Siemens
Industry
Online
Support

APPLICATION EXAMPLE

Control Module (CM) Technology - Efficient Engineering in SIMATIC PCS 7

SIMATIC PCS 7 V10.0

SIEMENS

Legal information

Use of application examples
Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability
Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information
Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (https://support.industry.siemens.com) shall also apply.

Security information
Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial security measures that may be implemented, please visit https://www.siemens.com/industrialsecurity.

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

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

Preface

Purpose of this document

This document describes the structure, scope of functions, configuration, typical scenarios, and advantages of utilizing the new SIMATIC PCS 7 Control Module Type concept.

Abbreviations

The following table lists the abbreviations and designations of the type models.

Abbreviation	English	Description
PT	Process tag	CFC according to the old type model
PTT	Process tag type	CFC-type template for instantiation according to the old type model
CM	Control Module	CFC according to the new type model
CMT	Control Module Type	CFC-type template for instantiation according to the new type model
BCM	Basic Control Module Type Library	Predefined Control Module types in form of a library
EMT	Equipment Module Type	EM type templates can contain several CMs in order to map a plant section.
EPHT	Equipment Phase Type	EPH type templates contain SFC and CM for standardized start-up and operation of a system section.

NOTE This document uses the terms and abbreviations Control Module (CM) and Control Module Type (CMT).

Applies to

The description refers to the use of CM technology from SIMATIC PCS 7 V9.0 SP3, but is, in principle, also applicable to earlier versions (as of PCS 7 V8.0) and later versions (up to PCS 7 V10.0).

The Basic Control Module Type Library is available for SIMATIC PCS 7 Versions V9.1 and V10.0.

Table of contents

1. Introduction

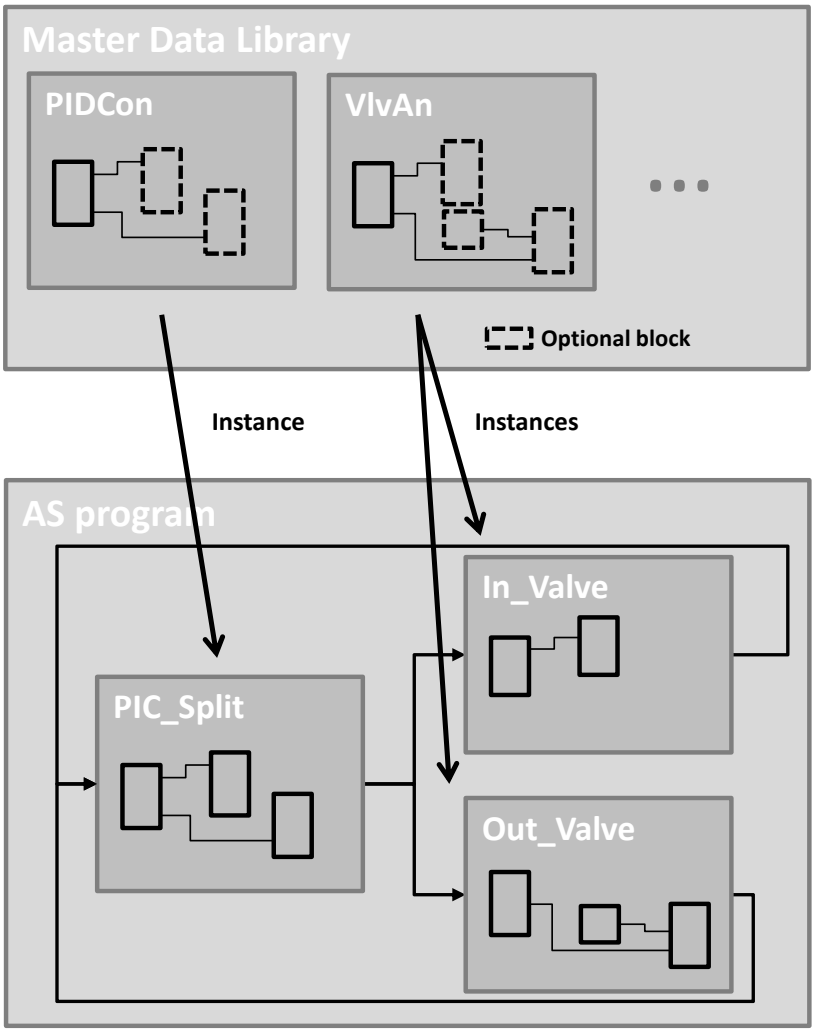
1.1. Overview

Standardization of engineering is an important instrument for the continuous improvement of competitiveness and for achieving higher planning quality. However, varied process steps and sequences, varied equipment, and flexibility in production make this task more difficult.

One approach to standardization is the consistent use of individual Control Module Types (CMT) to create an automation program. The ISA-88 standard contains a CMT, e.g. a valve from the user program, as well as the physical setup. CMTs can either be a component of a technical device, such as flow control, or a component of a sub-system, such as a stirring tank.

1.2. Principle of operation

This application example describes the handling of the Control Module technology in the environment of SIMATIC PCS 7 using individual technology components and typical applications. The use of CM technology results in additional improvements and increased efficiency of the SIMATIC PCS 7 engineering, i.e., the automation project can be continuously adapted to changing requirements, as shown in the following figure.



CM technology provides support during the typical engineering phases that influence the automation program:

- Concept: Development of a rough structure based on a piping and instrumentation flow chart (P&I diagrams)
- Development: Implementation of customer requirements, e.g., interlocks, process units, logic, etc.
- Engineering: With the new view in the technological list editor, CM technology also supports mass data engineering.
- Test procedure: Preparing the hardware connection
  (sensors and actuators)

The CM technology not only supports initial engineering, but also program extensions and the detection of program changes.

1.2.1. Control Module Types

Standardized engineering can be achieved through the consistent use of Control Module Types (CMTs). With the introduction of CMT technology, a clear type-instance concept will be implemented. Here, the CMT is the template, which is instantiated later in the project (CM).
By defining optional blocks in the CMT, a large number of different variants of this CMT can be instantiated in the project. One variant, for example, stands for an indicating measuring point for the input signal (4-20 mA, PA field device). A selectable function in turn refers to the program logic, such as a locking function.

The following figure shows a matrix with optional blocks for creating a variant and activating additional functions.

ValAn (CMT master data library)	BypassAct	Intlock	Permit	Protect	MV_Scale	IF_Ctrl#	RbkReturn	GSL	GSH	YC_FB	YC	YS	GI	
Variants	**Function**							**Channel block**						**Description**
ValAn_Std	o	o	o	o	x	x	x	o	o		x	o		Controls a valve without position feedback (analog signal)
ValAn_StdRbk	o	o	o	o	o	o		o	o		x	o	x	Controls a valve with position feedback (analog signal)
ValAn_FbRbk	o	o	o	o	o	o		o	o	x		o		Controls a valve with position feedback (fieldbus)

X = Selection for variant o = Selectable functions

All instances can be compared and matched with the type at any time.

The use of CMT offers the following benefits:

- Reduced test effort (type-based testing)
- Faster configuring through instantiation
- Reduced maintenance for libraries
- Change tracking by detecting deviations on the instance
  (Exception inserted blocks in an instance)

It is generally recommended to use only one basic technological module, such as valve, motor, controller, etc., per CMT; otherwise, the cooperation between command, status, and the SFC type is no longer guaranteed.

Care should be taken to ensure that the name is both appropriate and simple.
For example:

- Valve = Y
- Motor = N
- Indication = I

NOTE Additional tips on naming can be found in Section 2.9.

Several technological blocks per CMT are possible, but please note that commands/status can only be configured for one block in the CMT.

If a CMT contains several blocks with the S7_contact attribute (usually technological blocks), only those of the first block in alphabetical order are available for commands and status.

Since technological blocks are shown in the visualization, it is recommended to carry over the CFC comment (Section 3.5). That way, the CFC comment, block comment, and faceplate display are consistent and only have to be configured once.

1.2.2. Basic Control Module Type Library

The Basic Control Modules (BCM), in the form of a Type Library, are available for SIMATIC PCS 7 as a master data library, and contain typical, pre-configured, and tested CMTs. The BCM are created with CM technology and enable more efficient engineering through standardized program components.

The following benefits are achieved by using the BCM Type Library:

- Extensive library for different applications and industries
- Reduction of the configuration effort
- Reduced maintenance
- Standardized structures

The BCM Type Library offers typical components as a template for building automation solutions. The CMTs of the BCM Type Library contain all necessary function and channel blocks and can be adapted to the project-specific conditions by instantiation.

The BCMs are based on the SIMATIC PCS 7 Advanced Process Library (APL) and Industry Library (IL), are pre-configured independently of hardware, and have a modular structure.

The library "109475748_BCM_Lib_PCS7V10_0.zip" provides the following CMT groups:

- MonAn: Analog measured value display
- MonDi: Digital measured value display (binary signal)
- OpDi: Setting a binary value by the operator
- PIDCon: Controller for standard and cascade control loops
- Mot: Engine control with simple speed control
- Vlv: Valve actuation with two defined positions
- VlvAn: Valve control with analog control valve

NOTE For BCMs, the name of the central technology block of the APL is used.

NOTE A detailed description of each CMT with functional description, supported variants, and control elements is included in the library.

1.3. Updates

This document serves as a practical guide for the configuration of Control Module Types (CMTs). The handling of CMTs is optimized with the continued development of SIMATIC PCS 7.

The latest UpdateCollection can be obtained from the following SIOS entry:

https://support.industry.siemens.com/cs/ww/en/view/109972821

1.4. Components used

The following list contains all files and projects used in this example.

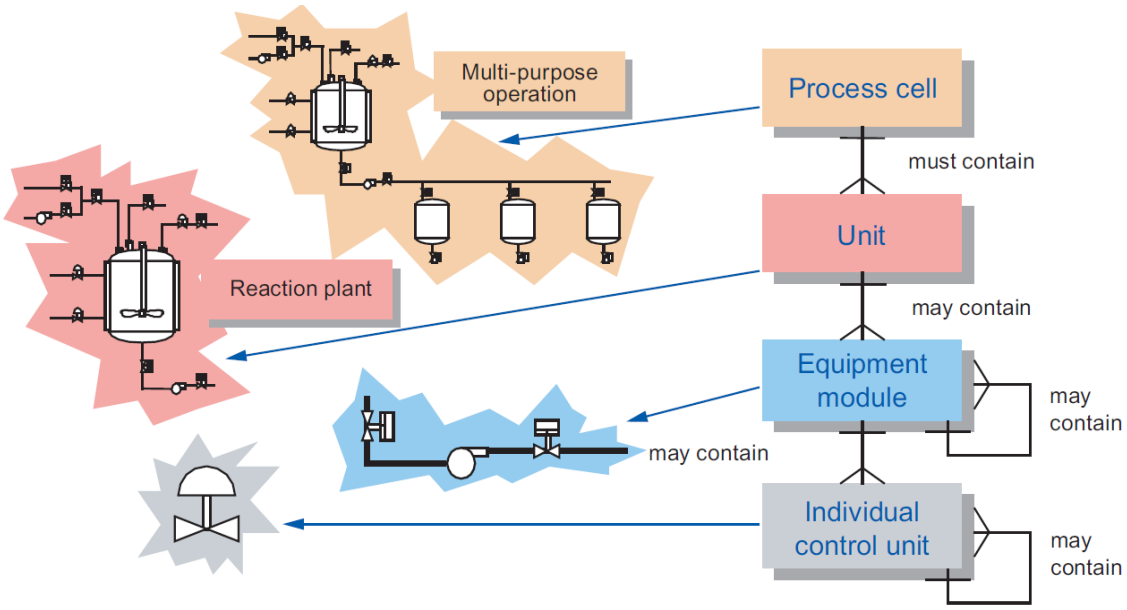
Component	Note
109475748_CMT_Engineering_DOC_PCS7V10_0_en.pdf	This document
109475748_BCM_Type_Library_DOC_PCS7V10_0_en.pdf	Document "Structure and Setup of the Best Practice Control Modules Type Library (BCM)"
109475748_BCM_Lib_PCS7V10_0.zip	Library available in the corresponding article: 109475748

2. Fundamentals

2.1. ISA-88 Standard (discontinuous mode)

The "ANSI/ISA-88" standard refers to batch-oriented operation in batch plants that are operated with SIMATIC BATCH, for example, and includes the relevant standards and terminology.

The following figure shows an asset structure based on CM (individual control units).



2.2. ISA-106 Standard (continuous mode)

The "ISA-106" standard refers to the structure of the automation solution for continuous process plants. The standard describes, among other things, the:

- "Physical Model": Represents the physical components of the system up to the actual field device.
- "Procedure Requirements Model": Contains the process-specific requirements for the individual plant components.
- "Procedure Implementation Model": Contains the implementation procedures for the individual plant components.

The CMs or CMTs must be assigned to the "Procedure Implementation Model". These are required in the automation program to connect or process the physical plant component.

The basic construction is similar to the ISA-88, but there is a difference in the way the system operates. In discontinuous operation, products are manufactured according to a recipe. Depending on the use case (phase), technical functions (equipment modules) are controlled differently and supplied with recipe-specific parameter sets. This means that the driving style depends on the product to be manufactured.

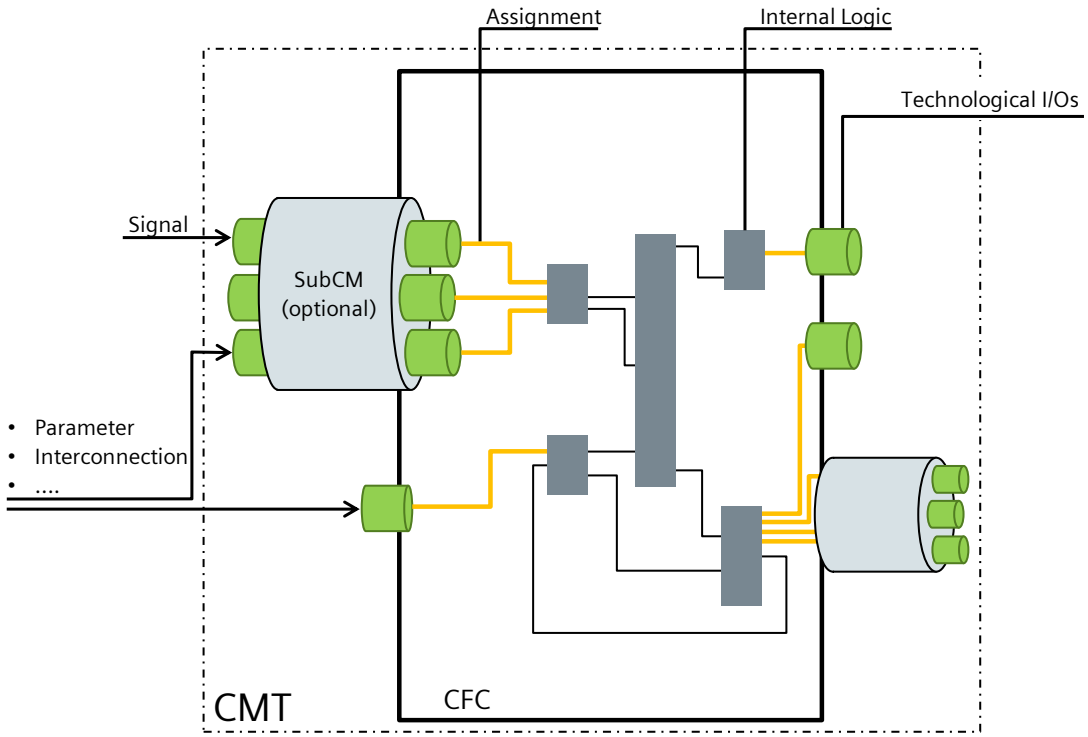
In continuous operation, the process is in the foreground, i.e. the plant is started up via successive process states. After reaching a stable and defined condition, a product is continuously produced with constant quality. The defined operating mode can react to abnormal conditions by means of defined measures such as a Safety Integrated System.

2.3. PT/PTT and CM/CMT structure

The Control Module Type (CMT) marks a new type of standardized software block that enables even more efficient engineering than classic measurement point types. A CMT can contain blocks, plans, control variables (block connections such as signals and parameters), and messages.

CMT Model

Control Module Types have detailed control logic inside and "Technological I/Os" outside. By assignment, the internal logic is linked to the "Technological I/Os".



- Assignment: The assignment is the linking of the logic in the CFC and the technological connections.
- Internal logic: The internal logic describes the behavior and functions of the CMT and is implemented in the CFC.
- Technological I/Os: The technological I/Os form the connection to other CM, I/O-HW and assigned parameter values. They offer a simplified view with all signals, connections, and parameters relevant for technological engineering. A SubCM combines several objects from the "Technological I/Os" and can be declared as optional.

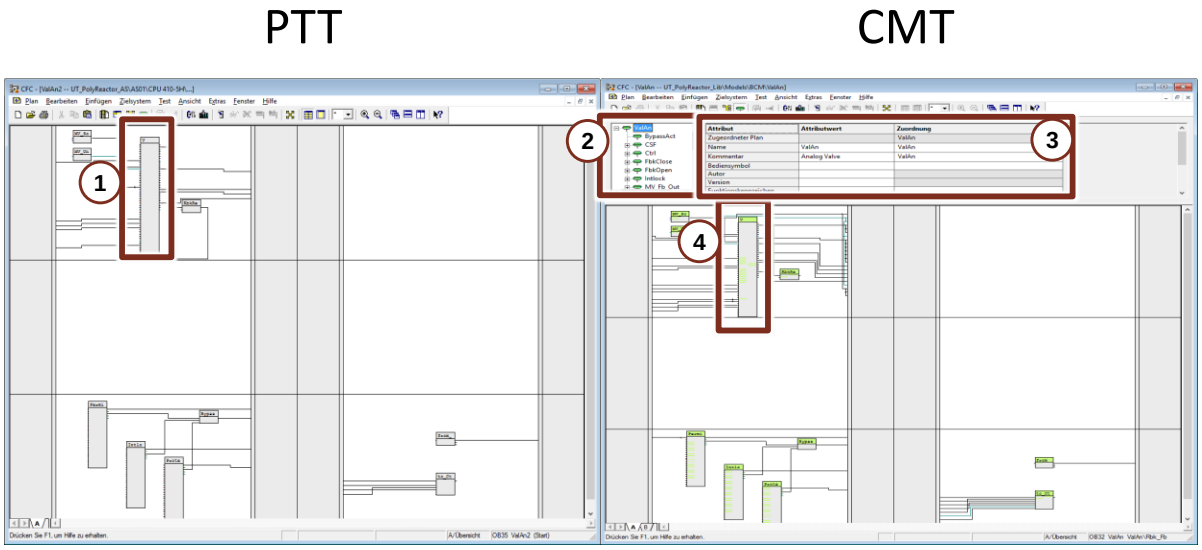
NOTE Function blocks can also be assigned a SubCM. This will later place or remove them in the instance (CM).

Technological engineering: The individual CMs are connected via the technological I/Os, in SIMATIC PCS 7 with the "Technological List Editor", CFC Editor and in COMOS or SIMATIC PCS 7 Plant Automation Accelerator via the "Function Diagram".
The assignment maps the technological planning information to the lower abstraction level of the CFC.

NOTE The internal logic and the assignment of the logic to the technological I/Os are only visible with the CFC editor. For the SIMATIC PCS 7 Plant Automation Accelerator, COMOS, and SIMIT, the properties are not displayed. Here, the CMT behaves like a "black box".

Realization of a PTT and CMT in PCS 7

The following figure shows the structure of a PTT and CMT using the example of an analog valve.



1. The blocks are shown in gray at PT/PTT. The "Technological I/Os" are not supported.
2. All created and defined objects (parameters, signal, messages, status, command) are displayed in the "Technological I/Os" area.
3. In the "Attributes" area, the "Technological I/Os" are linked to the internal logic.
4. The blocks and the "Technological I/Os" of the CM/CMT are shown in green.

NOTE The colors described for the display refer to the standard setting. The colors can be adjusted in the CFC via the menu item "Extras > Settings > Colors..." for each Engineering System or reset to the default values.

2.4. Comparison between PT/PTT and CM/CMT

The following table compares the features of PT/PTT and CM/CMT.

Functionality	PT/PTT	CM/CMT
Change tracking	Only with special tools	Yes, with the compare function in the file transfer dialog.
Variant support	No, because a PTT is needed for each variant	Yes, through CMT with selectable variants (options)
System-supported instantiation	IEA (Import/Export Assistant)	With COMOS, Plant Automation Accelerator, or from PCS 7 9.0 SP3 in the List Editor with IEA license.
Extending functions	Yes, by adapting the PTT and instantiating with the IEA **Caution**: The export file must be adapted to the new function. Specific changes to instances are lost if they are not read back.	Yes, very convenient by extending the functionality in the CMT and synchronizing it with the instances.
Type project planning	Easy, by placing and interconnecting the required blocks.	Somewhat more extensive, since the technological I/Os must also be defined.

2.5. Typical Changes of the CMTs and CM

To make the most of the instance and type concept with the synchronization function, we recommend that you make the changes to the type or instance depending on the type of change. The following table gives typical examples of changes of type or instance.

Change	Type or Instance	Execution
Insert block	Type	1. Insert block 2. Define as SubCM 3. Mark as "optional" 4. Synchronize type with instance
Insert block (without adapting the type)	Instance	1. Use of functions **Note:** See Section 2.9
Parameterization for multiple instances	Type	1. Adjust parameters 2. Synchronize type with instance **Note:** If the changed parameter is defined in the technological I/Os, the parameter is not adjusted and must be adjusted in the instances.
Parameterization for an Instance	Instance	1. Adjust parameters **Note:** The changed parameter must be defined in the technological connections so that the parameter is not overwritten during the next synchronization.
Interconnections between instances	Instance	1. Add interconnections **Note:** Both connections of the connection must be defined in the technological connections.
Connection in the instance	Type	1. Add interconnections Synchronize type with instance

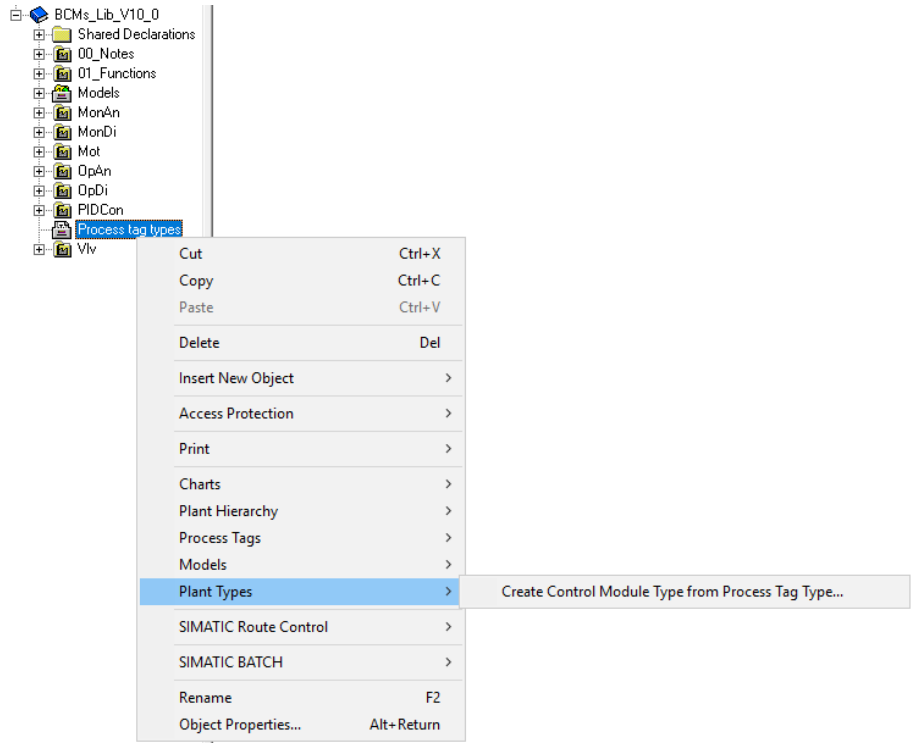
Instance-specific parameters

We recommend that all values that are to be adjusted in an instance-specific manner be configured as technological parameters (=green parameter). This setting is not necessary for parameters with the attribute S7_m_c = true, but it facilitates the entire technological configuration (technological list editor, PAA, COMOS, etc.). For this application, the use of the "pink parameters" is not recommended.

2.6. Conversion of a PTT into a CMT

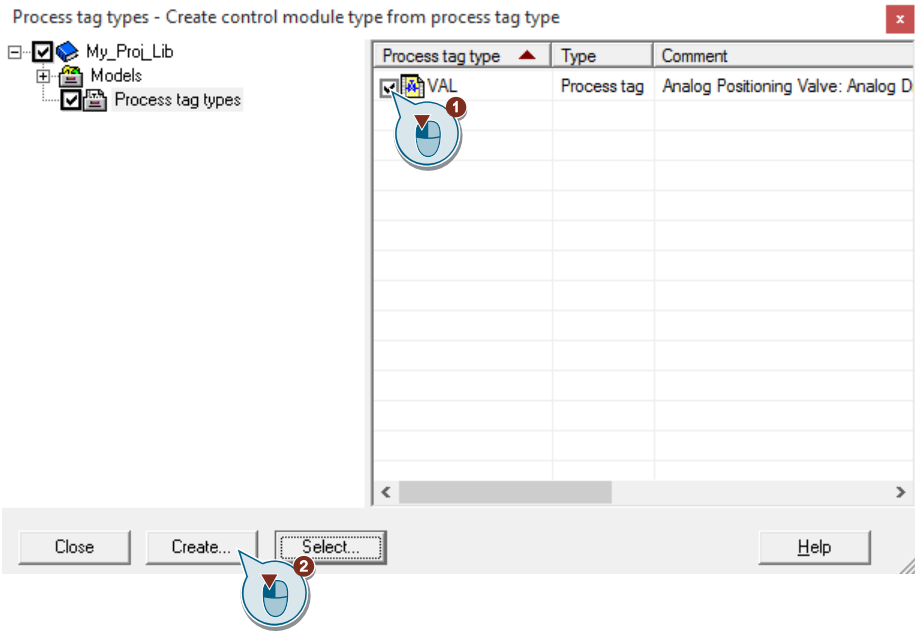
An existing PTT can be easily converted into a CMT without losing the configuration in the library. Below, you will find a step-by-step guide on how to convert a PTT into a CMT.

1. Right-click on the storage folder for the PTT and click on "Technological Types > Control Module Type from Process Tag Type..." in the context menu.



A new dialog window "Create Control Module Type from Process Tag Type..." opens.

2. Select the PTTs you want converted to CMTs and click the "Create..." button.
   This creates CMTs with the same names as the selected PTTs.



NOTE Alternatively, you can create CMTs from PTTs in other libraries, such as the APL library. To do this, click on the "Select..." button and select the library. Carry out step 2.

3. Open the newly created CMT and define the required subordinate individual control units, parameters, signals, messages, states, and commands in the technological I/Os.

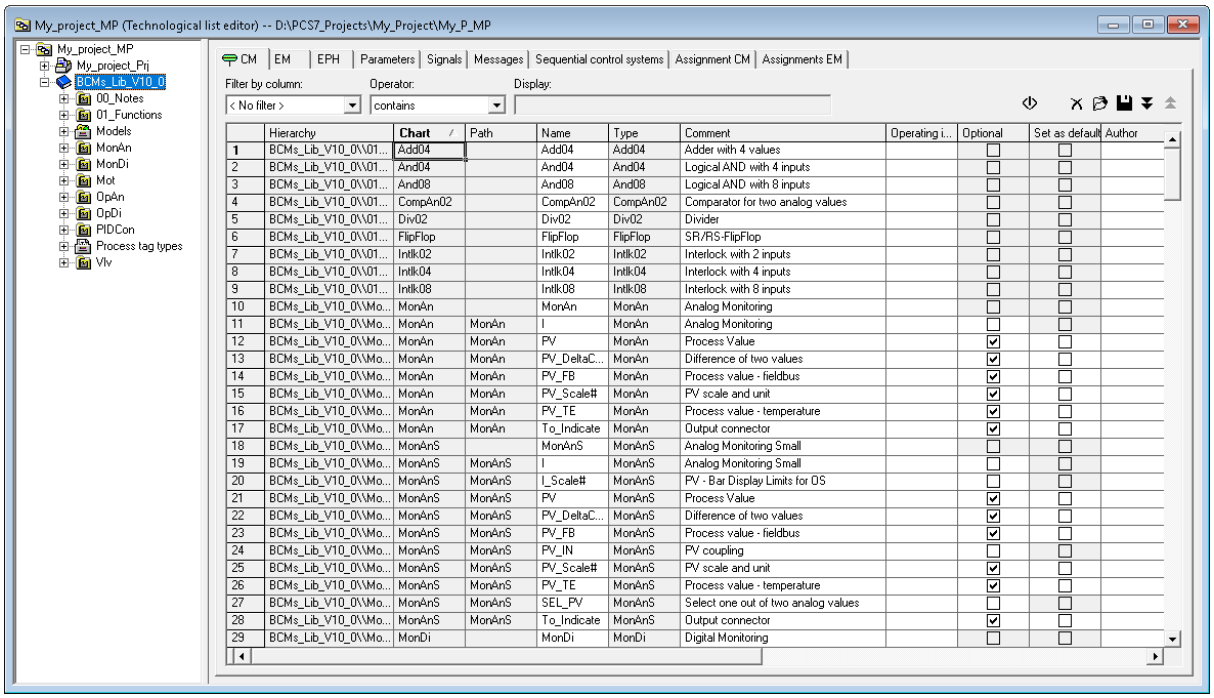
NOTE Instructions can be found in the Section 5.1. You can skip the point "Create a CMT", because the engineering of the CFC was taken over from the PTT.

2.7. Mass data engineering

With a modular engineering approach, the overall project efficiency can be increased and risks can be minimized. High standardization and simple configuration additionally save engineering time and costs.

Technological list editor

As of PCS 7 V9.0, a new view is available by using the "Technological List Editor". The "Technological List Editor" offers various displays, operations, and filter options in the tabs to edit the technological types or objects, with their properties and attributes in tables, or to create them in PCS 7 9.0 SP3 and higher.



	Hierarchy	Chart	Path	Name	Type	Comment	Operating i...	Optional	Set as default	Author
1	BCMs_Lib_V10_0\\01...	Add04		Add04	Add04	Adder with 4 values		☐	☐	
2	BCMs_Lib_V10_0\\01...	And04		And04	And04	Logical AND with 4 inputs		☐	☐	
3	BCMs_Lib_V10_0\\01...	And08		And08	And08	Logical AND with 8 inputs		☐	☐	
4	BCMs_Lib_V10_0\\01...	CompAn02		CompAn02	CompAn02	Comparator for two analog values		☐	☐	
5	BCMs_Lib_V10_0\\01...	Div02		Div02	Div02	Divider		☐	☐	
6	BCMs_Lib_V10_0\\01...	FlipFlop		FlipFlop	FlipFlop	SR/RS-FlipFlop		☐	☐	
7	BCMs_Lib_V10_0\\01...	Intlk02		Intlk02	Intlk02	Interlock with 2 inputs		☐	☐	
8	BCMs_Lib_V10_0\\01...	Intlk04		Intlk04	Intlk04	Interlock with 4 inputs		☐	☐	
9	BCMs_Lib_V10_0\\01...	Intlk08		Intlk08	Intlk08	Interlock with 8 inputs		☐	☐	
10	BCMs_Lib_V10_0\\Mo...	MonAn		MonAn	MonAn	Analog Monitoring		☐	☐	
11	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	I	MonAn	Analog Monitoring		☐	☐	
12	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	PV	MonAn	Process Value		☑	☐	
13	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	PV_DeltaC...	MonAn	Difference of two values		☑	☐	
14	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	PV_FB	MonAn	Process value - fieldbus		☑	☐	
15	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	PV_Scale#	MonAn	PV scale and unit		☑	☐	
16	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	PV_TE	MonAn	Process value - temperature		☑	☐	
17	BCMs_Lib_V10_0\\Mo...	MonAn	MonAn	To_Indicate	MonAn	Output connector		☑	☐	
18	BCMs_Lib_V10_0\\Mo...	MonAnS		MonAnS	MonAnS	Analog Monitoring Small		☐	☐	
19	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	I	MonAnS	Analog Monitoring Small		☐	☐	
20	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	I_Scale#	MonAnS	PV - Bar Display Limits for OS		☐	☐	
21	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	PV	MonAnS	Process Value		☑	☐	
22	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	PV_DeltaC...	MonAnS	Difference of two values		☑	☐	
23	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	PV_FB	MonAnS	Process value - fieldbus		☑	☐	
24	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	PV_IN	MonAnS	PV coupling		☐	☐	
25	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	PV_Scale#	MonAnS	PV scale and unit		☑	☐	
26	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	PV_TE	MonAnS	Process value - temperature		☑	☐	
27	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	SEL_PV	MonAnS	Select one out of two analog values		☐	☐	
28	BCMs_Lib_V10_0\\Mo...	MonAnS	MonAnS	To_Indicate	MonAnS	Output connector		☑	☐	
29	BCMs_Lib_V10_0\\Mo...	MonDi		MonDi	MonDi	Digital Monitoring		☐	☐	

In the "Technological List Editor", the signals, parameters, and messages of CMs can be parameterized and CMs can be interconnected via the technological I/Os. In addition to parameterization and interconnection, CMTs can also declare blocks as optional.

In addition, export and import to and from Microsoft Excel is supported. This enables engineering without system-specific knowledge.

NOTE

Further information on the "Technological List Editor" can be found in section 8.7 of the SIMATIC process control system PCS 7 Compendium Part A - Configuration Guide (V9.1) under the following link:
https://support.industry.siemens.com/cs/ww/en/view/109809015

COMOS and SIMATIC PCS 7 Plant Automation Accelerator

The "COMOS" and "SIMATIC PCS 7 Plant Automation Accelerator" (PAA) applications support the program-based generation of automation data (hardware configuration and automation program).

Under the following links you will find examples of mass data engineering:

- Application example:
  SIMATIC PCS 7 Plant Automation Accelerator using a practical example
  (https://support.industry.siemens.com/cs/ww/en/view/109742154)
- Application example:
  Integrated Engineering with COMOS and SIMATIC PCS 7 using a practical example
  (https://support.industry.siemens.com/cs/ww/en/view/70922226)

NOTE

To avoid inconsistencies in mass data engineering with "COMOS" and "SIMATIC PCS 7 Plant Automation Accelerator" or with the "SIMIT Simulation" simulation program, the following points must be observed:

- All connections of the CM must be routed via the technological I/Os. This means that the two ports of the connection must be defined in the technological connections.
- An extension of the CM with additional blocks is only permitted by selecting optional blocks or using functions.
  For more information on functions, refer to the Section 5.4.

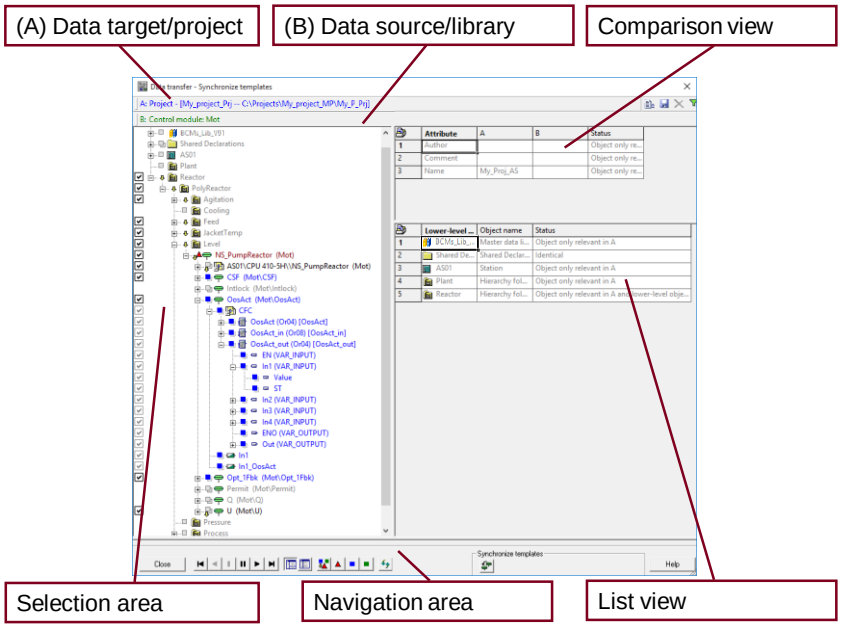
2.8. Automation Interface

The Automation Interface contains an abstract data model of the Control Module Types. The information of this data model is provided by the different data sources, PCS 7, PAA, etc. Therefore, with PCS 7, the configured information is provided by the "Technological I/Os".

The Automation Interface information is used to exchange and compare data, such as the "Data Transfer" dialog, when synchronizing CMs with CMTs.

During "data transfer", it shows which instances have been changed compared to the CMT, and exactly what has been changed. Changes are represented by different colors or objects, such as deviations that occurred when comparing the project status.

The following figure shows the detailed structure of the file transfer.



- (A) Data target/project: The data target corresponds to the project and contains all instantiated CMs. In the bar the project name and project path are displayed in blue font color.
- (B) Data source/library: The data source corresponds to the master data library and contains all CMTs of the library. The entire library or individual CMTs can be selected for comparison. In the bar, the library name and the CMTs are displayed in green.
- Comparison view: The comparison view shows differences between the folders/CFCs selected in the selection area and the comparison object (data source/library).
- Selection area: All instances found in the project are displayed in the selection area. On the left edge, instances that differ from the template can be selected or deselected for synchronization. All instances are selected by default.
- Navigation pane: You can switch between the individual data records (instances) in the navigation pane. The display can be switched between standard and tabular views and a prefiltered view (only deviations). The navigation pane also contains buttons for updating the project comparison as well as for starting the synchronization.
- List view: The list view lists the subordinate objects of the folders/CFCs that are selected in the selection area compared to the comparison object (data source/library).

NOTE

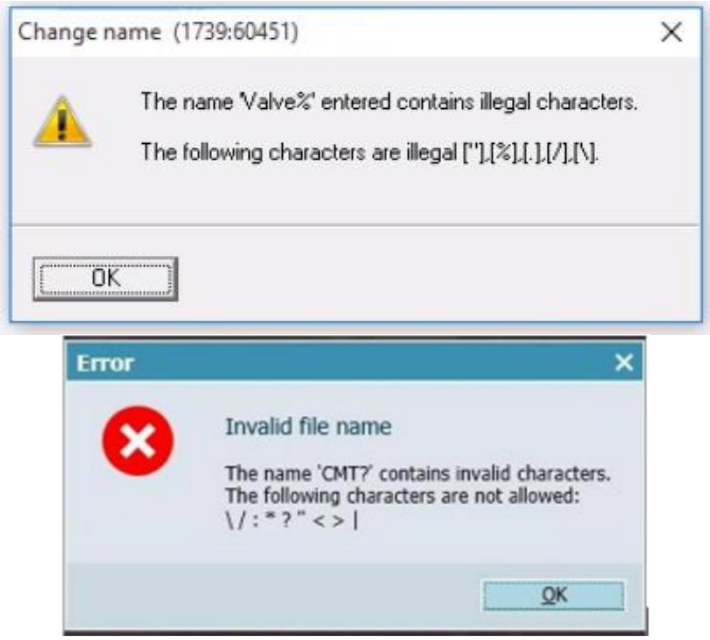
For more information on synchronizing CMT via the Automation Interface, refer to the article "Synchronizing Control Module Types" at the following link:
https://support.industry.siemens.com/cs/ww/en/view/109758382

2.9. Naming

A uniform naming concept with basic parts that identify the type or, in the case of different libraries, the respective library, is recommended (e.g., CMT_MonAn, BCM_Vlv). The name should not contain any instance-specific information.

As with CFCs, the name can be up to 22 characters long. Special characters such as " " % . / \ are not permitted as with CFC. Furthermore, the use of the following special characters : * ? " < > is not recommended.

The latter are not approved for use in SIMIT.



2.10. Functions

Functions are used for instance-specific adjustment within a CM.

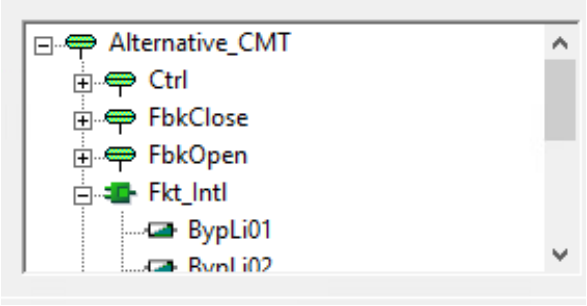
A function is created in the library as a CMT with only one sub-CM, and is declared as a "function" via option field. In contrast to the CMT, the function may not have any subordinate functions (further summarized objects, SubCMs), since the function is later instantiated as an additional SubCM in a CM. In a function with optional blocks, integration in a CMT would create a further hierarchy level in the technological I/Os that is not permitted.

They should always be marked with a prefix (e.g. "fkt_xxx"). This avoids a situation where a sub-CM has the same name. That situation could create an issue if a function and a SubCM with the same name are both used in an instance.

The created functions can be instantiated once or multiple times in a CM without the need to adapt the CMT. Connections to functions are treated as external connections and are, therefore, excluded from synchronization.

By using functions in PCS 7, the adjustments made to the CM are also visible outside of PCS 7 (e.g., when exporting to the PAA) without the need to adjust the CMTs.

In the Plant Automation Accelerator, functions offer the possibility to adapt the CM without changing the CMT and without having to use an additional CM. When exporting to PCS 7, the function is integrated into the instance (CM).



Note

A description of how to create and use a function with SIMATIC PCS 7 and the Plant Automation Accelerator can be found in section 5.4.

Individual complex calculations or logic can be centrally managed and adapted easily by functions.

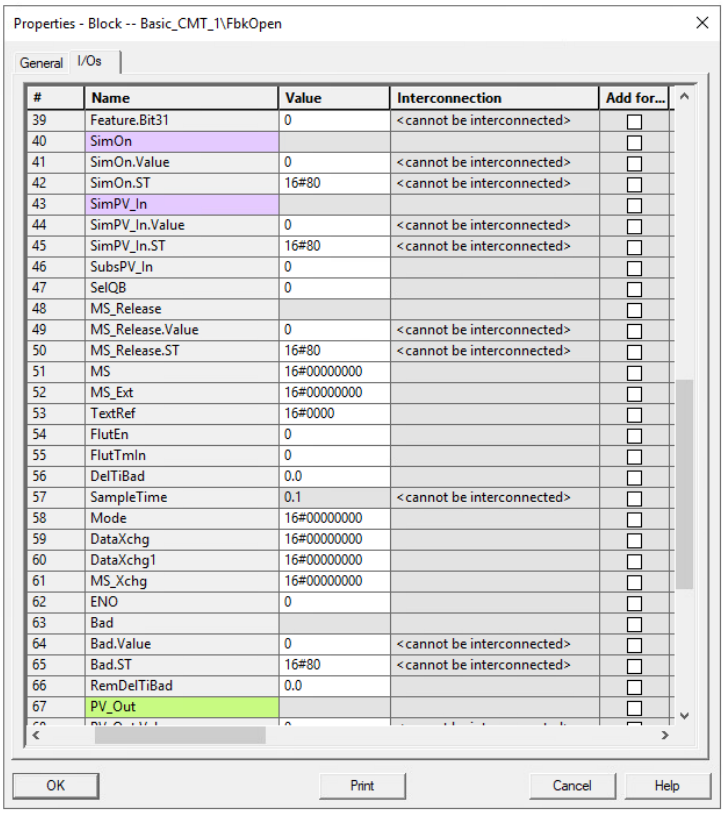
2.11. Parameters that should not be synchronized

Parameters that are excluded from type instance synchronization are highlighted in pink. As a result, the type instance concept for these parameters is suspended. For this reason, this function should only be used with caution and only in certain situations.

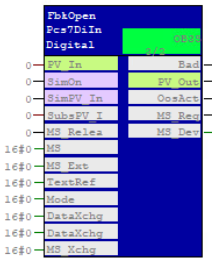
It may be useful to use it during commissioning. Usually, the inputs "SimOn" and "SimPV" of the APL blocks are not created as technological I/Os. This means that these cannot be simulated during commissioning in the CFC as they usually could.

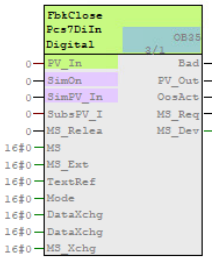
To be able to continue setting the inputs during setup, it is recommended to exclude these parameters from the synchronization (parameters marked in pink).

When commissioning is complete, these settings are reset centrally in the type. This has the advantage that the default values (e.g., SimOn=0) are transferred to the instances when a new synchronization is performed. Consequently, all channel simulations are terminated.



#	Name	Value	Interconnection	Add for...
39	Feature.Bit31	0	<cannot be interconnected>	
40	SimOn			
41	SimOn.Value	0	<cannot be interconnected>	
42	SimOn.ST	16#80	<cannot be interconnected>	
43	SimPV_In			
44	SimPV_In.Value	0	<cannot be interconnected>	
45	SimPV_In.ST	16#80	<cannot be interconnected>	
46	SubsPV_In	0		
47	SelQB	0		
48	MS_Release			
49	MS_Release.Value	0	<cannot be interconnected>	
50	MS_Release.ST	16#80	<cannot be interconnected>	
51	MS	16#00000000		
52	MS_Ext	16#00000000		
53	TextRef	16#0000		
54	FlutEn	0		
55	FlutTmIn	0		
56	DelTiBad	0.0		
57	SampleTime	0.1	<cannot be interconnected>	
58	Mode	16#00000000		
59	DataXchg	16#00000000		
60	DataXchg1	16#00000000		
61	MS_Xchg	16#00000000		
62	ENO	0		
63	Bad			
64	Bad.Value	0	<cannot be interconnected>	
65	Bad.ST	16#80	<cannot be interconnected>	
66	RemDelTiBad	0.0		
67	PV_Out			



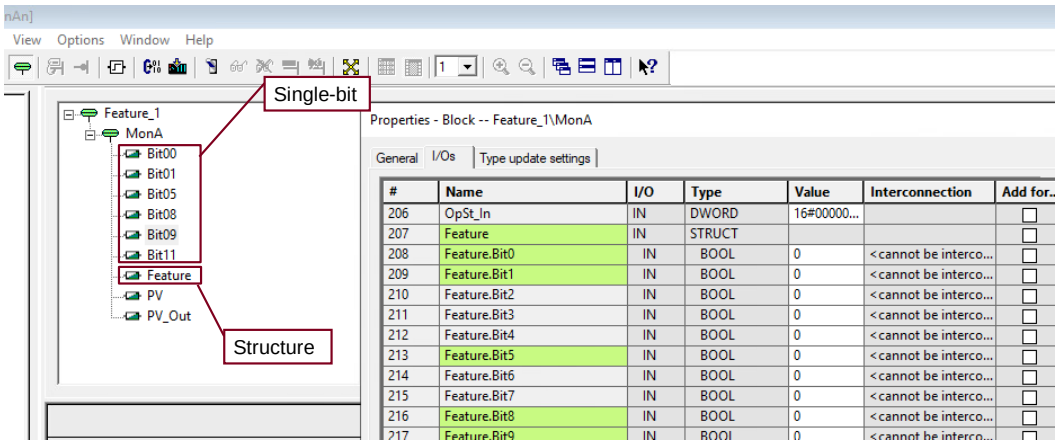


2.12. Feature bits/ OS-Perm

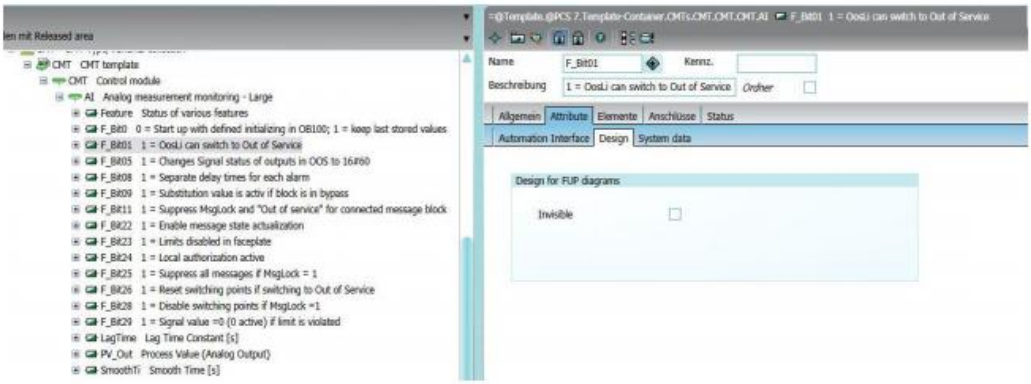
If the bits feature will be adapted instance-specifically, we recommend creating the structure and the individual bits as technological parameters.

To be able to change the bits within an instance in PCS 7, it is sufficient to create the structure only (STRUCT in the figure below). In this case, however, no technological access to the values of this structure is possible. This means that the bits can only be changed in the CFC, but not in the list editor. For COMOS and PAA, it is necessary to create individual feature bits (single bits in the figure).

The OS-Perm parameterization is usually defined in the type. If this is also to be changed in an instance-specific manner, the same specifications apply. In this case, it is helpful to prefix the names of the bits with a prefix such as "F" or "OS". This makes it easier to distinguish the individual bits. If single-digit bits are provided with a presented "0", the bits are also displayed in the correct technological order.



In order to keep the function plans (FBD) in COMOS/PAA clear, the bits can be switched to being invisible.



3. Principle of operation

3.1. Technological I/Os and variants

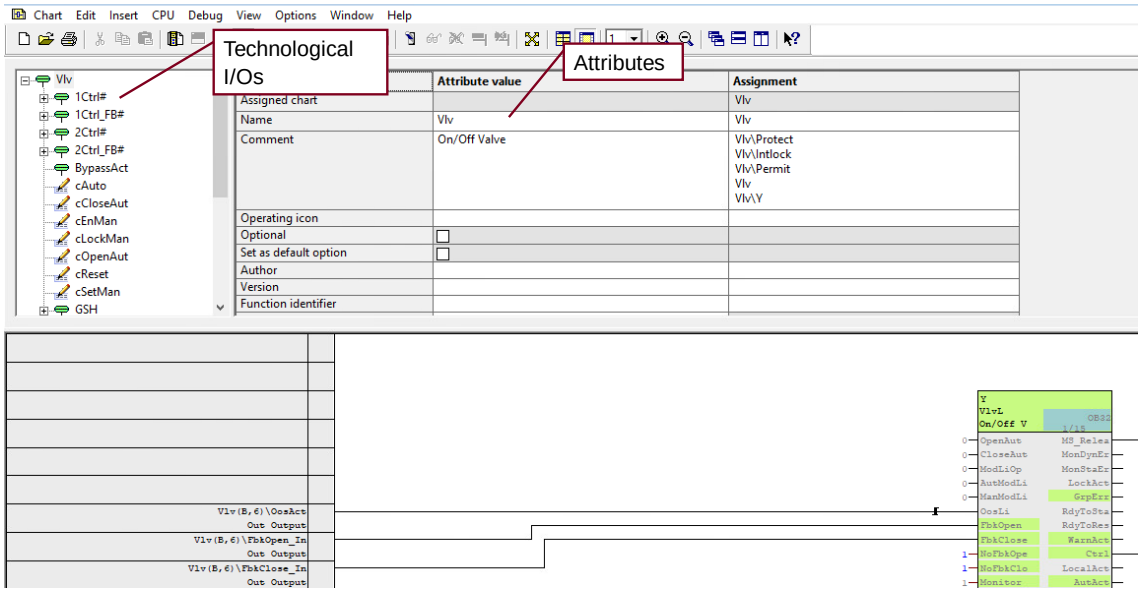
The technological I/Os are the interfaces of the CM to other CMs and provide a simplified view of the CM, with all signals, connections, and parameters that are relevant from a technological point of view. Due to the instance-specific parameterized or interconnected signals, interconnections, and attributes, the instance-specific changes are retained during the synchronization.

CAUTION

CM Engineering

Connections that are created at CFC level between non-technological I/Os of two CMs are not available for the abstract data model. This means that, when using the technological list editor, as in COMOS and PCS 7 PAA, they cannot be displayed or interconnected.

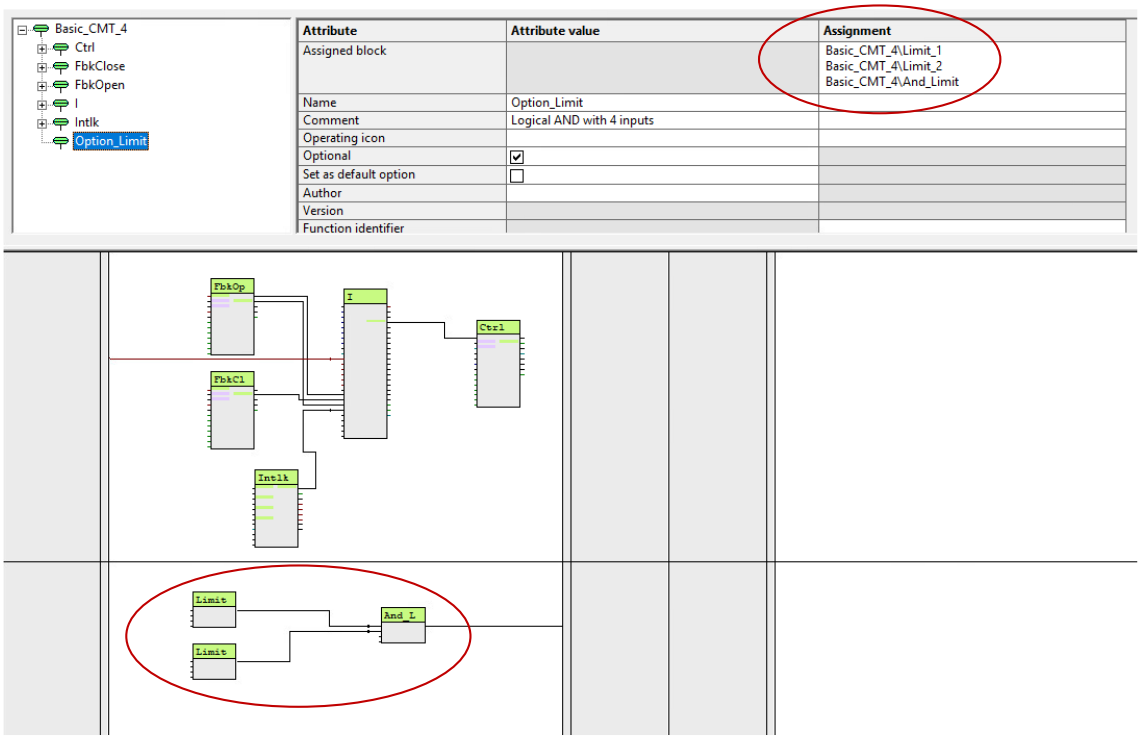
This can lead to undesired behavior in the case of a later change in the CMT and the synchronization with the instance. Therefore, additional wiring of the instance should always be routed via the technological I/Os.



The technological I/Os can be created and extended in a user-friendly way via drag&drop. An added object is assigned automatically.

The following objects are available in the technological I/Os:

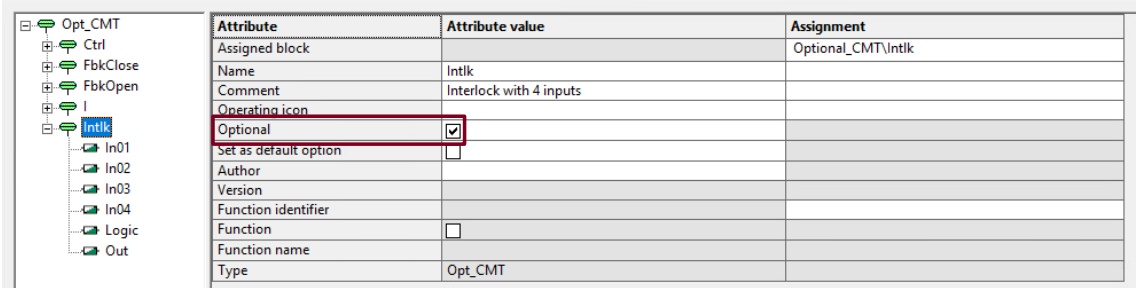
- Control Module: Topmost object in the structure tree of the technological I/Os. Represents the entire CMT or CM with the assigned CFC and subordinate objects of the technological I/Os.
- Sub Control Module: SubCMs are subordinate functions of the Control Module. One or more function blocks of the CFC can be assigned to a SubCM.
- Also, if there are blocks that are optional and, together, represent a logical function, they should be created as a SubCM. This option allows several blocks to be added and removed. In addition, a SubCM can be marked as optional to be switched on or off in the instance of the Control Module. This enables a CMT in PCS 7 to be varied at instance level. Optional SubCMs are thus the basis for variants of a CMT.



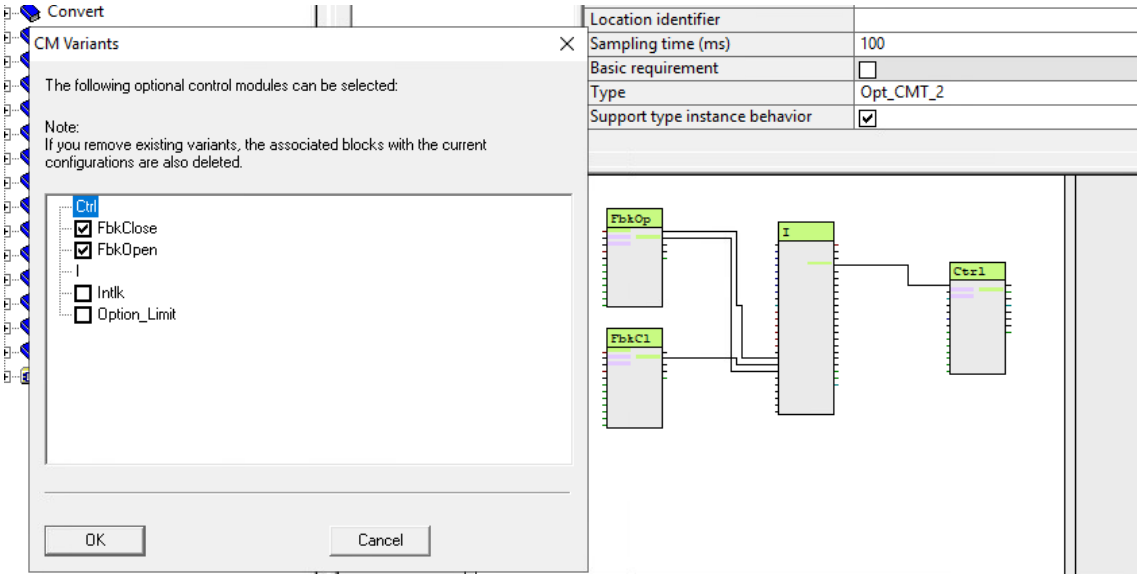
Variants

Variants allow you to define several components of a CMT as optional. These options can be enabled or disabled in an instance-specific manner.

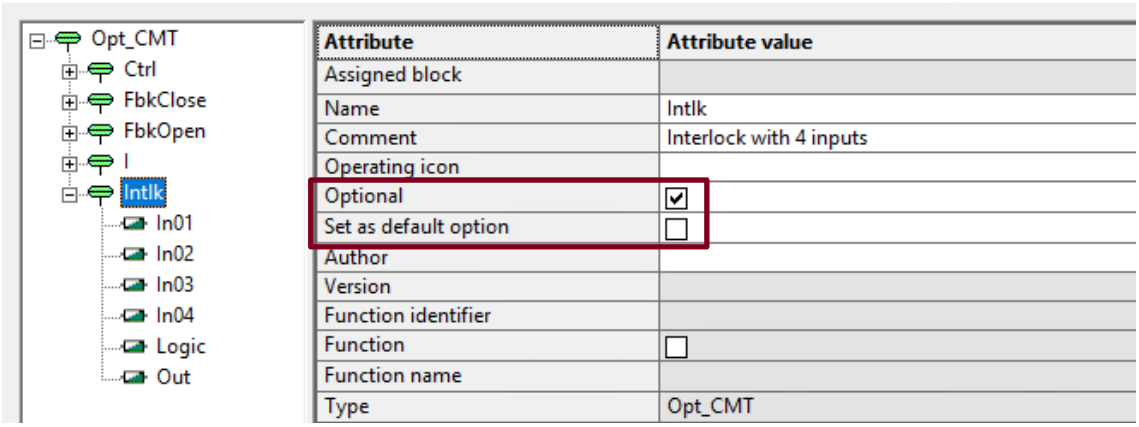
Example: Optional interlock block



Creating variants by activating and deactivating options



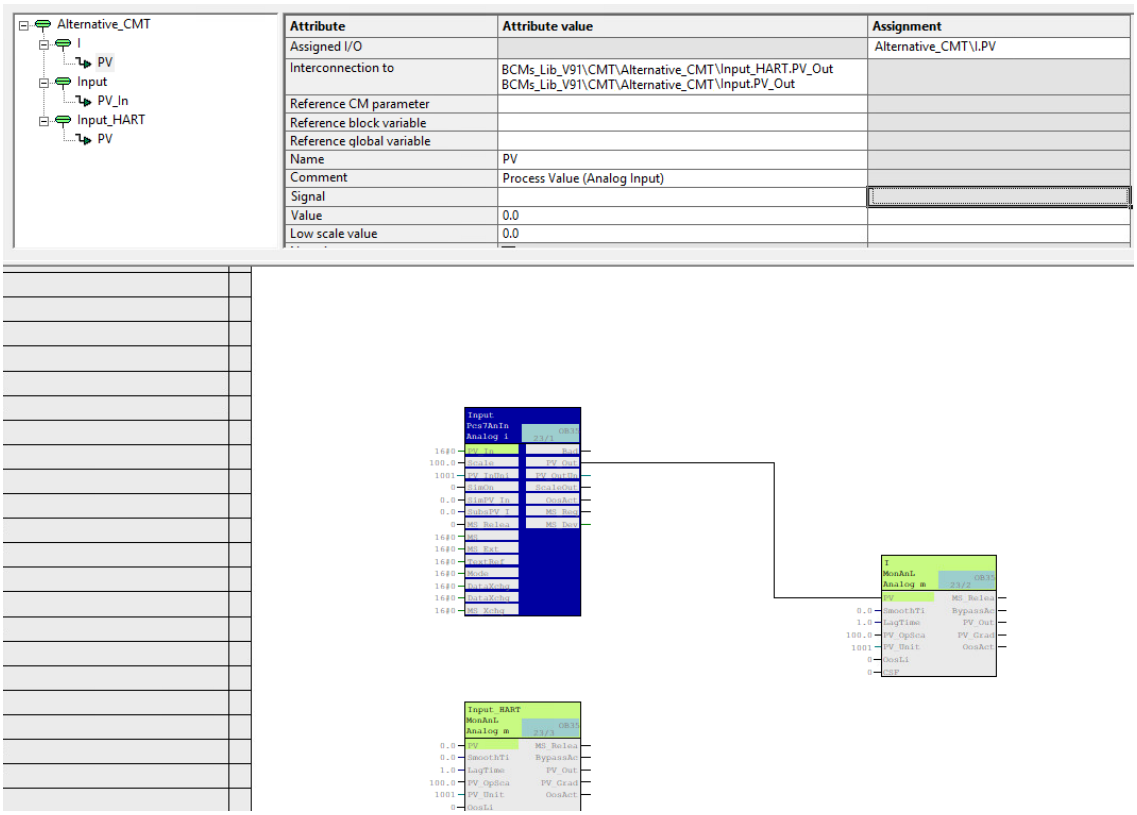
If an option is used with almost all instances, it can also be enabled by default. If an instance is created with "Set as default option" enabled, each new created instance will be activated with the option.



The alternative is a continuation of the variant. This allows you to configure "either/or variants". This is necessary if a block input will be configured with different connections depending on the variation. The instance must decide on an alternative.

Example: Peripheral signal from different sources → Different driver modules (hard driver, standard PCS 7 driver) are required.

NOTE In case of an alternative and the option "Set as default option", the interconnection with the "Set as default option" option must be in the technological connections as the top-level interconnection in the assignment. Otherwise, the connection will not be created by default when instantiating.



- Parameter: Block input or output with a pre-programmed value or an interconnection/multiple interconnection that can later be adapted in an instance-specific manner.
- Signal: Connection to input or output channels of the automation hardware.
- Messages: This object can be used to transfer the information and settings of messages from the function blocks of the CFC to the interface of the technological I/Os.
- For more information on creating notifications, see the following article "How can you incorporate messages in a CMT (Control Module Type) in SIMATIC PCS 7?": https://support.industry.siemens.com/cs/ww/en/view/63367955

NOTE When importing to COMOS/PAA, ensure that the language settings are identical.

- Status: The various individual conditions and OS comments are predefined in the status so that the status of the instance can be queried more easily by an SFC.
- Command: The various initializations, edits, and terminations are predefined in the command so that access to the instance is facilitated by an SFC.

NOTE All block parameters/connections assigned in the technological I/Os are displayed in green on the block and are not changed during synchronization.

Attributes

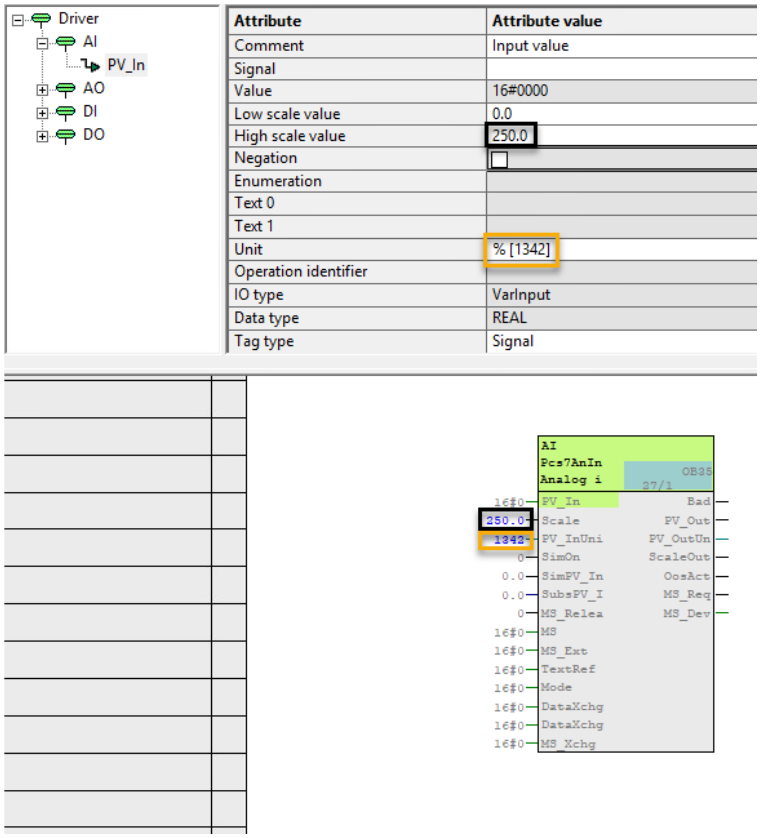
In the "Attributes" area, the available attributes of an object (name, option, value, unit, etc.) are displayed and assigned CM or linked connections are displayed. In addition, the assignment between the technological I/Os and the internal logic in the CFC is carried out in the attributes.

Each technological I/Os object has different attributes. While the designation for the connection is assigned to a block or block group or the "Optional" function is set, a process value and/or a unit can be preset for a lower-level parameter.

3.2. Peripheral Signals (Technological Inputs/Outputs) with APL Drivers

The properties of a signal (e.g., upper/lower limit and unit, see Figure) are automatically linked to the block's connections when APL drivers are used. When using COMOS/PAA, this technique is preferred for analog signals, since a signal in the database also has exactly three parameters.

When using SIMIT V10.1 or older Versions, it is necessary to create the upper and lower limits as independent parameters, otherwise, SIMIT cannot access the values.



NOTE

Further details about the units can be found in chapter 2 of the documentation of the FAQ "Which Units of measurement can be configured in the SIMATIC PCS 7 Plant Automation Accelerator?": https://support.industry.siemens.com/cs/ww/en/view/109780555

3.3. Internal Interconnections

In order to implement a clear type-instance concept, defining all interconnections firmly in the type is recommended. Basically, the following four variants of interconnections are possible.

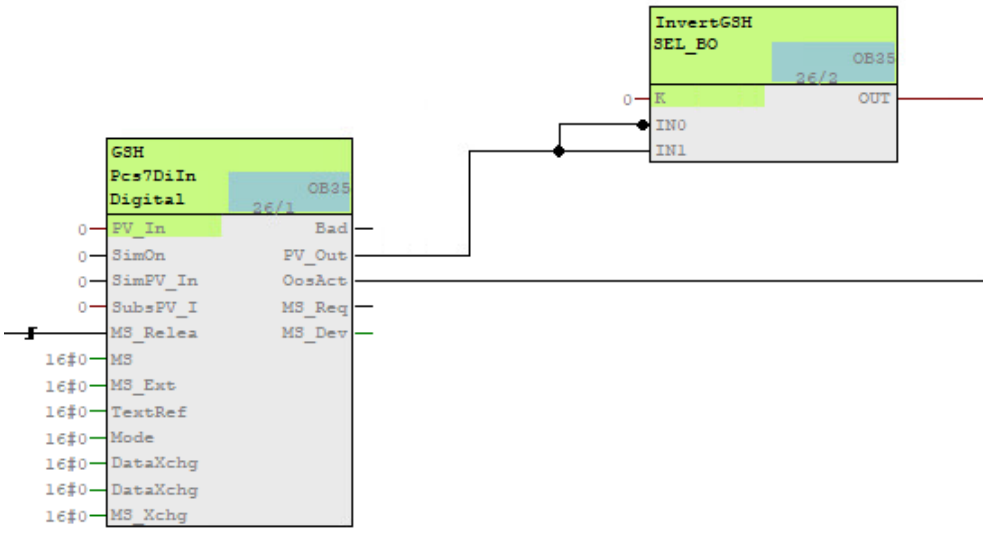
Input	Output	Interconnection can be changed in the CM?	Synchronization behavior	Application
Gray	Gray	No	Is adapted to type	Fixed wiring
Green	Green	No	Is adapted to type	Fixed technical interconnection (options/alternatives)
Green	Gray	Yes	Remains as in the instance	Instance-specific wiring
Gray	Green	No	Is adapted to type	No application known

3.4. Negations

For negations to technological I/Os (green connections), the use of the EMERGENCY block is recommended. Another suitable method is to use the negation parameter (e.g., interlock) on the target module and place it on the CMT interface. In contrast to the inversion at the port (CTRL+R), these are also clearly defined technologically and can be mapped in the PAA and COMOS. In addition, the plan is easier to read.

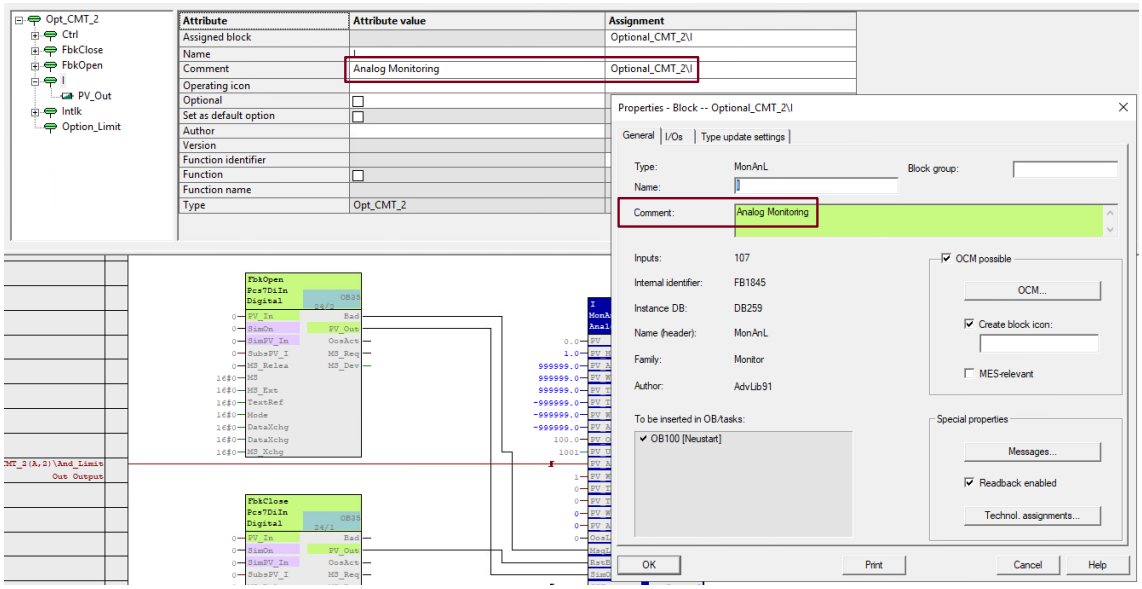
For connections that cannot be changed in the instance (gray-gray), the inversion at the channel can still be used.

Example: For the inversion of binary signals of different characteristics (Low - High, High - Low), an implementation similar to the BCM is recommended.



3.5. Comment Inheritance

The CFC comment can be transferred to the display module via the technological assignment. Therefore, the comment (e.g. "Agitator Tank 1") is automatically displayed on the faceplate.



3.6. Runtime Groups

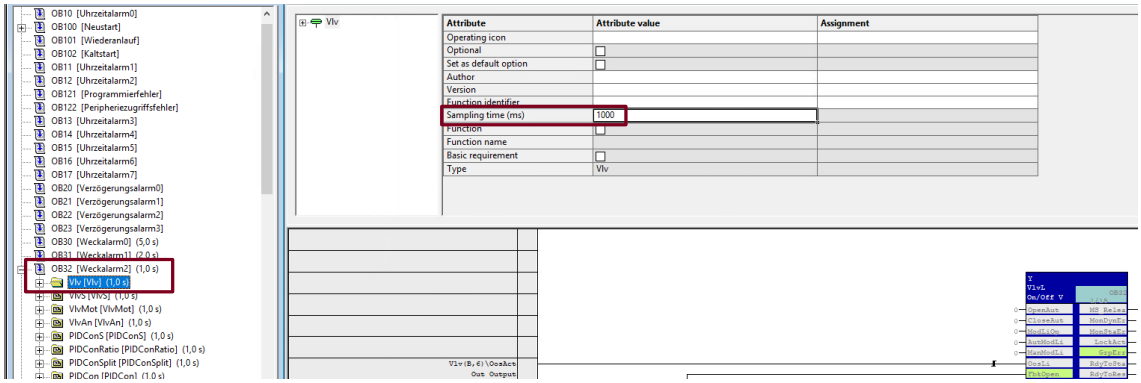
For CMT, a plan-oriented installation in runtime groups is always necessary and preset. This ensures that a plan lies within a single runtime group.

When an instance (CM) is created, the runtime group is included in the cyclic interrupt OB (e.g., OB33, OB 34, etc.) as in the library. The sampling time is adjusted according to the CPU configuration. If a scan time is configured in the CPU that differs from the library, the scan time of the CPU is used.

If the sampling time is changed in the CM, the runtime group is automatically installed in the corresponding cyclic interrupt OB. If a sampling time is entered for which there is no cyclic interrupt OB, it is replaced by the next later runtime.

NOTE The reduction or phase shifting of the CM should not be used in this case.

Example: If, in the configuration from the figure below, the sampling time is changed from 1000 ms to 1200 ms, the changed value will be replaced immediately by 2000 ms (OB31 – next higher cyclic interrupt).



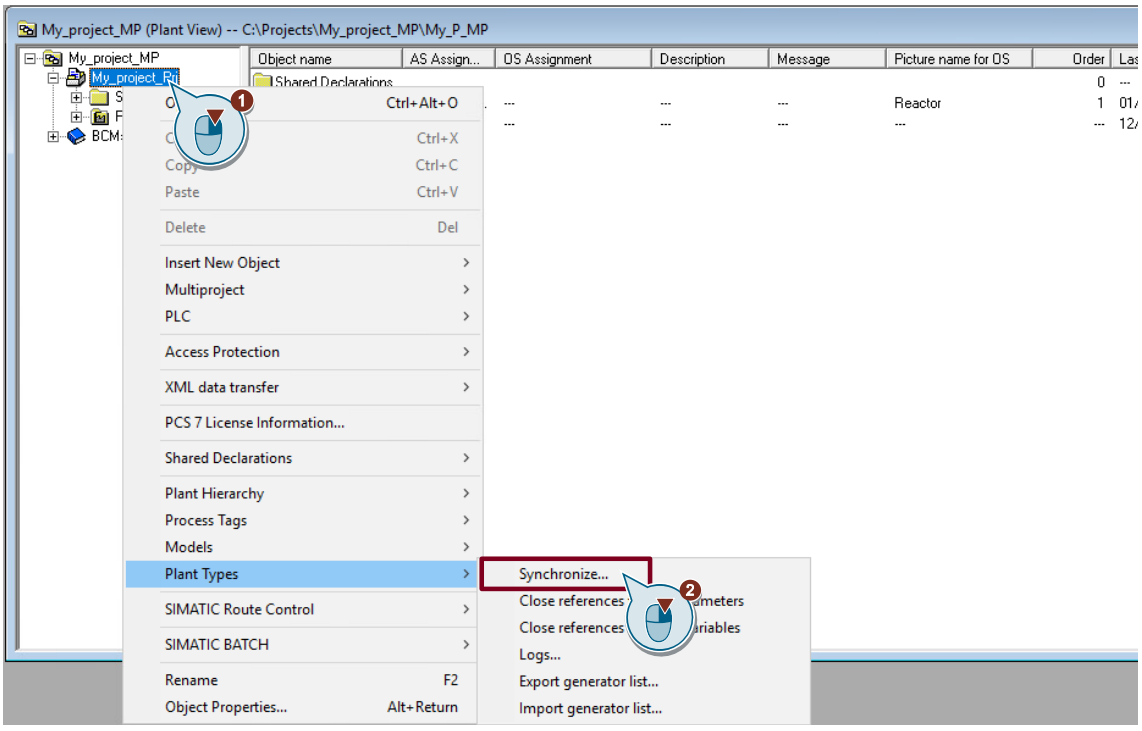
3.7. Synchronization Functionality

In PCS 7, the synchronization function is performed for exactly one project. All instances and detected changes between type and instance are displayed in the file transfer dialog. The user can deselect the differences that are not to be adjusted. For example, blocks that were added in an instance are left in the synchronization. While blocks or connections that have been added in the type are transferred to the instances during synchronization. Since the function "Synchronization of Control Modules" uses parts of the Version Cross Manager (VXM), a license (6ES7658-1CX58-2YA5) of the VXM on the Engineering Station is mandatory for using this function.

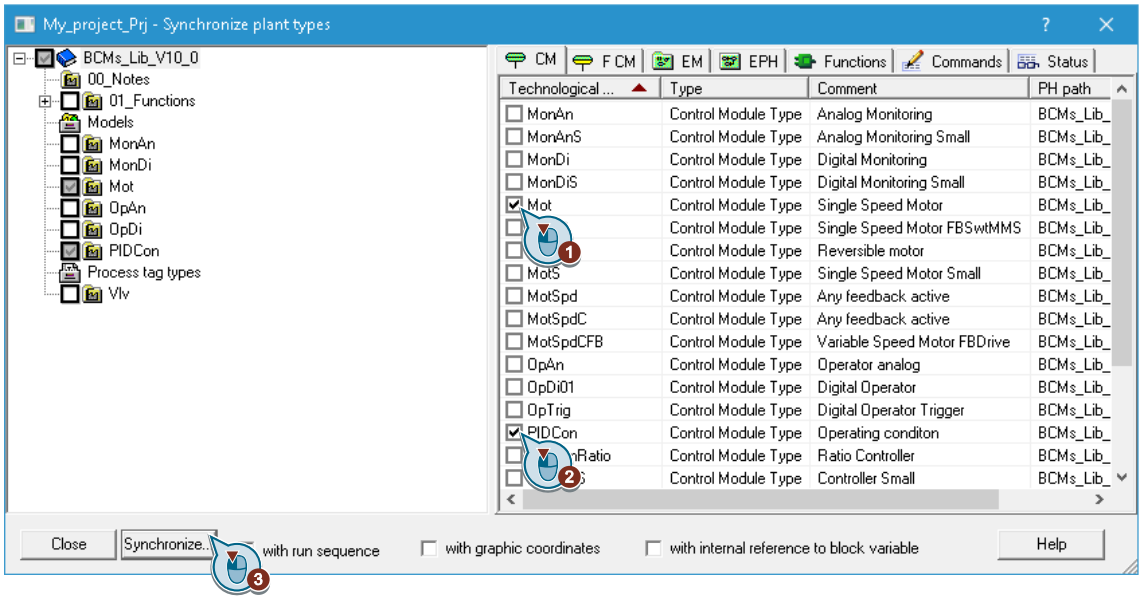
NOTE The instance-specific connections that conflict with a new connection in the CMT are replaced by the connections in the CMT.

NOTE Filter settings, such as deselecting instances or subordinate objects, are not retained after closing the file transfer dialog.

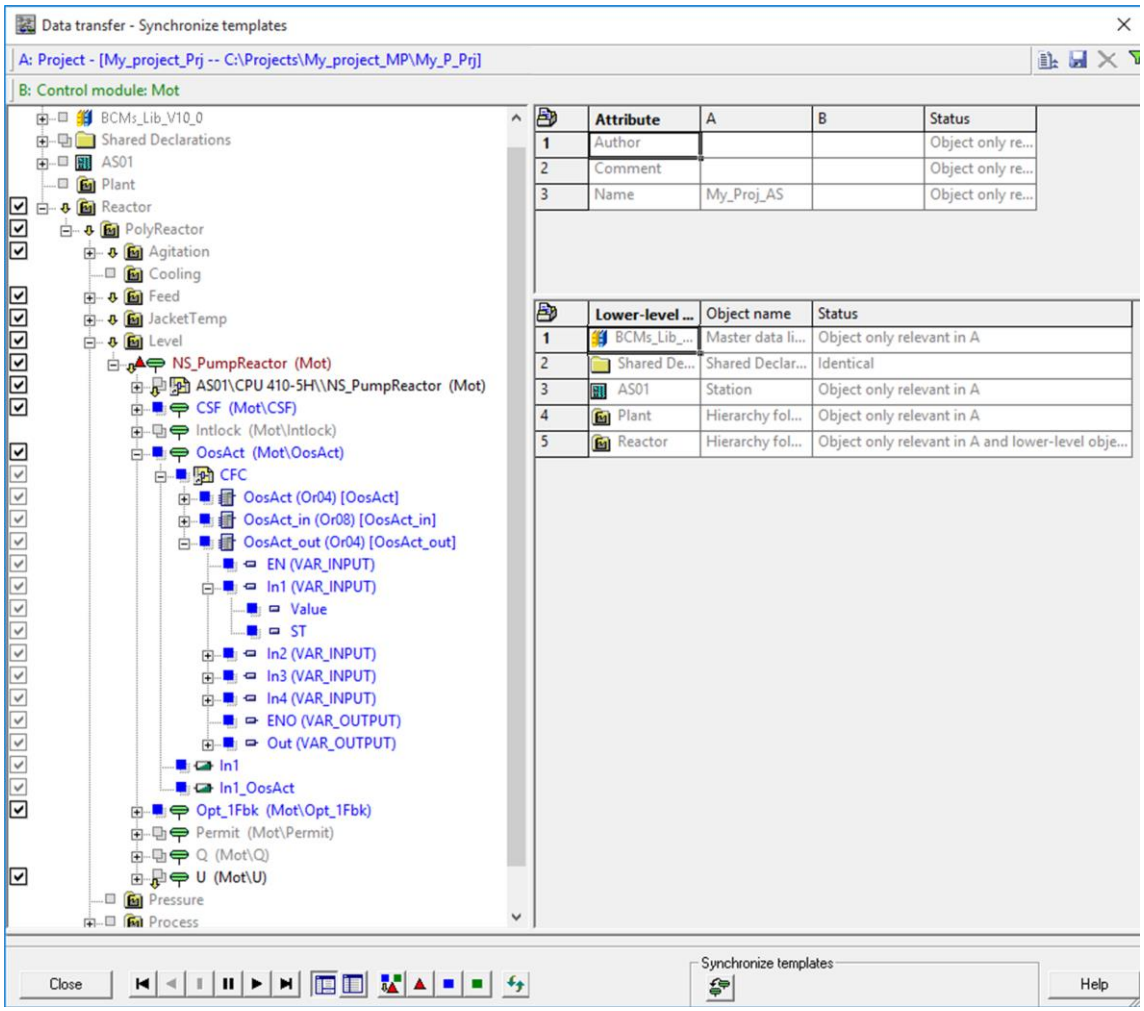
1. Select the AS project in the plant view of the SIMATIC Manager and select "Technological Types > Synchronize..." in the context menu.



2. Select the types you want to synchronize and click the "Synchronize..." button.



3. In the comparison dialog, all folders are displayed in which the previously selected CMTs were created as an instance. Changes made to connections that are not technological I/Os are displayed at the instance in question.



NOTE

If two blocks are connected together in the type and the target connection is defined as a "Technological I/Os ", the change is retained in the synchronization when the connection is deleted in the instance.
If both connections are not defined as "Technological I/Os" in the type and the connection is deleted in the instance, the original state is restored during the synchronization.

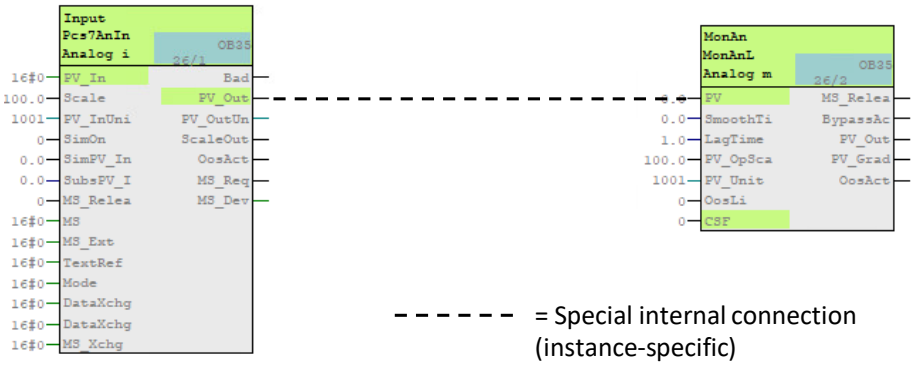
NOTE

For more details on synchronization, refer to the following link in the "Synchronization of individual control module types (PCS 7 V9.0 SP1)" user guide:
https://support.industry.siemens.com/cs/ww/en/view/109758382

4. Advanced Technological Configuration

4.1. Special Internal Connections (Instance-Specific)

If a connection will be instance-specific, this connection is not configured in the type or is left open. The affected inputs/outputs must be defined as technological I/Os in order to be connected in the instance. The connection must be configured individually for each instance.



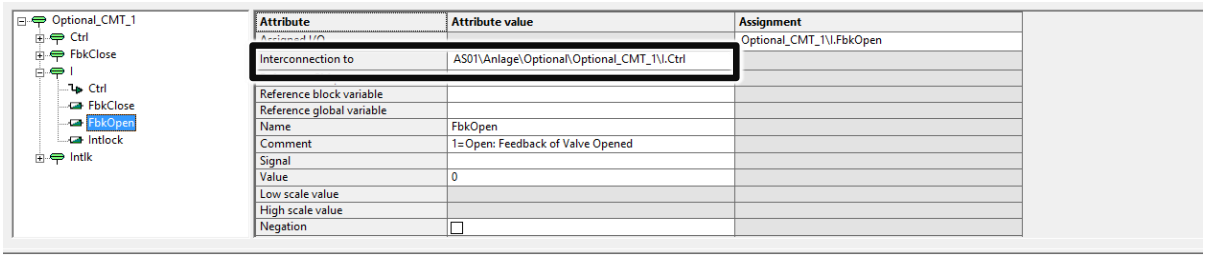
4.2. Reconnections

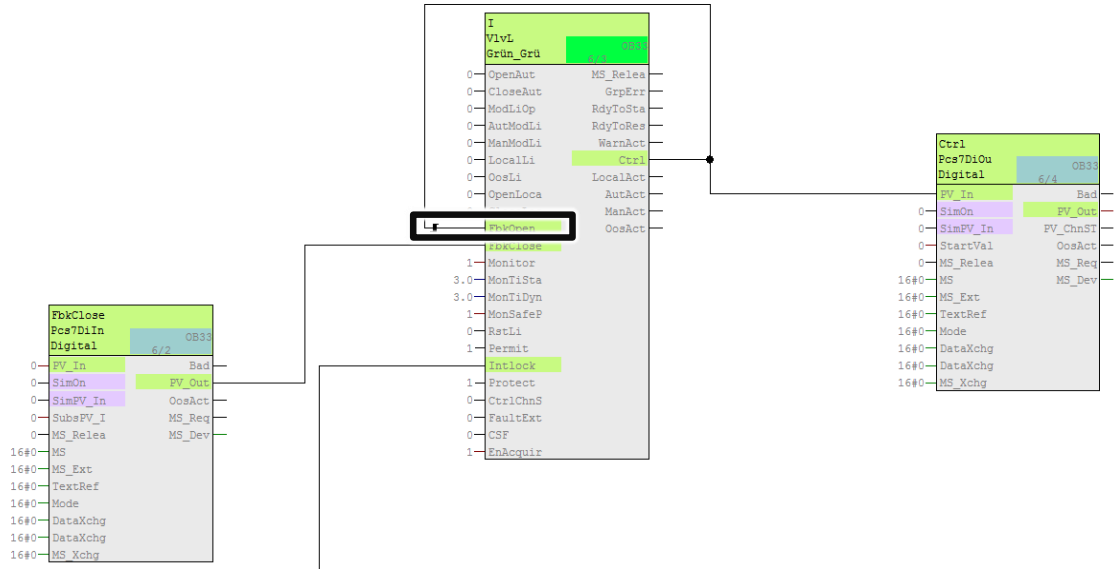
If an option within the CMT has fewer signals than the basic setup, it may be necessary to reconnect or short-circuit signals to ensure correct operation (see figure below).

NOTE If no driver modules are used, necessary back-connections in the instance must be added by the user.

Example: Valve > CTRL > FbkOpen

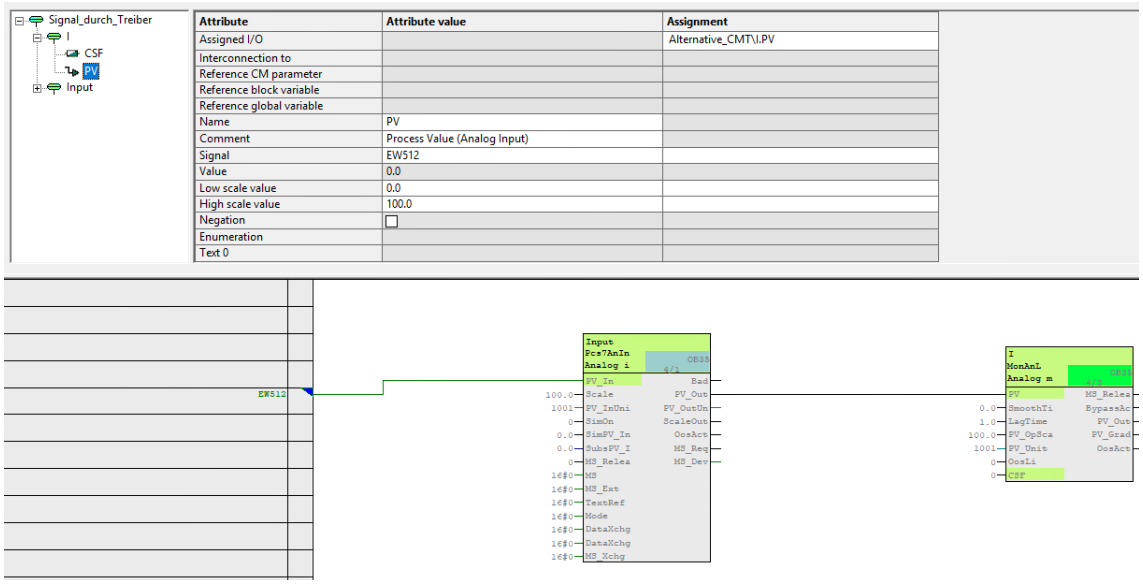
A backward connection is possible if the input is declared as technological I/Os. If both the input and the output are technological I/Os, the backward connection is technologically (e.g., in PAA, COMOS) visible ("connection to" in the technical editor).





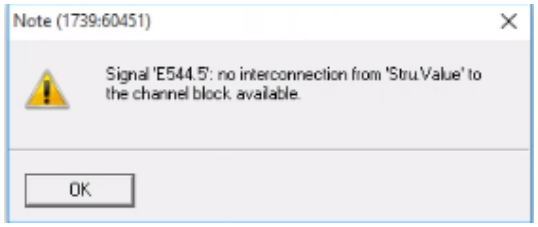
4.3. Signal Interconnection Directly at the Technological Block

When using APL drivers, signals can also be declared directly on the technical block. The operands are connected in the technological attribute value of the signal. The connection in the CFC via the edge bar (green connection) is automatically drawn. A direct connection between driver and block must be present in the type.



4.4. Peripheral Signals to "Non-APL Driver"

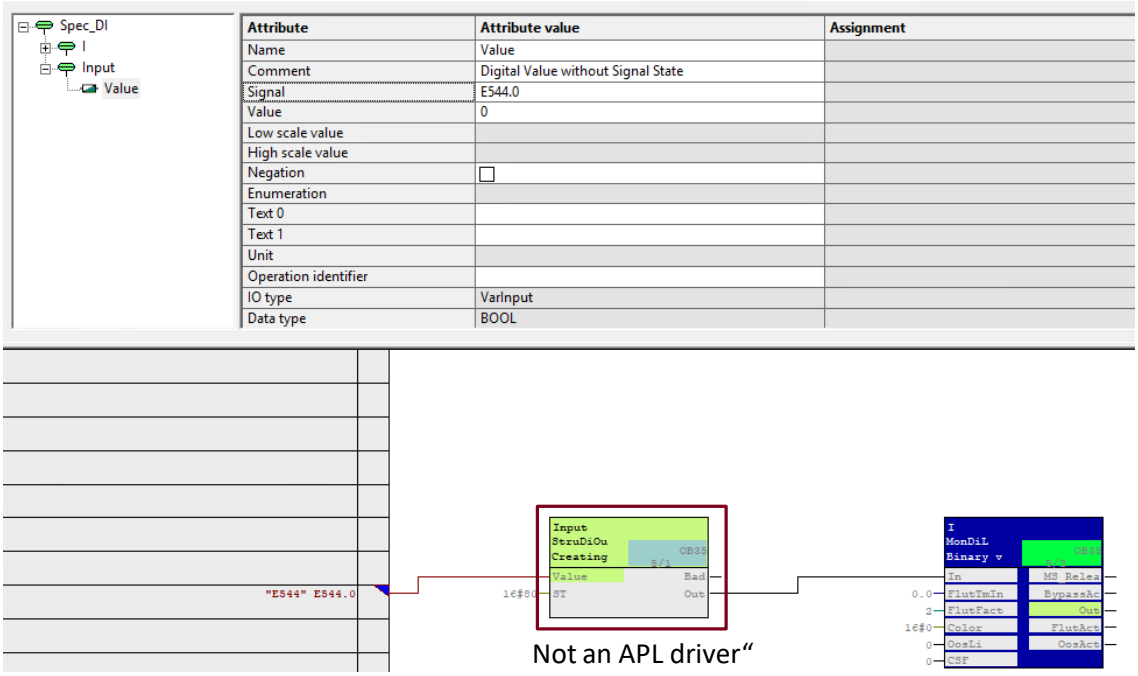
If "non-APL driver" blocks are connected directly to the peripherals, these must be made known to the system. Otherwise, a message is reported (figure below), because the system expects the same technique as in Section 4.3.



To make additional blocks known to the system as driver blocks, these must be entered in the XML "SignalBlocksAPL_90" (\Siemens\STEP7\S7data\SignalProcessing).

NOTE Before adapting the XML file, a backup copy of the original file should be created.

The syntax for entering new drivers is described in the manual "CFC for SIMATIC S7 (V10.0)", Section 11.1.13:
https://support.industry.siemens.com/cs/ww/en/view/109954532

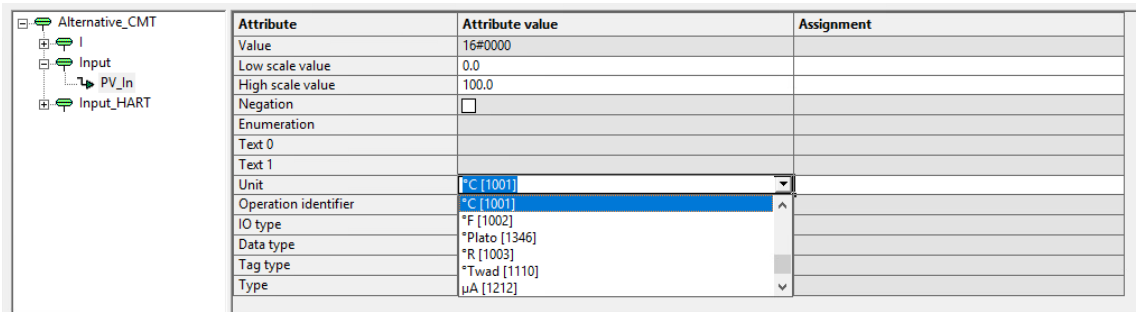


4.5. Units

As described in Section 3.2 "Peripheral Signals (Technological Inputs/Outputs) with APL Drivers", the unit is passed from the technological signal to the driver module.

The available units are stored in the file UnitMapping.xml (C:\Program Files (x86)\SIEMENS\STEP7\S7DATA\Units).

If user-defined units are required, they must be defined and entered in this file (UnitMapping.xml). For visualization in WinCC, the new units must be entered in APLCustomUnits.xml (\\OS\wincproj\\\GraCS). Only IDs up to 199 are permitted in WinCC.



NOTE

Further details about the units can be found in the entry "Which Units of measurement can be configured in the SIMATIC PCS 7 Plant Automation Accelerator?":
https://support.industry.siemens.com/cs/ww/en/view/109780555

5. Application scenarios

The following scenarios refer to the handling and engineering in PCS 7 by using the CM technology:

- CMT for measured value display with variants
- Configuring an equipment module with CMT
- Efficient function extension with APG through type matching
- Creating and using functions

5.1. Scenario A – Creating a CMT with variants

In this scenario, a CMT is configured for the measured value display. The CMT supports a large number of variants by means of optional SubCMs to which channel blocks are assigned. A CMT can be either generated from an existing measurement point in the project or built from scratch.

Initially, the user needs to think about the structure, static or variable parameters, block messages, as well as about possible variants.

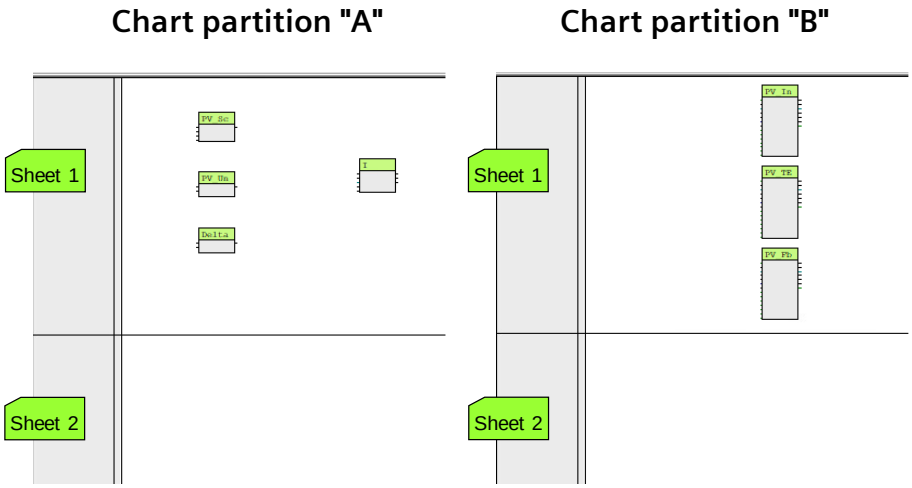
This example considers the following configuration:

- Measured value display via the "MonAnL" block
- 3 different channel drivers (analog, thermocouple and fieldbus) are supported
- As an alternative to the channel drivers, it is possible to choose a differential measurement
- Central parameterization of the scaling and unit
- Program logic and functionality in chart partition A and channel driver in chart partition B

Creating a CMT

In preparation, a new folder was created in the "Plant View" of an existing project library in the SIMATIC Manager. An empty CMT with the name "AMon" for measured value display has been added to the folder.

1. Open the CMT and create a second segment plan for channel drivers.
2. Add the following blocks with the corresponding names to the CFC.
    a. MonAnL block as "I" in chart partition A, sheet 1
    b. StruScOu block as "PV_Scale" in chart partition A, sheet 1
    c. DI_I-block as "PV_Unit" in Segment plan A page 1
    d. Sub02 block as "DeltaCalc" in chart partition A, sheet 1
    e. Pcs7AnIn block as "PV_In" in chart partition B, sheet 1
    f. Pcs7AnIn block as "PV_TE_In" in chart partition B, sheet 1
    g. FbAnIn block as "PV_Fb_In" in chart partition B, sheet 1



3. Switch the following block inputs and outputs visible or invisible.
   a. PV_Unit: Hides the inputs "In2", "SelMode", "Sel_In2" and the output "In2Selected"
   b. I: Display of inputs "PV_Hyst", all alarms, tolerance and warning limits "PV_xx_Lim", activation of limits "PV_xx_En", "MsgLock", "SelFp1", "SelFp2", "Feature", "MsgEvID1", "MsgEvID2" and outputs "PV_Grad", active limits "PV_xx_Act", "OosAct", "OnAct", "Status1" and "Status2".

Pre-configuration and interconnection

The following parameters are adjusted for the basic configuration:

- I: Deactivate all limit value messages "PV_xx_En" ("0")
- I: Preassign all upper limit values "PV_xH_Lim" to "99999.0"
- I: Preassign all lower limit values "PV_xL_Lim" to "-99999.0"
- PV_TE_In: Preset the scale to "0" (Low) and "1" (High)

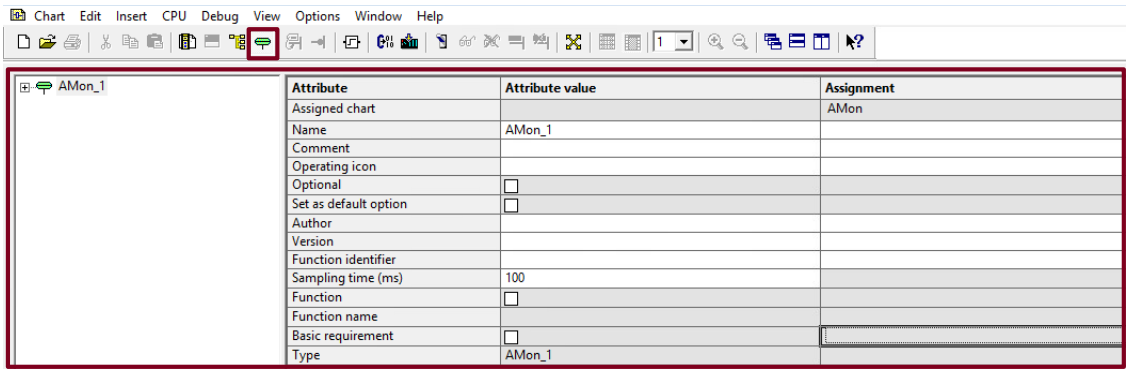
In addition, the following block interconnections are carried out:

Source (output)	Target (input)	Comment
PV_Scale.Scale	I.PV_OpScale PV_In.Scale PV_Fb_In.Scale	Central scaling of the process variable for display and channel driver
PV_Unit.Out	I.PV_Unit PV_TE_In.PV_InUnit PV_In.PV_Unit PV_Fb_In.PV_Unit	Central scaling of process unit for display and channel driver
PV_In.Bad	I.CSF	Display if process value is invalid
PV_In.PV_Out	I.PV	Display of the process value
PV_In.OosAct	I.OosLi	Indication when process device is in maintenance

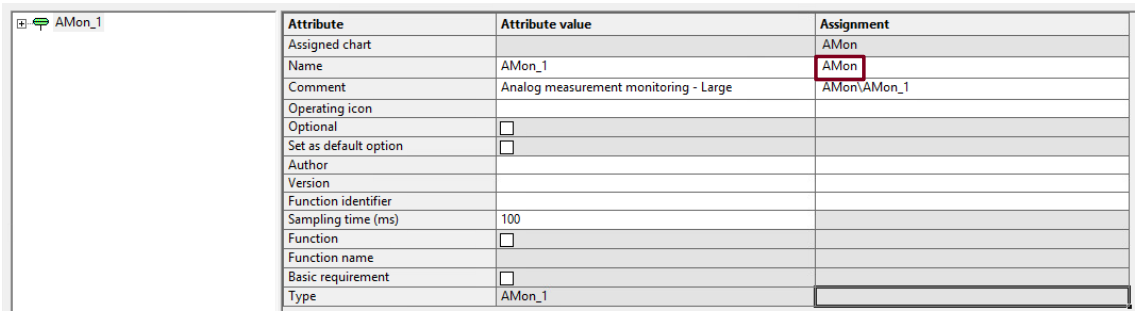
Synchronization parameters and messages

In the following, all parameters (inputs or outputs) and messages are created in the technological connections.

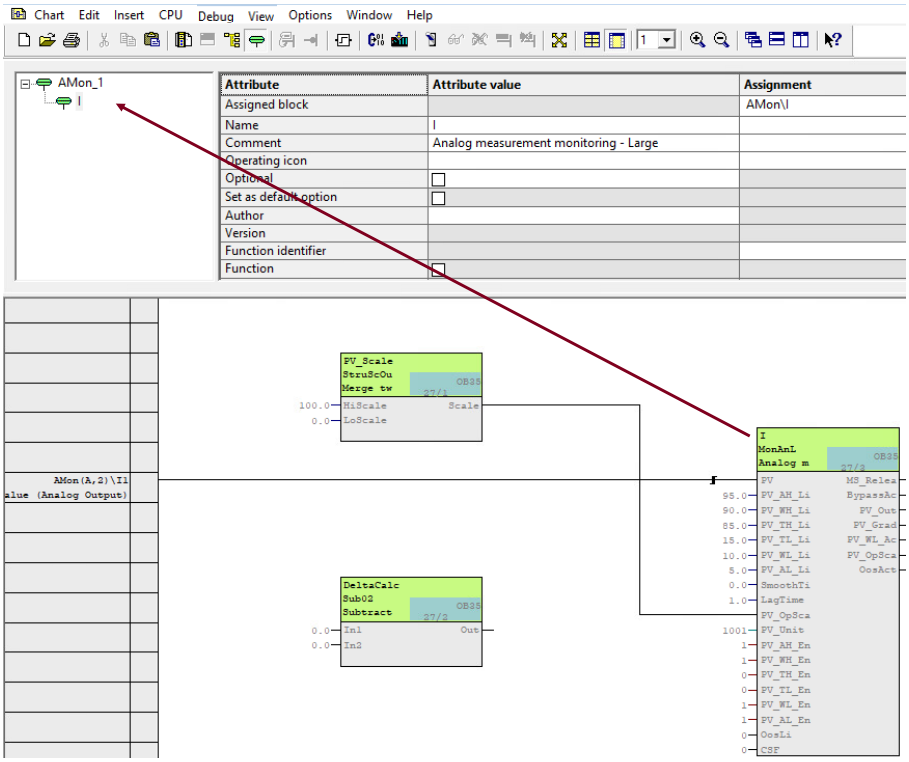
1. Open the "Technological I/Os" in the CMT.



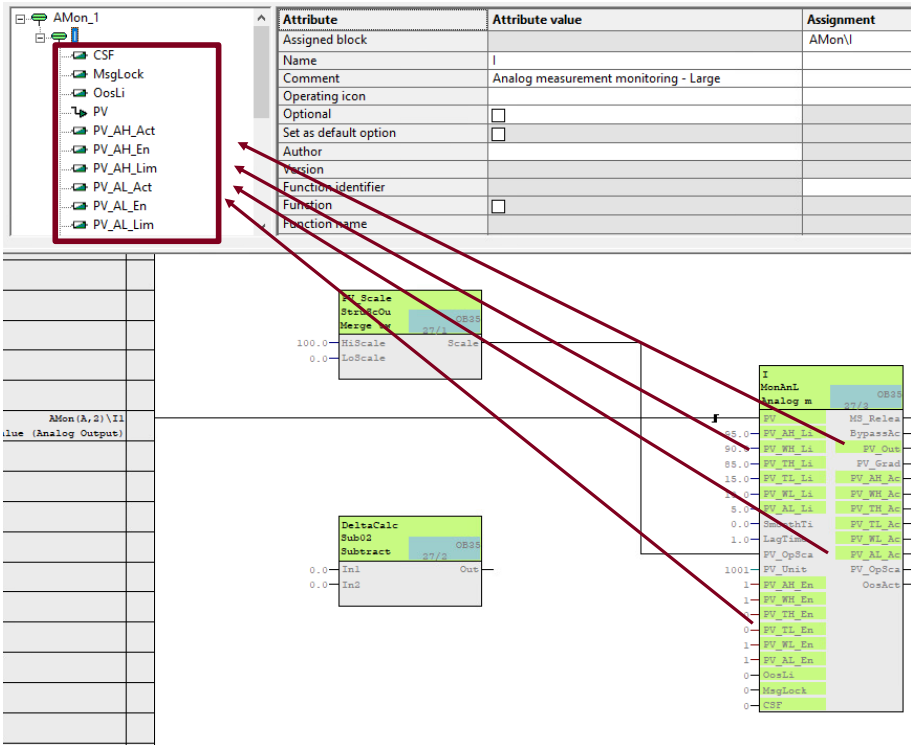
2. Enter the corresponding plan name in the "Assignment" column in the "Name" area.



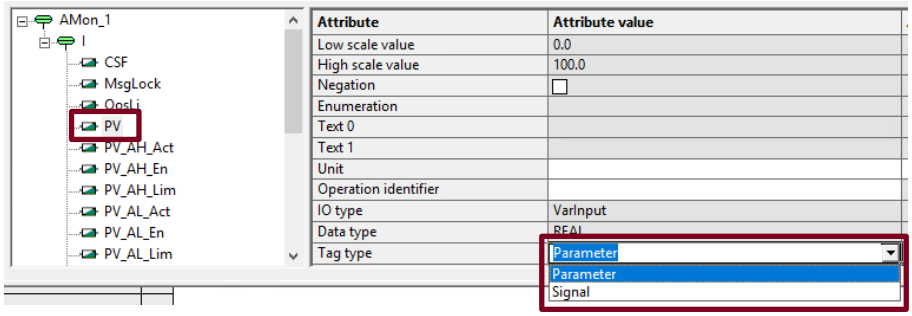
3. Add the display block "I" to the technological I/Os using drag&drop.
   A SubCM is created and the block "I" is assigned to it.



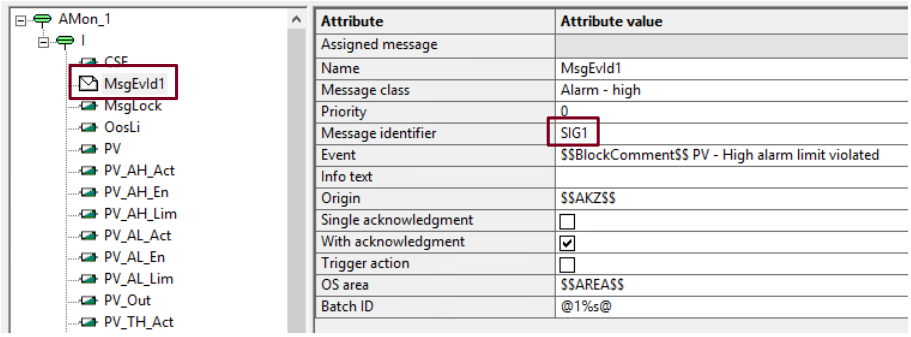
4. Drag each connection of the block (inputs and outputs) to the created CM "I" and update the screen display with the function key "F5".
   The assignment has been performed and the connected connections are displayed in green.



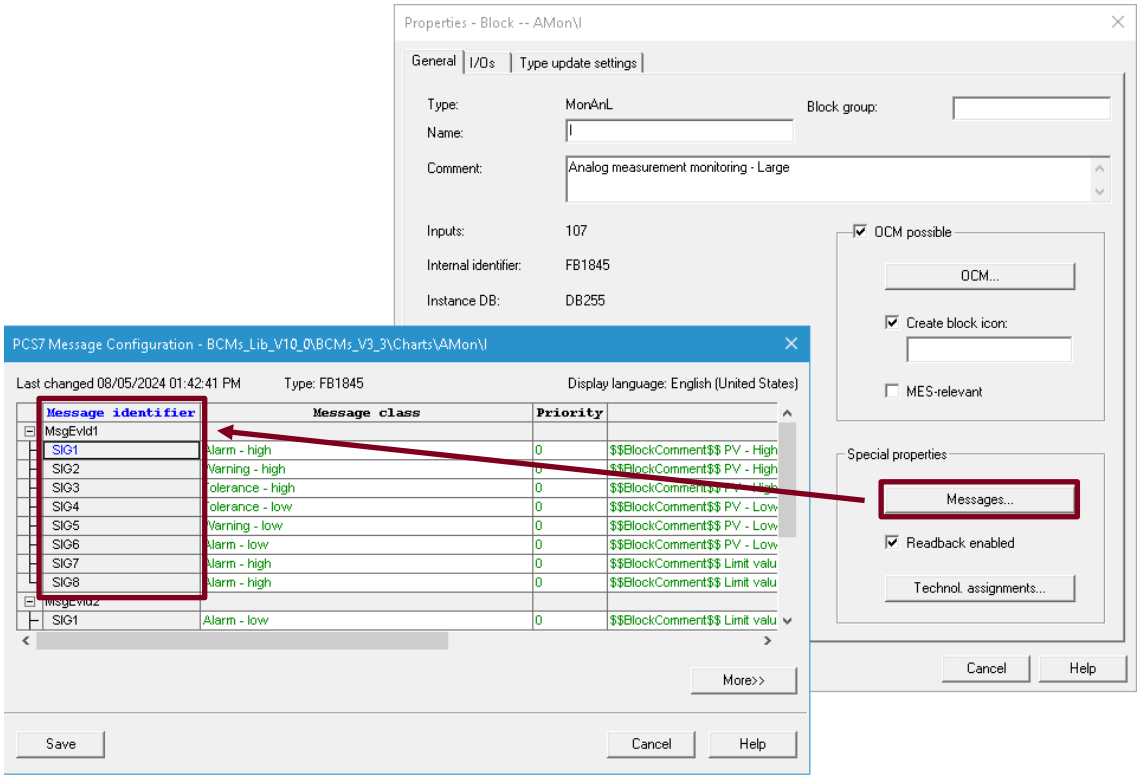
5. Change the "PV" attribute "Tag type" from Signal to Parameter. Only then can several connections be linked to the input.



6. Add the messages "MsgEvId1" to the technological I/Os by drag&drop. Enter the message identifier e.g. "SIG1" in the attributes of MsgEvID1.



NOTE In the object properties of the reportable block under "Messages..." you will find the available message identifiers. This is only possible with signalable blocks, such as display and controller modules.



7. Carry out step 5 for the message identifiers "SIG2" to "SIG8" and repeat the procedure for "MsgEvId2".
8. Add the following additional blocks and parameters to the technological I/Os:

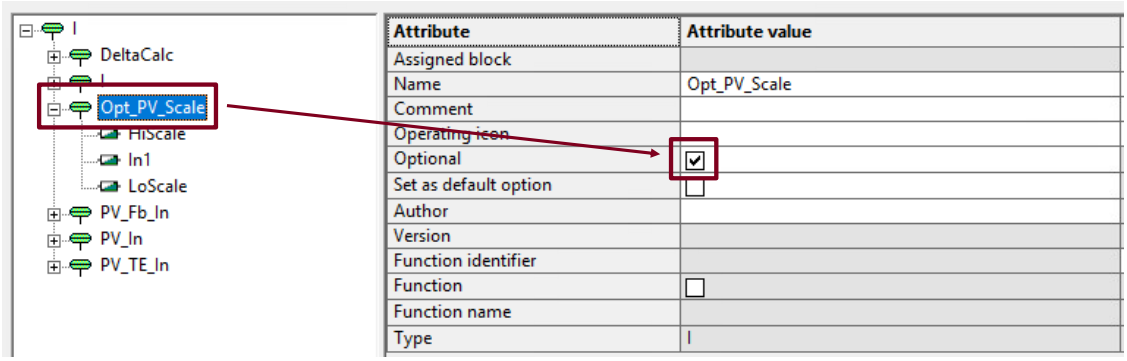
CM designation	Block with parameter
DeltaCalc	DeltaCalc.In1 DeltaCalc.In2 DeltaCalc.Out
Opt_PV_Scale **Note**: PV_Unit is also included in the group.	PV_Scale.HiScale PV_Scale.LoScale PV_Unit.In1
PV_Fb_In	PV_FB_In.PV PV_FB_In.PV_Li PV_FB_In.PV_ST PV_FB_In.Bad PV_FB_In.OosAct
PV_In	PV_In.PV_In PV_In.Bad PV_In.PV_Out PV_In.OosAct
PV_TE_In	PV_TE_In.PV_In PV_TE_In.Bad PV_TE_In.PV_Out PV_TE_In.OosAct

Multiple interconnections (variants)

Simple connections to a block input can be created as usual. For variants in which the selection of the interconnection partners changes, these options must be configured in the technological connections.

In the following, the CMT is preconfigured for the display of different process values (channel driver or difference formation). For this purpose, optional block or block groups, and the optional connections, are created in the technological I/Os.

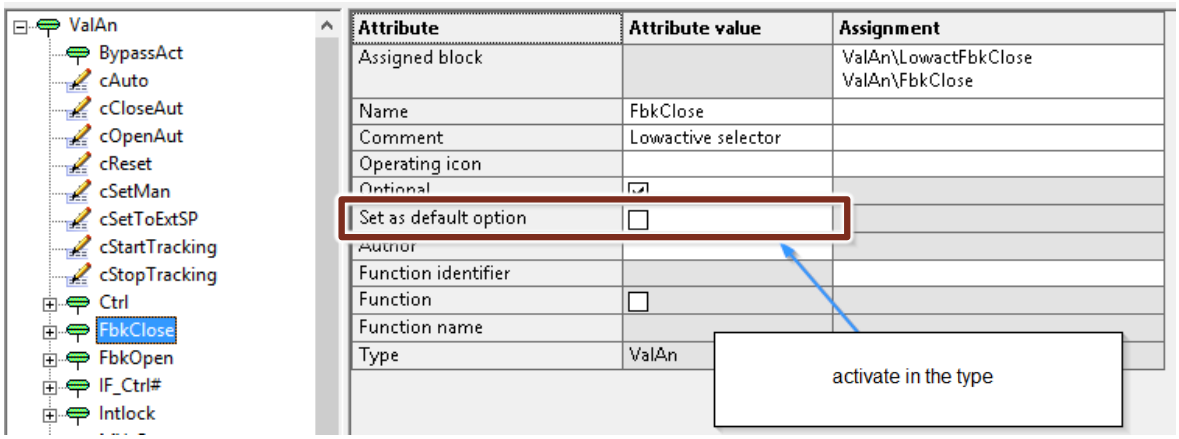
1. In the technological connectors, select the CM "Opt_PV_Scale" and enable the Optional attribute. This action can be used to deselect the central setting of the scaling and unit.



NOTE

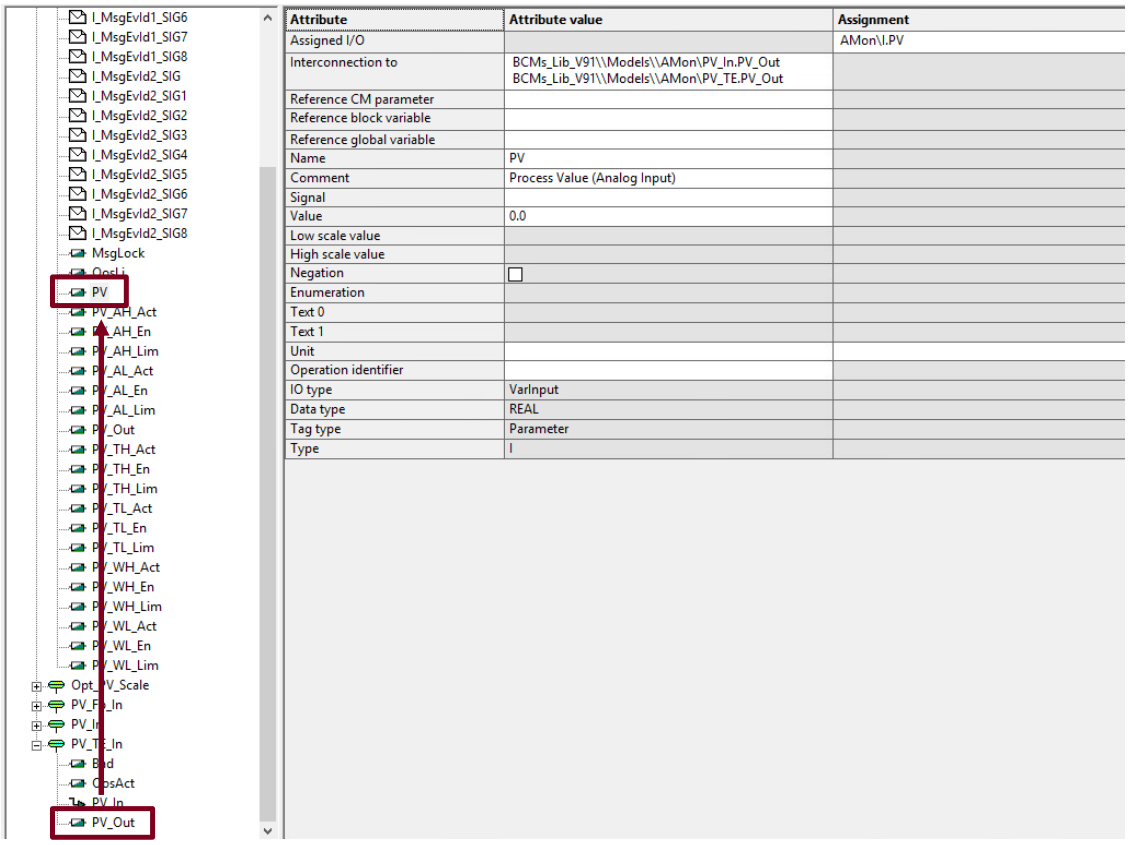
Setting the default option in the CMT allows an optional block to be selected by default for a new instance.

To do this, you only have to open the corresponding CMT in the master data library and select the option "Set as Default Option".

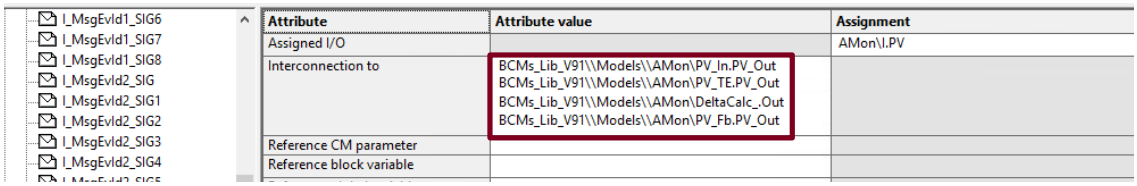


2. To create variants, activate the optional attribute for the CM "DeltaCalc", "PV_FB_In", "PV_In" and "PV_TE_In".

3. Link the other process value outputs of the channel drivers and the differential with the PV input of the display block using Drag&Drop.



NOTE All links or connections are listed in the attribute "Interconnected to".



4. Perform the wiring for the following parameters in the technological I/Os from the source to the destination:

Parameter source	Parameter target
PV_Fb_In.Bad PV_In.Bad PV_TE_In.Bad	I.CSF
PV_FB_In.OosAct PV_In.OosAct PV_TE_In.OosAct	I.OosLi

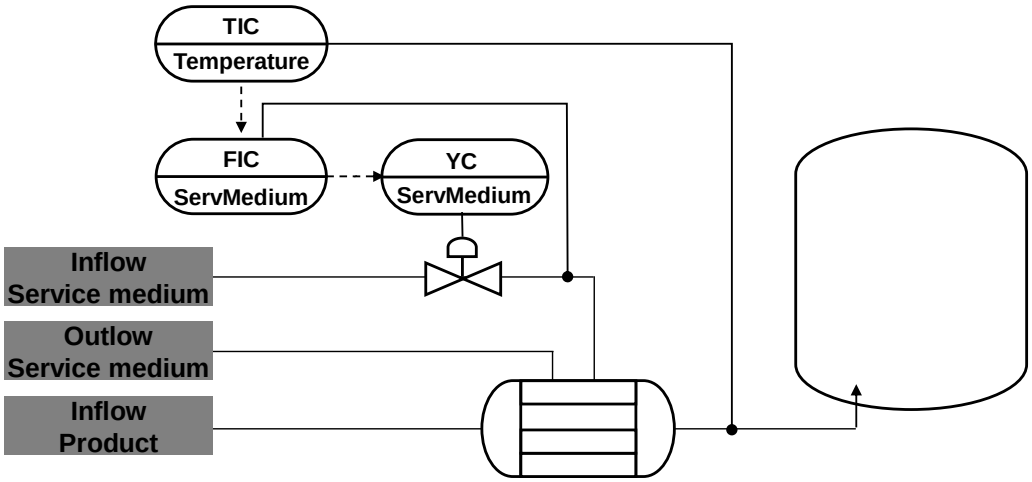
NOTE The multiple interconnections for variant creation only work if the attribute "Optional" was activated for all variant blocks (CM) at the beginning.

The connections from the parameter source to the parameter target are only made in the technological I/Os.

NOTE The display CMT is a component of the Basic Control Module library and the project "Equipment Modules for SIMATIC PCS 7 using the example of the Chemical Industry". You can download the sample project under the link https://support.industry.siemens.com/cs/ww/en/view/53843373

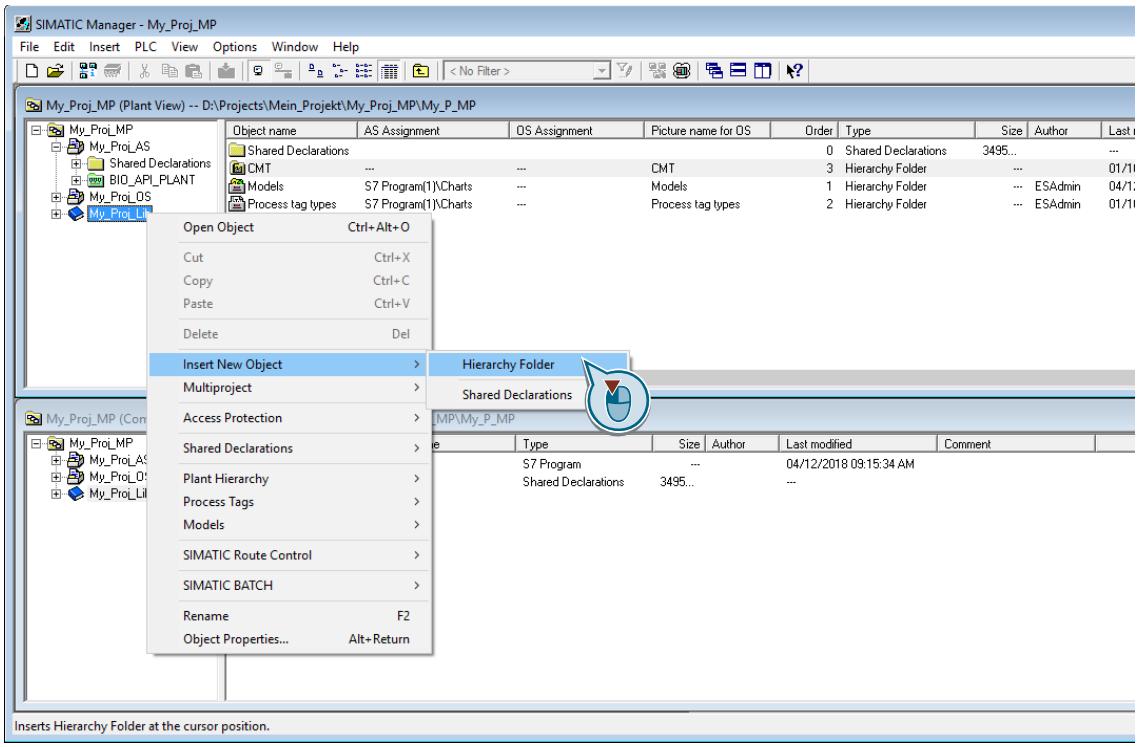
5.2. Scenario B - Configuring a cascade control with CMT

The basis for sustainable engineering in PCS 7 is the use of a master data library with CMT. For the following scenario, the Basic Control Module Type Library (BCM library) is used to create a temperature-flow cascade control "Temperature-Flow-Cascade". Cascade control is used for applications where variations within the auxiliary control loop (from the flow controller) need to be compensated or where the actuator has a non-linear valve characteristic. The following figure shows the P&I diagram of a temperature flow cascade.



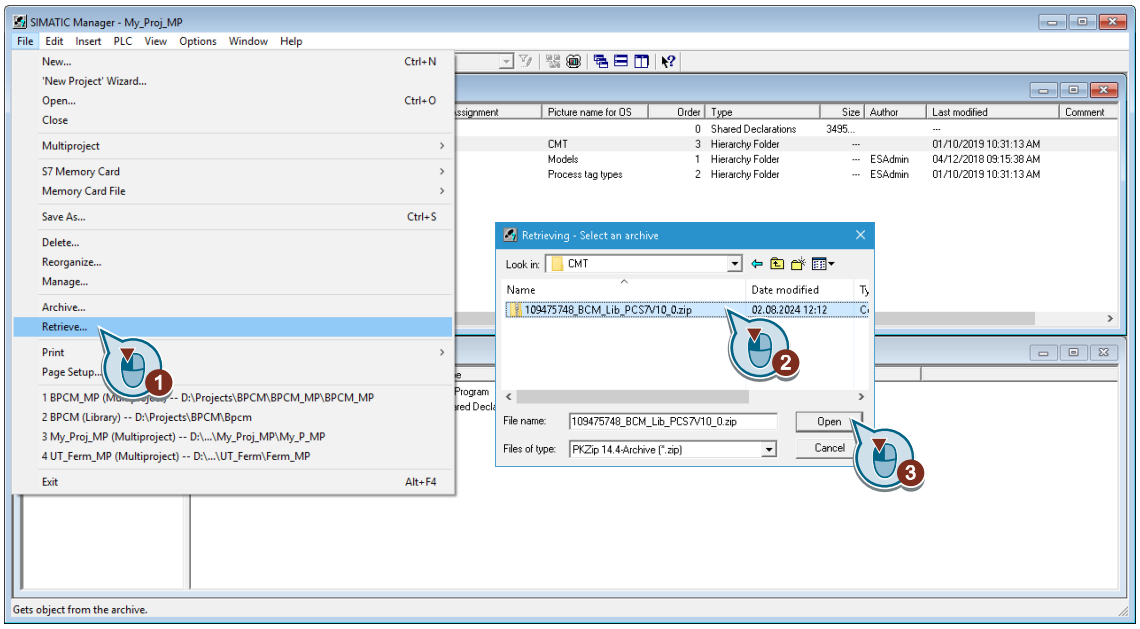
NOTE The project/multiproject was created according to the procedure in the manual "SIMATIC Process Control System PCS 7 Compendium Part A - Configuration Guide (V9.1)". You can find the configuration guide at https://support.industry.siemens.com/cs/ww/en/view/109809015

1. Change to the "Plant View" of your project in the SIMATIC Manager and create a hierarchy folder with the name "CMT" in your master data library.

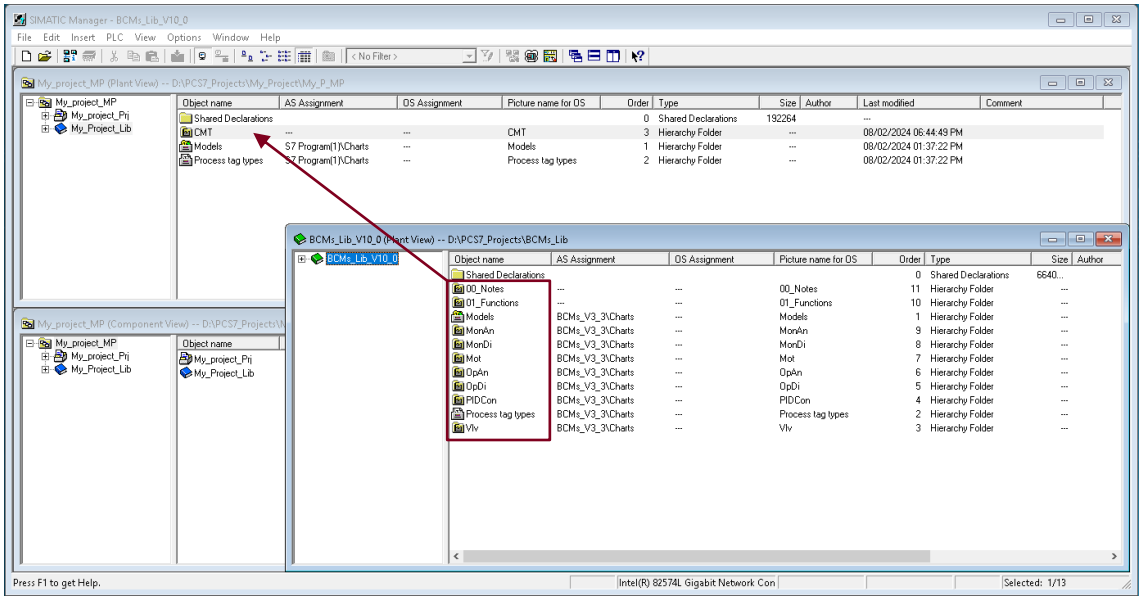


NOTE The folder name is not binding. Even the Process Tag Type folder can be used.

2. Unarchive the library "109475748_BCM_Lib_PCS7V91.zip" and switch to the "Plant View".

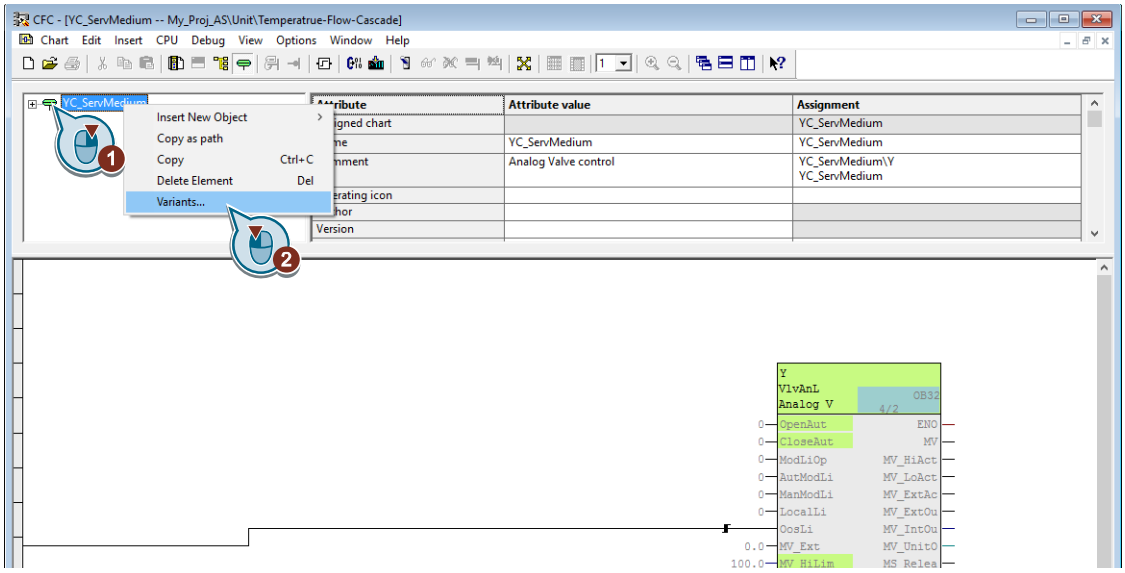


3. Drag the contained CMT folders into the master data library of your project.

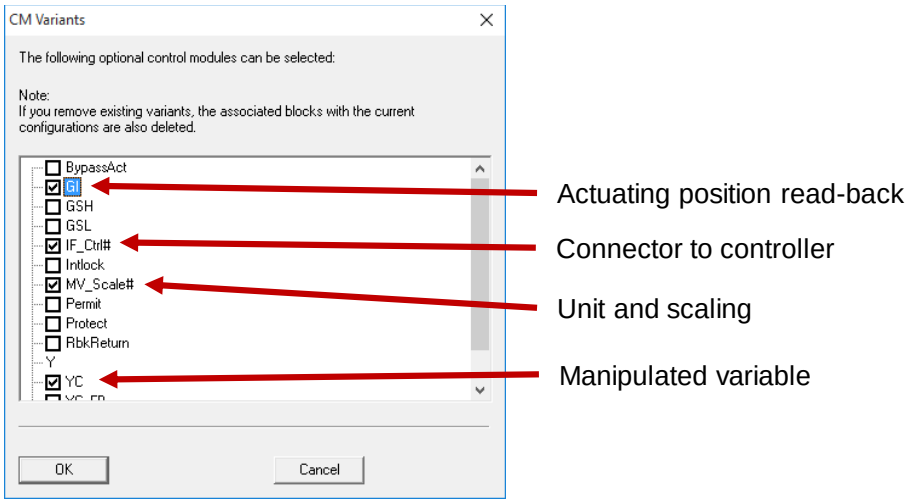


NOTE All the necessary blocks are adopted when transferring the CMT.

4. Change the folder names in the AS project for the hierarchy folder subsystem "Unit", and subordinate the technical function "Temperature-Flow-Cascade".
5. Copy two controller CM "BCM_PIDCon" and one valve CM "ValAn" from the master data library into the folder "Temperature-Flow-Cascade".
6. Change the names of the CM "BCM_PIDCon" to "TIC_Temperature", "BCM_PIDCon(1)" to "FIC_ServMedium", and "VlvAn" to YC_ServMedium".
7. Open the CFC "YC_ServMedium", show the "Technological I/Os" and select "Variants..." from the context menu.



8. In this example, the valve receives the manipulated variable from the controller, the range of adjustment and the unit are configured centrally and the valve (actuator) delivers an analog signal including readback of the manipulated position.



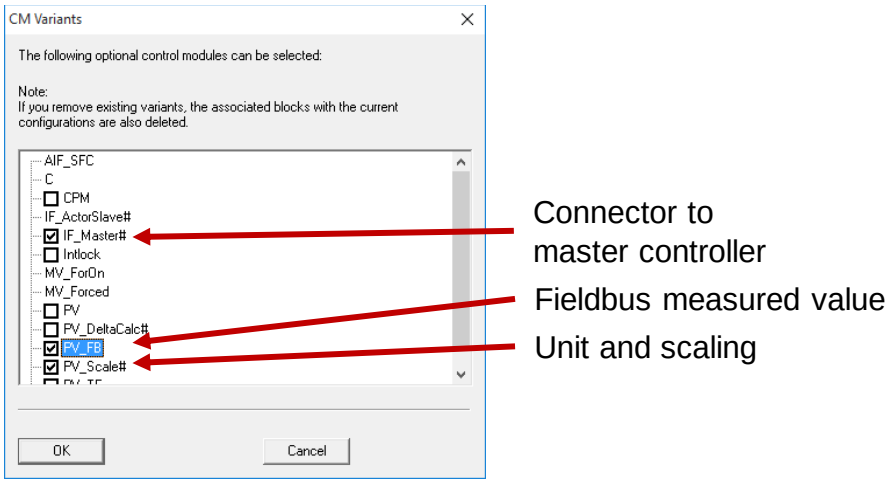
If you cannot imagine an exact function under the individual selection points, then you receive a complete overview of the interconnected blocks and configured technological I/Os in the CMT. You can find the assigned CMT in the object properties of the CFC.

9. If necessary, change the setting range at the block "MV_Scale" and the unit at the block "MV_Unit". The default is 0% to 100%.

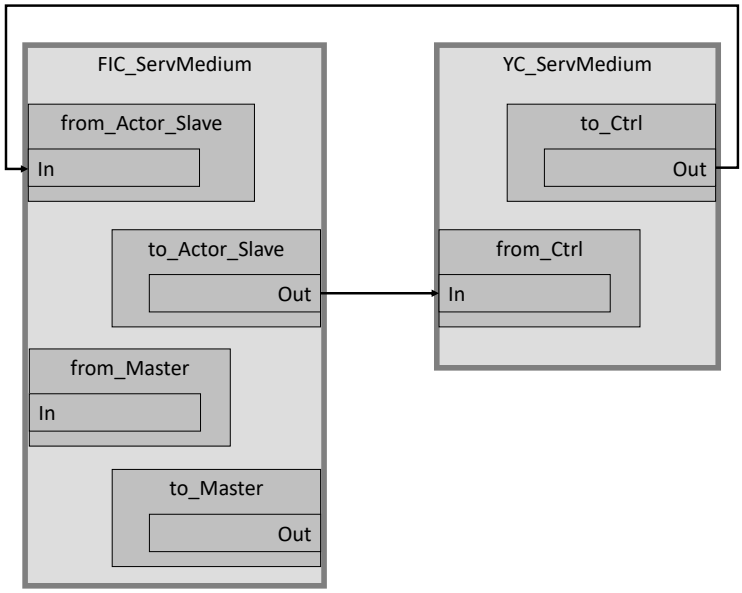
NOTE If the communication interfaces of the actuator are not known, the variant can be subsequently changed in the CFC.

10. Open the CFC "FIC_ServMedium", show the "Technological I/Os" and select "Variants..." from the context menu.

11. In this example, the controller receives its setpoint from a master controller and defines the manipulated variable for the valve. The process size and unit are set centrally and the flow rate is measured by a field device with digital communication.

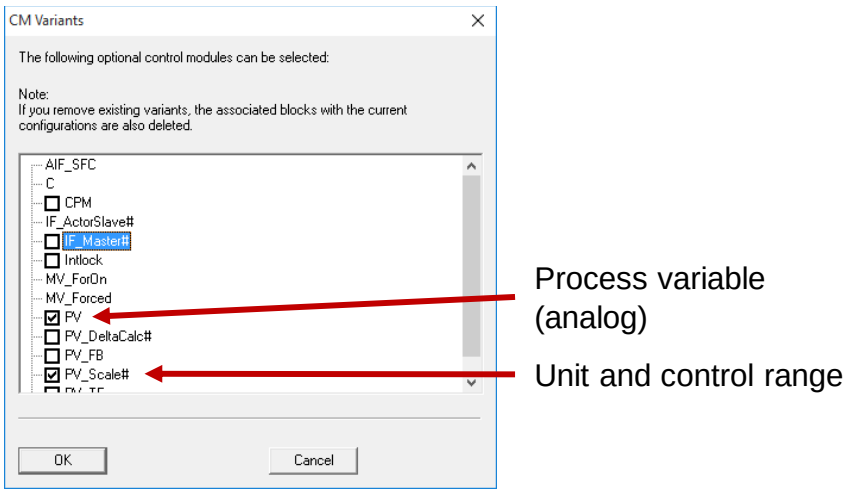


12. For the process variable, set the range 0 to 10 on the "PV_Scale" block and the unit 1328 (t/h) on the "PV_Unit" block.
13. Connect the block output "to_Actor_Slave" to the block input "from_Ctrl" of the valve CM and the block output "to_Ctrl" to the block input "from_Actor_Slave".

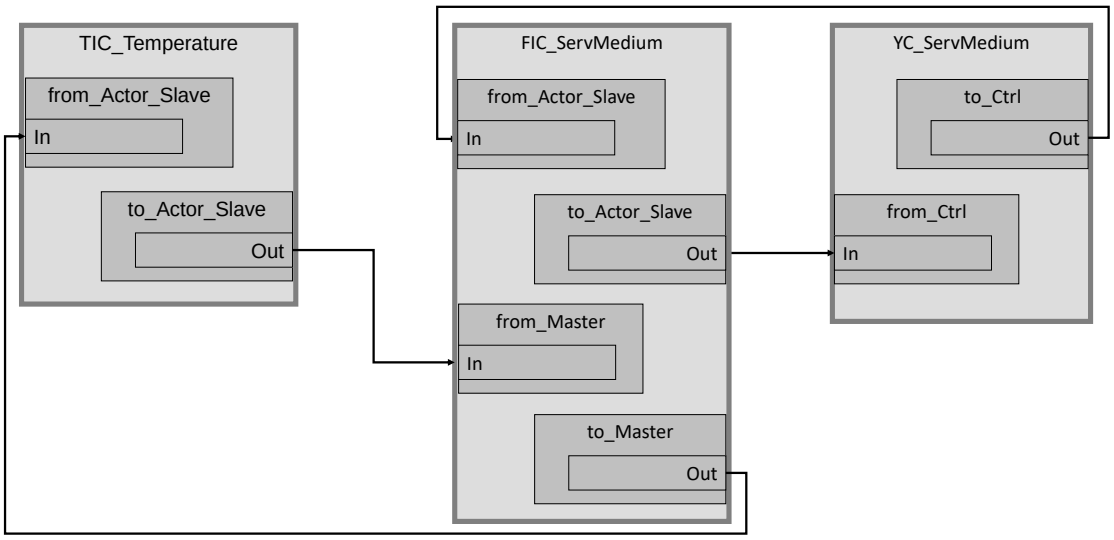


14. Open the CFC "TIC_Temperature", show the "Technological I/Os" and select "Variants..." from the context menu.

15. In this example, the master controller "TIC_Temperature" gives the setpoint to the slave controller "FIC_ServMedium". The process variable and unit are set centrally and the temperature is recorded as an analog measured value.



16. For the process variable, set the range 0 to 200 on the "PV_Scale" block and the unit 1001 (°C) on the "PV_Unit" block.
17. Connect the block output "to_Actor_Slave" with the block input "from_Master" of the slave controller "FIC_ServMedium" and the block input "from_Actor_Slave" of the master controller "TIC_Temperature" with the block output "to_Master" of the slave controller "FIC_ServMedium".



NOTE In addition, you must link the individual channel drivers of the CM with the respective periphery via the symbolic name. In the second step, the two control loops must be set, starting with the flow control "FIC_ServMedium" and then the temperature control "TIC_Temperature".

NOTE The preconfigured and simulated "Temperature Flow Cascade" is part of the project "Equipment Modules for SIMATIC PCS 7 using the example of the Chemical Industry" under the link: https://support.industry.siemens.com/cs/ww/en/view/53843373

5.3. Scenario C – Efficient functional enhancement to APG via type matching

The following shows how the existing "Unit Template Distillation Column" project can be expanded to include optimized process operation by APG (Advanced Process Graphics). The project is structured, i.e. it contains the levels subsystem, technical function (Equipment Module=EM) and Control Module (CM), and is based on the Control Module Library.

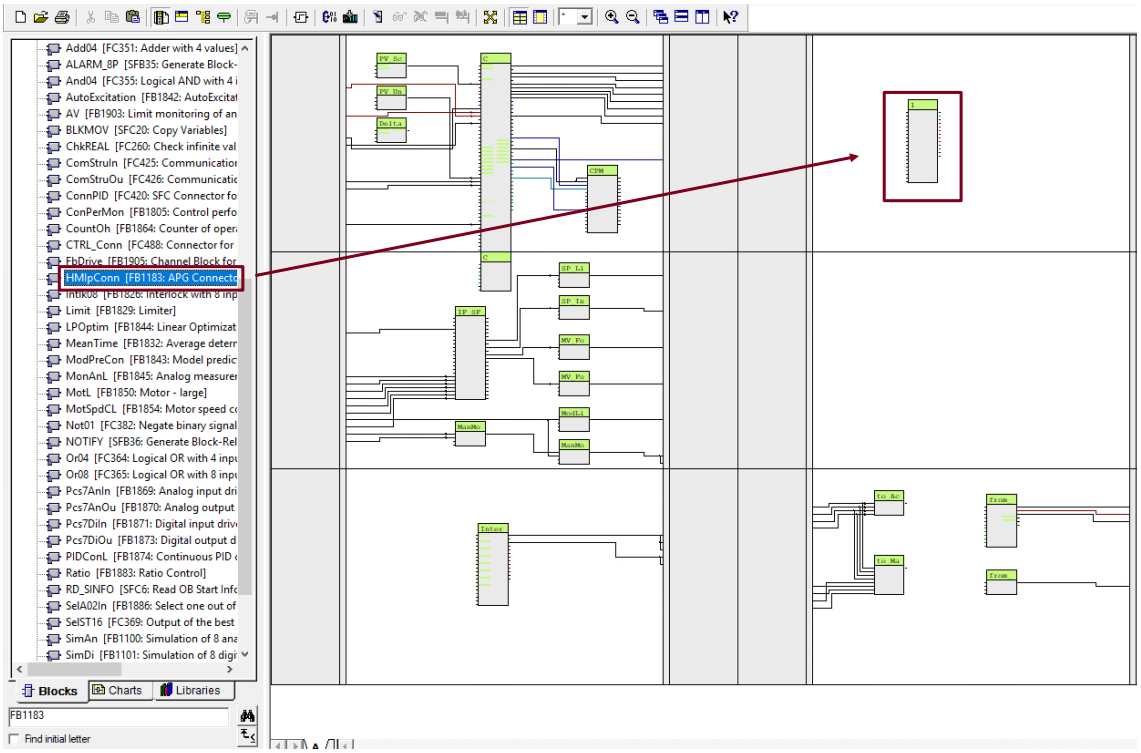
APG provides both an AS object for connection to the automation software and some OS objects for displaying the process information. In this scenario, the focus is on efficient AS project planning, which can be carried out quickly and with the necessary flexibility with the help of CM technology. The APG Connector block required for the display is added and preconfigured once in each type and is then available to each instance as an option.

Preparation

1. Dearchive the sample project "Distillation column" in the SIMATIC Manager and switch to the "Plant view" of your project.
2. Add the APG Connector block "HMIpConn" to the master data library.

NOTE The block is available with the installation of Advanced Process Graphics. A description of the installation and integration can be found in the application example "Integration of Advanced Process Graphics in SIMATIC PCS 7" at https://support.industry.siemens.com/cs/ww/en/view/89332241.

3. Add the "HMIpConn" block with the designation "HMI" to the CMT "Ctrl".

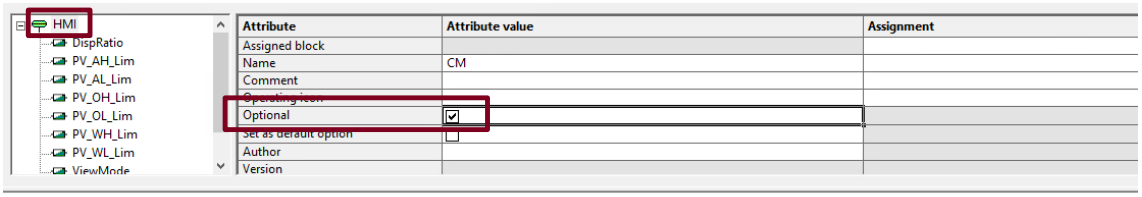


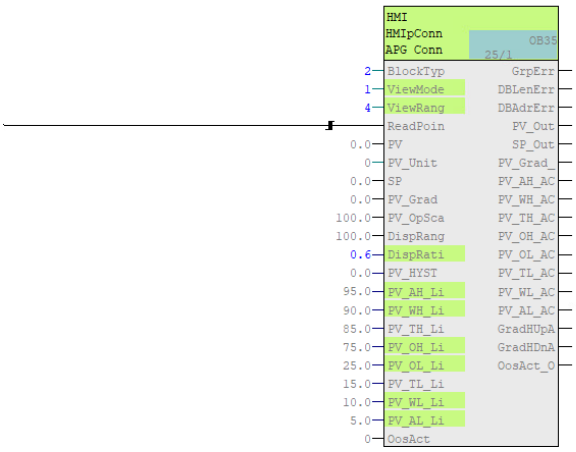
4. Connect the "ReadPointer" input of the "HMIpConn" block to an output of the "PIDConL" block.

NOTE

Use an unused output of the source block for the interconnection. For this configuration example, the output "Status2" of the controller block was made visible and connected.

5. Parameterize the "HMIpConn" block with the following parameterization:
   - "BlockType": 2 as a suitable representation of PIDConL
   - "ViewMode": 1 for absolute value range
   - "ViewRange": 4 to display the working range
   - "DispRatio": 0.6 ratio of display to ViewRange
6. Use drag&drop to add the APG block "HMIpConn" with the designation "APG" to the technological I/Os.
7. Drag the block inputs "ViewMode", "ViewRange", "DispRatio" and the working and limit value monitoring "PV_Xx_Li" onto the CM "APG" and update the screen display with the function key "F5".
8. Select the CM "APG" in the technological I/Os and activate the optional attribute. Thanks to this action, the visualization can be selected when needed.





9. Repeat steps 3 to 7 for the CMT "AMon" with "BlockType": 1 as a suitable representation of MonAnL.

NOTE

Configuration is carried out for both controller and display CMT with "MonAnL" blocks. The output "Status2" of the "MonAnL" device can be connected to the APG Connector block.
The "ENO" output of the block must not be used for the "ReadPointer" connection.

If you configure a different range for the "ViewRange" parameter, e.g. the alarm range, you must also configure the corresponding "PV_Xx_Li" limits.

Synchronization

1. Select the AS project in the plant view of the SIMATIC Manager and select "Technological Types > Synchronize..." in the context menu.
2. Select the two types "AMon" and "Ctrl" and press the "Synchronize..." button.
3. Click on the "Synchronize template" button in the comparison dialog.

NOTE After propagating the change, the "APG" option is available in the instances. The option is not selected by default.

Instance adjustment

1. Activate the option "HMI" for the following instances:
   a. Controller: "FIC_Feed", "FIC_Reflux", "FIC_Vapor", "PIC_ColuHead", "LIC_Bottom", "LIC_RefluxDrum"
   b. Display: "TI_Head", "TI_HeadPacking", "TI_AboveFeed", "TI_BelowFeed ", "TI_BottomPacking", "TI_Bottom", "PI_ColuBottom", "FI_Disitl", " FI_Bottom "
2. The final steps are:
   a. Setting the work areas "PV_OL_Li" and "PV_OH_Li" in each instance
   b. Create a process image with APG objects using the templates "@Template_APG.pdl" and "@Examples_APG.pdl"
   c. Link the process screen objects with the relevant instance using the Dynamic Wizard

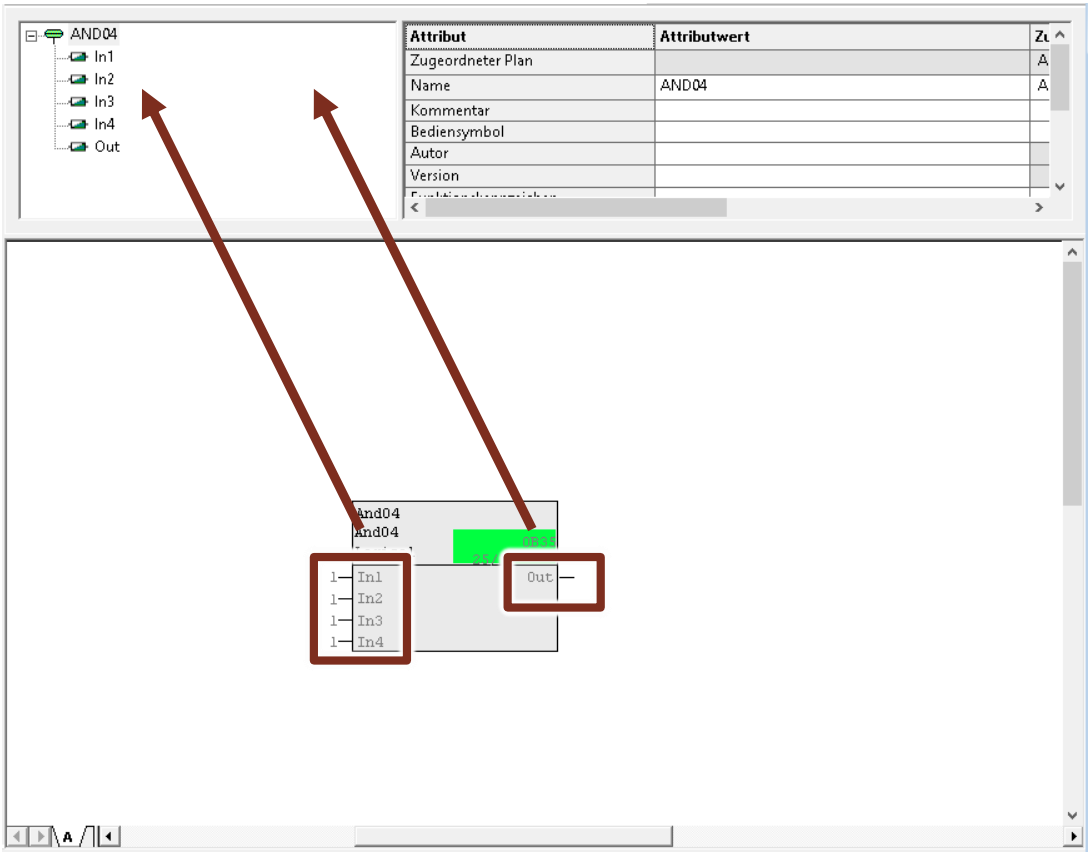
NOTE A detailed description for configuring the process image and interconnecting the APG objects (AS-OS connection) can be found in the application example "Integration of Advanced Process Graphics in SIMATIC PCS 7" under the following link:
https://support.industry.siemens.com/cs/ww/en/view/89332241

5.4. Scenario D – Creating and using functions

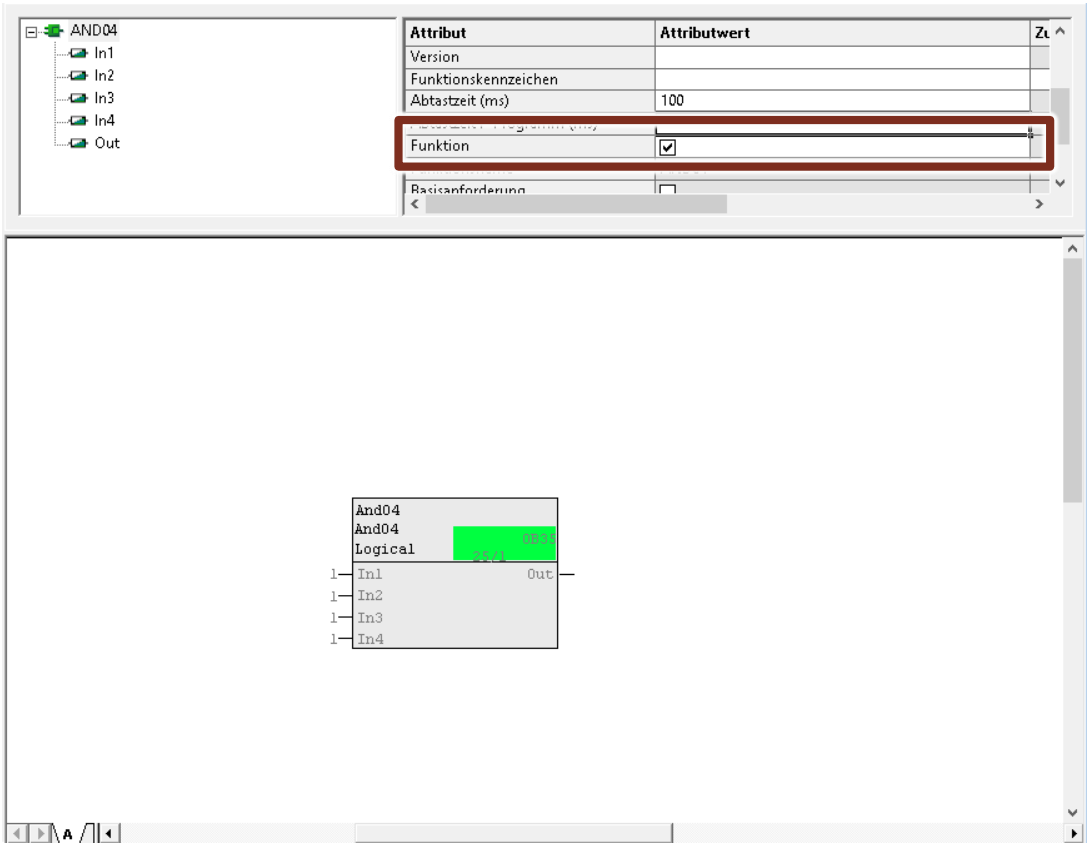
For general information about functions in CMTs, see Section 2.10.

Creating a function

1. In the plant view, right-click on the CMT storage folder and click on "Create New Object > Control Module Type"
2. Enter a sensible name for the individual control unit type and open it.
3. Use Drag&Drop to drag the required blocks into the CFC and assign them sensible names.
   If several blocks are used, connect the blocks.
4. Define the technological I/Os using drag&drop to the required parameters and signals into the "Technological I/Os" field.
   This creates the corresponding objects in the "technological I/Os" and links them to the block.



5. Activate the "Function" option in the technological I/Os.



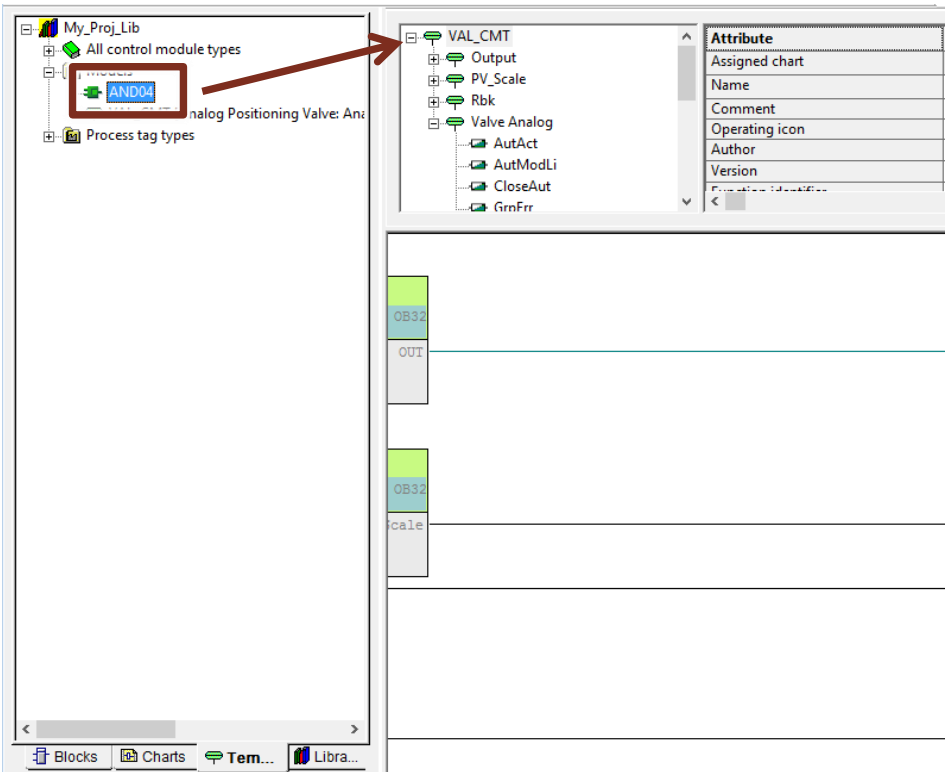
By activating the "Function" option, the symbol changes from CMT to symbol of a function .

NOTE

The "Function" field can only be selected if no SubCMs are used. With a function with SubCM, a further hierarchy level would be created in the technological I/Os with the integration in a CMT that is not allowed.

Using Functions in PCS 7

1. Open the plan of the instance where you want to insert the function.
2. Select the project library with the CMTs from the "Templates" tab and drag the created function into the "Technological I/Os" window of the opened instance.



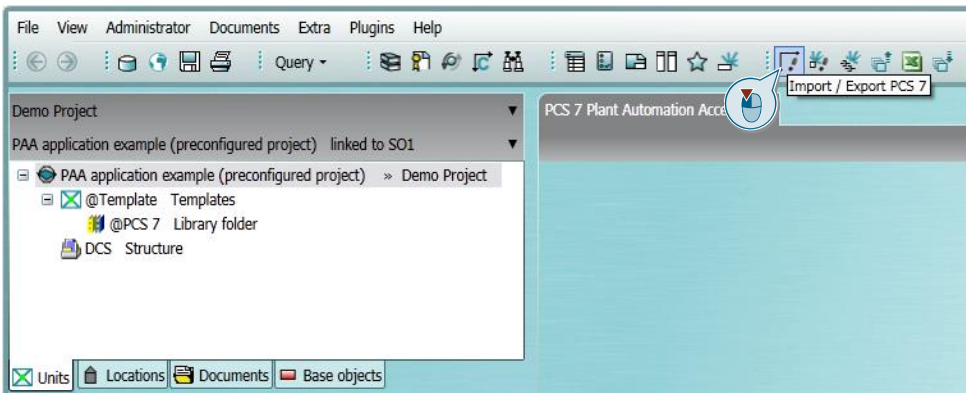
This creates the function in an empty CFC subplan.

3. Drag the function to the desired mounting position.
4. Connect and parameterize the function.

The adaptation of the instance (CM) known when importing into the mass data engineering tools (COMOS and (PAA)) and SIMIT.

Use of Functions in PAA

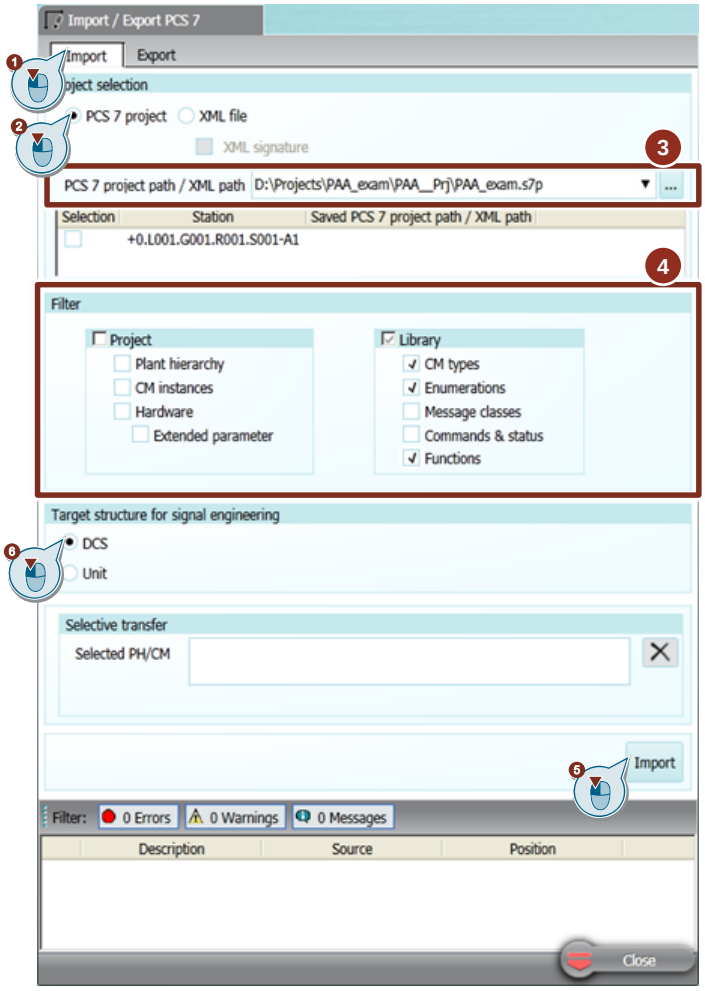
1. Click on the "Import/Export PCS 7" button in the PAA menu bar.



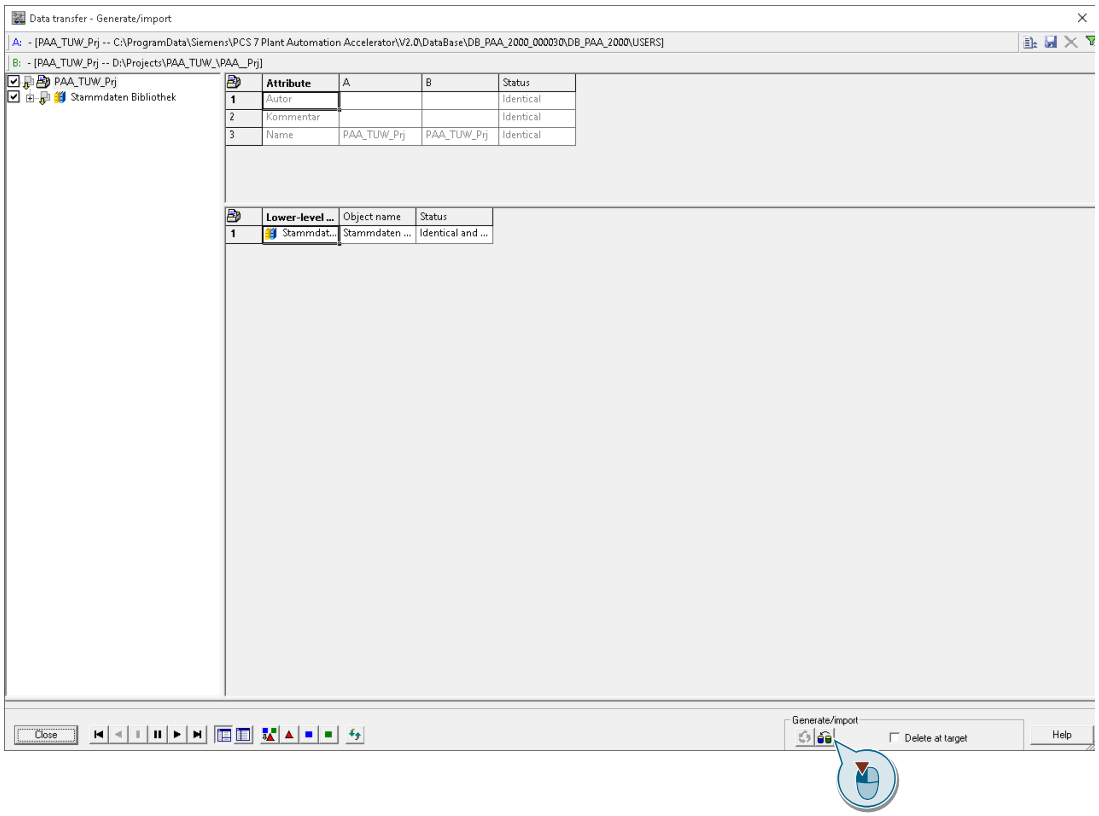
The dialog "Import / Export PCS 7" opens in the working view of PAA.

2. Open the "Import" tab.
3. Select the PCS 7 project.
4. Select your PCS 7 project.
5. Activate the check boxes "CM Types", "Enumerations" and "Functions" in the "Filter" directory and deactivate the remaining option fields.

6. Select option "DCS".
7. Click on the "Import" button.

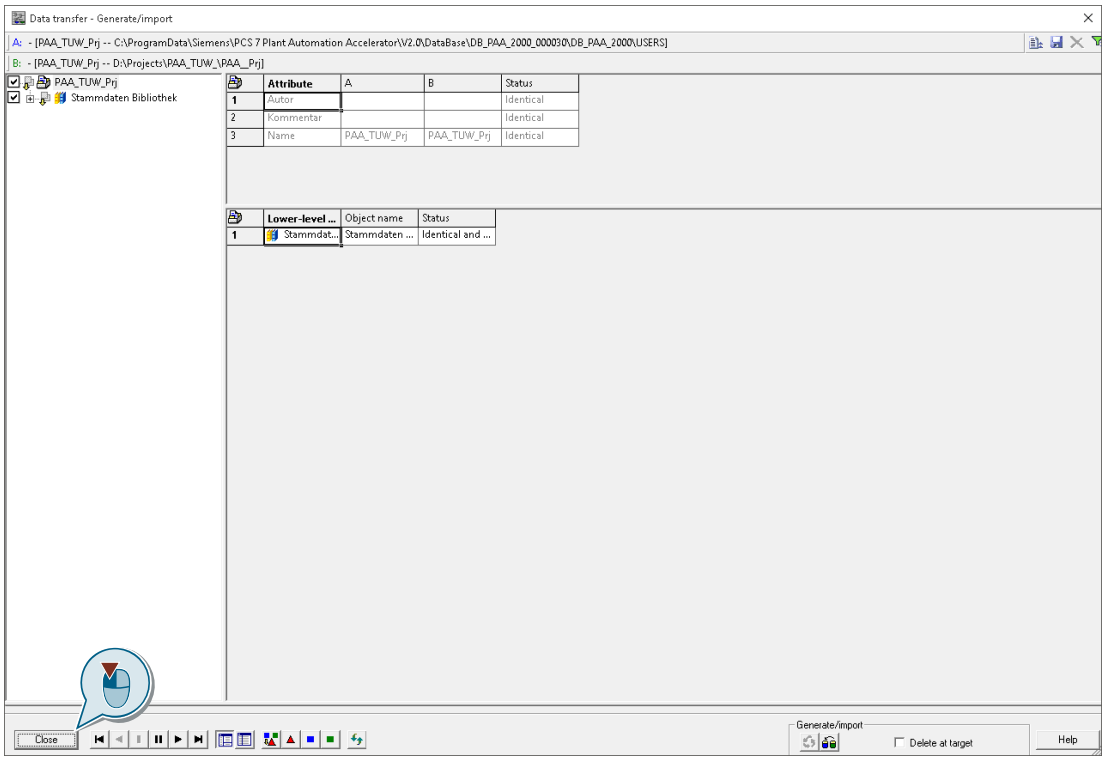


8. Click on the "Import from B to A" button to start the import from PCS 7 to PAA.



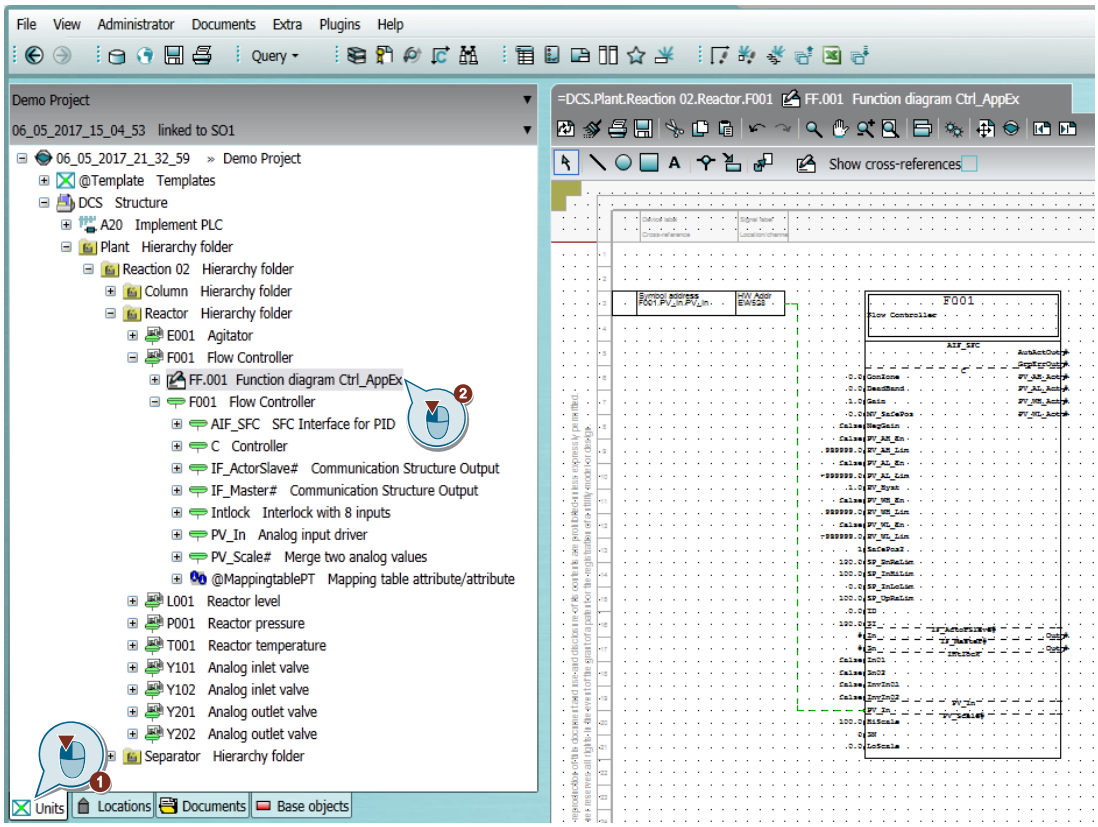
NOTE The "Import from B to A" button is only enabled if differences are found between the PAA and PCS 7 projects. Only changes are ever imported.

9. After the data transfer, click on the "Close" button to close the import dialog.

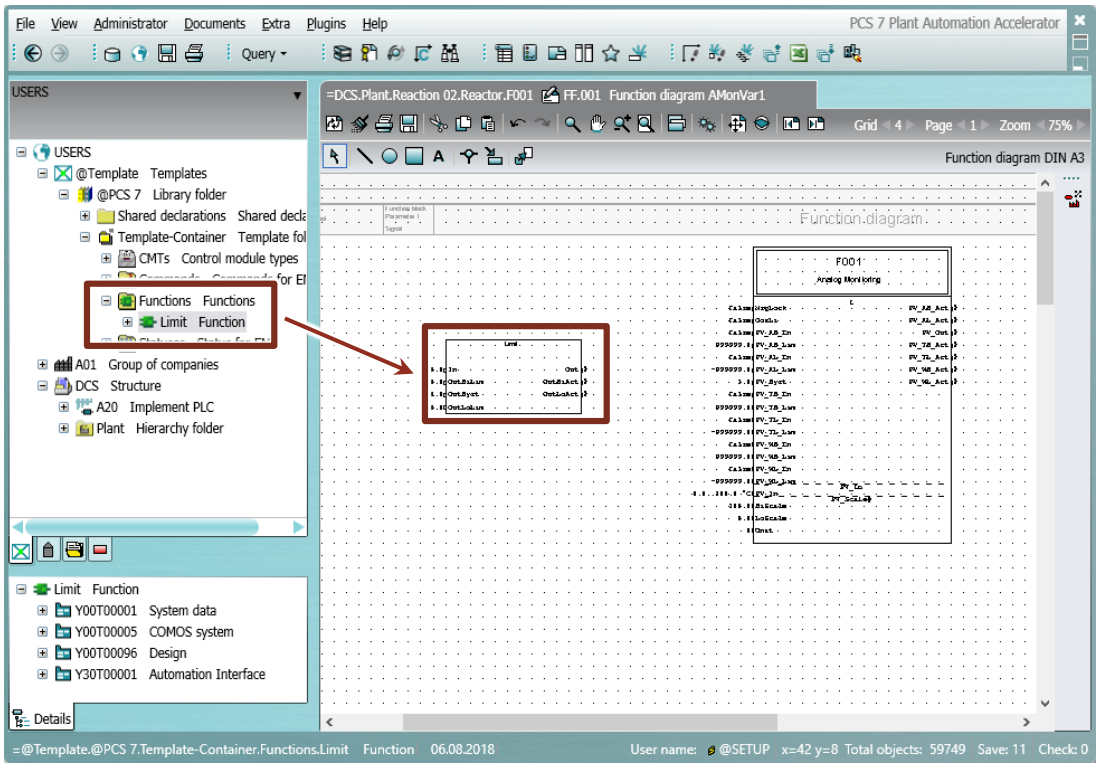


10. Open the "Units" view.

11. Navigate to the Control Module where you want to use the function and open the subordinate objects.
12. Double-click to open the respective "Function diagram".



13. Drag the function from the folder "<Project name> > Templates > Template container > Functions" into the "Function diagram".



14. Connect and parameterize the function.
    The adjustments of the CM are made directly in the Plant Automation Accelerator without changing the CMT. When exporting to PCS 7, the function is integrated into the CM.

6. Appendix

6.1. Service and support

SiePortal

The integrated platform for product selection, purchasing and support - and connection of Industry Mall and Online support. The SiePortal home page replaces the previous home pages of the Industry Mall and the Online Support Portal (SIOS) and combines them.

- Products & Services
  In Products & Services, you can find all our offerings as previously available in Mall Catalog.
- Support
  In Support, you can find all information helpful for resolving technical issues with our products.
- mySieportal
  mySiePortal collects all your personal data and processes, from your account to current orders, service requests and more. You can only see the full range of functions here after you have logged in.

You can access SiePortal via this address: sieportal.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts.
Please send queries to Technical Support via Web form: support.industry.siemens.com/cs/my/src

SITRAIN – Digital Industry Academy

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.
For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page: siemens.com/sitrain

Industry Online Support app

You will receive optimum support wherever you are with the "Industry Online Support" app. The app is available for iOS and Android:

6.2. Industry Mall

The Siemens Industry Mall is the platform on which the entire siemens Industry product portfolio is accessible. From the selection of products to the order and the delivery tracking, the Industry Mall enables the complete purchasing processing – directly and independently of time and location:
mall.industry.siemens.com

6.3. Links and literature

No.	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to this entry page of this application example https://support.industry.siemens.com/cs/ww/en/view/109475748
\3\	Synchronizing of individual control module types (PCS 7 V9.0 SP1) https://support.industry.siemens.com/cs/ww/en/view/109758382
\4\	SIMATIC PCS 7 Overview (link collection to FAQ, manuals, compendium, forum, application examples and videos) https://support.industry.siemens.com/cs/ww/en/view/63481413
\5\	SIMATIC PCS 7 Plant Automation Accelerator using a practical example https://support.industry.siemens.com/cs/ww/en/view/109742154
\6\	Integrated Engineering with COMOS and SIMATIC PCS 7 using a practical example https://support.industry.siemens.com/cs/ww/en/view/70922226
\7\	Engineering efficiency in the interaction of SIMATIC PCS 7 Plant Automation Accelerator, SIMATIC PCS 7 and SIMIT Simulation https://support.industry.siemens.com/cs/ww/en/view/109770538
\8\	SIMATIC PCS 7 Information Center with numerous videos and technical information on PCS 7 including APL, APG. etc. https://support.industry.siemens.com/cs/ww/en/view/109760496
\9\	Numerous videos about the Advanced Process Library on YouTube https://www.youtube.com/results?search_query=SIMATIC+PCS+7+APL

6.4. Change documentation

Version	Date	Change
V1.0	09/2015	First version
V2.0.	03/2019	Update to V9.0 SP1, Additional sections:2.5, 2.6, 2.10, and 5.4.
V2.1	04/2020	Correction
V3.0	11/2021	Update to V9.0 SP3. Addition of practical tips in Sections 2, 3, and 4.
V4.0	07/2022	Update to BCM_Lib_PCS7V91; update of the CMT table for ValAn in section 1.2.1.
V5.0	08/2024	Update to V10.0; update to BCM_Lib_PCS7V10_0.

