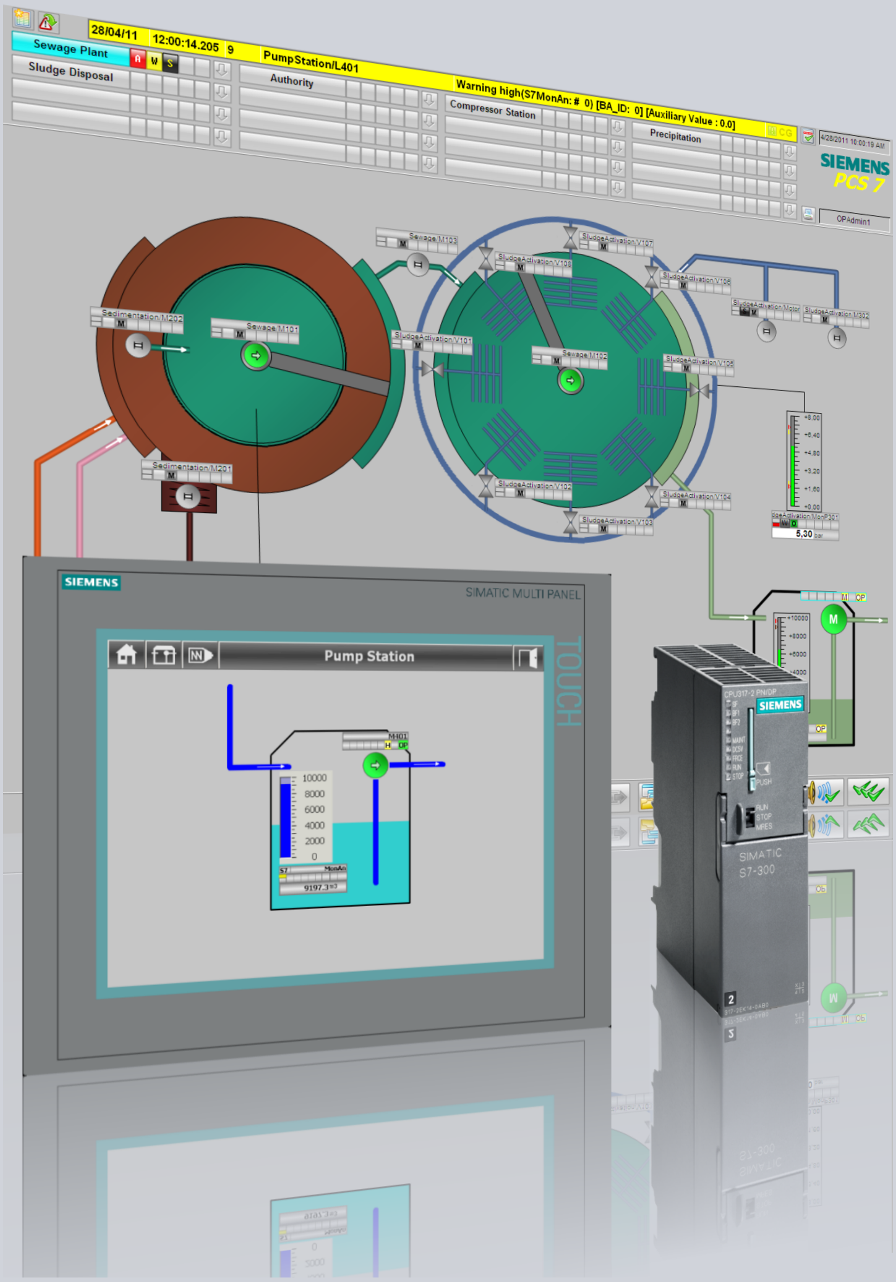
Integration of S7-300 Package Units in SIMATIC PCS 7 with PCS 7 Industry Library

SIMATIC PCS 7

Application • July 2012



Applications & Tools

Answers for industry.

SIEMENS

Siemens Industry Online Support

This article is taken from the Siemens Industry Online Support. The following link takes you directly to the download page of this document:

http://support.automation.siemens.com/WW/view/en/50708061

If you have any questions about this document, please send an e-mail to:

online-support.automation@siemens.com

SIEMENS

SIMATIC PCS 7

Integration of S7-300 Package Units in a PCS 7 System

Application Example

Warranty and Liability

Notes

The application examples are not binding and do not claim to be complete with regard to configuration, equipment or any contingencies. The application examples do not represent customer-specific solutions; they are only intended to provide support for typical applications. You are solely responsible for the correct operation of the described products. These application examples do not relieve you of your responsibility to use sound practices in application, installation, operation and maintenance. When using these application examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these application examples at any time without prior notice. If there are any deviations between the recommendations provided in this application example and other Siemens publications (e.g. catalogs), the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.

Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). Damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change in the burden of proof to your detriment.

It is not permissible to transfer or copy these Application Examples or excerpts thereof without express authorization from Siemens Industry Sector.

Table of Contents

1 Automation Task

Introduction

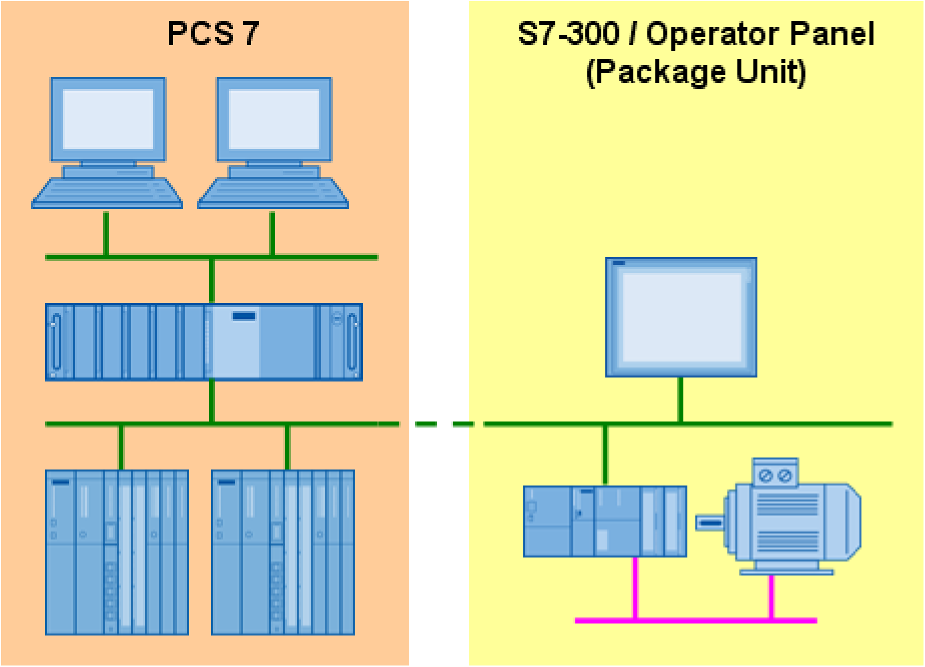
Apart from the components directly related to the engineering process, many production plants include a series of further machines and aggregates. In many cases, these so-called ‘package units’ (PU) are complete subsystem models for specific production tasks, or they are equipped with further plant components which are not part of the standard PCS 7 system, such as S7-300 CPUs or operator panels, for example. Package Units being able to be e. g. centrifuges, dryers or pondering stations.

Problems may arise when connecting these package units to the control system. The use of various types of operation, display and processing schemes, as well as different types of alarm systems make the integration of package units into the higher-level process control system more complicated.

Description of the automation task

This document provides different options for the integration of S7-300 automation systems and operator panels (HMI devices) in a PCS 7 project, whereby the configuration settings shall meet the PCS 7 standards as close as possible.

Figure 1-1



2 Automation Solutions

Using of PCS 7 Industry Library (IL)

The “IL for S7” Library offer a standardized procedure for the integration of S7-300 CPUs and operator panels. They include technological blocks for the processing of actuators and sensors which are also suitable for operation in S7-300 automation systems.

These blocks support the messaging functions (Alarm_DQ) and can be operated and monitored via the OS. Visualization on the operator panel can be configured with the help of the associated interface blocks and WinCC flexible faceplates.

Benefits of the solution with PCS 7 Industry Library

- Enables integrated and harmonic solutions of a process control task and thus an optimal operation of the whole process by reducing risk of operation failures
- Amount of user specific functions will be reduced, this saves costs over the complete life cycle
- Synergy effects regarding training and know-how transfer

Scenarios for the connection of an S7-300 package unit (PU)

This document presents two different scenarios for connection as follows:

- Communication between S7-400 and S7-300 CPUs by using the blocks “S7Get” and “S7Put”
  (see chapter 5 “Communication Management in the S7-400 CPU”)
- Configuration of an S7-300 package unit incl. panel
  (see chapter 6 “Configuration of the S7-300 CPU with PCS 7 Industry Library”)

Selection of the best solution depends on whether the PU to be integrated is a fixed and complete unit and whether the control program of the PU can be modified or supplemented to some extent.

Delimitation

The following issues are not considered in this document:

- Connection to S7-200, S7-1200, TIA portal
- Connection of controllers from third-party providers.
  For details on this automation task, please refer to the following document which includes a possible solution:
  http://support.automation.siemens.com/WW/view/en/49740087
- PROFIBUS connection
  This document refers only to Ethernet connections. PROFIBUS may be used as an alternative and differs only with respect to the connection configuration.
- Programming for S7 function blocks
- Creation of faceplates in the OS and on the operator panel.
  For more information on this issue, please refer to the relevant PCS 7 and WinCC flexible documentations.

Required knowledge

Basic knowledge in the fields of system configuration with PCS 7 and configuration with WinCC flexible is assumed.

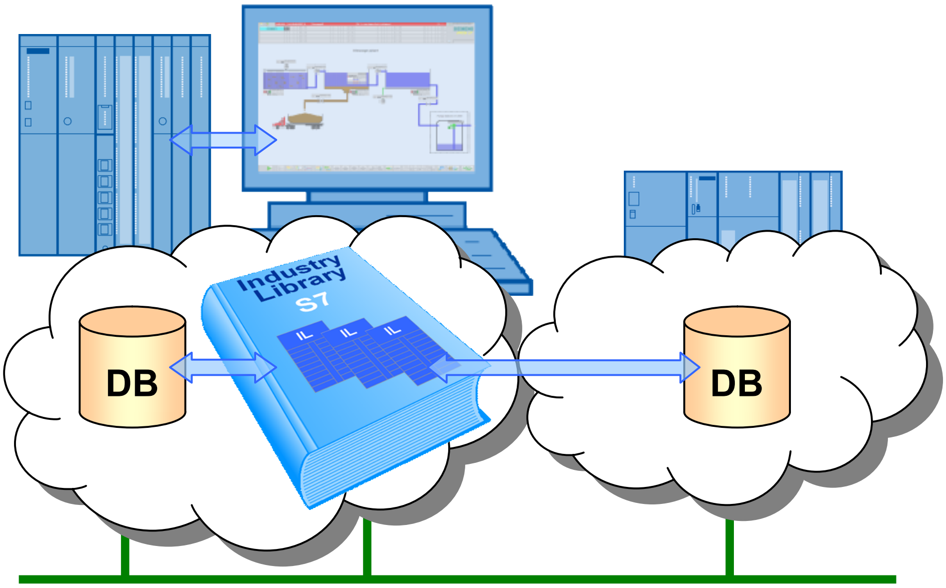
Alternatives

As an alternative to PCS 7 Industry Library you may also use the blocks of the "S7 standard library", as well as your own blocks for the S7-300 program configuration. This approach, however, requires extensive effort in preparing the control program and visualization in the OS and on the operator panel. Furthermore, compliance with the PCS 7 standards cannot be guaranteed and might lead to further problems.

2.1 Communication management in the S7-400 CPU

The block “S7Get” is used to read the values of a DB from an S7-300 CPU and to transfer them to the DB of an S7-400 CPU in the PCS 7 system. The data is then processed in the S7-400 CPU and retransferred to the DB of the S7-300 unit. The processing of data in the S7-400 CPU enables easy integration in the PCS 7 OS.

Figure 2-1

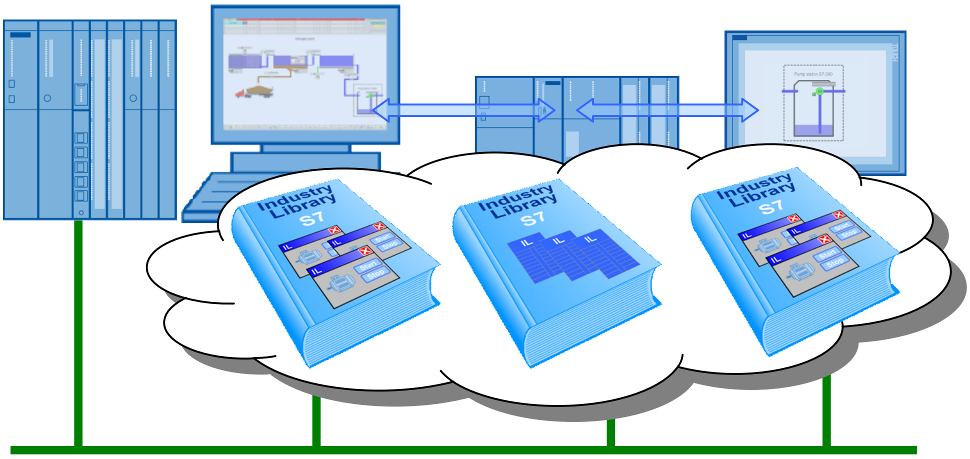


This solution does not require any intervention in the configuration of the S7-300 control program since the “S7Get” block as well as the “S7Put” block are processed in the S7-400 CPU. If required, some defined interfaces may be integrated in the form of data blocks, if necessary at all.

2.2 Configuring the S7-300 CPU with Industry Library

In this scenario, the package unit program is processed exclusively in the S7-300 CPU. Visualization in the PCS 7 OS and on the operator panel is realized with the help of technological blocks and IL for S7 interface blocks. In this approach, the control program of the PU is created anew with IL elements.

Figure 2-2



3 Basic Elements

3.1 Industry Library

The block library IL for S7 used in this application includes communication blocks, monitoring blocks, technological blocks, operating blocks and simulation blocks. These blocks are operated and monitored by means of the associated faceplates.

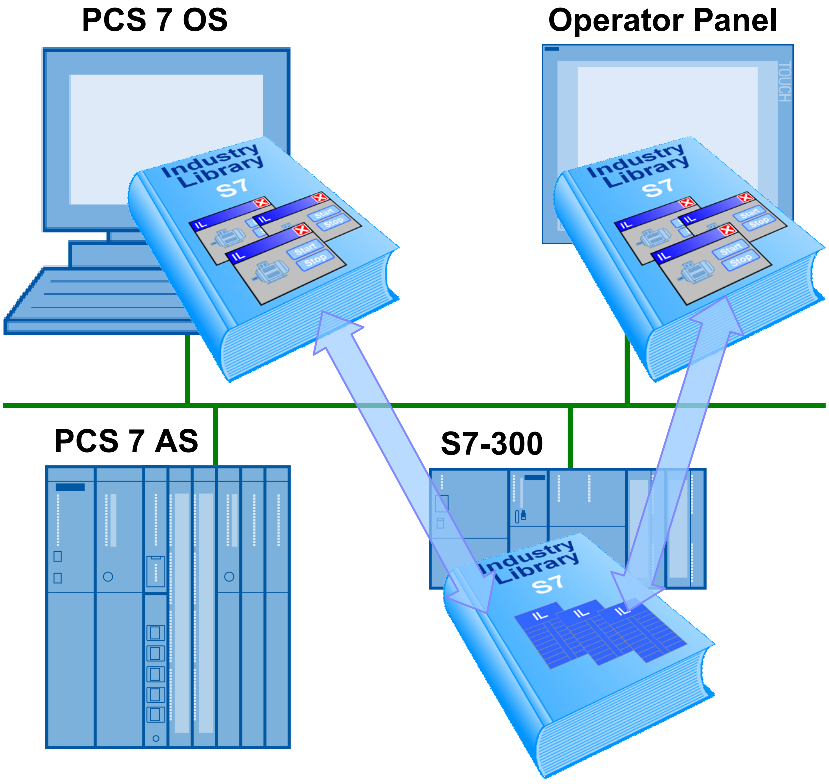
The IL for S7 alarm concept is based on the system function block “ALARM_DQ”. The technological functions are complemented by interface blocks for visualization on the operator panel.

The integration of S7-300 CPUs and operator panels in the PCS 7 environment is realized with the help of the following libraries:

Table 3-1

Library	Description
PCS 7 Industry Library V8.0 (IL for S7)	Technological blocks for use in S7-300 CPUs and faceplates for visualization in the OS. WinCC flexible faceplate library for display and operation on the operator panel.

Figure 3-1



Reasons for the use of Industry Library

- minimize risks through standardization
- 'look and feel' in coherence with PCS 7 APL
- easy integration of S7-300 CPUs and operator panels
- direct access from PCS 7 server to S7-300 CPUs (no subnetting)
- reduce the time and expense required for development
- prepared very well for upgrading to higher PCS 7 versions

System requirements when using Industry Library

The PCS 7 Industry Library V8.0 is compatible with the following configuration software:

Table 3-2

Library	Configuration software
IL for S7	SIMATIC STEP 7 V5.5 SIMATIC S7 CFC SIMATIC WinCC V7.0 SP3 or SIMATIC PCS 7 V8.0 and WinCC flexible Advanced 2008 SP3
IL for PCS 7	SIMATIC PCS 7 V8.0 SIMATIC PCS 7 APL V8.0 WinCC flexible Advanced 2008 SP3

The following hardware, or later versions, should be used:

Table 3-3

Library	Hardware
IL for S7	Ab CPU-314C-2 PN/DP Firmware ≥ V3.1 oder IM 151-8PN
IL for PCS 7	Same system requirements as for PCS 7 V8.0
IL for S7 / IL for PCS 7 (WinCC flexible)	MP 277 or MP 377 (display-size ≥ 10 inch) SIMATIC IPC277D (display size ≥ 12 inch)

Recommendations for the calculation of a CPU-315 PN/DP with Industry Library

The CPU-315 PN/DP provides the following relevant resources:

- main memory: 384kB
- simultaneous messages: 300

(See device manual “CPU 31xC and CPU 31x: technical data”
http://support.automation.siemens.com/WW/view/en/12996906)

Assuming the following program parts in mixed configuration to be controlled by a CPU-315 PN/DP, the degree of utilization is approx 60%.

- 20 process tags
- 20 drives
- 2 aggregates
- 2 controllers

3.2 Time synchronization

In PCS 7 systems the clock times of all components – such as PC stations, automation systems or other peripheral devices – need to be synchronized. This is important to ensure the correct succession of processes or the archiving of messages in the correct time order.

Procedure

Time synchronization can be performed in different ways as, for example, by defining a domain server or a central system clock (SICLOCK) as time master.

For further details on time synchronization, please refer to the manual "SIMATIC Process Control System PCS 7 time synchronization"

Integration of operator panels

The operator panels should also be synchronized, so as to avoid time inconsistencies. Operator panels, however, cannot be synchronized with a SIMATIC or NTP procedure.

Operator panels are synchronized by defining area pointers which can be used to synchronize the system time of the panel with that of the controller. The control program uses the function "READ_CLK" to provide the area pointer with the current system time.

The configuration settings for time synchronization are described further below in this documentation.

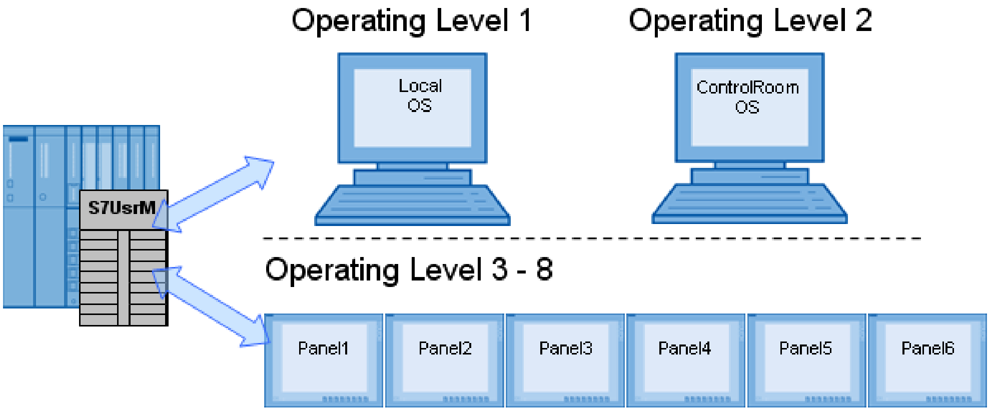
Note

Automation systems with an integrated Ethernet interface require a NTP procedure (NTP = Network Time Protocol) for synchronization.

3.3 Multi-user operation

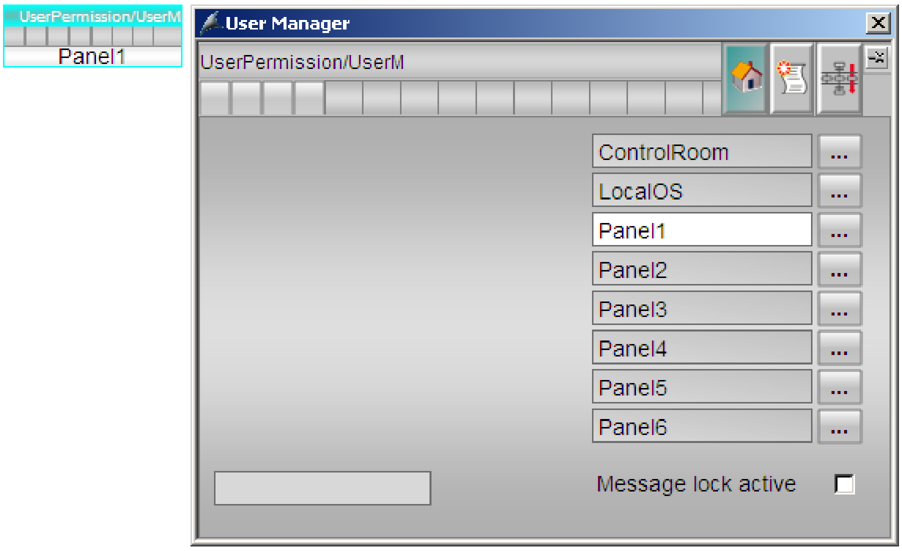
In order to avoid inconsistencies caused by control operations from various stations, a multi-user operation function has been introduced. Among other things this concept includes a 2-stage hierarchy of operations. The levels 1 and 2 are reserved for operation with the OS in the control room. Levels 3 to 8 are intended for operation via operator panels at the plant. If required, the 8 operating levels may also be configured individually.

Figure 3-2



The different operating levels are managed by the block "S7UsrM" which is integrated in the control program and interconnected with the technological function blocks and the interface blocks of the operator panel. Selection of the operating level is effected via the faceplate in the OS or a connection to the input "KS_DEVICE". When the operator management function is activated, the faceplate can be operated only by a logged-in user with authorization for "higher process controlling" operations. Selection of the operating level on the operator panel is not possible.

Figure 3-3



In the OS, the operating level is defined by the internal tag “@Permission”. The operating level of the operator panels is configured in the interface blocks at the input “OP_PERMIS”.

The configuration settings for multi-user operation are described later in this document.

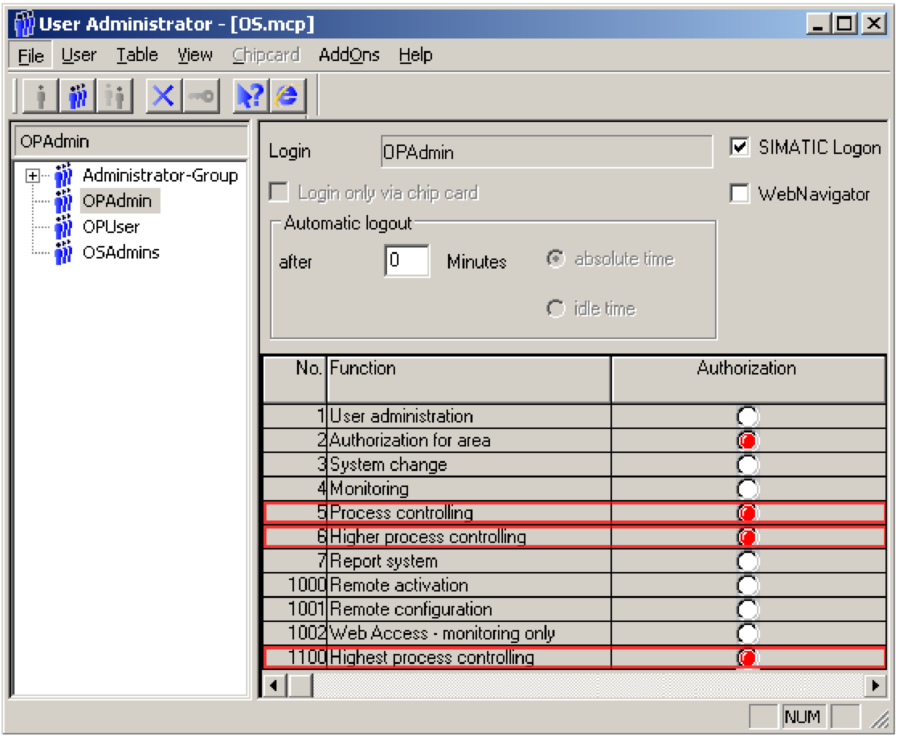
3.4 User authorization and user specification

The PCS 7 faceplates for process control provide three standard authorization levels as follows:

- Level 5: Process controlling
  Allows normal control operations, e.g. switchover between manual/automatic mode.
- Level 6: Higher process controlling
  Allows control operations with significant effects on the process, e.g. adjusting the limit values of a controller.
- Level 1100: Highest process controlling
  Allows the simulation of process values or the release of operating equipment for maintenance purposes.

The levels 1101 and 1102 are free project-specific operator authorizations.

Figure 3-4



OPAdmin

- Administrator-Group
- OPAdmin
- OPUser
- OSAdmins

Login: OPAdmin

☑ SIMATIC Logon

☐ Login only via chip card

☐ WebNavigator

Automatic logout

after 0 Minutes

● absolute time

○ idle time

No.	Function	Authorization
1	User administration	○
2	Authorization for area	●
3	System change	○
4	Monitoring	○
5	Process controlling	●
6	Higher process controlling	●
7	Report system	○
1000	Remote activation	○
1001	Remote configuration	○
1002	Web Access - monitoring only	○
1100	Highest process controlling	●

For further details on user hierarchies in PCS 7, please refer to the “PCS 7 OS process control” manual.
http://support.automation.siemens.com/WW/view/en/36195920

The faceplate configurations of the operator panel only allow operations up to level 5. Higher and highest process control operations can be performed only at the OS.

Further restriction of the access to IL-faceplates on the operator panel can be defined by specifying the user administration rights for WinCC flexible Runtime. Some devices offer SIMATIC Logon as an option for user administration.

The following devices recommended for the use of the Industry Library also support SIMATIC Logon:

- MP 277, MP 377, SIMATIC IPC
- PC platforms with WinCC flexible advanced 2008 Runtime

For all other operator panels without SIMATIC Logon the definition of a local user administration system is required. The procedure for configuring of a user administration system is described in the documentation for PCS 7, WinCC flexible and SIMATIC Logon.

3.5 Signaling concept

The technological blocks of the IL for S7 library use the system function block “Alarm_DQ” to signal group errors to the OS and to the operator panel. The alarm messages can be acknowledged at both stations.

In contrast to the PCS 7 standard-type alarm signaling block “Alarm_8P”, the signaling function “Alarm_DQ” is also available on S7-300 CPUs and on HMI devices based on WinCC flexible. However, only one message per call can be generated with “Alarm_DQ”.

Type and size of alarms and messages:

Table 3-4

	PCS 7 with S7-400	PCS 7 with S7-300
Alarm signaling block	ALARM_8P / ALARM_DQ	ALARM_DQ
Number of messages	up to 1000	up to 300

For further information on alarm signaling blocks, please refer to the manual “System Software for S7-300/400 System and Standard Functions”.

http://support.automation.siemens.com/WW/view/en/44240604

Note

The generation of operator messages via the operator panel is presently not supported. If this functionality is required, it must be configured separately, e.g. with the help of an alarm signaling block which registers the switching signals from the interface block of the operator panel and issues an operator message.

4 Basic Work Steps

Note The work steps described in this chapter are not relevant to the communication configurations on the S7-400 CPU. Please continue with chapter 5 “Communication Management in the S7-400 CPU”.

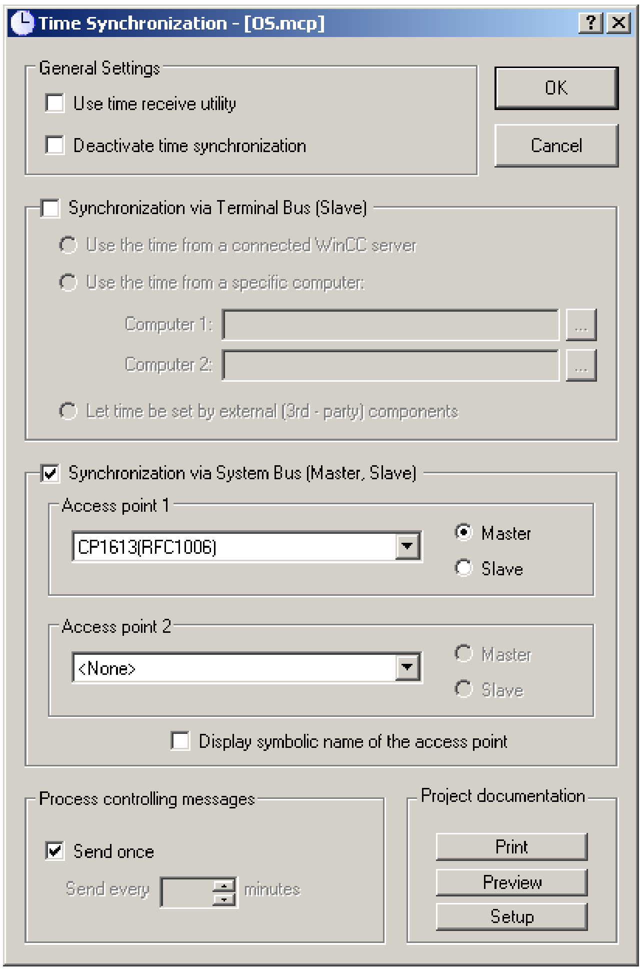
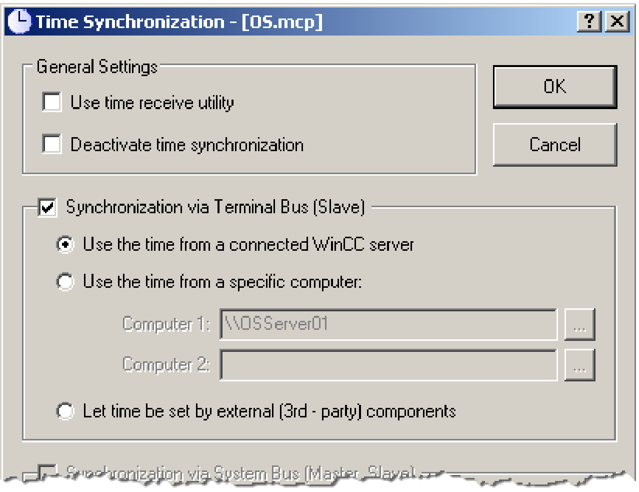
4.1 Configurations for time synchronization

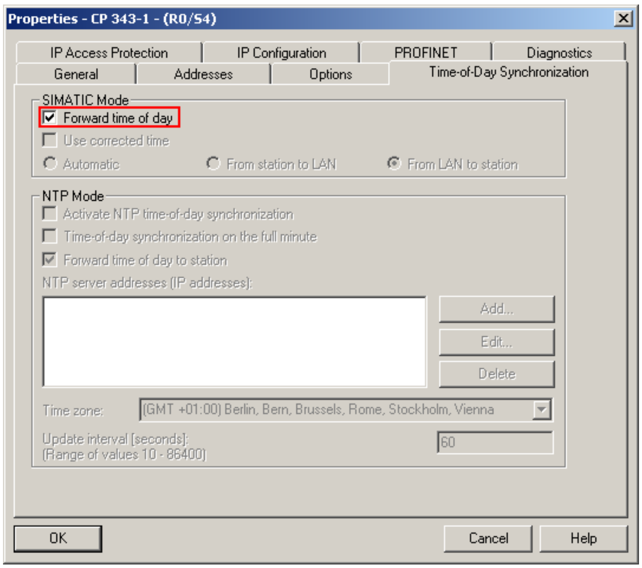
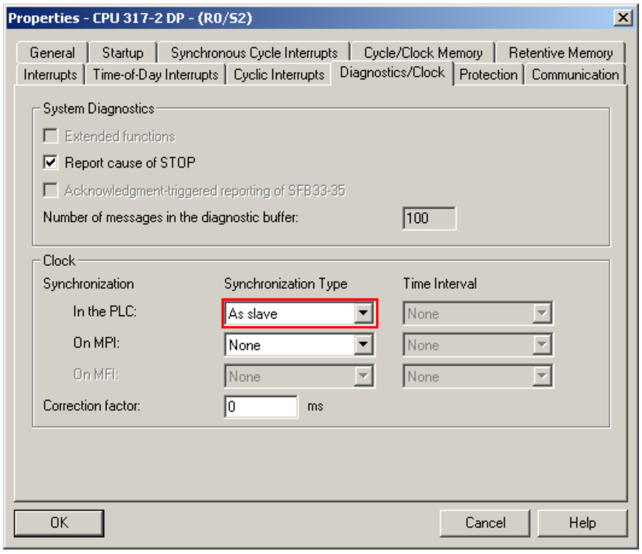
Note The following chapter refers only to time synchronization in the SIMATIC environment. Please note that time synchronization in the SIMATIC environment may not be supported by all components. Automation systems with an integrated Ethernet interface can be synchronized only by using an NTP procedure. Since the operator panels support none of these two procedures they must be synchronized with the help of an area pointer, whereby the current system time is to be supplied by the controller.

The OS server is defined as the time master. OS clients and automation systems are defined as time slaves. Operator Panels becomes synchronized with the AS via an area pointer. The clock time of the entire system should be set to UTC (coordinated universal time).

4.1.1 Time synchronization in the SIMATIC environment

Table 4-1

No.	Action	Display
1.	**Setting the time master** • Open the OS project of the server to be configured as time master. • Open the “Time Synchronization” editor. • Select the option “Synchronization via System Bus”. • Choose an access point and define it as “Master”. Select the CP of your system bus. • If required, you can define a further access point as “Master”. • Save your changes and download the OS.	 Time Synchronization - [OS.mcp] General Settings □ Use time receive utility □ Deactivate time synchronization □ Synchronization via Terminal Bus (Slave) ○ Use the time from a connected WinCC server ○ Use the time from a specific computer: Computer 1: Computer 2: ○ Let time be set by external (3rd - party) components ☑ Synchronization via System Bus (Master, Slave) Access point 1: CP1613(RFC1006) ● Master ○ Slave Access point 2: <None> ○ Master ○ Slave □ Display symbolic name of the access point Process controlling messages ☑ Send once Send every minutes Project documentation
2.	**Configuring time synchronization for OS clients** • Open, one-by-one, the projects of all OS clients. • Open the “Time Synchronization” editor. • Select the option “Synchronization via Terminal Bus”. • Select the option “Use the time from a connected WinCC server”. • Save your changes and download the OS.	 Time Synchronization - [OS.mcp] General Settings □ Use time receive utility □ Deactivate time synchronization ☑ Synchronization via Terminal Bus (Slave) ● Use the time from a connected WinCC server ○ Use the time from a specific computer: Computer 1: \\OSServer01 Computer 2: ○ Let time be set by external (3rd - party) components □ Synchronization via System Bus (Master, Slave)

No.	Action	Display
3.	**Time synchronization in the automation systems** • Open the hardware configuration of the AS to be configured. CP settings: • Open the Properties dialog for the CP and select the tab “Time-of-Day Synchronization”. • Select the option “Forward time of day“ in the SIMATIC Mode field. CPU settings: • Open the Properties dialog for the CPU and select the tab “Diagnostics/Clock”. • Choose the synchronization type “As slave”. • Save and compile your changes and download the hardware configuration.	 Properties - CP 343-1 - (R0/S4) IP Access Protection | IP Configuration | PROFINET | Diagnostics | General | Addresses | Options | Time-of-Day Synchronization SIMATIC Mode ☑ Forward time of day □ Use corrected time ○ Automatic ○ From station to LAN ● From LAN to station NTP Mode □ Activate NTP time-of-day synchronization □ Time-of-day synchronization on the full minute ☑ Forward time of day to station NTP server addresses (IP addresses): Add... Edit... Delete Time zone: (GMT +01:00) Berlin, Bern, Brussels, Rome, Stockholm, Vienna Update interval [seconds]: (Range of values 10 - 86400) 60 OK Cancel Help  Properties - CPU 317-2 DP - (R0/S2) General | Startup | Synchronous Cycle Interrupts | Cycle/Clock Memory | Retentive Memory | Interrupts | Time-of-Day Interrupts | Cyclic Interrupts | Diagnostics/Clock | Protection | Communication System Diagnostics □ Extended functions ☑ Report cause of STOP □ Acknowledgment-triggered reporting of SFB33-35 Number of messages in the diagnostic buffer: 100 Clock Synchronization | Synchronization Type | Time Interval In the PLC: As slave | None On MPI: None | None On MFI: None | None Correction factor: 0 ms OK Cancel Help

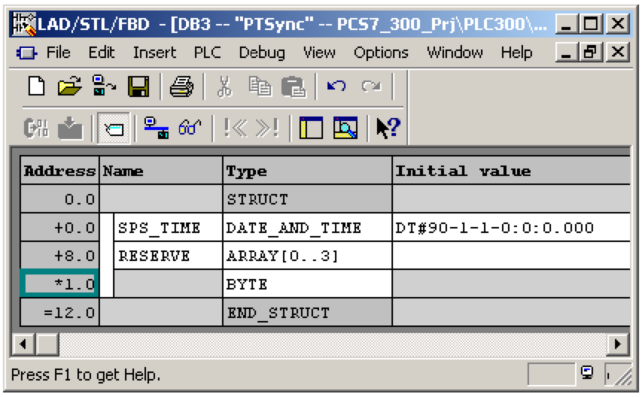
4.1.2 Time synchronization of the operator panels

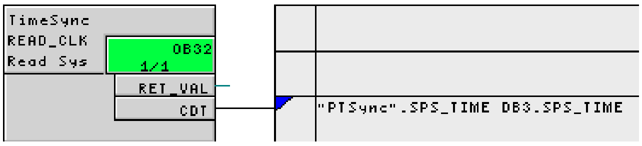
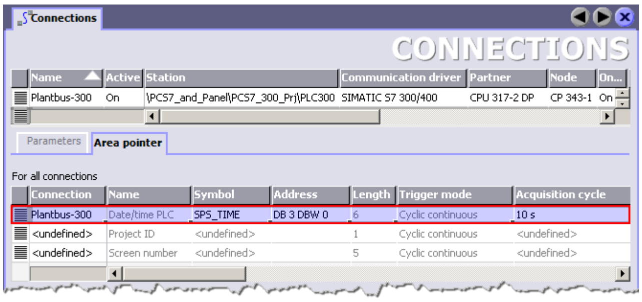
Time synchronization of the operator panels requires:

- the system function block “READ_CLK”
- a 12-byte data block
- the area pointer “Date/time PLC” for the operator panel

Table 4-2

<table>
<tr><th>No.</th><th>Action</th><th>Display</th></tr>
<tr><td>1.</td><td><b>Creating a data block</b>

The panel area pointer requires a data area of 12 bytes. Create a data block with the following parameters:<ul><li>1 parameter of type “DATE_AND_TIME”</li><li>4 reserve bytes</li></ul></td><td>
LAD/STL/FBD - [DB3 -- "PTSync" -- PCS7_300_Prj\PLC300\...
<table>
<tr><th>Address</th><th>Name</th><th>Type</th><th>Initial value</th></tr>
<tr><td>0.0</td><td></td><td>STRUCT</td><td></td></tr>
<tr><td>+0.0</td><td>SPS_TIME</td><td>DATE_AND_TIME</td><td>DT#90-1-1-0:0:0.000</td></tr>
<tr><td>+8.0</td><td>RESERVE</td><td>ARRAY[0..3]</td><td></td></tr>
<tr><td>*1.0</td><td></td><td>BYTE</td><td></td></tr>
<tr><td>=12.0</td><td></td><td>END_STRUCT</td><td></td></tr>
</table>
Press F1 to get Help.</td></tr>
<tr><td>2.</td><td><b>Reading the clock time</b>

The clock time is read with the help of the SFC “READ_CLK” and written to the DB parameter “SPS_TIME”.<ul><li>Create a new CFC chart.</li><li>Add the SFC “READ_CLK” to the chart.</li><li>Link the “CDT” output to the parameter “SPS_TIME” of the DB.</li><li>Compile and download the control program.</li></ul><b>Note</b>
It is sufficient if the block is called at 1-second intervals (OB32).</td><td>
</td></tr>
<tr><td>3.</td><td><b>Configuring the area pointer</b>
<ul><li>Open the Communication folder in the WinCC flexible project and select the CPU connection to be used for synchronization.</li><li>Open the “Area pointer” tab.</li><li>Assign the area pointer “Date/time PLC” to the connection and address of the previously created data block.</li><li>Transfer the WinCC flexible project.</li></ul></td><td>
Connections
CONNECTIONS
<table>
<tr><th>Name</th><th>Active</th><th>Station</th><th>Communication driver</th><th>Partner</th><th>Node</th><th>On...</th></tr>
<tr><td>Plantbus-300</td><td>On</td><td>\PCS7_and_Panel\PCS7_300_Prj\PLC300</td><td>SIMATIC S7 300/400</td><td>CPU 317-2 DP</td><td>CP 343-1</td><td>On</td></tr>
</table>

For all connections
<table>
<tr><th>Connection</th><th>Name</th><th>Symbol</th><th>Address</th><th>Length</th><th>Trigger mode</th><th>Acquisition cycle</th></tr>
<tr><td>Plantbus-300</td><td>Date/time PLC</td><td>SPS_TIME</td><td>DB 3 DBW 0</td><td>6</td><td>Cyclic continuous</td><td>10 s</td></tr>
<tr><td>&lt;undefined&gt;</td><td>Project ID</td><td>&lt;undefined&gt;</td><td></td><td>1</td><td>Cyclic continuous</td><td>&lt;undefined&gt;</td></tr>
<tr><td>&lt;undefined&gt;</td><td>Screen number</td><td>&lt;undefined&gt;</td><td></td><td>5</td><td>Cyclic continuous</td><td>&lt;undefined&gt;</td></tr>
</table></td></tr>
</table>

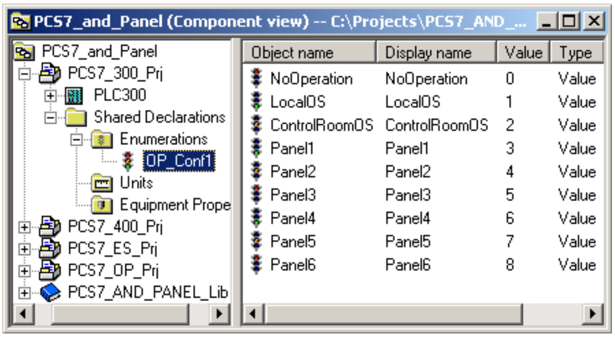
4.2 Configurations for multi-user operation

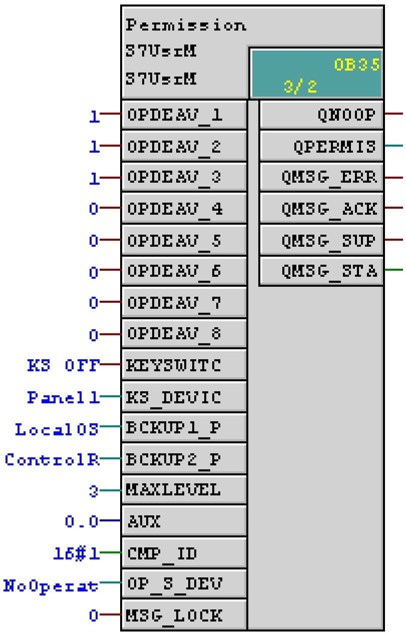
This chapter describes all steps required to configure the multi-user operation function.

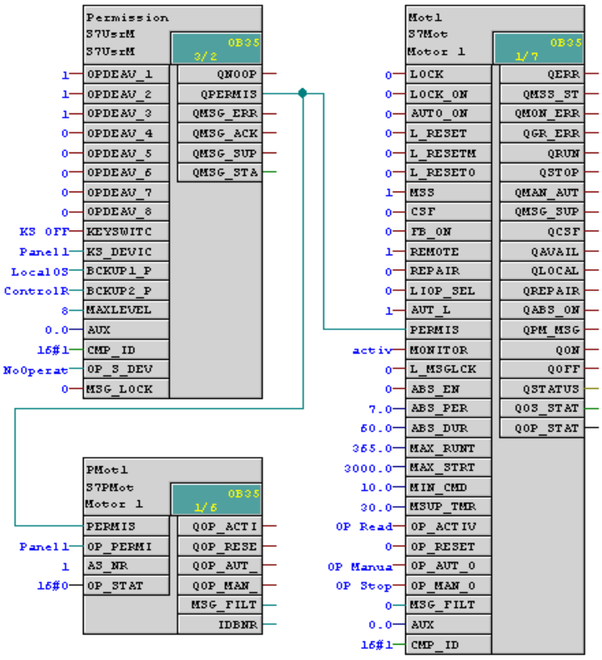
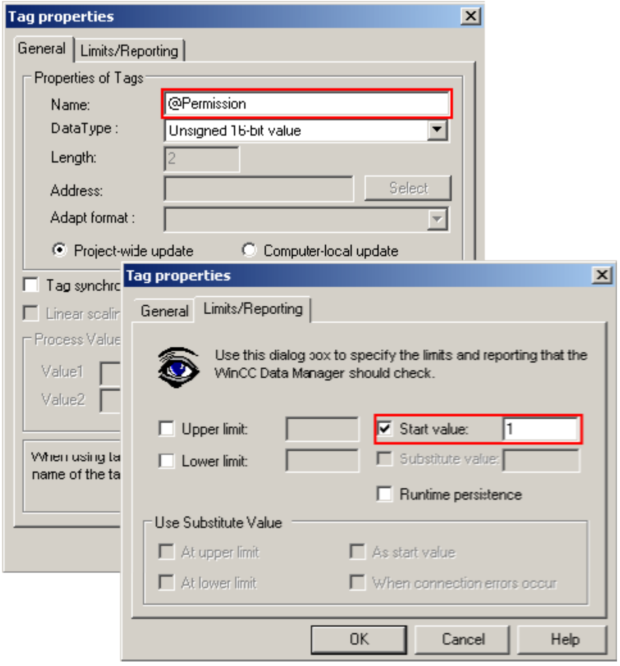
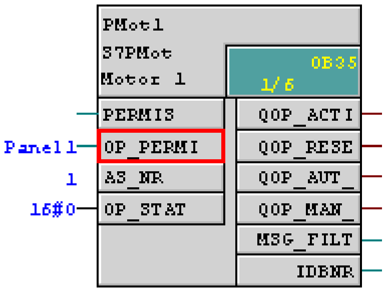
Proceed as follows.

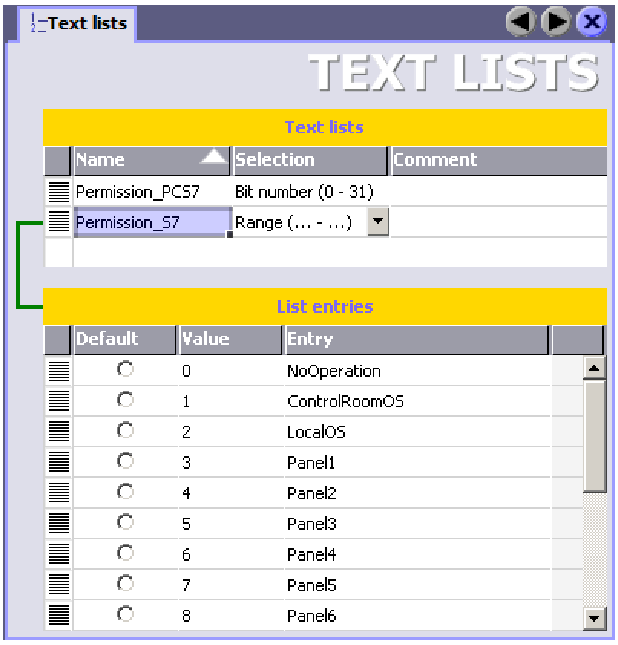
Table 4-3

<table>
<tr><th>No.</th><th>Action</th><th>Display</th></tr>
<tr><td>1.</td><td><b>Create an enumeration list</b>

In the operating-level connections, the blocks of the IL S7 library are preconfigured in the enumeration list “OP_Conf1”. Here, the block connections in the CFC chart are not just displayed as numeric values, but show the name of the operating level. In addition, text references to the OS are generated and indicated by a symbol and in the faceplate. The names of the operating levels can be freely selected.
• Create the enumeration list “OP_Conf1”.
• Configure the list objects for the values 0-8.
• Specify the value “0” as “not operated”.
• Define the names of the operating levels using the values “1-8”.</td><td>
PCS7_and_Panel (Component view) -- C:\Projects\PCS7_AND_...
PCS7_and_Panel
PCS7_300_Prj
PLC300
Shared Declarations
Enumerations
OP_Conf1
Units
Equipment Prope
PCS7_400_Prj
PCS7_ES_Prj
PCS7_OP_Prj
PCS7_AND_PANEL_Lib
<table>
<tr><th>Object name</th><th>Display name</th><th>Value</th><th>Type</th></tr>
<tr><td>NoOperation</td><td>NoOperation</td><td>0</td><td>Value</td></tr>
<tr><td>LocalOS</td><td>LocalOS</td><td>1</td><td>Value</td></tr>
<tr><td>ControlRoomOS</td><td>ControlRoomOS</td><td>2</td><td>Value</td></tr>
<tr><td>Panel1</td><td>Panel1</td><td>3</td><td>Value</td></tr>
<tr><td>Panel2</td><td>Panel2</td><td>4</td><td>Value</td></tr>
<tr><td>Panel3</td><td>Panel3</td><td>5</td><td>Value</td></tr>
<tr><td>Panel4</td><td>Panel4</td><td>6</td><td>Value</td></tr>
<tr><td>Panel5</td><td>Panel5</td><td>7</td><td>Value</td></tr>
<tr><td>Panel6</td><td>Panel6</td><td>8</td><td>Value</td></tr>
</table></td></tr>
<tr><td>2.</td><td><b>Configuring “S7UsrM”</b>

• Add the function block “S7UsrM” to a new or previous CFC chart.
• Define whether a level can be selected or not at the inputs “OPDEAV_1” to “OPDEAV_8”.
• Use the input “KEYSWITCH” to define whether the operating level shall be selected at the OS or be predefined at the block input “KS_DEVICE”.
• Use the input “MAXLEVEL” to define the number of the levels activated above.</td><td>
Permission
S7UsrM
S7UsrM
OB35
3/2
1 — OPDEAV_1
1 — OPDEAV_2
1 — OPDEAV_3
0 — OPDEAV_4
0 — OPDEAV_5
0 — OPDEAV_6
0 — OPDEAV_7
0 — OPDEAV_8
KS OFF — KEYSWITC
Panel1 — KS_DEVIC
LocalOS — BCKUP1_P
ControlR — BCKUP2_P
3 — MAXLEVEL
0.0 — AUX
16#1 — CMP_ID
NoOperat — OP_S_DEV
0 — MSG_LOCK
QNOOP
QPERMIS
QMSG_ERR
QMSG_ACK
QMSG_SUP
QMSG_STA</td></tr>
</table>

No.	Action	Display
3.	**Interconnecting S7UsrM** Interconnect the output "QPERMIS" with the input "PERMIS" of the panel interface block and with the input "PERMIS" of the technological block. Repeat this step for each block used in your configuration.	 Permission S7UsrM S7UsrM OB35 3/2 1 OPDEAV_1 1 OPDEAV_2 1 OPDEAV_3 0 OPDEAV_4 0 OPDEAV_5 0 OPDEAV_6 0 OPDEAV_7 0 OPDEAV_8 KS OFF KEYSWITC Panel1 KS_DEVIC Local0S BCKUP1_P ControlR BCKUP2_P 8 MAXLEVEL 0.0 AUX 16#1 CMP_ID NoOperat OP_S_DEV 0 MSG_LOCK QNOOP QPERMIS QMSG_ERR QMSG_ACK QMSG_SUP QMSG_STA Mot1 S7Mot Motor 1 OB35 1/7 0 LOCK 0 LOCK_ON 0 AUTO_ON 0 L_RESET 0 L_RESETM 0 L_RESETO 1 MSS 0 CSF 0 FB_ON 1 REMOTE 0 REPAIR 0 LIOP_SEL 1 AUT_L PERMIS activ MONITOR 0 L_MSGLCK 0 ABS_EN 7.0 ABS_PER 60.0 ABS_DUR 365.0 MAX_RUNT 3000.0 MAX_STRT 10.0 MIN_CMD 30.0 MSUP_TMR OP Read OP_ACTIV 0 OP_RESET OP Manua OP_AUT_O OP Stop OP_MAN_O 0 MSG_FILT 0.0 AUX 16#1 CMP_ID QERR QMSS_ST QMON_ERR QGR_ERR QRUN QSTOP QMAN_AUT QMSG_SUP QCSF QAVAIL QLOCAL QREPAIR QABS_ON QPM_MSG QON QOFF QSTATUS QOS_STAT QOP_STAT PMot1 S7PMot Motor 1 OB35 1/6 PERMIS Panel1 OP_PERMI 1 AS_NR 16#0 OP_STAT QOP_ACTI QOP_RESE QOP_AUT_ QOP_MAN_ MSG_FILT IDBNR
4.	**Defining the operating levels** The OS operating level is defined by the internal tag "@Permission". The operating level for the operator panel is defined at the interface block of the relevant operator panel. - Create an "@Permission" tag for each OS and define the value of the relevant operating level as start value. - Specify the operating level of the panel at the input "OP_PERMIS" of the interface block. **Note** The faceplate can be operated in the OS, if the "PERMIS" value of the technological block complies with the tag value "@Permission". The faceplate can be operated on the operator panel, if the values shown at the inputs "PERMIS" and "OP_PERMIS" at the interface block are identical.	Tag properties General | Limits/Reporting Properties of Tags Name: @Permission DataType : Unsigned 16-bit value Length: 2 Address: Select Adapt format : ● Project-wide update ○ Computer-local update □ Tag synchro □ Linear scalir Process Value Value1 Value2 When using la name of the ta Tag properties General | Limits/Reporting Use this dialog box to specify the limits and reporting that the WinCC Data Manager should check. □ Upper limit: □ Lower limit: ☑ Start value: 1 □ Substitute value: □ Runtime persistence Use Substitute Value □ At upper limit □ As start value □ At lower limit □ When connection errors occur OK Cancel Help  PMot1 S7PMot Motor 1 OB35 1/6 PERMIS Panel1 OP_PERMI 1 AS_NR 16#0 OP_STAT QOP_ACTI QOP_RESE QOP_AUT_ QOP_MAN_ MSG_FILT IDBNR

No.	Action	Display
5.	**Configuring the operating level texts for the operator panel** The insertion of IL faceplates is accompanied by the creation of text lists. These texts show the currently selected operating level in the faceplate. If you have named your operating levels differently, you can reconfigure these text lists accordingly. • Open the text lists of WinCC-flexible stored in the folder "Text and Graphics Lists". • Select the text list "Permission_S7". • Rename the operating levels in compliance with the corresponding values.	

Text lists

TEXT LISTS

Text lists

Name	Selection	Comment
Permission_PCS7	Bit number (0 - 31)	
Permission_S7	Range (... - ...)	

List entries

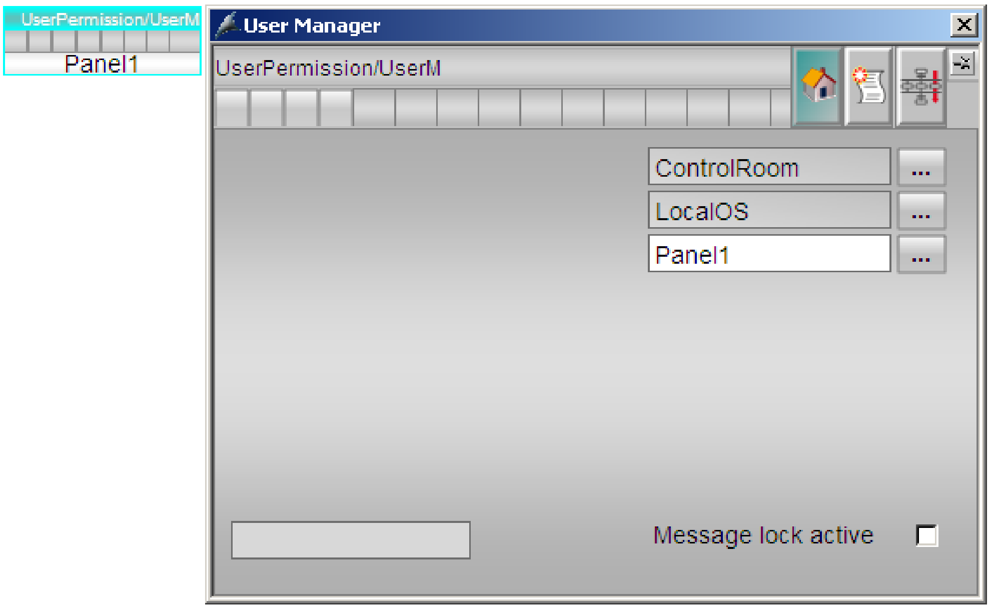
Default	Value	Entry
○	0	NoOperation
○	1	ControlRoomOS
○	2	LocalOS
○	3	Panel1
○	4	Panel2
○	5	Panel3
○	6	Panel4
○	7	Panel5
○	8	Panel6

Operation in Runtime

In Runtime you can select the operating level for the OS. If the user administration function is activated, this can only be performed by a registered user with "highest process controlling" authorization.

Only levels activated at the relevant block can be selected (inputs from "OPDEAV_1" to "OPDEAV_8").

Figure 4-1



UserPermission/UserM

Panel1

User Manager

UserPermission/UserM

ControlRoom ...

LocalOS ...

Panel1 ...

Message lock active □

Display in the faceplate

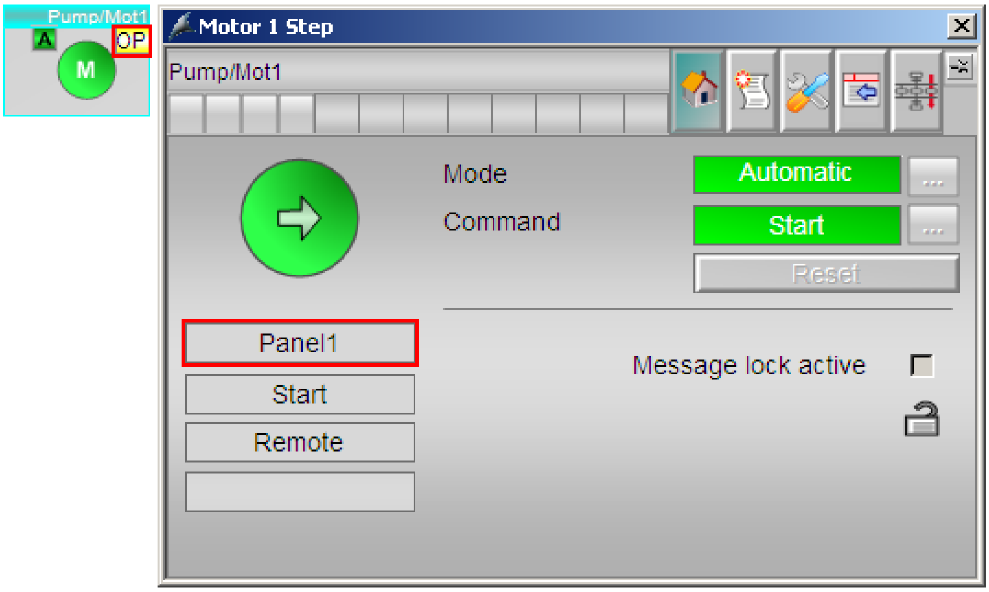
The block icon and the faceplate of the OS show the station with current operating priority.

The block icon can indicate the following:

- Level 1: no display – local OS
- Level 2: CR – control room
- Level 3-8: OP – operator panel

The faceplate shows the text stated in the enumeration list.

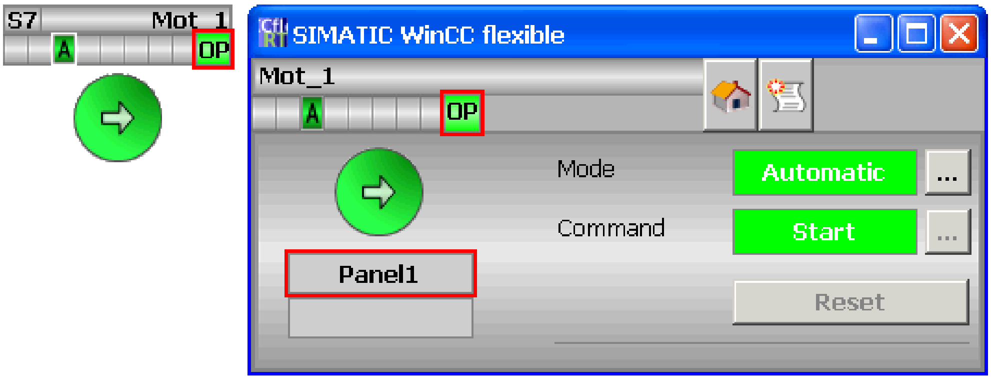
Figure 4-2



Mode Automatic ...
Command Start ...
Message lock active ☐

The selected authorization level is also shown in the faceplates of the operator panels in compliance with the OS display.

Figure 4-3



Mode Automatic ...
Command Start ...

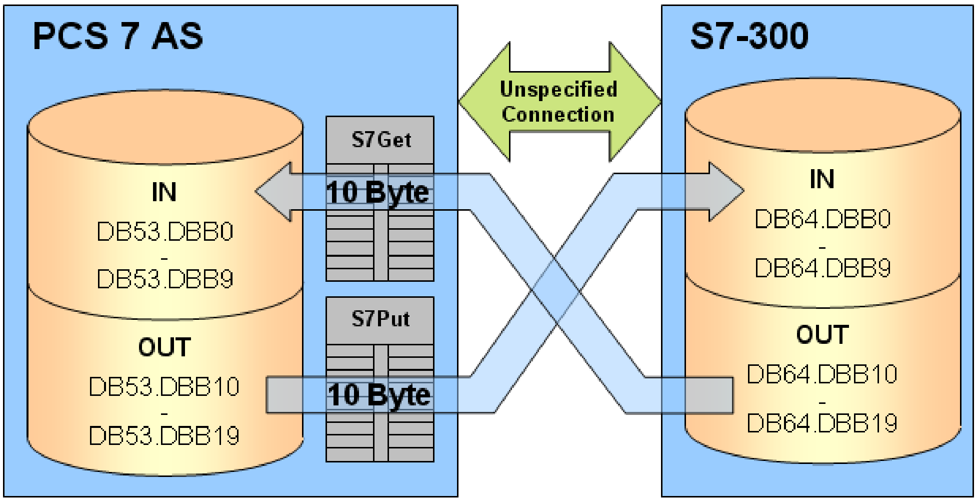
5 Communication Management in the S7-400 CPU

5.1 Description of the core functionality

In this example, the communication blocks “S7Put” and “S7Get” from the IL S7 library are used. These blocks are integrated in the S7 program of the automation system (AS) and supply a DB in the AS with all data required by the S7-300 CPU. This data is then processed individually. This chapter does not deal with the configurations for time synchronization and multi-user operation.

The values of a DB from an S7-300 CPU are read with the help of the “S7Get” block and then transferred to the DB of an S7-400 CPU integrated in the PCS 7 system. The data is processed in the S7-400 CPU and then retransferred to the DB of the S7-300 CPU via the block “S7Put”.

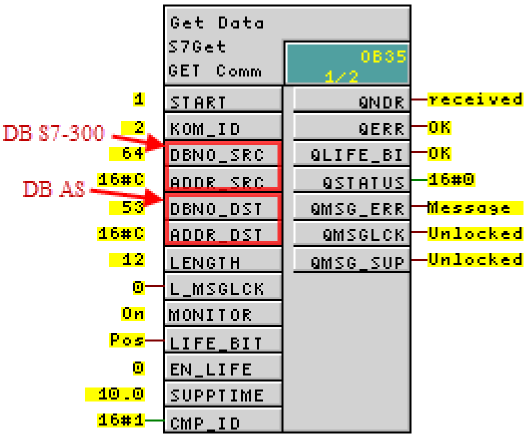
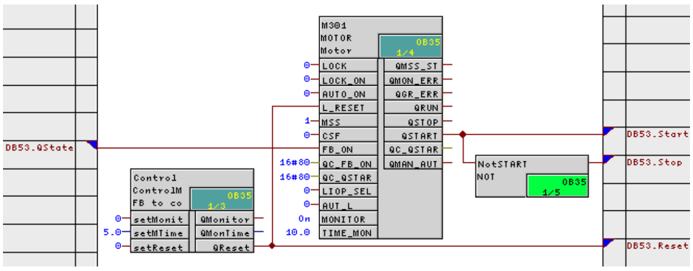
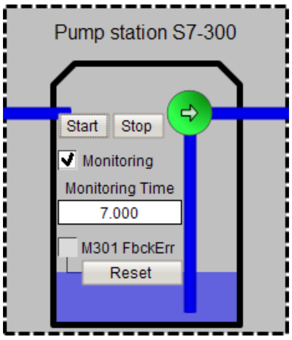
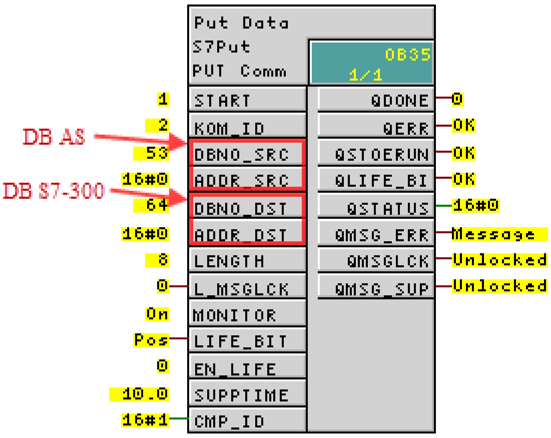
Figure 5-1



“S7Put” and “S7Get” are message-type blocks which transmit information about the connection status to the OS. Since the data is processed in the AS, the partial process unit can be easily visualized.

Principle of the core functionality

Table 5-1

No.	Action	Display
1.	The values from the DB of the S7-300 CPU are transferred to a DB of the AS with the help of the block "S7Get".	 Get Data S7Get GET Comm OB35 1/2 DB S7-300 DB AS 1 START 2 KOM_ID 64 DBNO_SRC 16#C ADDR_SRC 53 DBNO_DST 16#C ADDR_DST 12 LENGTH 0 L_MSGLCK On MONITOR Pos LIFE_BIT 0 EN_LIFE 10.0 SUPPTIME 16#1 CMP_ID QNDR received QERR OK QLIFE_BI OK QSTATUS 16#0 QMSG_ERR Message QMSGLCK Unlocked QMSG_SUP Unlocked
2.	The process values received are processed in the AS of the PCS 7 system and then available in the OS for display and operation.	 M301 MOTOR Motor OB35 1/4 0 LOCK 0 LOCK_ON 0 AUTO_ON L_RESET 1 MSS 0 CSF FB_ON 16#80 QC_FB_ON 16#80 QC_QSTAR 0 LIOP_SEL 0 AUT_L On MONITOR 10.0 TIME_MON QMSS_ST QMON_ERR QGR_ERR QRUN QSTOP QSTART QC_QSTAR QMAN_AUT Control ControlM FB to co OB35 1/3 0 setMonit 5.0 setMTime 0 setReset QMonitor QMonTime QReset NotSTART NOT OB35 1/5 DB53.QState DB53.Start DB53.Stop DB53.Reset  Pump station S7-300 Start Stop ☑ Monitoring Monitoring Time 7.000 □ M301 FbckErr Reset
3.	The block "S7Put" is used to retransfer the current values to the S7-300 CPU.	 Put Data S7Put PUT Comm OB35 1/1 DB AS DB S7-300 1 START 2 KOM_ID 53 DBNO_SRC 16#0 ADDR_SRC 64 DBNO_DST 16#0 ADDR_DST 8 LENGTH 0 L_MSGLCK On MONITOR Pos LIFE_BIT 0 EN_LIFE 10.0 SUPPTIME 16#1 CMP_ID QDONE 0 QERR OK QSTOERUN OK QLIFE_BI OK QSTATUS 16#0 QMSG_ERR Message QMSGLCK Unlocked QMSG_SUP Unlocked

Advantages of this solution

The partial process unit does not need to be reconfigured. Only the data block of the S7-300 CPU which contains the data to be processed must be known. There is no need to install a special block library on the configuration computer for the S7-300 CPU.

5.2 Hardware and software components used

This application was generated with the following components:

Hardware components

Table 5-2

Component	Qty.	MLFB/order number	Remarks
CPU 416-3 PN/DP	1	6ES7 416-3ER05-0AB0	
CPU 417-4H	1	6ES7 417-4HT14-0AB0	
CPU 317-2 DP	1	6ES7-317-2AJ10-0AB0	

Standard software components

Table 5-3

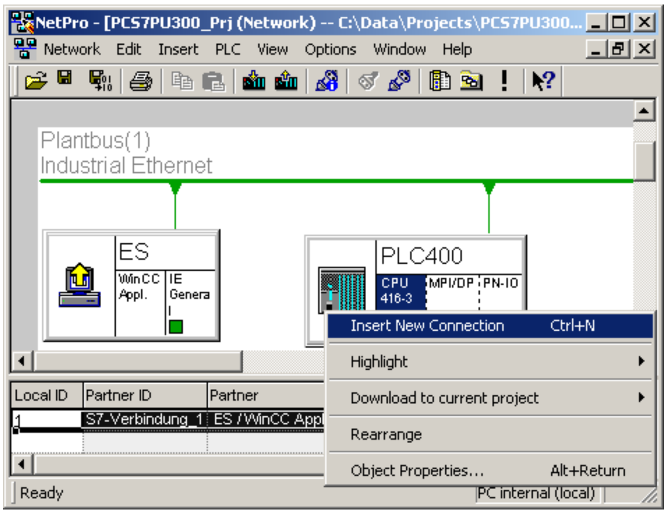
Component	Qty.	MLFB/order number	Remarks
PCS 7 V7.1 SP2	1	6ES7-658-5AC17-0YA5	
PCS 7 Industry Library V8.0	1	6DL5-410-8AA08-0YA0	

5.3 Configurations for an S7-400 Single CPU

5.3.1 Configuring an unspecified S7-connection

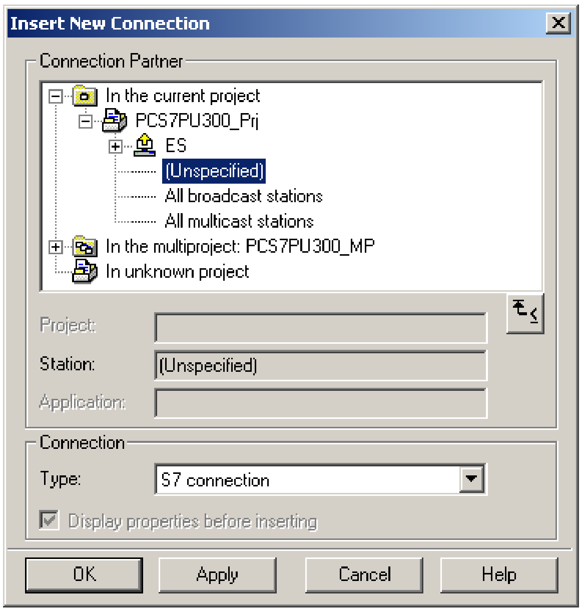
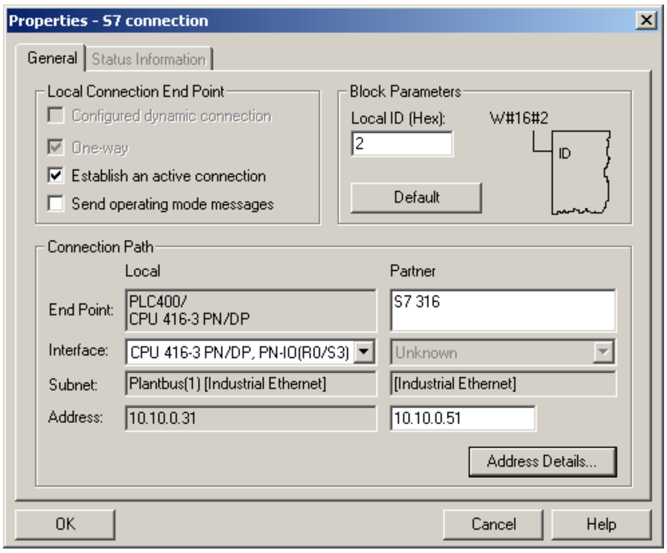
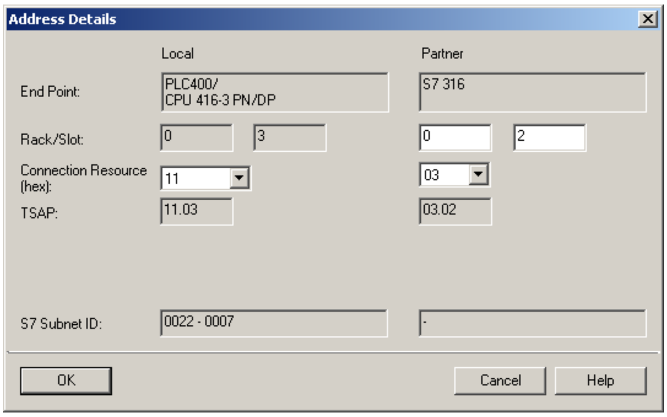
Before you begin, make sure that the two system networks are physically connected or that both CPUs are integrated in the same system network, respectively. The type of connection is of minor importance (Profibus or Industrial Ethernet), as it only differs with regard to the interface and address to be selected. Proceed as follows to produce a connection between the AS and an S7-300 CPU.

Table 5-4

No.	Action	Comments
1.	Open the “NetPro” editor for the current PCS-7 project.	Click the “Configure network” icon in the toolbar of the SIMATIC Manager.
2.	Specify the CPU to be connected with the S7-300 CPU and select “Insert New Connection” from the context menu.	

NetPro - [PCS7PU300_Prj (Network) -- C:\Data\Projects\PCS7PU300...
Network Edit Insert PLC View Options Window Help
Plantbus(1)
Industrial Ethernet
ES
WinCC Appl.
IE General
PLC400
CPU 416-3
MPI/DP
PN-IO
Insert New Connection Ctrl+N
Highlight
Download to current project
Rearrange
Object Properties... Alt+Return
Ready
PC internal (local)

Local ID	Partner ID	Partner
1	S7-Verbindung_1	ES / WinCC Appl

No.	Action	Comments
3.	Select the entry “(Unspecified)” as connection partner in the PCS-7 project. Select the connection type “S7 connection”. Click “OK” to confirm your settings.	 Insert New Connection Connection Partner In the current project PCS7PU300_Prj ES (Unspecified) All broadcast stations All multicast stations In the multiproject: PCS7PU300_MP In unknown project Project: Station: (Unspecified) Application: Connection Type: S7 connection ☑ Display properties before inserting OK Apply Cancel Help
4.	Open the dialog “Properties – S7 connection” and specify the name and network address of the connection partner. The local ID will be set automatically and does not need to be edited. This ID will later be used in the S7 program to specify the communication block. Click the “Address Details…” button.	 Properties - S7 connection General · Status Information Local Connection End Point ☐ Configured dynamic connection ☑ One-way ☑ Establish an active connection ☐ Send operating mode messages Block Parameters Local ID (Hex): 2 W#16#2 Default ID Connection Path: Local, Partner End Point: PLC400/ CPU 416-3 PN/DP; S7 316 Interface: CPU 416-3 PN/DP, PN-IO(R0/S3); Unknown Subnet: Plantbus(1) [Industrial Ethernet]; [Industrial Ethernet] Address: 10.10.0.31; 10.10.0.51 Address Details... OK Cancel Help
5.	Specify the rack number and the slot of the partner CPU in the "Address Details" dialog. The connection resource “03” is set by default and represents a one-side connection to an unspecified connection partner. Click OK to confirm your settings before you save/compile and download the connections.	 Address Details Local, Partner End Point: PLC400/ CPU 416-3 PN/DP; S7 316 Rack/Slot: 0 3; 0 2 Connection Resource (hex): 11; 03 TSAP: 11.03; 03.02 S7 Subnet ID: 0022 - 0007; - OK Cancel Help

5.3.2 Creating a data block

The data read from the DB are processed in the S7 program of the AS and the result will then be retransferred. To ensure correct data transfer from the S7-300 CPU you need a user-defined DB in the AS which complies with the data areas to be read/written in the S7-300 CPU.

ATTENTION

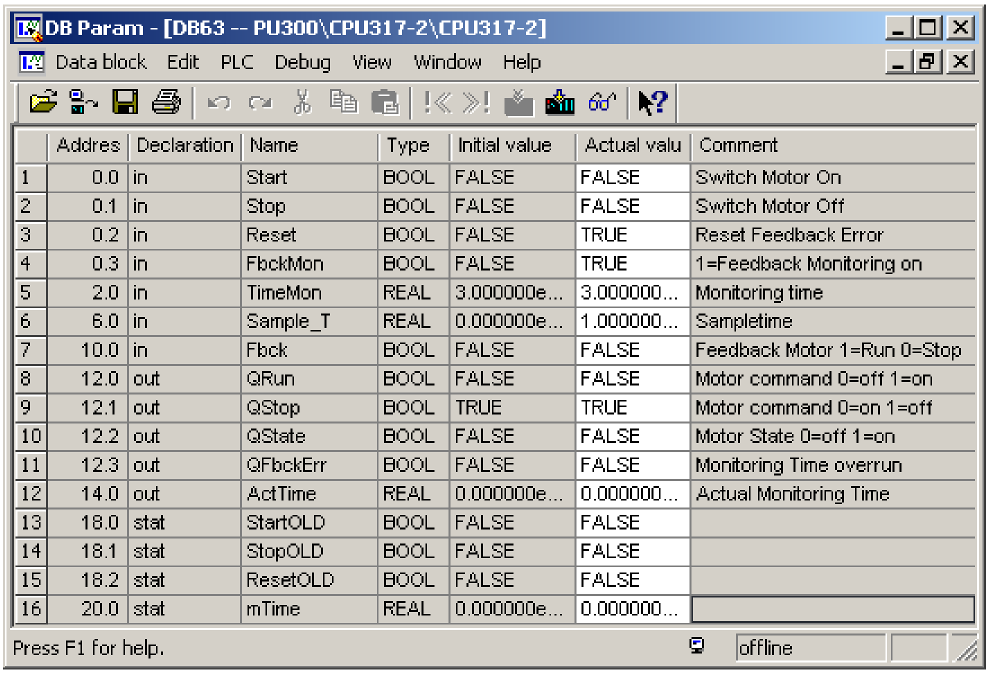
When using the block “S7Put” all data will written to the data blocks of the S7-300 CPU without further verification. Please take this in mind when configuring the program, as it may lead to unexpected system conditions.

It also may occur that the values will be overwritten if the target block is interconnected. Interconnected block inputs and outputs should not be overwritten by S7Put.

Example

A function block for the control of a simple motor has been integrated in the S7 program of the S7-300 CPU. The figure below shows the parameters of the associated data block.

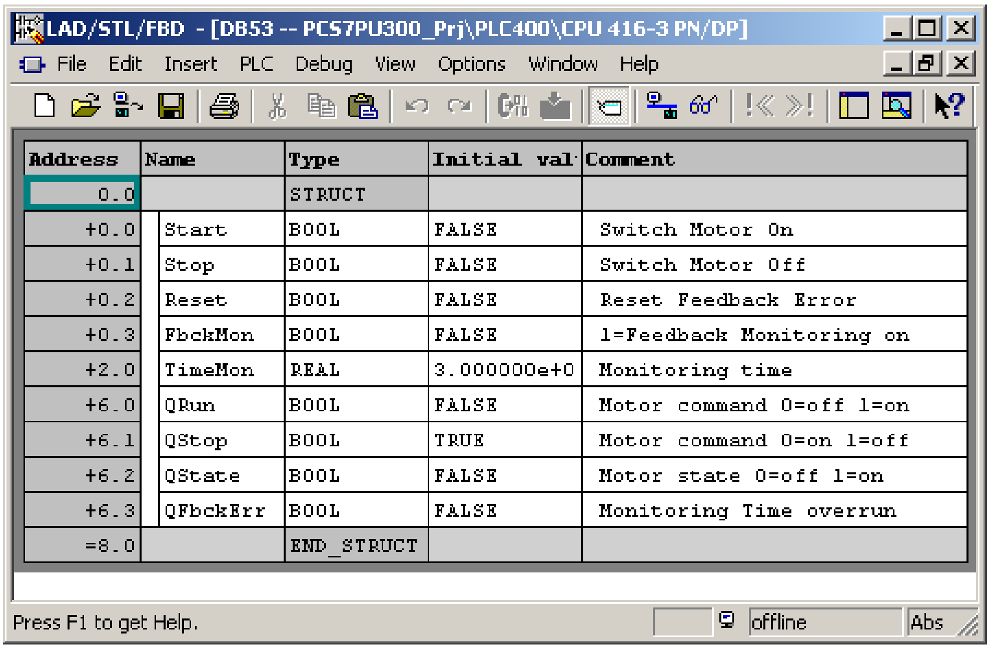
Figure 5-2



	Addres	Declaration	Name	Type	Initial value	Actual valu	Comment
1	0.0	in	Start	BOOL	FALSE	FALSE	Switch Motor On
2	0.1	in	Stop	BOOL	FALSE	FALSE	Switch Motor Off
3	0.2	in	Reset	BOOL	FALSE	TRUE	Reset Feedback Error
4	0.3	in	FbckMon	BOOL	FALSE	TRUE	1=Feedback Monitoring on
5	2.0	in	TimeMon	REAL	3.000000e...	3.000000...	Monitoring time
6	6.0	in	Sample_T	REAL	0.000000e...	1.000000...	Sampletime
7	10.0	in	Fbck	BOOL	FALSE	FALSE	Feedback Motor 1=Run 0=Stop
8	12.0	out	QRun	BOOL	FALSE	FALSE	Motor command 0=off 1=on
9	12.1	out	QStop	BOOL	TRUE	TRUE	Motor command 0=on 1=off
10	12.2	out	QState	BOOL	FALSE	FALSE	Motor State 0=off 1=on
11	12.3	out	QFbckErr	BOOL	FALSE	FALSE	Monitoring Time overrun
12	14.0	out	ActTime	REAL	0.000000e...	0.000000...	Actual Monitoring Time
13	18.0	stat	StartOLD	BOOL	FALSE	FALSE	
14	18.1	stat	StopOLD	BOOL	FALSE	FALSE	
15	18.2	stat	ResetOLD	BOOL	FALSE	FALSE	
16	20.0	stat	mTime	REAL	0.000000e...	0.000000...	

In this case, a DB with the required parameters must be created in the S7 program of the AS. The following DB was created in the AS program:

Figure 5-3


LAD/STL/FBD - [DB53 -- PCS7PU300_Prj\PLC400\CPU 416-3 PN/DP]
File Edit Insert PLC Debug View Options Window Help
Press F1 to get Help.
offline
Abs

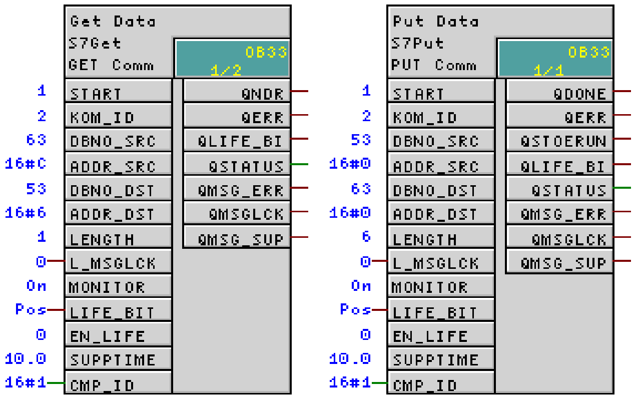
Address	Name	Type	Initial val	Comment
0.0		STRUCT		
+0.0	Start	BOOL	FALSE	Switch Motor On
+0.1	Stop	BOOL	FALSE	Switch Motor Off
+0.2	Reset	BOOL	FALSE	Reset Feedback Error
+0.3	FbckMon	BOOL	FALSE	1=Feedback Monitoring on
+2.0	TimeMon	REAL	3.000000e+0	Monitoring time
+6.0	QRun	BOOL	FALSE	Motor command 0=off 1=on
+6.1	QStop	BOOL	TRUE	Motor command 0=on 1=off
+6.2	QState	BOOL	FALSE	Motor state 0=off 1=on
+6.3	QFbckErr	BOOL	FALSE	Monitoring Time overrun
=8.0		END_STRUCT		

To enable motor control via the AS, the parameters “QRun, QStop, QState and QFbckErr” are captured with the help of “S7Get”, and the block “S7Put” is used to write the parameters “Start, Stop, Reset, FbckMon and TimeMon” to the S7-300 CPU.

5.3.3 Configuring communication blocks

In this example, communication between the CPUs is effected by using the blocks “S7Put” and “S7Get” from the Industry Library. The steps below describe how these blocks are integrated and configured in the S7 program of the AS.

Table 5-5

No.	Action	Comments
1.	Open an existing or new CFC chart. Add the blocks “S7Put” and “S7Get” to this chart. To minimize the communication load in the network, these blocks should be called by a cyclic interrupt OB with a larger cycle time (e.g. OB33 at 500ms).	 Get Data S7Get GET Comm OB33 1/2 1 START 2 KOM_ID 63 DBNO_SRC 16#C ADDR_SRC 53 DBNO_DST 16#6 ADDR_DST 1 LENGTH 0 L_MSGLCK On MONITOR Pos LIFE_BIT 0 EN_LIFE 10.0 SUPPTIME 16#1 CMP_ID QNDR QERR QLIFE_BI QSTATUS QMSG_ERR QMSGLCK QMSG_SUP Put Data S7Put PUT Comm OB33 1/1 1 START 2 KOM_ID 53 DBNO_SRC 16#0 ADDR_SRC 63 DBNO_DST 16#0 ADDR_DST 6 LENGTH 0 L_MSGLCK On MONITOR Pos LIFE_BIT 0 EN_LIFE 10.0 SUPPTIME 16#1 CMP_ID QDONE QERR QSTOERUN QLIFE_BI QSTATUS QMSG_ERR QMSGLCK QMSG_SUP
2.	Determine the local ID of the unspecified S7 connection and the DB number of the partner station.	The local ID is indicated in the “Block parameters” in the “Properties – S7 connections” dialog box. Open the project of the S7-300 CPU to determine the DB.
3.	Configure “S7Get” using the determined values.	In this example, the following values have been determined: • KOM_ID = 2 (connection ID) • DBNO_SRC = 63 (number of source DB) • ADDR_SRC = 16#C (data area starting from byte 12) • DBNO_DST = 53 (number of target DB) • ADDR_DST = 16#6 (data area starting from byte 6) • LENGTH = 1 (transmission of only 1 byte)
4.	Configure “S7Put” using the determined values.	In this example, the following values have been determined: • KOM_ID = 2 (connection ID) • DBNO_SRC = 53 (number of source DB) • ADDR_SRC = 16#0 (data area starting from byte 0) • DBNO_DST = 63 (number of target DB) • ADDR_DST = 16#0 (data area starting from byte 0) • LENGTH = 6 (transmission of 6 bytes)

Note

Data transmission with the communication blocks “S7Put” and “S7Get” can be performed only byte-wise, i.e. the source and target data bits are always transmitted in units of at least 8 bits.
When specifying the parameters for a DB, a new data type always starts at the byte boundary (byte, Bool and char) or at the word boundary (all other data types). If data transfer shall not include all Boolean parameters of the DB, just create a parameter of a different type in between.

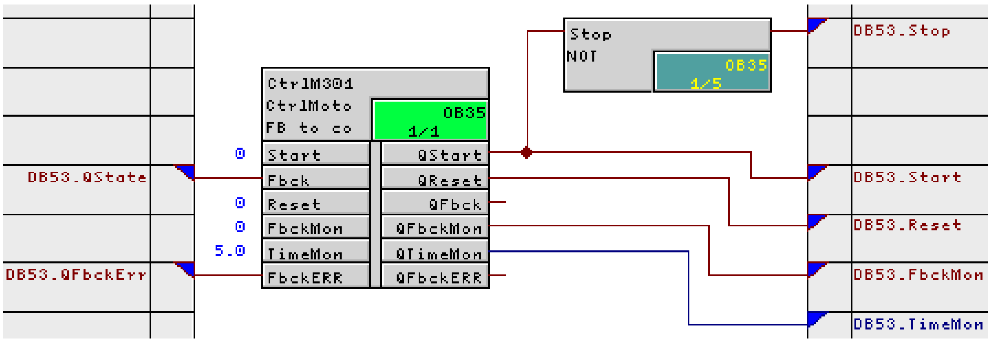
Example:
At first, two Boolean parameters with the start addresses 0.0 and 0.1 are created, followed by a “byte”-type parameter with the start address 1.0. Two further “Bool”-type parameters are provided with the addresses 2.0 and 2.1. The total length of the data block is 4 bytes.

5.3.4 Program logic and visualization

The DB parameters in the control program of the AS can be processed as desired. To enable operation in compliance with the PCS 7 standard, you should create a suitable function block with a corresponding OS faceplate. This procedure is described in the manual “Programming Instructions for Blocks” which is part of the PCS 7 documentation.

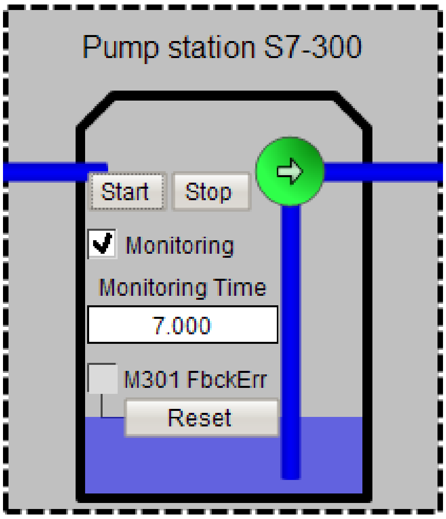
For this application a simple function block was created which is used to create variables for display and control in the OS and which issues a message, if an error occurs (feedback). This FB was then interconnected directly with the parameters of the user-defined DB. When included in the CFC chart, the block looks as follows:

Figure 5-4



The OS process image was provided with an area for the motor of the S7-300 CPU. This area could be displayed as follows:

Figure 5-5



As an alternative, a PCS 7 standard block for the program logic may be used (e.g. MOTOR – FB66). However, this block has the disadvantage of not providing all required control signals in the form of outputs, e.g. for error reset. Consequently, the "RESET" button of the OS faceplate cannot be used to reset an error in the S7-300 CPU. This function must then be implemented in a different way.

5.4 Configurations for an S7-400H CPU

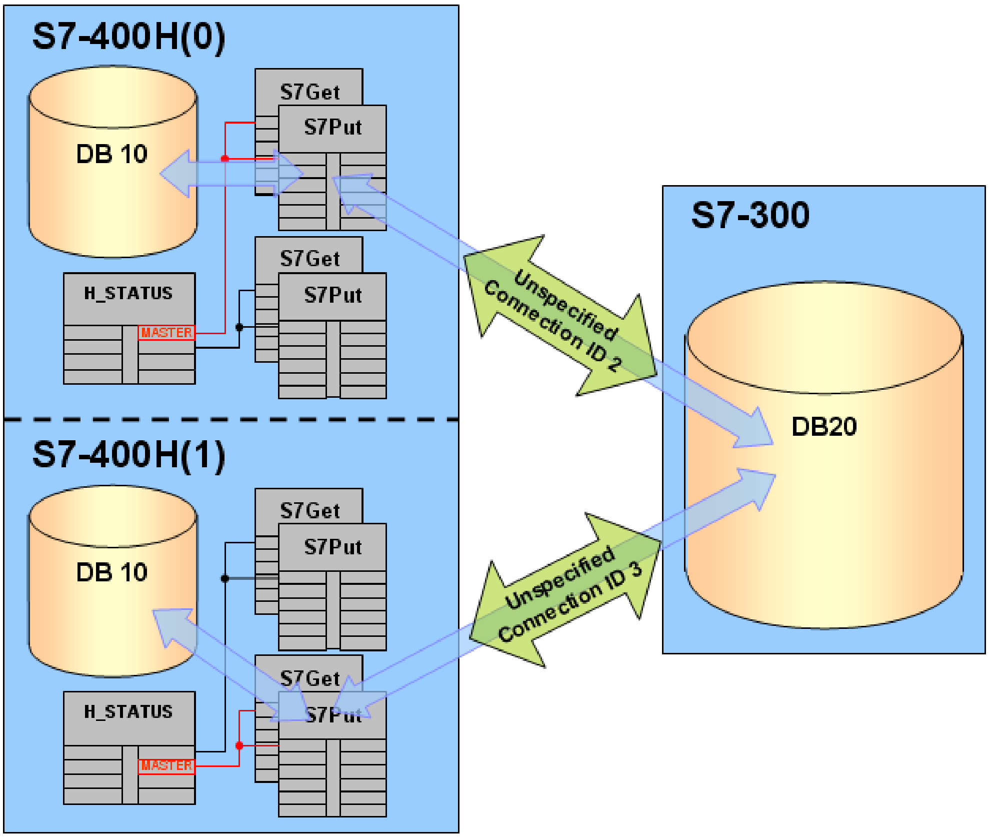
The H-system requires two unspecified connections, since the configuration of unspecified connections in the SIMATIC Manager does not provide any highly available connections. The blocks S7Put and S7Get are integrated in the S7-program in two-fold.

Data transmission is always executed by the station currently defined as master station. Switchover between the S7Put and S7Get blocks for data transfer is effected by means of “H_STATUS”. This block is available for download under

http://support.automation.siemens.com/WW/view/en/19537149

The function principle is illustrated in the following schematic diagram:

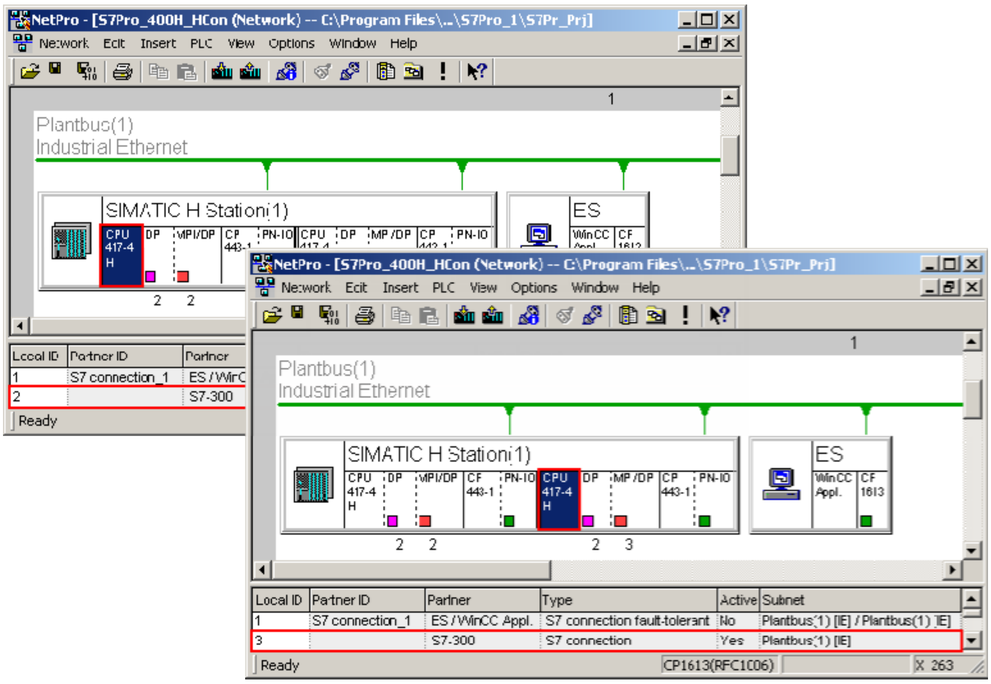
Figure 5-6



5.4.1 Configuring unspecified connections in the S7-400H CPU

Configure one connection each for the two S7-400H CPUs as described in chapter 5.3.1 “Configuring an unspecified S7-connection”. Configuration of the target station is identical for both connections. Each connection of the H-CPU is assigned to a separate connection ID.

Figure 5-7


NetPro - [S7Pro_400H_HCon (Network) -- C:\Program Files\...\S7Pro_1\S7Pr_Prj]
Network Edit Insert PLC View Options Window Help
1
Plantbus(1)
Industrial Ethernet
SIMATIC H Station(1)
CPU 417-4 H
DP
MPI/DP
CP 443-1
PN-IO
ES
WinCC Appl.
CP 1613
2
2
3

Local ID	Partner ID	Partner
1	S7 connection_1	ES / WirC
2		S7-300

Ready

Local ID	Partner ID	Partner	Type	Active	Subnet
1	S7 connection_1	ES / WinCC Appl.	S7 connection fault-tolerant	No	Plantbus(1) [IE] / Plantbus(1) [IE]
3		S7-300	S7 connection	Yes	Plantbus(1) [IE]

Ready CP1613(RFC1006) X 263

5.4.2 Creating a data block

Configuration of the data block is identical to the procedure for a Single CPU as described in chapter 5.3.2 “Creating a data block”.

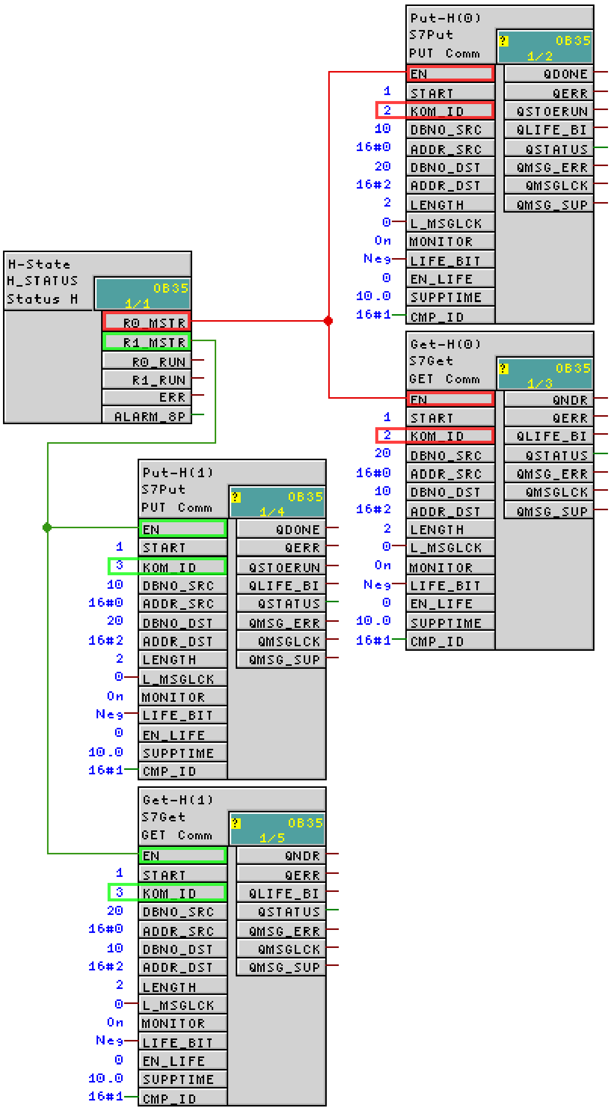
5.4.3 Configuring communication blocks in the S7-400H CPU

Since the parameter “KOM_ID” for the connection ID of the blocks “S7Put” and “S7Get” cannot be changed during operation, you need a double set of these blocks for the H-CPU.

“S7Put” and “S7Get” are configured in the same way for both connections; only the value of the parameter “KOM_ID” is different.

Changeover between the send and receive blocks is effected by the block “H_STATUS”. This block is used to output the operating states RUN/STOP and MASTER/RESERVE in the H-system.

Table 5-6

	Action	Comments
1.	Add the blocks H_STATUS, S7Put and S7Get to the CFC-chart.	
2.	Unhide the inputs “EN” of the function blocks S7Put and S7Get.	
3.	Configure the blocks as described in chapter 5.3.3 “Configuring communication blocks ”.	
4.	Then copy the blocks and specify the correct connection ID for the parameter “KOM_ID”.	
5.	Interconnect the output “R0_MSTR” of the status block to the “EN” inputs of the communication blocks for the CPU connection configured in rack 0.	
6.	Interconnect the output “R1_MSTR” of the status block to the “EN” inputs of the communication blocks for the CPU connection configured in rack 1.	

This configuration ensures that communication will be effected by the CPU currently defined as master.

5.4.4 Program logic and visualization

The remaining control program and OS configuration are the same as for a Single CPU. Please refer to chapter 5.3.4 “Program logic and visualization ”.

6 Configuration of the S7-300 CPU with PCS 7 Industry Library

6.1 Description of the core functionality

This package unit is integrated as a sub-project in the PCS 7 multi-project. Configuration of the S7-300 CPU is effected with the help of Industry Library and the process data is displayed at an OS and on a connected operator panel. The actual PCS 7 program on the S7-400 CPU remains unchanged.

The program of the S7-300 CPU is created with the help of blocks from the IL S7 library. To do so, the blocks for the technological functions (e.g. motor, valve, measuring point, …) are included and interconnected in a CFC-chart. Visualization on the operator panel is effected by integrating the associated interface blocks in the CFC-chart which are then interconnected with the relevant function blocks.

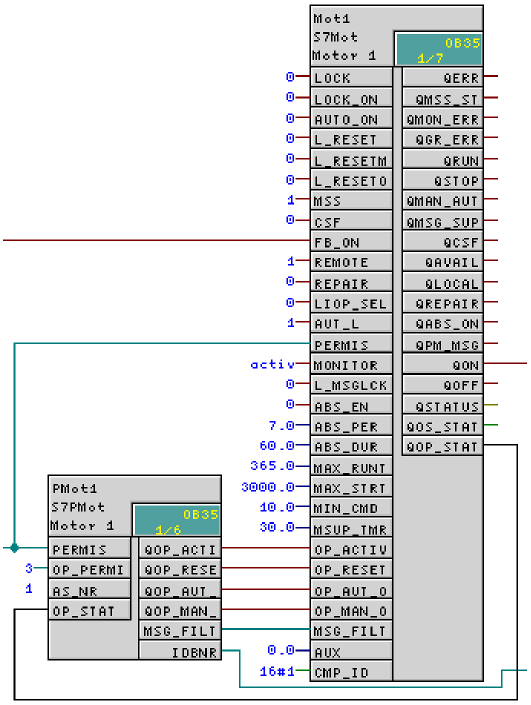
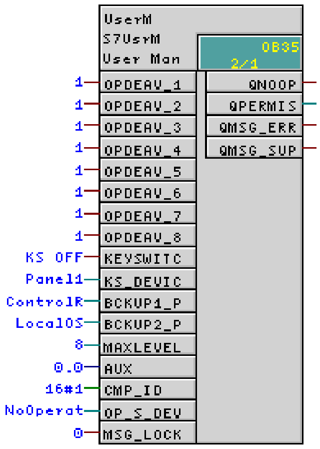
Furthermore, a multi-user operation function has been implemented. This function enables operation either via the OS or the operator panel.

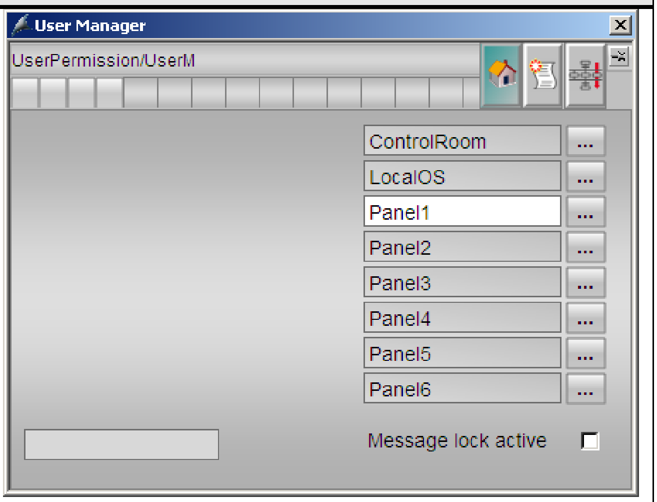
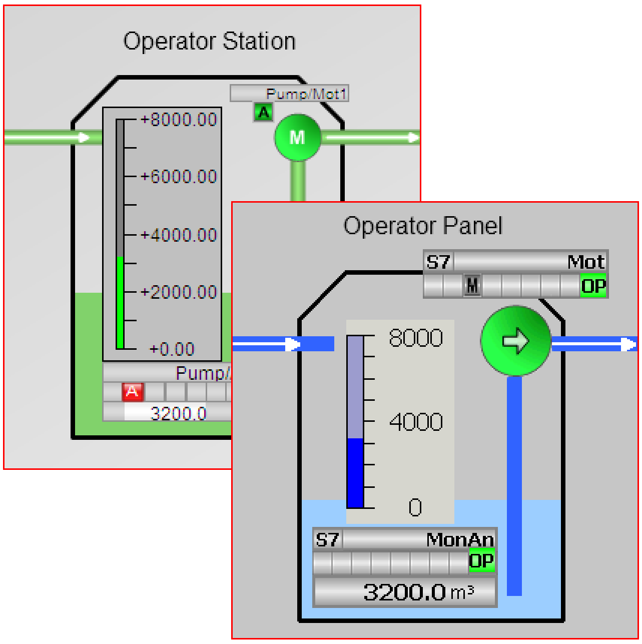
The technological blocks use ALARM_DQ (SFC 107) for the issue of group errors. These messages are then displayed and acknowledged at the OS and on the operator panel.

Note For example, the multi-user operation function follows a hierarchical two-stage station control concept which is broken down into 8 levels. Levels 1 and 2 are used for operation at the OS (control station) and levels 3 to 8 are intended for operation via the operator panels at the plant. If required, the 8 operating levels may also be configured individually.

Principle of the core functionality

Table 6-1

No.	Action	Display
1.	The actuators and sensors of the package unit are controlled by the program of the S7-300 CPU. This program is created with the help of the IL S7 library. The technological blocks (e.g. S7Mot) are assigned to interface blocks (e.g. S7PMot) for data transmission which are then integrated in the operator panel and interconnected.	
2.	The operating level is selected by the user manager block “S7UsrM” which is connected to the technological blocks and the interface blocks. It is then operated at the OS or by the block input “KS_DEVICE”. This block cannot be operated via the operator panel.	

No.	Action	Display
		 User Manager UserPermission/UserM ControlRoom LocalOS Panel1 Panel2 Panel3 Panel4 Panel5 Panel6 Message lock active
3.	The technological blocks of the IL S7 library are already provided with a faceplate for the PCS 7 OS. The package unit for visualization at the operator panel is configured with the help of the faceplate library included in WinCC flexible.	 Operator Station Pump/Mot1 A M +8000.00 +6000.00 +4000.00 +2000.00 +0.00 Pump/ A 3200.0 Operator Panel S7 Mot M OP 8000 4000 0 S7 MonAn OP 3200.0 m³

6.2 Hardware and software components

This application was generated with the following components:

Hardware components:

Table 6-2

Component	Qty.	MLFB/order number	Notes
CPU 416-3 PN/DP	1	6ES7 416-3ER05-0AB0	
CPU 317-2 DP	1	6ES7-317-2AJ10-0AB0	
Multipanel MP 377	1	6AV6-644-0AB01-2AX0	

Software components:

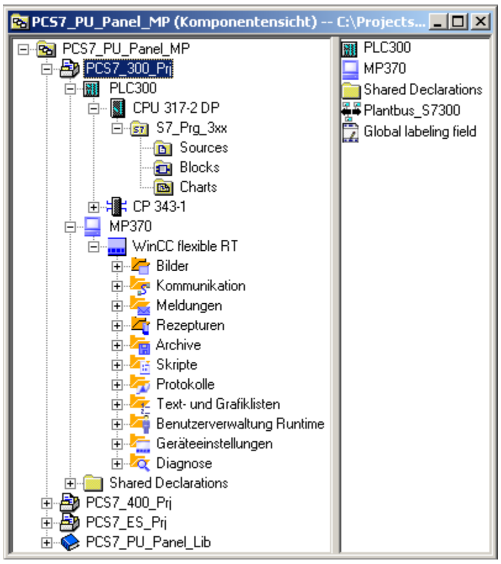
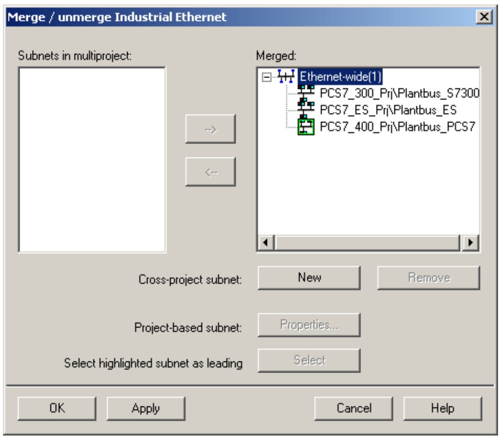
Table 6-3

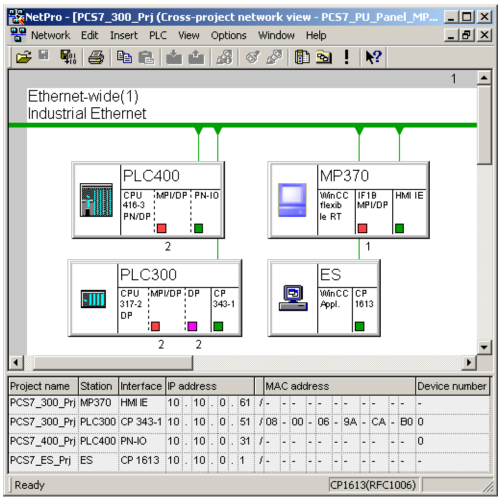
Component	Qty.	MLFB/order number	Notes
PCS 7 V8.0 Upd1	1	6ES7-658-1AF08-0YA6	
WinCC flexible 2008 SP3	1	6AV6-613-0AA51-3CA5	
PCS 7 Industry Library V8.0	1	6DL5-410-8AA08-0YA0	

6.3 Configuration and parameter assignment

6.3.1 Creating a new project for the package unit

Table 6-4

No.	Action	Display
1.	Open the PCS 7 multi-project. Use the menu commands “File > Multiproject > Add to Multiproject…” to create a project that includes the hardware and the program of the package unit. Configure the SIMATIC 300 station and the operator panel as required for your automation system. **Note** Before specifying the operator panel in the multi-project, make sure that WinCC flexible has been installed on the ES. As an alternative, the operator panel may also be configured on a separate computer on which WinCC flexible is installed.	 PCS7_PU_Panel_MP (Komponentensicht) -- C:\Projects... PCS7_PU_Panel_MP PCS7_300_Prj PLC300 CPU 317-2 DP S7_Prg_3xx Sources Blocks Charts CP 343-1 MP370 WinCC flexible RT Bilder Kommunikation Meldungen Rezepturen Archive Skripte Protokolle Text- und Grafiklisten Benutzerverwaltung Runtime Geräteeinstellungen Diagnose Shared Declarations PCS7_400_Prj PCS7_ES_Prj PCS7_PU_Panel_Lib PLC300 MP370 Shared Declarations Plantbus_S7300 Global labeling field
2.	To combine the subnetworks of the multi-project, open NetPro and select the menu commands “Edit > Merge / Unmerge Subnets > …”.	 Merge / unmerge Industrial Ethernet Subnets in multiproject: Merged: Ethernet-wide(1) PCS7_300_Prj\Plantbus_S7300 PCS7_ES_Prj\Plantbus_ES PCS7_400_Prj\Plantbus_PCS7 --> <-- Cross-project subnet: New Remove Project-based subnet: Properties... Select highlighted subnet as leading Select OK Apply Cancel Help

No.	Action	Display
3.	After having combined the subnets, select the menu commands “View > Cross-Project Network View” to display the cross-project network view. **Note** Connection to the S7-300 CPU cannot be established via the interface “PC internal (local)". This interface can be loaded only, if the corresponding PC/PC interface is set. This should also be considered when configuring the OS connection.	

Project name	Station	Interface	IP address		MAC address	Device number
PCS7_300_Prj	MP370	HMI IE	10 . 10 . 0 . 61	/	- - - - - - - - - - -	-
PCS7_300_Prj	PLC300	CP 343-1	10 . 10 . 0 . 51	/	08 - 00 - 06 - 9A - CA - B0	0
PCS7_400_Prj	PLC400	PN-IO	10 . 10 . 0 . 31	/	- - - - - - - - - - -	0
PCS7_ES_Prj	ES	CP 1613	10 . 10 . 0 . 1	/	- - - - - - - - - - -	-

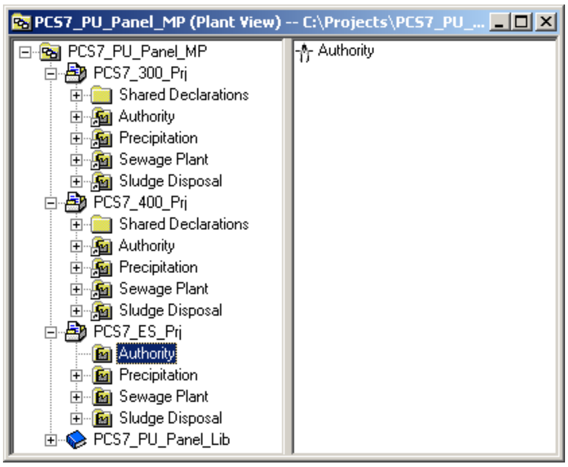
Ready CP1613(RFC1006)

6.3.2 Configuring the technological hierarchies

The technological hierarchy (TH) for each sub-project of the PCS 7 multiproject is set by default. Usually, the TH is established in the project of the OS and then transferred to the AS projects by using the function “Update in Multiproject”.

This chapter describes the multi-user operation function of the package unit in a separate OS area. The actual function of the package unit will be displayed in a previously defined area.

Table 6-5

No.	Action	Display
1.	Add a new hierarchy folder on the level of the OS area. Create a new OS image in this folder. Use the function “Update in Multiproject …” to transfer the technological hierarchy to the AS projects.	

6.3.3 Creating an S7 program

The steps below describe how you can use the IL S7 library to configure a motor (S7Mot), so that this motor will be displayed at the OS and on the operator panel. Interconnection with peripheral units or the configuration settings for further signal processing can be adapted to your requirements.

The motor block is interconnected with the block “S7UsrM”, so as to enable selection of the operating level (operator panel or OS),.

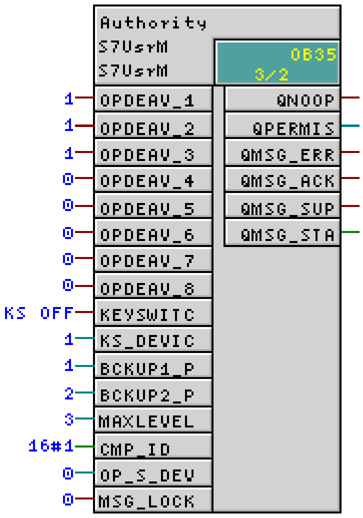
Table 6-6

<table>
<tr><th>No.</th><th>Action</th><th>Display</th></tr>
<tr><td>1.</td><td>Creating the multi-user operation function in the program of the S7-300 CPU.

- Open the technological view and select the project of the S7-300 CPU.
- Create a new CFC-chart in the operator authorization folder (in this example “Authority”).
- Add the block “S7UsrM” to this CFC-chart. This block will later be used for the interconnection of technological blocks.

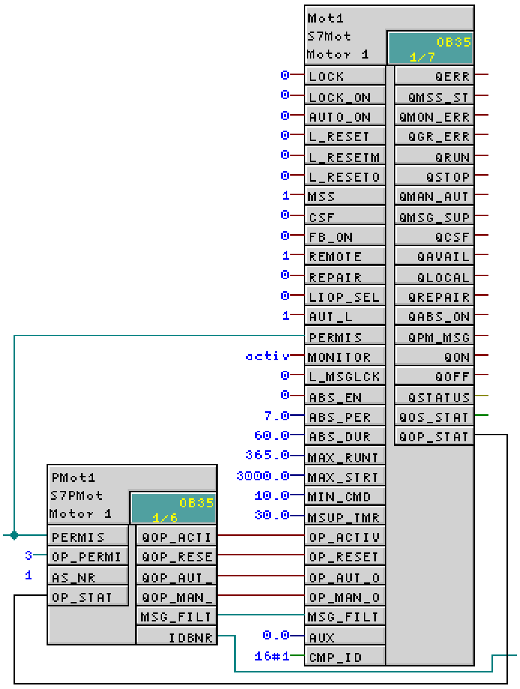
Note

The inputs “OPDEAV_1” to “OPDEAV_8” are set to activate the relevant operating levels which can then be selected at the OS.

In addition, the number of operating levels must be specified at the input “MAXLEVEL”.</td><td>
</td></tr>
<tr><td>2.</td><td>Creating the motor control function in the program of the S7-300 CPU.

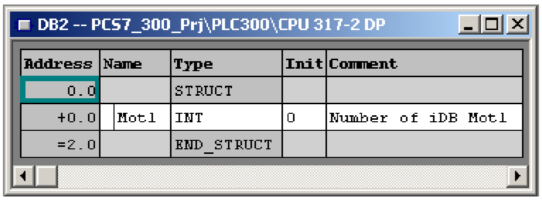
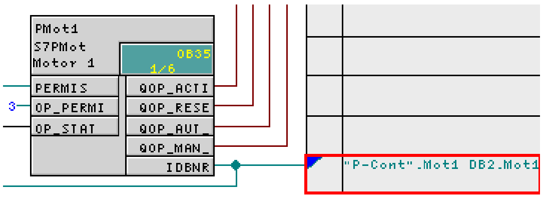
- Create a CFC-chart in the relevant folder of the TH.
- Add the blocks “S7Mot” and “S7PMot” to the chart.
- Interconnect the blocks as shown on the right.
- Set the input “OP_PERMIS” of “S7PMot” to operating level 3.

Note

The “PERMIS” inputs of the blocks are connected with the output “QPERMIS” of the “S7UsrM” block. The output “IDBNR” of the block “S7Pmot” is additionally connected with a data block parameter (see step 3).</td><td>
</td></tr>
</table>

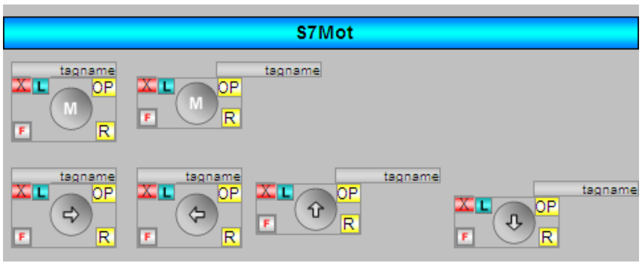
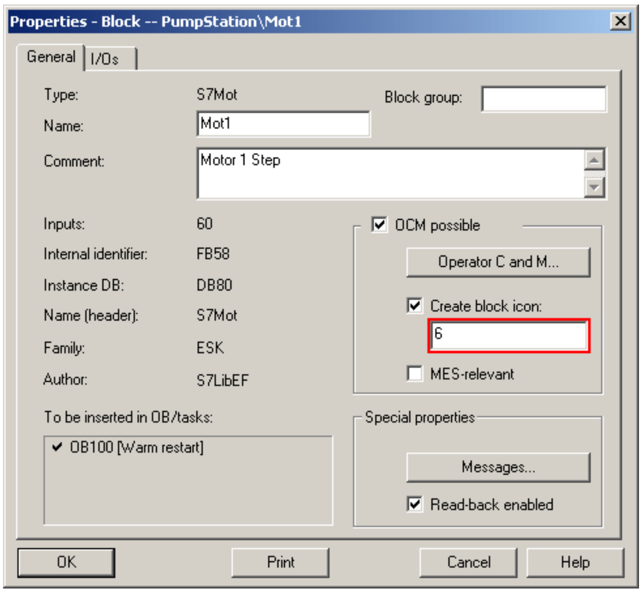
<table>
<tr><th>No.</th><th>Action</th><th>Display</th></tr>
<tr><td>3.</td><td>Creating a data block in the S7-300 program. This block will be used as a container for parameters and includes the instance data block numbers of the function blocks used.

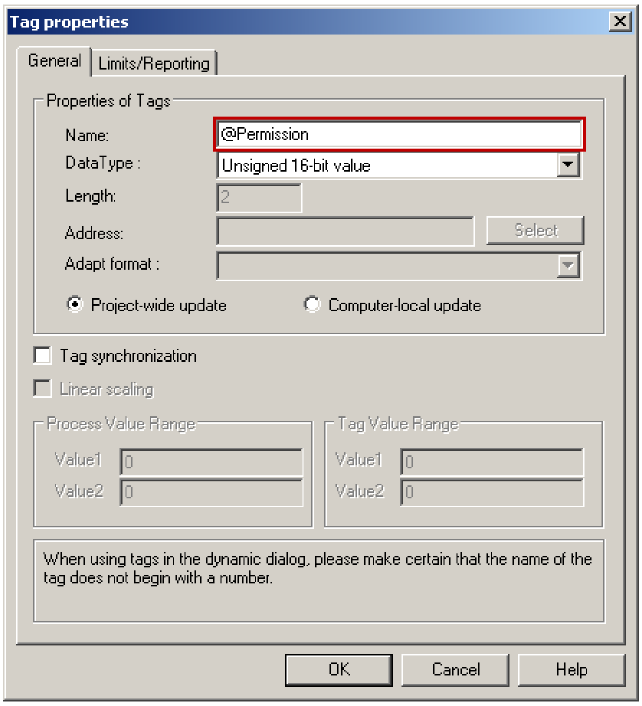
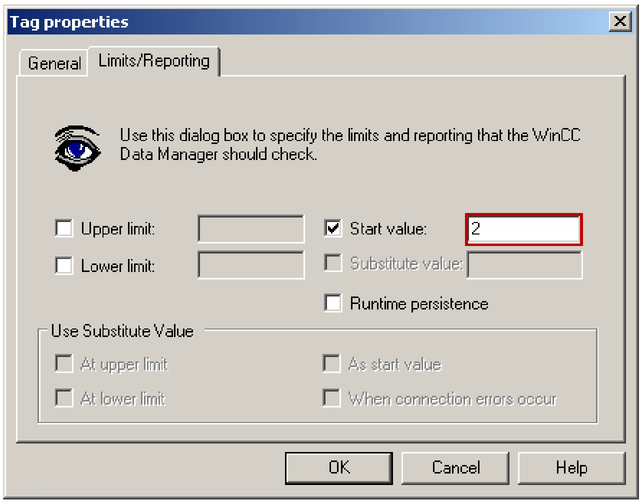
• Change to the “Blocks” folder of the S7-300 program.
• Create a new data block. Make sure to assign a unique number that will not be used by the CFC compiler. The settings for reserved areas and other applications can be found in the CFC Editor under “Options > Settings > Compile/Download…”.
• Specify a parameter of type “INT”. This parameter will then be provided with the DB number of the corresponding block.
• Connect the output “IDBNR” of the panel interface block with the parameter of the DB.

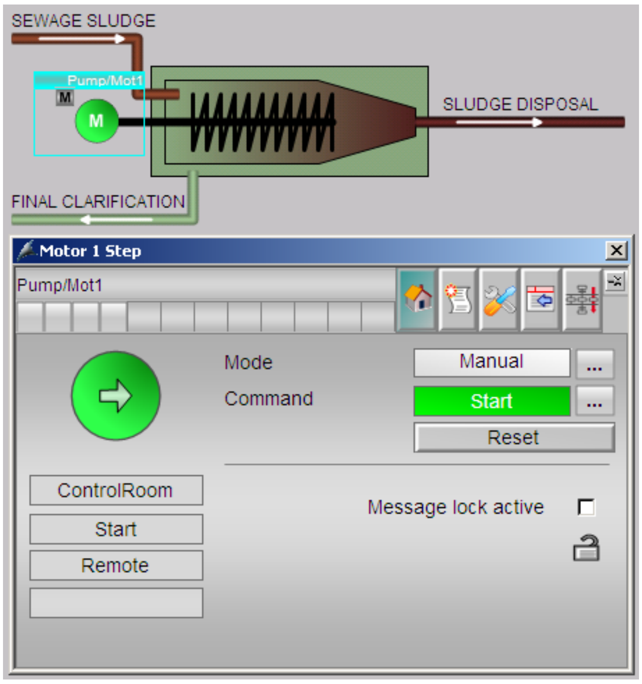
<b>Note</b>
You only need a single DB which includes all parameters for the IDB numbers of the panel blocks used.</td><td>
DB2 -- PCS7_300_Prj\PLC300\CPU 317-2 DP
<table>
<tr><th>Address</th><th>Name</th><th>Type</th><th>Init</th><th>Comment</th></tr>
<tr><td>0.0</td><td></td><td>STRUCT</td><td></td><td></td></tr>
<tr><td>+0.0</td><td>Mot1</td><td>INT</td><td>0</td><td>Number of iDB Mot1</td></tr>
<tr><td>=2.0</td><td></td><td>END_STRUCT</td><td></td><td></td></tr>
</table>

</td></tr>
<tr><td>4.</td><td>Compile the S7 program and download it to the controller.</td><td></td></tr>
</table>

6.3.4 Configuring the OS

Table 6-7

No.	Action	Display
1.	In the OS image, the icon “@S7Mot/1” for the “S7Mot” block will be included as a standard item. You may, however, select other icons from a series of block icons stored in the OS under “@PCS7Typicals_IL_S7”. Open the relevant OS image to determine the ID of the desired block icon and to see the associated properties. As an alternative, you can move your mouse pointer over an icon until the tooltip appears.	
2.	If you want to use another block icon, open the Properties dialog for the relevant technological block (e.g. “S7Mot”) in the CFC-chart and enter the ID of the desired icon in the input field “Create block icon”. If not specified otherwise, the “@S7Mot/1” icon will be used.	 Properties - Block -- PumpStation\Mot1 General | I/Os Type: S7Mot Block group: Name: Mot1 Comment: Motor 1 Step Inputs: 60 Internal identifier: FB58 Instance DB: DB80 Name (header): S7Mot Family: ESK Author: S7LibEF ☑ OCM possible Operator C and M... ☑ Create block icon: 6 □ MES-relevant To be inserted in OB/tasks: ✓ OB100 [Warm restart] Special properties Messages... ☑ Read-back enabled OK Print Cancel Help
3.	Select the function “Compile OS” and open the OS project.	

No.	Action	Display
4.	When using the mutli-station control function, each OS server must be assigned to a specific operating level. This is effected by means of the internal tag “@Permission”. The level of this tag is set as start value by default. If the operating level selected complies with the “@Permission” value, the interconnected blocks of the IL can be operated via this OS. • Create a new internal tag named “@Permission” and specify the data type as “Unsigned 16-bit value”. • Activate the option “Start value” and enter the operating level to be used for this OS.	 Tag properties General | Limits/Reporting Properties of Tags Name: @Permission DataType : Unsigned 16-bit value Length: 2 Address: Select Adapt format : ● Project-wide update ○ Computer-local update □ Tag synchronization □ Linear scaling Process Value Range: Value1 0, Value2 0 Tag Value Range: Value1 0, Value2 0 When using tags in the dynamic dialog, please make certain that the name of the tag does not begin with a number. OK Cancel Help  Tag properties General | Limits/Reporting Use this dialog box to specify the limits and reporting that the WinCC Data Manager should check. □ Upper limit: □ Lower limit: ☑ Start value: 2 □ Substitute value: □ Runtime persistence Use Substitute Value □ At upper limit □ As start value □ At lower limit □ When connection errors occur OK Cancel Help

No.	Action	Display
5.	Open the plant image which includes the generated block. Position the block icon and configure the plant image according to your requirements. The example in the picture on the right shows what the configured motor could look like in the plant image.	 SEWAGE SLUDGE Pump/Mot1 M M SLUDGE DISPOSAL FINAL CLARIFICATION Motor 1 Step Pump/Mot1 Mode Manual ... Command Start ... Reset ControlRoom Start Remote Message lock active □

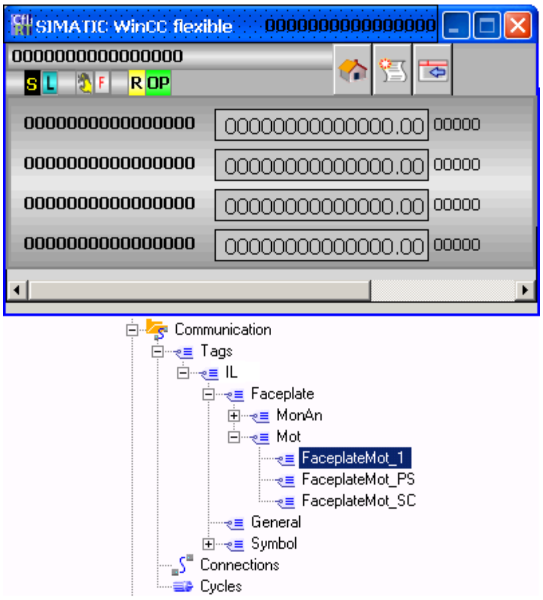
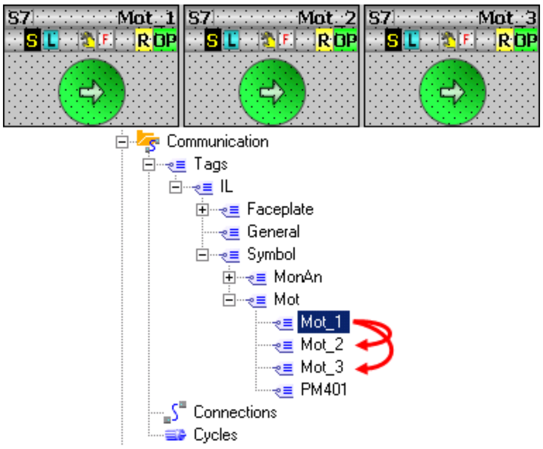
6.3.5 Configuring the operator panel

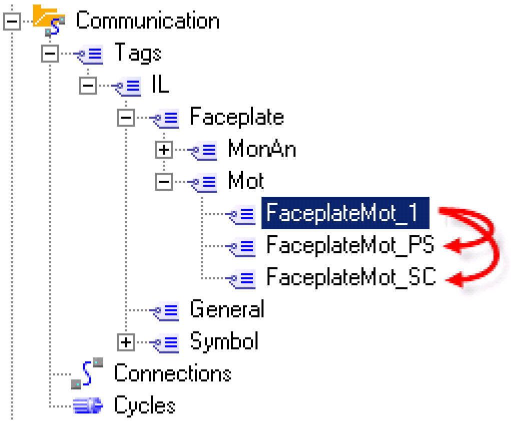
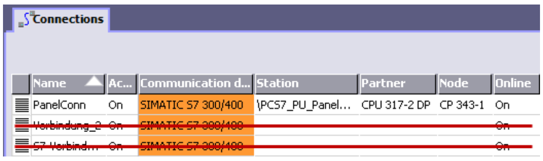
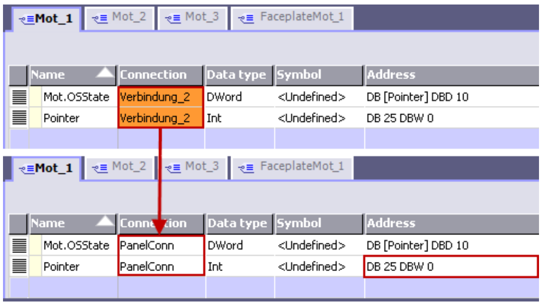
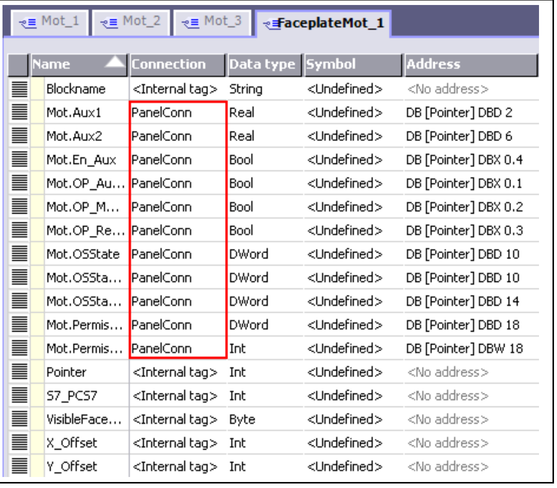
The Industry Library for WinCC flexible includes a preconfigured collection of faceplates. The block icons and faceplates are already provided with all necessary tags and functions. When placed in the plant image, not only the graphic objects, but also the associated tags, connections, graphics and text lists will be generated. You only need to adapt them to the project-specific situation. Please note that each faceplate and each block icon requires a separate tag folder.

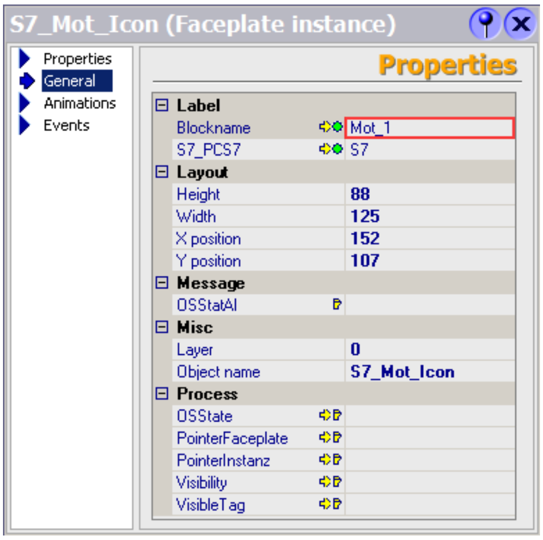
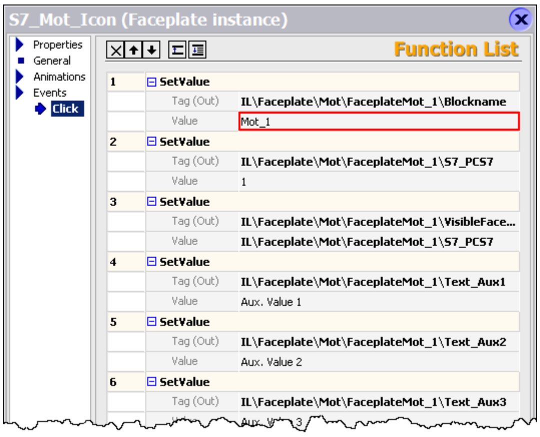
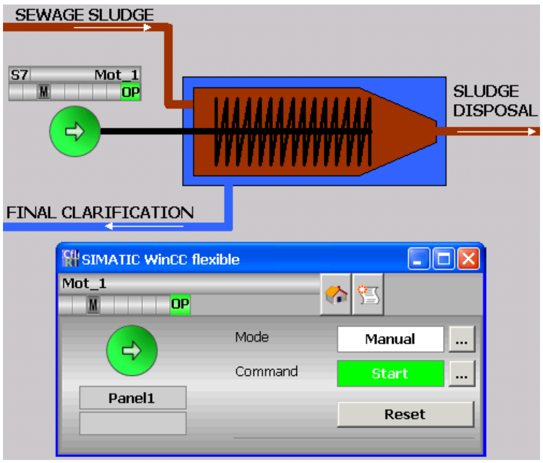
The process values of several technological blocks of the same type can be displayed in one screen display. The process values to be displayed for the faceplate can be defined by clicking the corresponding block icon.

If you want to configure several block icons and faceplates of the same type, please proceed as described in the following table.

Table 6-8

No.	Action	Display
1.	Add the desired faceplate (e.g. MotL) from the library to the image. The tag folder “IL > Faceplate > Mot > FaceplateMot_1” will be generated automatically. **Note** When the same faceplate is assigned anew, the previous tag folder will be overwritten. No further tags will be generated.	 SIMATIC WinCC flexible S L F R OP Communication Tags IL Faceplate MonAn Mot FaceplateMot_1 FaceplateMot_PS FaceplateMot_SC General Symbol Connections Cycles
2.	Add the associated block icon (e.g. MotLIcon) from the library to the image. The tag folder “IL > Symbol > Mot > Mot_1” will be generated automatically. Rename the tag folder (e.g. Motor_1) to avoid that the created tags will be overwritten by the next block icon. Repeat this step for all block icons of the same type and the process values of which shall be indicated in the same image display. **Note** Some of the tags used by the block icons are stored in the faceplate container. These tags are required only once for each faceplate and will be overwritten when the icon is added to the tag folder.	 S7 Mot_1 S7 Mot_2 S7 Mot_3 S L F R OP Communication Tags IL Faceplate General Symbol MonAn Mot Mot_1 Mot_2 Mot_3 PM401 Connections Cycles

<table>
<tr><th>No.</th><th>Action</th><th>Display</th></tr>
<tr><td>3.</td><td>Rename the tag folder created for the faceplate in step 1.
Repeat the steps 1 to 3 for each further faceplate.</td><td>
Communication
Tags
IL
Faceplate
MonAn
Mot
FaceplateMot_1
FaceplateMot_PS
FaceplateMot_SC
General
Symbol
Connections
Cycles</td></tr>
<tr><td>4.</td><td>Open WinCC flexible in project view and change over to the folder “Communication > Connections”.
Delete all connections which were created when adding the faceplates and block icons but which are no longer needed now.</td><td>
Connections<table>
<tr><th>Name</th><th>Ac...</th><th>Communication d...</th><th>Station</th><th>Partner</th><th>Node</th><th>Online</th></tr>
<tr><td>PanelConn</td><td>On</td><td>SIMATIC S7 300/400</td><td>\PCS7_PU_Panel...</td><td>CPU 317-2 DP</td><td>CP 343-1</td><td>On</td></tr>
<tr><td>~~Verbindung_2~~</td><td>~~On~~</td><td>~~SIMATIC S7 300/400~~</td><td></td><td></td><td></td><td>~~On~~</td></tr>
<tr><td>~~S7 Verbind...~~</td><td>~~On~~</td><td>~~SIMATIC S7 300/400~~</td><td></td><td></td><td></td><td>~~On~~</td></tr>
</table></td></tr>
<tr><td>5.</td><td>Open, one by one, all newly created block icon tag folders.<ul><li>Change all invalid tag connections. Use the selection list to specify the correct connection to the CPU.</li><li>Address the “Pointer” tag in compliance with the parameter of your newly created DB.</li></ul></td><td>
Mot_1 Mot_2 Mot_3 FaceplateMot_1<table>
<tr><th>Name</th><th>Connection</th><th>Data type</th><th>Symbol</th><th>Address</th></tr>
<tr><td>Mot.OSState</td><td>Verbindung_2</td><td>DWord</td><td><Undefined></td><td>DB [Pointer] DBD 10</td></tr>
<tr><td>Pointer</td><td>Verbindung_2</td><td>Int</td><td><Undefined></td><td>DB 25 DBW 0</td></tr>
</table>
Mot_1 Mot_2 Mot_3 FaceplateMot_1<table>
<tr><th>Name</th><th>Connection</th><th>Data type</th><th>Symbol</th><th>Address</th></tr>
<tr><td>Mot.OSState</td><td>PanelConn</td><td>DWord</td><td><Undefined></td><td>DB [Pointer] DBD 10</td></tr>
<tr><td>Pointer</td><td>PanelConn</td><td>Int</td><td><Undefined></td><td>DB 25 DBW 0</td></tr>
</table></td></tr>
<tr><td>6.</td><td>Open, one by one, all newly created faceplate tag folders.<ul><li>Change all invalid tag connections. Use the selection list to specify the correct connection to the CPU.</li></ul><b>Note</b>
The faceplate tags need not be addressed. The process tags are addressed via the “Pointer” tag. The internal tag values are available when clicking the block icon.</td><td>
Mot_1 Mot_2 Mot_3 FaceplateMot_1<table>
<tr><th>Name</th><th>Connection</th><th>Data type</th><th>Symbol</th><th>Address</th></tr>
<tr><td>Blockname</td><td><Internal tag></td><td>String</td><td><Undefined></td><td><No address></td></tr>
<tr><td>Mot.Aux1</td><td>PanelConn</td><td>Real</td><td><Undefined></td><td>DB [Pointer] DBD 2</td></tr>
<tr><td>Mot.Aux2</td><td>PanelConn</td><td>Real</td><td><Undefined></td><td>DB [Pointer] DBD 6</td></tr>
<tr><td>Mot.En_Aux</td><td>PanelConn</td><td>Bool</td><td><Undefined></td><td>DB [Pointer] DBX 0.4</td></tr>
<tr><td>Mot.OP_Au...</td><td>PanelConn</td><td>Bool</td><td><Undefined></td><td>DB [Pointer] DBX 0.1</td></tr>
<tr><td>Mot.OP_M...</td><td>PanelConn</td><td>Bool</td><td><Undefined></td><td>DB [Pointer] DBX 0.2</td></tr>
<tr><td>Mot.OP_Re...</td><td>PanelConn</td><td>Bool</td><td><Undefined></td><td>DB [Pointer] DBX 0.3</td></tr>
<tr><td>Mot.OSState</td><td>PanelConn</td><td>DWord</td><td><Undefined></td><td>DB [Pointer] DBD 10</td></tr>
<tr><td>Mot.OSSta...</td><td>PanelConn</td><td>DWord</td><td><Undefined></td><td>DB [Pointer] DBD 10</td></tr>
<tr><td>Mot.OSSta...</td><td>PanelConn</td><td>DWord</td><td><Undefined></td><td>DB [Pointer] DBD 14</td></tr>
<tr><td>Mot.Permis...</td><td>PanelConn</td><td>DWord</td><td><Undefined></td><td>DB [Pointer] DBD 18</td></tr>
<tr><td>Mot.Permis...</td><td>PanelConn</td><td>Int</td><td><Undefined></td><td>DB [Pointer] DBW 18</td></tr>
<tr><td>Pointer</td><td><Internal tag></td><td>Int</td><td><Undefined></td><td><No address></td></tr>
<tr><td>S7_PCS7</td><td><Internal tag></td><td>Int</td><td><Undefined></td><td><No address></td></tr>
<tr><td>VisibleFace...</td><td><Internal tag></td><td>Byte</td><td><Undefined></td><td><No address></td></tr>
<tr><td>X_Offset</td><td><Internal tag></td><td>Int</td><td><Undefined></td><td><No address></td></tr>
<tr><td>Y_Offset</td><td><Internal tag></td><td>Int</td><td><Undefined></td><td><No address></td></tr>
</table></td></tr>
</table>

No.	Action	Display
7.	To make sure that the block icon shows a clear and unique name, it can be adapted in an icon properties dialog. • Select the icon to be edited in the process image. • Open the Properties dialog and select the "General" folder. • In the “Block name” field you can define a characteristic name (e.g. name of the process tag). • The “S7_PCS7” field shows “S7” to indicate that an S7MotL block is connected.	
8.	In the icon’s properties dialog you can specify that the name of the block icon shall be indicated in the faceplate. • Mark the icon to be edited in the process image. • Open the Properties dialog and select the folders “Events > Click”. • Enter the same name as used above in the event "SetValue" line for the “block name” tag of the faceplate. **Note** If you have performed configuration in the same order as described above, the events will already include the correct faceplate tags. If you wish to use another faceplate, the tags to be set are to be changed here.	
9.	Position the block icon and the faceplate in the process view as desired. Configure the plant image according to your requirements. The illustration on the right shows an example of a configured motor.	

S7_Mot_Icon (Faceplate instance)

Properties — General — Animations — Events

Properties

⊟ Label		
Blockname		Mot_1
S7_PCS7		S7
⊟ Layout		
Height		88
Width		125
X position		152
Y position		107
⊟ Message		
OSStatAI		
⊟ Misc		
Layer		0
Object name		S7_Mot_Icon
⊟ Process		
OSState		
PointerFaceplate		
PointerInstanz		
Visibility		
VisibleTag		

S7_Mot_Icon (Faceplate instance)

Properties — General — Animations — Events — Click

Function List

1	⊟ SetValue	
	Tag (Out)	IL\Faceplate\Mot\FaceplateMot_1\Blockname
	Value	Mot_1
2	⊟ SetValue	
	Tag (Out)	IL\Faceplate\Mot\FaceplateMot_1\S7_PCS7
	Value	1
3	⊟ SetValue	
	Tag (Out)	IL\Faceplate\Mot\FaceplateMot_1\VisibleFace...
	Value	IL\Faceplate\Mot\FaceplateMot_1\S7_PCS7
4	⊟ SetValue	
	Tag (Out)	IL\Faceplate\Mot\FaceplateMot_1\Text_Aux1
	Value	Aux. Value 1
5	⊟ SetValue	
	Tag (Out)	IL\Faceplate\Mot\FaceplateMot_1\Text_Aux2
	Value	Aux. Value 2
6	⊟ SetValue	
	Tag (Out)	IL\Faceplate\Mot\FaceplateMot_1\Text_Aux3
	Value	Aux. Value 3

Mode: Manual

Command: Start

Reset

7 Links & Literature

Internet links

The following list is by no means complete and only presents a selection of related sources.

Table 7-1

	Topic	Title
\1\	Reference to this document	http://support.automation.siemens.com/WW/view/en/50708061
\2\	Siemens Industry Online Support	http://support.automation.siemens.com
\3\	Sales/Delivery Release SIMATIC PCS 7 Industry Library V8.0	http://support.automation.siemens.com/WW/view/en/60982306
\4\	Integration of third-party systems with SIMATIC PCS 7/OPEN OS	http://support.automation.siemens.com/WW/view/en/49740087
\5\	“PCS 7 OS Process Control” manual	http://support.automation.siemens.com/WW/view/en/36195920
\6\	How to read out the status of an H system	http://support.automation.siemens.com/WW/view/en/19537149

8 History

Table 8-1

Version	Date	Revisions
V1.0	12.08.2011	Publication (Intranet)
V2.0	22.06.2012	Changed to IL, Publication (Internet)