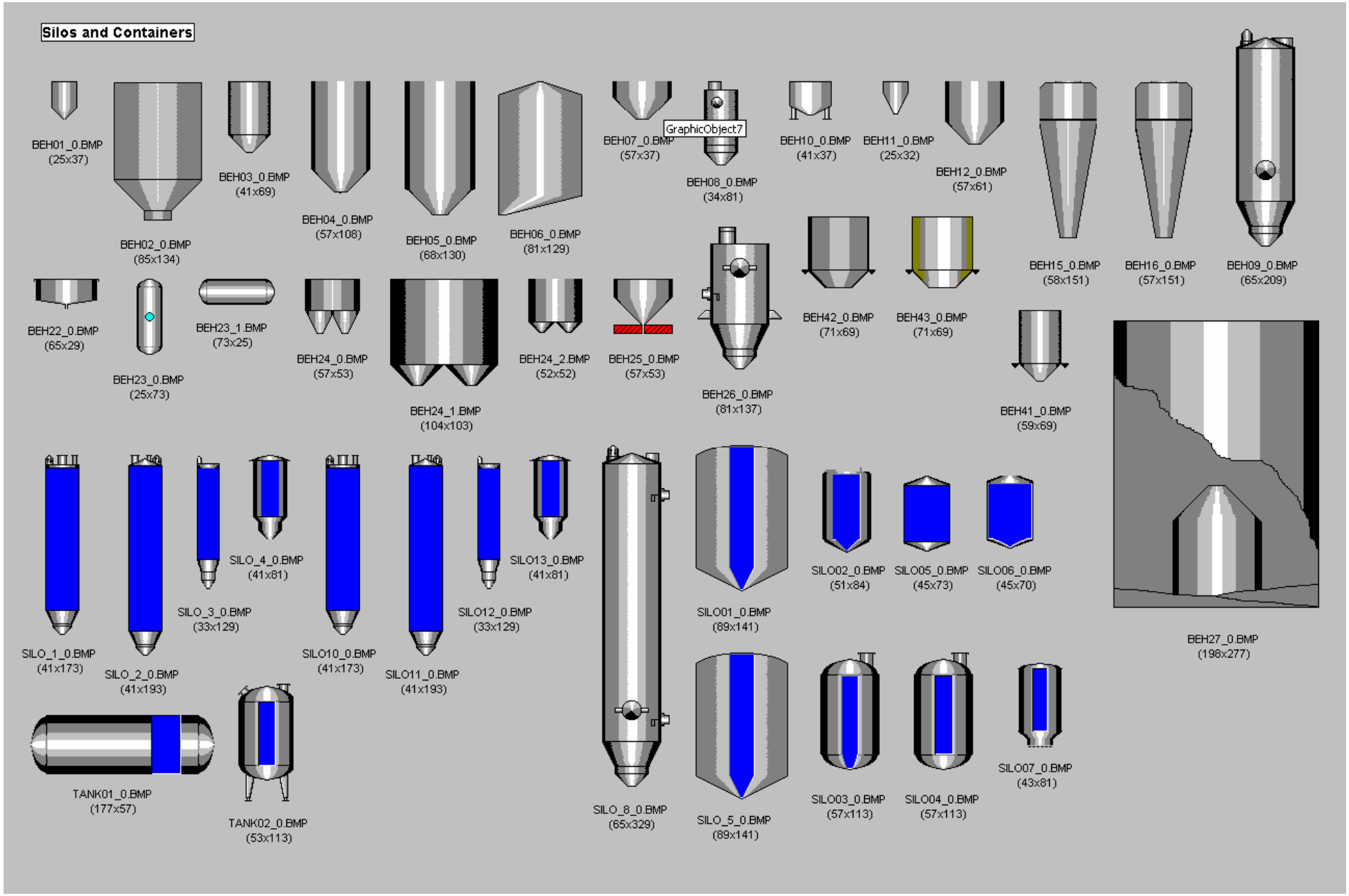
The following pages show the existing Bitmaps. After the installation of CEMAT V6 these bitmaps are located in directory D:\Cem_V6\Bitmaps in zipped format.

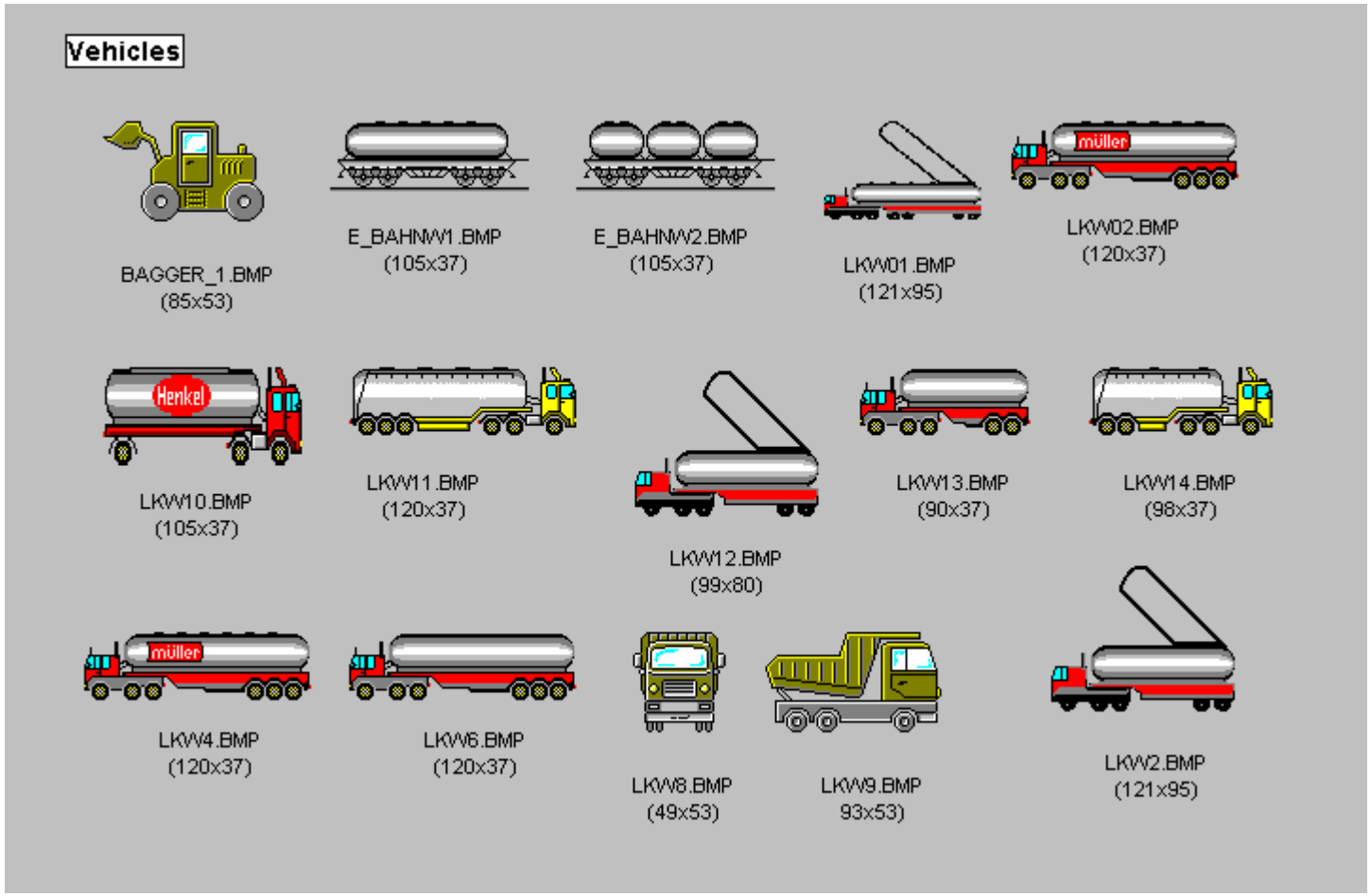












Tips&Tricks

Table of Contents

Edition: 01.03.2005
D:\CEM_V6\DOCU\English\Engineering\14_Tips_Tricks.doc

Notes and Tips for the Use of WinCC / PCS7 V6

The following notes and tips for the use of WinCC/PCS7 V6 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 PCS7 V6 Manuals
- PCS7 Readme
- PCS7 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 Annunciation list

In the CEMAT Annunciation list 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:\CEM_V6\Config\C_Messages.cfg under `[MsgProperty]`.

The default value `MinusSec=86400` corresponds to 86400 seconds = 24 hours.

```
[MsgProperty]
;MinusSec wird vom aktuellen Datum abgezogen.
;Z.B.: 86400 = 1 Tag; 28800 = 8 Stunden; 3600 = 1 Stunde
MinusSec=86400
```

Engineering of the Messages in CFC

The Message class “Operating Message” is not available when in the column “with acknowledged” 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 acknowledged” all boxes.

After that in CFC select Options, Block Types and click on the FB1010 and press New Version.

After that check the messages of one Group again. The messages must have the message class “Operating Messages – Standard.

Replace standard PCS7 INTERLOCK by CEMAT INTERLOCK

The Interlock block from the PCS7 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 PCS7 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 PLC must be completely reloaded:

1. Delete FB 75 = INTERLOK from the PCS7 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!!!!!

Update Information

Contents

Edition : 06.04.2005
D:\CEM_V6\DOCU\English\Engineering\15_Update-Information.doc

Update CEMAT V6.0 / V6.0 SP1 / V6.0 SP2 / V6.0 SP3 to CEMAT V6.1

You have received a CEMAT V6.1 CD and would now like to update your PCS7 project. The following section describes the necessary steps.

Requirements

CEMAT V6.1 requires PCS7 V6.1. Before the installation of CEMAT, you have to update your PCs to PCS7 V6.1.

Caution: The update to CEMAT V6.1 requires a complete reloading of the PLC and consequently cannot be carried out while the plant is running.

Follow the PCS7 V6.1 manual „PCS 7 - Software Update with utilization of the new functions“, specially the chapter 5.4 “Updating a PCS 7 Project”. The following section refers to this manual.

Please pay attention to the hints in OS migration in chapter 5.4.2: The server projects should be migrated directly on the server PCs, in order to you keep the runtime data (curve values and alarms).

Please install CEMAT V6.1 before you migrate your PCS7 project. If you have OS projects with migrated S5 PLCs (CEMAT V2, V3, V4), you have to install all parts for the S5 migration on your PC and register all necessary OCX controls before you start the update to CEMAT V6.1. If you convert PDL files to PCS7 V6.1 and the PDL files contain OCX controls which are not registered, then the PDL files are destroyed! Save all your standard PDLs for he S5 migration and all process mimics with S5 connections.

Consider also the project rules from the CEMAT V6.1 manual „Installation of a PCS7 Project“ = 03_PCS7_Project.DOC

Save Modifications

In case of modifications in the directory CEM_V6 or in the libraries ILS_CEM or PRO_CEM (e. g. CFG-Files or S7 blocks), these modifications will be overwritten during the CEMAT Update.
Save these data except the directory CEM_V6/BIN before starting the update procedure.

Caution: Never save directory „CEM_V6/BIN“!!!

Please save the file C_Config.cfg from all stations.

Modified/new functions

The modified/new functions are listed in file readme_c.wri on the CEMAT CD.

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 PCS7 library V6. You may still use the PTE controllers for existing applications, if your project (and your PCs) already contains the controller functions because of the previous installation of CEMAT V6.0 SP1.

Installation of CEMAT V6.1

Please remove the old CEMAT version by Settings → Contol Panel → Add or Remove Programs → Remove CEMAT.

Restart the computer after the CEMAT program is removed from the computer.

Start the Setup program from the CD-ROM “CEMAT V6.1” and follow the provided instructions.

Installation path: 'D:\CEM_V6\BIN'.

Keys for project standards
000 = CEMAT
006 = Dyckerhoff
007 = Heidelberger Zement
024 = Bushehr
025 = Caima
026 = Alsen
028 = Rossi

After the installation, for all PC stations (ES, SERVER and CLIENT) the following steps have to be carried out:

1. During the installation of PCS7 V6.1 the Siemens directory has been updated.
   For this reason copy all files under d:\Cem_v6\WinCC\Siemens to c:\....\Siemens.
2. During the installation of CEMAT the Config-Files (among others) have been overwritten.

   Adapt the settings in the configurations files (such as Directory of the Acrobat Reader) according to the settings in the backup files. Don’t copy the old configuration files!

   In any case you have to edit file C_config.cfg.
3. If you use the S5 CEMAT migration, then you have to install the CEMAT S5 OS part (OCX files, PDLs, scripts, CFG files) before you migrate your PCS7 Project. Otherwise the standard S5 PDLs (@PG_C_S5??????.PDL) will be wrong converted and can not be used any more.
   Copy all necessary files for the S5 migration to D:\CEM_V6\ into the corresponding directories. Register the OCX files, using the BAT file. Refer to the S5 migration manual.
   The file „pcs7_opengroupdisplay_v6_EwS5.fct“ must be copied to D:\CEM_V6\WinCC\Library and into your OS projects to the \Library directory.
   Please, if exists, delete the file „pcs7_opengroupdisplay_v6_EwS5.fct“ from C:\.....\Siemens\WinCC\aplib\FaceplateDesignerV6.
4. 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 PCS7 Project. Otherwise the standard PDLs (@PG_C_??????.PDL) will be wrong converted and can not be used any more.
   Copy all necessary files for the PTE-Controller function to D:\CEM_V6\ into the corresponding directories. Register the OCX files using the BAT file. Refer to the PTE-Controller instructions.

Update of S7 Program

Please refer to the chapter 5.4.8. of the manual „PCS 7 - Software Update with utilization of the new functions“.

Message numbers and message texts

The setting in PCS7 V5 for the message number assignment was always project-related unique message numbers.
Since PCS7 V6 there are two methods for the message number assignment:
- project-related unique
- CPU-wide unique (default-setting since V6)
During the migration from PCS7 V5 to V6 the free message texts got lost, if you had changed to CPU-wide unique message numbers. Therefore in our CEMAT V6 migration manual we recommended to stay at project-related unique message numbers.
During the migration of a project from V6.0 (which was migrated from V5) to V6.1 and change-over from project-related unique message numbers to CPU-wide unique message numbers we detected no problems regarding the message texts. When we updated again a block with an interface change we did not loose any message texts.

In the new CEMAT blocks a short description is used as event text instead of the abbreviation (e. g. „EBM“ is changed to „Overload“). If you like to retain the old event texts, then you have to change the event texts on the new blocks before you update the chart container. Please keep in your mind, that this change must be done for every CEMAT block update.

Update of the CEMAT Blocks, Symbols and Charts

The CEMAT installation transfers a new version of the CEMAT Library ILS_CEM to the C:\Siemens\Step7\S7libs directory.

This library contains the current CEMAT symbol list, blocks and the system chart.

In case of a project standard (other than “000”) the library PRO_CEM is also transferred into C:\Siemens\Step7\S7libs directory. This library contains all blocks which differ from the normal CEMAT standard.

Carry out the following steps for each PLC:

Symbol list

Open the CEMAT Library ILS_CEM in the Simatic Manager and export the symbols from the CEM_ALL S7 program.

Then open the symbol list of the PLCs in your project and import the new standard symbols. Analyze the error list of the import editor and correct any errors (e.g. non-unique symbols). There may be a conflict of FB501 CYC_MSS with the symbol TE_STWD and of FB596 CYC_MSE with the symbol TE_LIMHL. Please use the new symbols CYC_MSS and CYC_MSE.

CEMAT Blocks

Attention: If at the start of the project you changed the system attributes of the blocks in the block folder (e.g. by setting the parameter hidden), you must repeat these changes in the block folder after the update. The same applies to the message configuration. The existing instances are not affected, but only the new instances.

1. Copy all blocks from the S7 program of the CEMAT ILS_CEM library into the block folder of your PLC (overwrite = yes).
   In case of a project standard copy after that also all blocks from PRO_CEM library into the block folder of your PLC (overwrite = yes).
2. Delete in the block folder of your PLC all blocks that have ES_MAP as author. (These are all blocks that were created by the CFC.)
3. Carry out a general reset of the PLC.
4. Load the system data into the PLC and start it.
5. Open the CFC editor by opening any chart.
6. You must now use Options -> Block Types to update all blocks of the chart folder with the "New Version" key or transfer the following listed blocks from the offline block folder into the chart folder.
7. Please carry out the following modifications in CFC for all listed objects. Please refer the particular object documentation.
   - C_DRV_1D, C_DRV_2D: Delete the block C_SPEEDM and connect the pulse input directly to the motor block. The PLC cycle must be set to 100 ms in the hardware configuration.
   - 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 not longer supported. Please use the blocks C_DRV_1D, C_DRV_2D, C_DAMPER and C_VALVE with the SIMOCODE adapter block C_SIMO_A.
   - The block C_ANNUN8 is reduced to 7 alarms. The function / reaction of the block has also changed. Please check your interconnections.
   - C_SILOP: Delete the block C_ SPCNT and connect the pulse input directly to the motor block. The PLC cycle must be set to 100 ms in the hardware configuration.
   - 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 any more or the assignment has been changed.
8. Replacement of the System chart:
   The system chart has been changed and you must replace the existing system chart by the new system chart. Therefore copy the system chart SYSPLC00 from the library ILS_CEM within the “Plant View” into the chart folder of your PLC. Carry out the necessary settings in the new system chart. After that delete the old system chart and rename the new chart (PLC no.). Refer the CEMAT manual „03_PCS7_Projekt.DOC“. Please check the “Run Sequence”!
   If your project has the message number concept “project-related unique message numbers” you can not copy the system chart. In this case you have to modify your existing system chart like the model from the library ILS_CEM. Please check the “Run Sequence”.

List of the blocks in the chart folder that must be updated:

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
FB1009	C_ROUTE	OB1
FB1010	C_GROUPE	OB1
FB1011	C_SILOP	OB1
FB1013	C_SELECT	OB1
FB1015	C_COUNT	OB35
FB1016	C_RUNNT	OB1
FB1020	C_FB_PLC	OB1
FB1026	C_MEAS_I	OB1
FB1033	C_SIMO_A	OB1
FB1052	C_PLC_SEND	OB1
FB1053	C_PLC_RECEIVE	OB1
FB1054	C_PLC_PLC	OB1/OB35

Absolute	Symbol	Task
FB1075	C_INTERL	OB1
FB1076	C_INTER5	OB1
FB61	CTRL_PID	OB35
FB70	RATIO_P	OB35
FB76	CTRL_S	OB35
FC527	OB100_SYS1	OB100
FC528	OB1_SYS2	OB1
FC529	OB35_SYS1	OB35
FC530	OB35_SYS2	OB35
FC1017	C_MUX	OB1
FC1018	C_ADAPT	OB1
FC1088	C_PUSHBT	OB1
FC1102	OB1_SYS1	OB1
FC1103	C_OB1SY1	OB1

Non CEMAT Blocks

If you use blocks from another library (e. g. PCS7 driver blocks) you have to update also this blocks.
Attention: If you exchange one driver block you have to exchange all driver blocks.
A mix of V5 / V6 and V6.1 driver blocks is not allowed.

Update of the WinCC Project

CEMAT Update

After the CEMAT installation, the D:\Cem_V6\WinCC directory contains the current version of the global scripts and the system pictures. These must be transferred to the WinCC project.

Before you start the update, you have to check which type of message area selection you will use, the area selection like in CEMAT V6.0 or the standard message area selection of PCS7 (area selection with user rights). If the areas and the picture tree are created manually and the areas are different from the plant view, then you must use the old CEMAT V6.0 message selection (area selection with the L-Button).

Procedure, if the area and picture tree is generated automatically from the plant view and the message area selection of PCS7 V6.1 is used

Carry out the following steps for each WinCC Project

1. Open the WinCC Explorer.
2. Open, save and close the OS project editor. By this procedure all standard mimics (area view, buttons, alarm line, etc) are replaced by the new standard PDLs of V6.1.
3. Close WinCC Explorer.
4. Delete all CEMAT standard scripts in your PCS7 project '<OS>\Library'. Keep your own scripts in PCS7 project '<OS>\Library'. Copy all global scripts from 'D:\CEM_V6\WinCC\Library' into PCS7 project '<OS>\Library' (incl. subdirectories).
5. The CEMAT standard global scripts in PCS7 project '<OS>\Pas are not longer used and can be deleted.
6. Copy all files from "D:\CEM_V6\WinCC\GraCS' into the PCS7 project '<OS>\GraCS'.
7. Open the WinCC project.
8. Open 'Global Script' and regenerate the header. (Menu/Options/Regenerate Header)
9. 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.
10. Probably you modified the template pictures @PCS7Typicals.pdl and @Template.pdl for your project. You can further use the old template pictures. Additionally you can use the new template pictures @C_PCS7Typicals_V61.pdl and @C_Template.pdl.
    Please check the project rules from the CEMAT manual „08_OS_Projektierung.doc" regarding template pictures. Reduce the objects in the template pictures to a necessary number. This will increase the speed for generating symbols in process mimics.
    The old symbols from V6 are still working (except the damper positioner), but if you like to use the new functions, you have to use the new symbols. In order to use the new functions in the existing process pictures, you have to replace the symbols in your process mimics.
11. Please carry out your settings in OS project editor according to CEMAT manual „03_PCS7_Projekt.DOC":
    Select in the register "Message display" the item "Acknowledgeable messages in separate list" and select "Create / update group display".
    In the register "Basic data" the standard CEMAT PDLs are listed, which should not be overwritten by the PCS7 V6.1 standard:
    @Overview1.pdl, @AlarmOneLine.pdl, @Buttons11.pdl, @CSIGQuit.pdl, @HornQuit.pdl.
    For the Server please select to overwrite! For all other station types don't select.
12. Horn configuration – Refer CEMAT manual „03_PCS7_Projekt.DOC".
13. Create internal variable C_ServerName - Refer CEMAT manual „03_PCS7_Projekt.DOC".

14. Create users with area specific rights on ES, OS single station and OS client. Check all users! Maybe you have to change the user right properties of the block symbols. Refer CEMAT manual „03_PCS7_Projekt.DOC“ and CEMAT manual „08_OS_Engineering.doc“ regarding user rights and instance specific user rights.
15. Replace all symbols for damper positioner by the new damper positioner symbol. All other old symbols from CEMAT V6.0 are working, but they do not support the new symbol functions of CEMAT V6.1 (e. g. mark active object for loop-in-alarm function, positioning of faceplates, etc.).

Procedure, if the areas and picture tree is created manually and the areas are different to the plant view and could not be generated from the plant view

In this case, the new message area selection and the new alarm mimic from CEMAT V6.1 can not be used. You have to use the message selection method from CEMAT V6.0.

Carry out the following steps for each WinCC Project

1. Open the WinCC Explorer.
2. Open, save and close the OS project editor. By this procedure all standard mimics (area view, buttons, alarm line, etc) are replaced by the new standard PDLs of V6.1.
3. Close WinCC Explorer.
4. Delete all CEMAT standard scripts in your PCS7 project '<OS>\Library'. Keep your own scripts in PCS7 project '<OS>\Library'. Copy all global scripts from 'D:\CEM_V6\WinCC\Library' into PCS7 project '<OS>\Library' (incl. subdirectories).
5. For all PC Stations except the Stations without Alarm line (Server) copy file PlaySoundForAlarm.pas from 'D:\CEM_V6\WinCC\Pas' into the PCS7-Projekt '<OS>\Pas'
6. Copy all files from ''D:\CEM_V6\WinCC\GraCS' into the PCS7 project '<OS>\GraCS'.
7. Open the WinCC project.
8. Open 'Global Script' and regenerate the header. (Menu/Options/Regenerate Header)
9. If you have modified for your project system PDL’s and scripts you have to carry out the modification again in the new system files. Refer to your backup.
10. Probably you modified the template pictures @PCS7Typicals.pdl and @Template.pdl for your project. You can further use the old template pictures. Additionally you can use the new template pictures @C_PCS7Typicals_V61.pdl and @C_Template.pdl.
    Please check the project rules from the CEMAT manual „08_OS_Projektierung.doc“ regarding template pictures. Reduce the objects in the template pictures to a necessary number. This will increase the speed for generating symbols in process mimics.
    The old symbols from V6 are still working (except the damper positioner), but if you like to use the new functions, you have to use the new symbols. In order to use the new functions in the existing process pictures, you have to replace the symbols in your process mimics.
16. Please carry out your settings in OS project editor according to CEMAT manual „03_PCS7_Projekt.DOC“:
    Select in the register “Message display” the item “Acknowledgeable messages in separate list”.
    Difference to the CEMAT manual „03_PCS7_Projekt.DOC“: Do not select “Create / update group display”.
    In the register “Basic data” the standard CEMAT PDLs are listed, which should not be overwritten by the PCS7 V6.1 standard:
    @Overview1.pdl, @AlarmOneLine.pdl, @Buttons11.pdl, @CSIGQuit.pdl, @HornQuit.pdl.
    For the Server please select to overwrite! For all other station types don’t select.
11. **Difference to the CEMAT manual „03_PCS7_Projekt.DOC“:**
    No horn configuration.

12. Create internal variable C_ServerName - Refer CEMAT manual „03_PCS7_Projekt.DOC“.
13. Create users with area specific rights on ES, OS single station and OS client. Check all users! Maybe you have to change the user rights properties of the block symbols. Refer CEMAT manual „03_PCS7_Projekt.DOC“ and CEMAT manual „08_OS_Engineering.doc“ regarding user rights and instance specific user rights.
14. Replace all symbols for damper positioner by the new damper positioner symbol. All other old symbols from CEMAT V6.0 are working, but they do not support the new symbol functions of CEMAT V6.1 (e. g. mark active object for loop-in-alarm function, positioning of faceplates, etc.).
15. Carry out the adaptations in the standard PDLs @AlarmOneLine.pdl, @CSIGQuit.pdl, @HornQuit.pdl for the standard pictures for the message system like described in the old manual for CEMAT V6.0. Please follow for the details the old CEMAT V6 manual „03_PCS7_Projekt.DOC“. Change the acknowledge functions in the corresponding events for PLC acknowledgement and for horn acknowledgement. The old commands are existing as comments. Reactivate the old functions by removing of the comment characters. Call for each PLC the function AcknowledgeAlarm(<AS-Nr.>) and AcknowledgeHorn(<AS-Nr.>) in the corresponding events.
    Create for a OS client the trigger for the action “AcknowledgeHorn“ like for CEMAT V6.
    In the standard PDL @Buttons11.pdl modify the events for the buttons “CEMAT alarm mimic” and “return to previous process mimic” to the old CEMAT alarm mimic.
    Under Events you have to replace the picture name (@C_AlarmListing.PDL into @C_AlarmList.PDL). The button “return to previous process mimic” is located behind the button for the CEMAT alarm mimic.

Procedure, if migrated S5 PLCs exists

In this case, the new message area selection and the new alarm mimic from CEMAT V6.1 can not be used. You have to use the message selection method from CEMAT V6.0.

Procedure like for a PCS7 project where the areas and the picture tree are created manually.
Additionally we recommend the following procedure:
Because of the early conversion of pictures by the OS migrator it can happen, that some PDL files are converted wrong. E. g. if OCX controls are in the PDL files or if for the script „pcs7_opengroupdisplay_v6_EwS5.fct“the header was not yet generated. The failure is: The OCX controls are not present or the group module, route module and the selection module the faceplate can not be opened. The mouse event was deleted.
Workaround: After a complete OS migration (see step 1 to 16 before) copy the process pictures from the backup in the V6.0 version again the GraCS directory. Start in the WinCC explorer the conversion of the die PDL files again. Now the S5 OCX controls should be integrated correctly and the faceplate call from group, route and selection module should work.

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 V6.0 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.

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 Serverdata (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.

Transfer Changed CEMAT Standard Blocks

One or more S7 blocks have been changed because of en extension / error correction and must now be copied into the PCS 7 project. You have received a library that contains the modified blocks.

Note: This procedure is possible only if no interface changes have been made to the blocks. If at the start of the project you changed in the block folder the system attributes of the affected blocks (e.g. by setting the parameter hidden), you must repeat these changes in the block folder after the update. The same applies to the message configuration. The existing instances are not affected, but only the new instances.

Proceed as follows to update the S7 blocks:

1. Copy all blocks from the S7 program of the CEMAT "Update" library into the block folder of your PLC.
2. Open the CFC editor by opening any chart.
3. You must now replace the changed blocks in the chart folder of the CFC with the new version from the offline block folder (Options -> Block Types).
4. Compile CFC.
5. Load changes in the program into the PLC.