

**SIEMENS**



# SCE Training Curriculum

Siemens Automation Cooperates with Education (SCE) | 09/2015

**PA Module P02-02**  
**SIMATIC PCS 7 – Alarm Engineering**

Cooperates  
with Education

Automation

**SIEMENS**

## Matching SCE Trainer Packages for these curriculum

- **SIMATIC PCS 7 Software block of 3 packages**  
Order No. 6ES7650-0XX18-0YS5
- **SIMATIC PCS 7 Software block of 6 packages**  
Order No. 6ES7650-0XX18-2YS5
- **SIMATIC PCS 7 Software Upgrade block of 3 packages**  
Order No. 6ES7650-0XX18-0YE5 (V8.0 → V8.1) or 6ES7650-0XX08-0YE5 (V7.1 → V8.0)
- **SIMATIC PCS 7 Hardware Set including RTX Box**  
Order No. 6ES7654-0UE13-0XS0

Please note that these trainer packages may be replaced with subsequent packages.  
An overview of the available SCE packages is provided at: [siemens.com/sce/tp](https://www.siemens.com/sce/tp)

## Continuing education

For regional Siemens SCE continuing education, contact your regional SCE contact partner.  
[siemens.com/sce/contact](https://www.siemens.com/sce/contact)

## Additional information relating to SIMATIC PCS 7 and SIMIT

In particular, Getting Started, videos, tutorials, manuals and programming guide.  
[siemens.com/sce/pcs7](https://www.siemens.com/sce/pcs7)

## Additional information relating to SCE

[siemens.com/sce](https://www.siemens.com/sce)

## Note on Usage

The training curriculum for the integrated automation solution Totally Integrated Automation (TIA) was prepared for the program "Siemens Automation Cooperates with Education (SCE)" specifically for training purposes at public educational and R&D facilities. Siemens AG is not liable for the contents.

This document may only be used for initial training on Siemens products/systems. This means it may be copied entirely or partially and handed to trainees for use within the scope of their training. Passing on or copying this document and communicating its contents is permitted within public training and continuing education facilities for training purposes.

Exceptions require written permission by Siemens AG. Contact person: Roland Scheuerer  
[roland.scheuerer@siemens.com](mailto:roland.scheuerer@siemens.com).

Violators are subject to damages. All rights including translation rights are reserved, particularly in the event a patent is granted or a utility model or design is registered.

Usage for industrial customer courses is explicitly not permitted. We do not agree to the commercial utilization of these documents.

We would like to thank the Technical University Dresden, particularly Prof. Dr. Leon Urbas and Annett Krause, MS, as well as the Michael Dziallas Engineering Corporation and those who provided support in preparing this SCE training document.

# ALARM ENGINEERING

## TRAINING OBJECTIVE

In this module, the students are familiarized with the fundamentals of an alarm system. They understand the purpose and the areas of application of alarm and signaling systems, and they know the requirements for such systems resulting from this. They are familiarized with the possibilities of representation, and interactions with messages and alarms. This enables the students to design a suitable and useable alarm management in **PCS 7**.

## THEORY IN BRIEF

Alarm systems play an extremely important role in modern process control concepts for the economic operation of process plants. Designed ergonomically, they inform the operating personnel specifically if there are unintended deviations of the process state from a defined normal state (refer also to the chapter Functional Safety). Alarm systems make it possible for the operator to localize the cause of the malfunction directly, and to adjust the process control strategy through suitable intervention in a way that, despite the fault, either specification-conforming products continue to be manufactured, or the process is stabilized in a way that the fault causes minimal production outage.

The control system **PCS 7** includes a number of technical resources for implementing an alarm system. The palette ranges from function blocks for generating messages, icons for representing alarm states, group alarms along the plant hierarchy to components for representing and managing messages in lists (refer to Figure 1).

By following a number of design rules for specifying message texts and assigning priorities, a very efficient alarm system can be implemented that meets all the requirements of the national and international standards and guidelines currently valid.

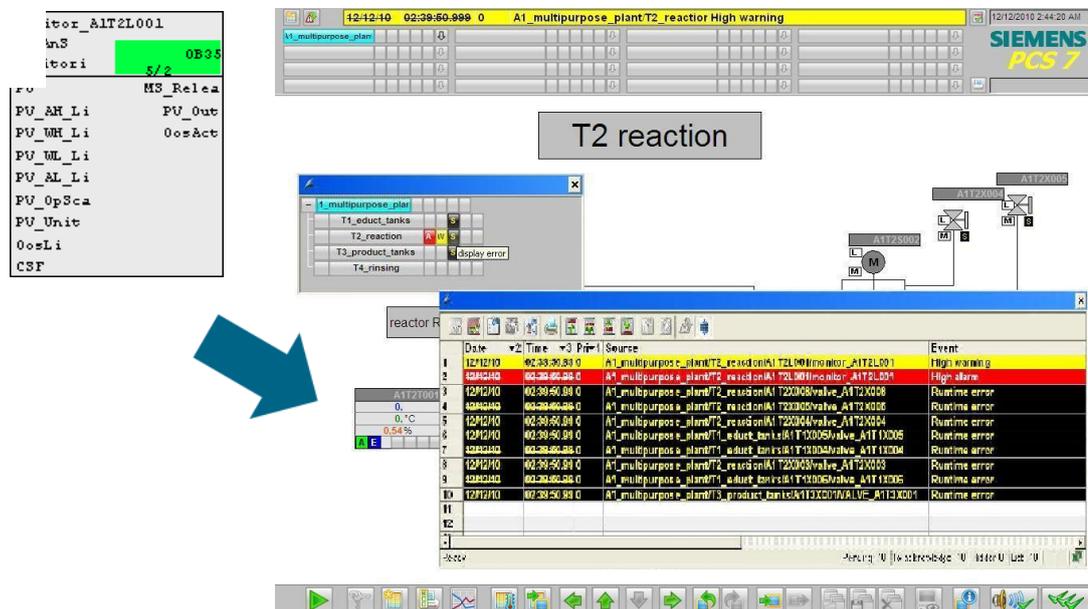


Figure 1: From alarm block to visualization at alarm lists

## THEORY

### ALARM SYSTEMS

Through the consistent use of modern process control engineering, process plants are highly automated and optimized regarding safety. For this reason, the operator of such a plant monitors a largely automated process that requires operations only if, because of a fault of the process or of the plant, manual intervention is necessary. The objective of such manual intervention is always to take the process back to the normal state (refer also to the chapter 'Functional Safety') before the automatic protection equipment is activated.

Since in general, protection equipment takes the monitored technical facility to a safe state, this usually causes a loss in product quality, production delays or even the standstill of the entire production. This has a considerable negative effect on the economic efficiency of the plant. For that reason, the possibility that an impermissible fault activates the protection equipment has to be detected early so that it can be prevented through suitable manual intervention. In addition, the operator has to be informed of protection equipment being activated so that he can monitor the consequences.

The alarm system is the central interface between the operator and the monitored process, and makes available all setups for managing messages and alarms in the control system [2]. The alarm system makes it possible for the operator to detect deviations from specified conditions within the range of the intended operation early, and to specifically counteract it. Figure 2 shows the four phases of interaction between the operator and the alarm system of the process control system.

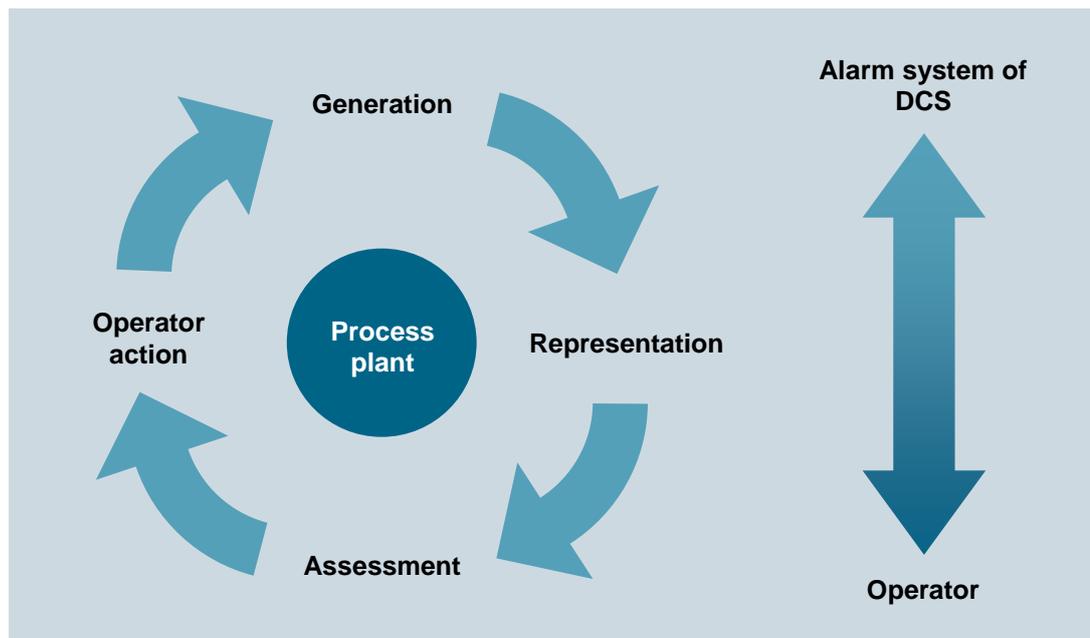


Figure 2: Phases of interaction according to [2]

This means the alarm system has to provide the operator with the capability and the opportunity to respond suitably to a signaled event. To attain this, the system has to meet a number of requirements. Messages and alarms have to be represented **clearly**, **transparently** and **consistently**.

The operator has to be supported in the situation-oriented evaluation of a message or an alarm as well as in the selection of suitable intervention. To this end, a suitable action prompt has to be provided, depending on the process state.

To prevent the operator from being overburdened, the number as well as the frequency of messages and alarms has to be minimized. In addition, the operator's workload should be kept as low as possible when messages and alarms occur. Beyond this, the operator can be supported in his work with suitable tools for documenting and evaluating messages and alarms.

When designing an alarm system, the performance capability limits of the future operators have to be taken into account. The totality of the tasks that an alarm system requires of an operator to be performed must not exceed human performance limits either for short periods of time nor continuously.

On the one hand, a sudden rise in the number of alarms or of the alarm rate can cause the operator to be briefly overloaded (alarm shower). It has to be taken into account here that on the average, an operator cannot process, as a maximum, more than 7 items of information at the same time (**7±2 rule**).

On the other hand, a continuously high workload can cause the operator to be permanently overloaded through a constant high number of arriving alarms. This leads to an increasing drop of the operator's performance and reliability.

An alarm system has to be designed that utilizes the characteristic features of human perception and takes their limits into account. Important alarms have to be emphasized to be noticed quickly. Events that occur rarely have to be presented in a special manner to draw the attention of the user. Important information should be presented redundantly to facilitate it being noticed. Moreover, several sense channels should be addressed for transmitting information (for example, using acoustical warning signals).

Only if an alarm system meets these requirements can it actually support the operator in his tasks: to monitor and control the plant.

## **ALARMS AND MESSAGES**

Alarm systems are used to manage messages and alarms in control systems. In general, 'Message' means every report and every display referring to the occurrence of a specific event. However, in the narrower sense the term is used only for such messages that do not require immediate operator reaction [1]. Otherwise, the term alarm is used; i.e., the term 'message' is used as a generic term as well as subsidiary term. Below, the following definitions will be used consistently:

- **Alarm:** Refers to a display or report that an event occurred that requires immediate operator reaction. The reaction may be an activity; for example, performing an operation. But it can also be only a mental reaction; for example, increased attention.
- **Message:** Refers to a display or report that an event occurred; no immediate operator action is required.

Alarms signal deviations of the process or of the plant from the desired state and make it possible for the operator to avert a danger situation or economic damage. To fulfill this task, good alarms have to have the following features [3]:

- **Relevant:** The alarm is justified and valuable to the operator.
- **Clear:** The alarm contains information for the operator. It does not repeat another alarm.
- **Timely:** The alarm arrives close to the time when an intervention is necessary. It arrives in time for the operator to intervene.
- **Prioritized:** The alarm refers to the urgency of operator reaction.
- **Can be understood:** The alarm contains information that can be understood easily.
- **Diagnostic:** The alarm makes it possible for the operator to identify the problem that occurred.

- **Instructional:** The alarm provides suitable action instructions to solve the problem that occurred
- **Focusing:** The alarm guides the attention to the most important problems

Alarms should always be used purpose-oriented. We should clarify **what** is monitored, **how** it is done and **when** an alarm is triggered. Furthermore, we have to define how the operator can react to the alarm. Based on these criteria, alarms can be subdivided into a variety of alarm types (refer to [3]). The most important types are:

- **Absolute alarm:** The alarm is generated when a specified limit is exceeded or dropped below.
- **Time delayed alarm:** The alarm is generated when the alarm criterion for a specified time span is met.
- **Process alarm:** The control system itself generates an alarm that requires immediate operator reaction.

## ALARM PROCESSING BY THE OPERATOR

The operator processes alarms in three phases: First, the operator has to recognize that a problem occurred (1st phase: **recognition**). To this end, the alarm system has to draw the operator's attention to the problem. Then, with the aid of the control system, the operator has to identify the problem (2nd phase: **identification**). After the operator located the cause, he can initiate steps to remove the fault and to compensate for the consequences of the problem (3rd phase: **problem recovery**).

During each of these phases, the alarm system has to support the operator in a suitable manner. Table 1 lists the most important support capabilities of the alarm system.

Table 1: Options of the alarm system to support alarm processing

Phase	Supporting options of the alarm system
Recognition	<ul style="list-style-type: none"> <li>– Effective guidance of attention</li> <li>– Suitable presentation of information</li> <li>– Pre-processing and sampling information</li> </ul>
Identification	<ul style="list-style-type: none"> <li>– Significant description of errors</li> <li>– Tools for investigation of errors</li> <li>– Jumping to the appropriate operator display of PCS</li> </ul>
Troubleshooting	<ul style="list-style-type: none"> <li>– Significant instructions for solving the problem</li> <li>– Jumping to the appropriate operator display of PCS for operator intervention</li> </ul>

To make appropriate alarm processing possible for the operator of a plant, alarms that occur have to be suitably managed by the alarm system. Alarm management supports all phases of interaction between the operator and the alarm system of the process control system.

### Generation

Messages and alarms are generated process-oriented in the devices of the control system equipment of the plant. Generation may be linked to certain conditions (for example, timing conditions, and hysteresis) and is carried out always with time-synchronous stamping.

When defining messages and alarms, the operator's reaction time has to be taken into account. After a message or an alarm occurred, the operator has to have sufficient time to remove the indicated problem before the next alarm is triggered.

This can be easily illustrated using the overflow protection of a reactor as an example. Corresponding to the inflow rate of the reactor, a defined time elapses between the overflow indication and the corresponding overflow alarm. If the operator's counter measures don't take effect in time, the indication is not useful to the operator since the alarm and the automated protection function are triggered in any case.

### Prioritization

Large process plants have a considerable number of alarm sources available that in turn can trigger different types of alarms. To keep this diversity controllable for the operator, it is advisable to structure the alarm system. A suitable method for this is alarm prioritization. This refers to the unambiguous classification of all alarms of an alarm system according to their importance and urgency [2]. If several alarms are accumulating, a processing sequence can be suggested to the operator based on alarm priorities.

Response time	Seriousness		
	Shutdown	Off spec	delay in production
< 5 min	High	Medium	Low
5 - 20 min	Medium	Medium	Low
> 20 min	Low	Low	Low

Figure 1: Example of a prioritization matrix according to [2]

To this end -as shown in Figure 1- a **prioritization matrix** can be set up. It depends on the requirements of the respective process plant and is usually used for the entire plant. Corresponding to this matrix, each individual alarm is assigned a priority (**static prioritization**).

Alternatively, alarms can also be prioritized depending on the current plant situation and the combination of other queued alarms (**dynamic prioritization**). Usually, alarms are marked by color regarding their priority.

Prioritization is designed in a way that the long term average alarm rate for each operator station in normal operation is not more than one alarm every ten minutes [2]. For that reason, useful priority distribution is the aim. For example:

- 5 % Priority **High**
- 15 % Priority **Medium**
- 80 % Priority **Low**

The reduction of operator load resulting from this avoids overload consequences and ensures necessary time spans for monitoring and controlling [2].

### *Representation*

The representation of alarms is of essential importance to the usability of an alarm system. The representations below have proven themselves in practice and have prevailed [2]:

- **Area overview of alarms:** Arrangement of the alarms in an overall view that cannot be covered up (also called **common status display**). The alarms are arranged in a way that they can be assigned directly to the corresponding plant units. Using corresponding jump functions, the assigned process and plant displays are directly accessible.
- **Representing alarms using an alarm list:** Arranging the queued alarms in list form. The list can be sorted and filtered in many ways. Often, jump functions are provided to the assigned process and plant displays in this representation also.
- **Alarm representation in schematic flow diagrams:** Alarms are signaled through saturated colors (preferably red or yellow) of the corresponding symbols in the process and plant display.
- **Initial message system:** If alarms accumulate, the system determines the primary alarm and filters out the subsequent alarms resulting from it. For the operator, this decreases the number of alarms he has to process.

Often, the graphic representations are supplemented by optical or acoustical signal transmitters. They inform the operator in addition of an occurring alarm.

The operator has to acknowledge alarms and messages that occurred; i.e., he documents that he took notice of the state change,

### *Evaluation*

In order to evaluate an alarm or a message, the operator has to be able to interpret the current process and plant state correctly. He is assisted in doing this by the types of representation just discussed, by suitable message texts and alarm descriptions as well as suitable tools for pre-processing larger alarm volumes.

### *Operator Intervention*

After the operator has evaluated the state of the plant and the consequences, he has to decide what to do appropriate to the situation, and implement it. This takes place within the control system but outside the alarm system. For that reason it is extremely helpful to the operator if he can jump from the alarm system directly to the corresponding screen where the necessary action can be taken. Often, alarm systems provide corresponding jump functions. In addition, decisions regarding operator action are supported with help texts that are assigned to the different alarms.

## ALARM MANAGEMENT IN PCS 7

**PCS 7** has an efficient alarm system. It informs the plant operator about occurring events and indicates them in the process mode in the form of message lists and a group display. An additional list shows the operator interventions. The display for messages is configured in **WinCC**.

**PCS 7** differentiates three message classes [4]:

- **Control system messages:** are generated in **PCS 7** by driver blocks if they detect faults on their own components (AS, OS etc.) These messages don't have to be configured.
- **Process messages:** signal events of the automated process such as limit violations and operational messages. These messages don't have to be configured. However, messages texts and message priority can be changed if needed.
- **Operation messages:** are generated when process variables are operated; for example, when the operating mode is changed. Operation messages are generated automatically if **PCS 7** faceplates (picture blocks) from the **Advanced Process Library** are used, or own blocks that are configured in conformance with **PCS 7**.

Messages for the AS and distributed periphery are configured when the CFCs are generated, or in the process object view. Messages of block types or individual block instances can be changed and own message texts can be configured. SFCs. Types and instances can also generate messages.

Messages for the OS are configured using the application **Alarm Logging** in **WinCC Explorer**. There, the triggering event is specified for a message.

When configuring messages, different aspects have to be taken into account:

- **Message text:** Blocks with signaling behavior have preset message texts with the corresponding message class and message type. These texts and attributes can be adapted depending on requirements. In addition, information from the process or the block comment can be added.
- **Message number:** During compilation, each message configured in the ES is automatically assigned a unique message number in alarm logging. The message number range is specified when the project is set up. Message numbers are assigned either project wide or CPU wide and are unique. The latter is the precondition for assigning message priorities.
- **Message priority:** A message can be assigned a priority between 0 (the lowest) and 16 (the highest). Message lists can be sorted and filtered according to their priority. The message line in the overview area always displays the message that has the highest priority and was not acknowledged.

Technological blocks that are visualized on the OS have the function **Loop-In-Alarm**. It makes it possible to change directly from the message list to the corresponding faceplate for process and control system messages.

**PCS 7** uses a central acknowledgement concept. If a message is acknowledged on the OS, the acknowledgement is sent first to the triggering block and from there forwarded to all relevant OSs.

## LITERATURE

- [1] VDI/VDE 3699 (Ed. 2014-01): Process control using display screens
- [2] NAMUR NA 102 (Ed. 2008-10): Alarm Management.
- [3] EEMUA 191 (Ed. 2013, Edition 3): Alarm Systems.
- [4] SIEMENS (2015): Process Control System PCS7: Engineering System (V8.1). A5E32712311-AC. (<http://support.automation.siemens.com/WW/view/en/90663380>)

## STEP BY STEP INSTRUCTIONS

### TASK

In this task, alarms and warnings are set up for the operator system (OS). As an example, we program level monitoring for reactor A1T2R001, and have the alarms and warnings we set up there displayed in **WinCC**.

### TRAINING OBJECTIVE

In this chapter, the students

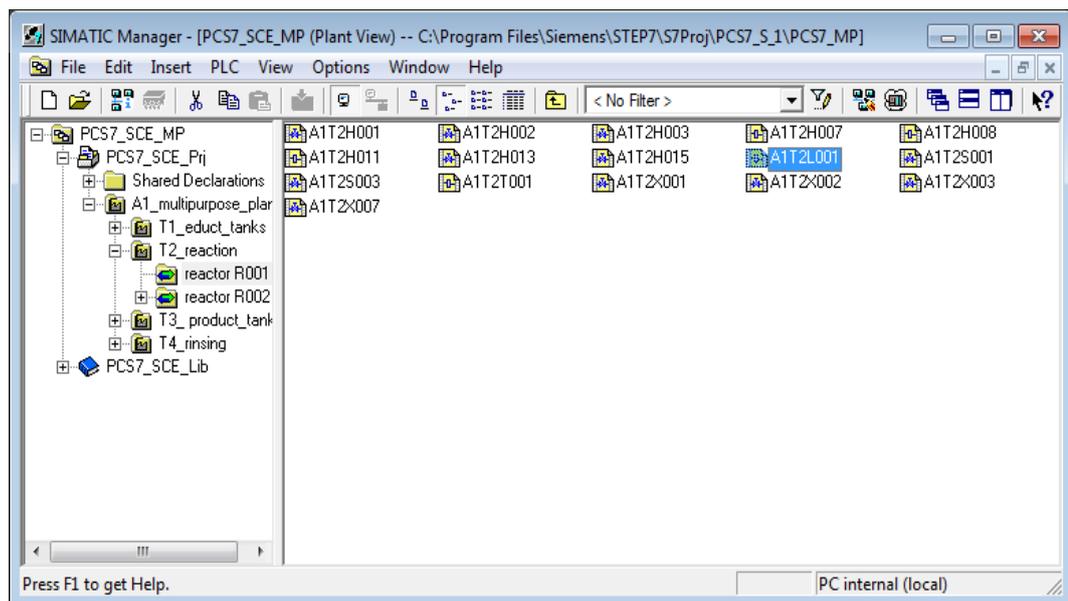
- Learn to integrate monitoring and alarm blocks in the CFC
- Become familiarized with the **WinCC** message system
- Learn representing alarms and warnings in the operator system (OS)
- Are familiarized with the **WinCC Graphics Designer**.

These instructions are based on, PCS7\_SCE\_0201\_Ueb\_R1305\_en.zip'.

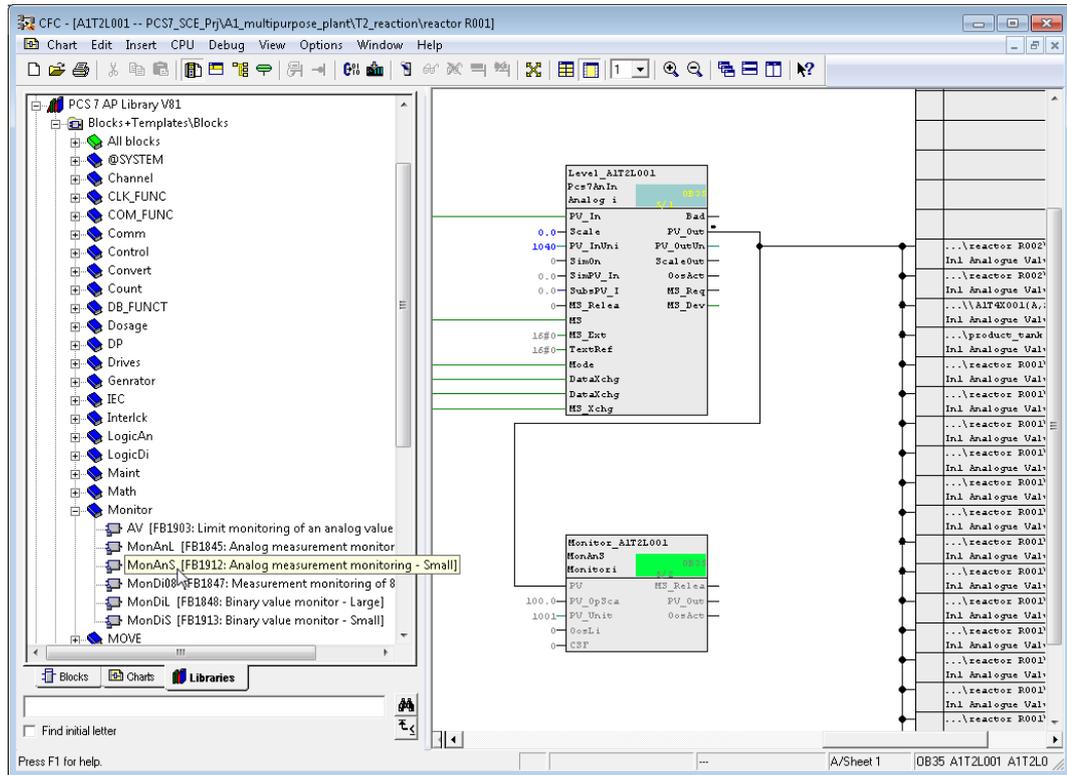
### PROGRAMMING

1. To program level monitoring, we first open the existing CFC A1T2L001 for the level of reactor A1T2R001.

(→ A1\_multipurpose\_plant → T2\_Reaction → Reactor R001 → A1T2L001)



- From the folder 'Monitor' of the **PCS 7 Advanced Process Library V81** in the library catalog, enter the block MonAnS and interconnect it as shown. (→ Libraries → PCS 7 AP Library V81 → Blocks+Templates/Blocks → Monitor → MonAnS)

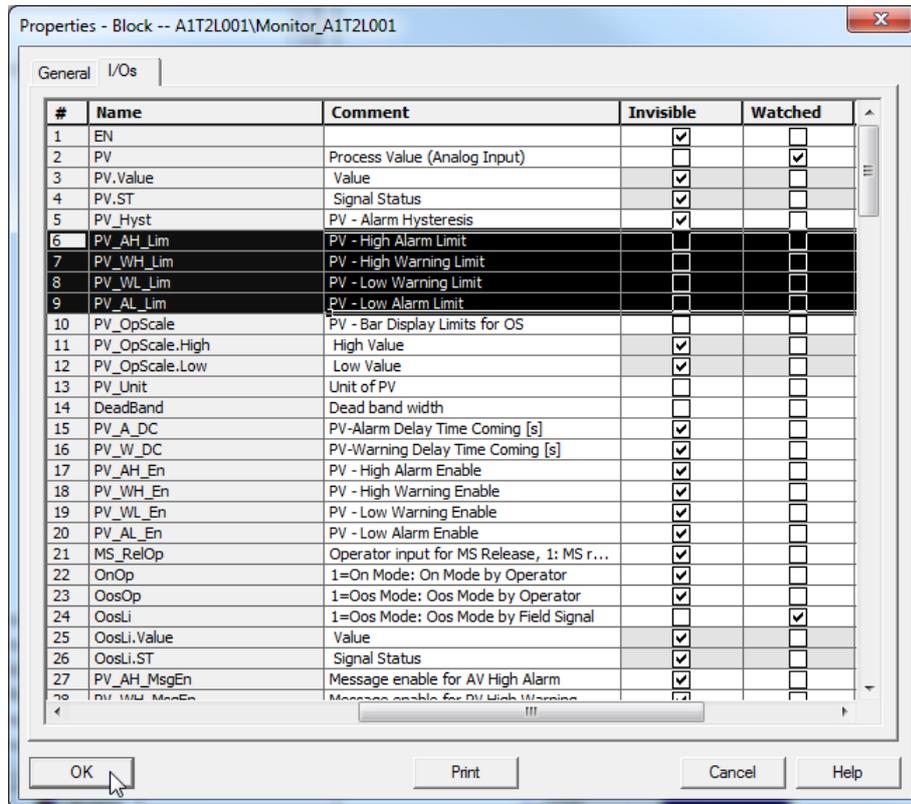


**Note:** The block MonAnS is used to monitor a measured value (analog signal) for the limit pairs:

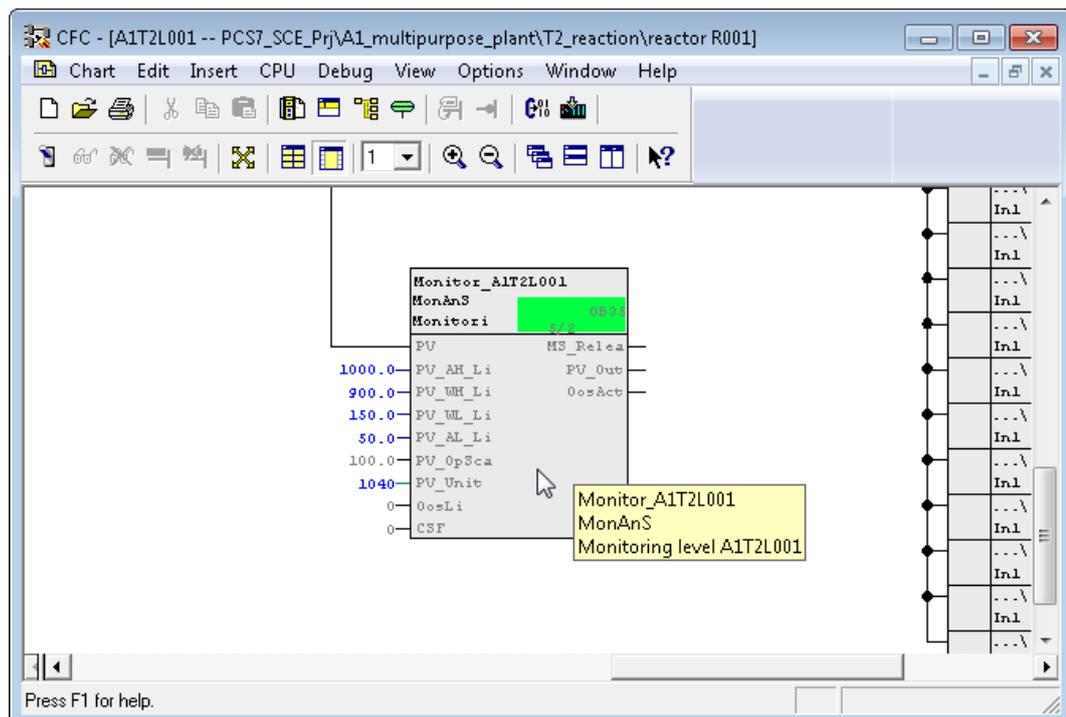
- Warning limit (high/low)
- Alarm limit (high/low)

- In the block properties, the connections shown for setting the upper and lower alarm/warning limits are made visible.

(→ PV\_AH\_Lim → PV\_WH\_Lim → PV\_WL\_Lim → PV\_AL\_Lim → Visible(Invisible))



- At the connections that are now visible, the warning and alarm limits shown are set and the value of connection PV\_Unit is set to '1040'. It specifies the unit that is displayed on the screen of the OS or in the faceplate. An excerpt from Siemens documentation for possible units is also shown below.



Value	Display	Description
1000	K	Kelvin
1001	°C	Degrees Celsius
1002	°F	Degrees Fahrenheit
1005	°	Degree
1006	'	Minute
1007	"	Second
1010	m	Meter
1013	mm	Millimeter
1018	ft	Foot
1023	m <sup>2</sup>	Square meter
1038	L	Liter
1041	hl	Hectoliter
1054	s	Second
1058	min	Minute
1059	h	Hour
1060	d	Day
1061	m/s	Meters per second
1077	Hz	Hertz
1081	kHz	Kilohertz
1082	1/s	Per second
1083	1/min	Per minute

Here, the changes in chart 'A1T2L001Sheet1' are listed once more:

Table 2: New blocks in chart 'A1T2L001 Sheet1'

Block	Catalog/Folder	Number of connectors
MonAnS	Libraries/PCS 7 AP Library V81/ Blocks+Templates\Blocks/Monitor	

Table 3: Input wiring in chart 'A1T2L001 Sheet1'

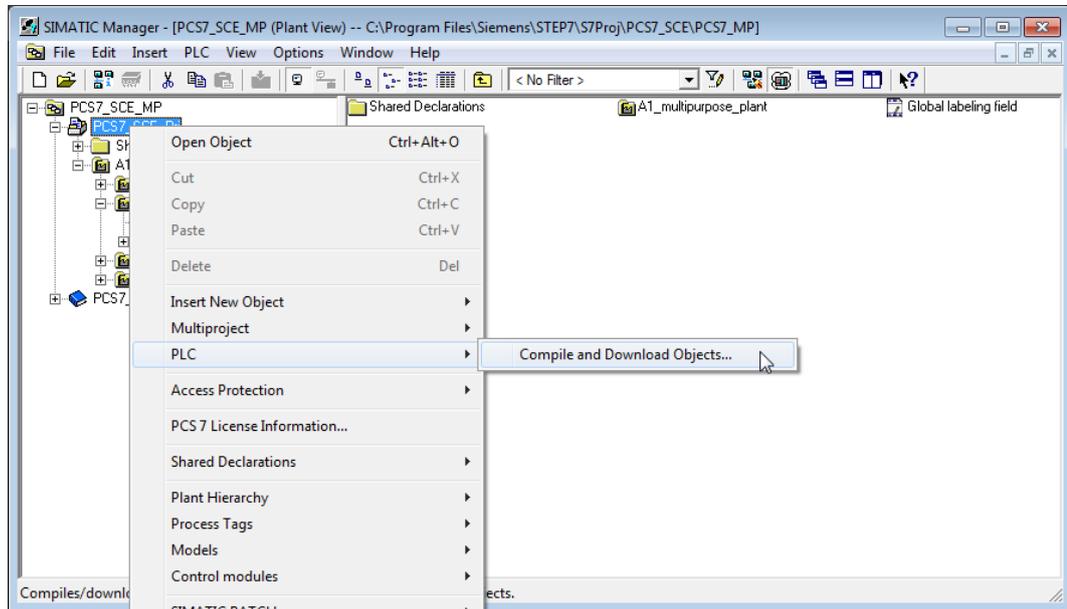
Input	Wiring to	Inverted
MonAnS.PV	Pcs7AnIn.PV_Out	
MonAnS.PV_AH_Lim	1000.0	
MonAnS.PV_WH_Lim	900.0	
MonAnS.PV_WL_Lim	150.0	
MonAnS.PV_AL_Lim	50.0	
MonAnS.PV_Unit	1040	
Pcs7AnIn.PV_InUnit	1040	

Table 4: Output wiring in chart 'A1T2L001 Sheet1'

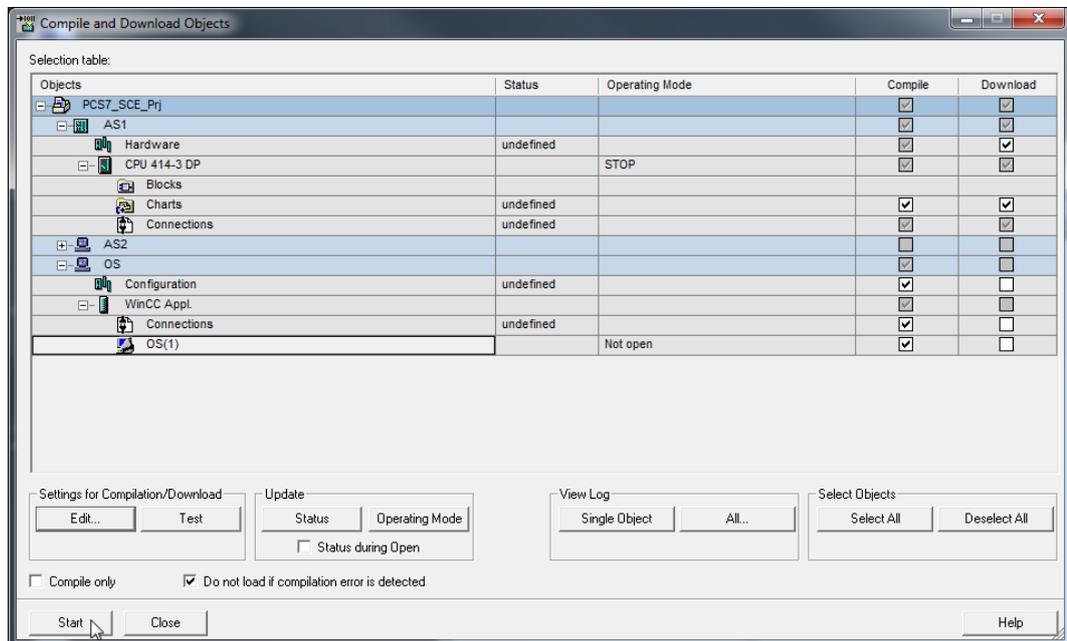
Output	Wiring to	Inverted
none	none	none

- To compile and download AS and OS at the same time, highlight the project in the component view of the **SIMATIC Manager**. Then select for the PLC 'Compile and Download'.

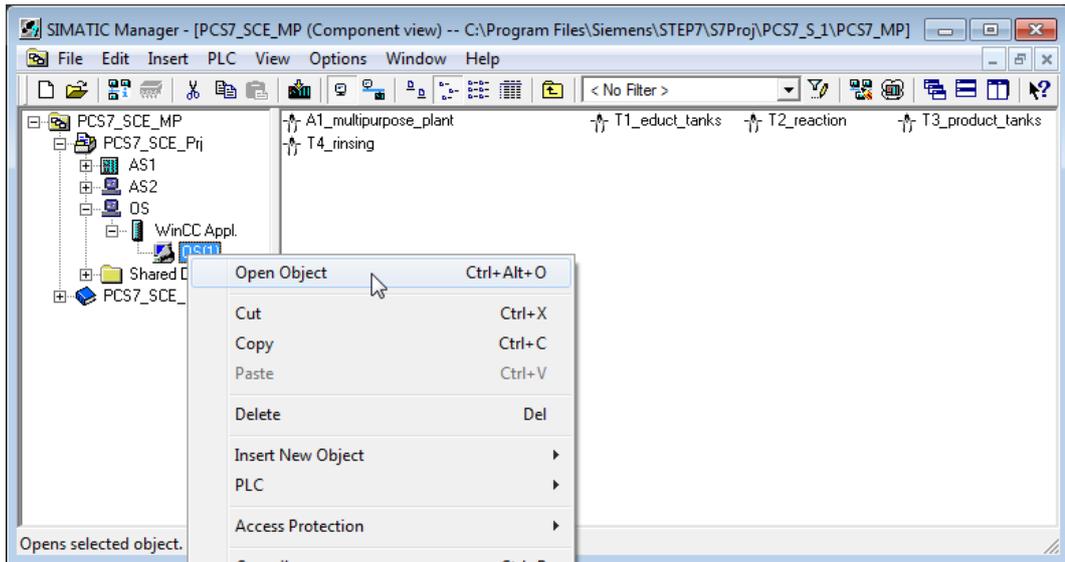
(→ SCE\_PCS7\_Prj → PLC → Compile and Download Objects)



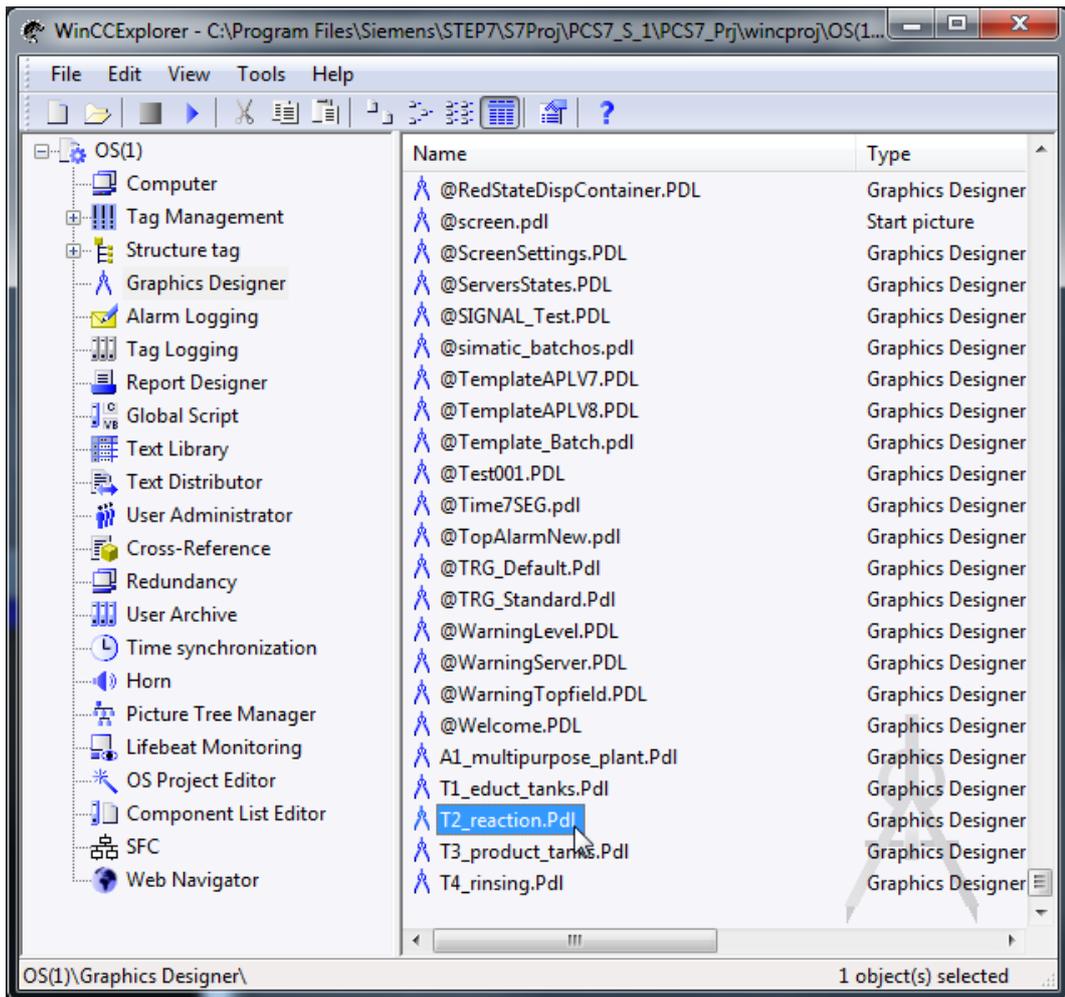
- Next—as shown here—select the objects for compiling and start the process as you learned in the previous chapters. (→ Start)



7. After the successful compilation, open the OS. (→ OS(1) → Open Object)

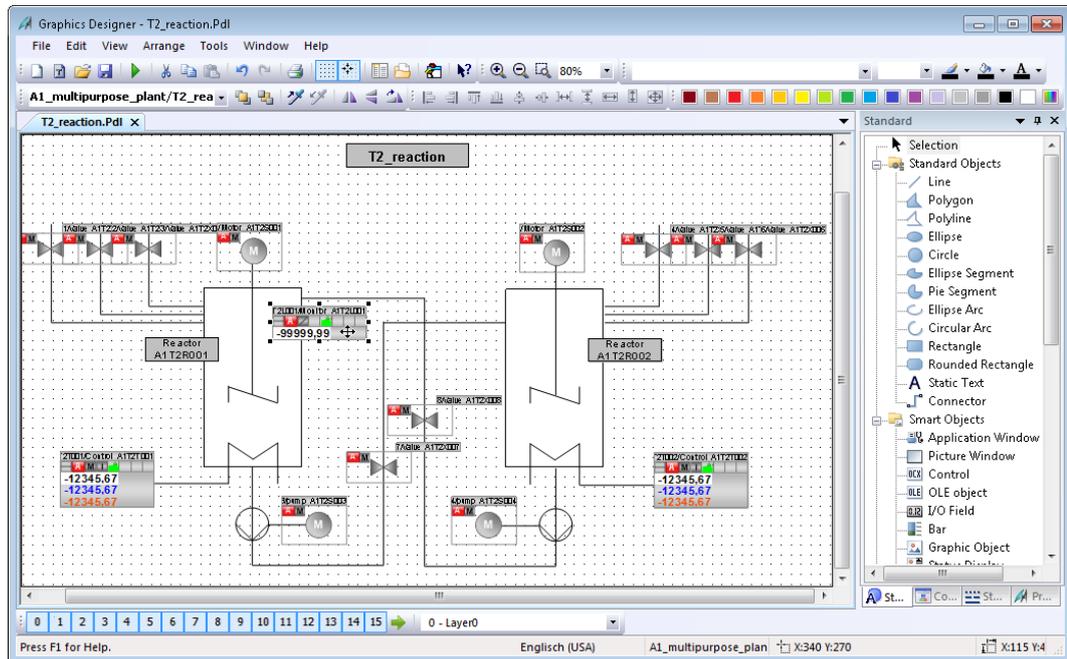


8. Within WinCC, open in the Graphics Designer the picture 'T2\_Reaction.Pdl'. (→ Graphics Designer → T2\_Reaction.Pdl)



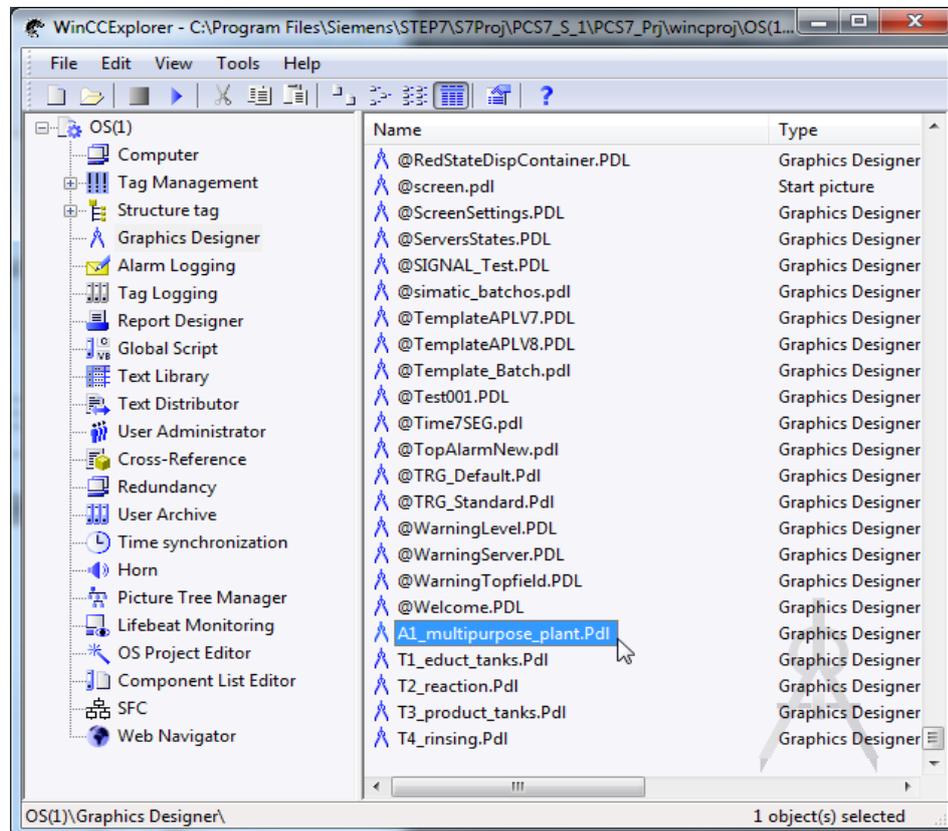
9. In this picture, the block symbol for the MonAnS block 'A1T2L001' was set up through the compilation run. Position it to the right of the reactor and save the picture.

(→ A1T2L001 → )

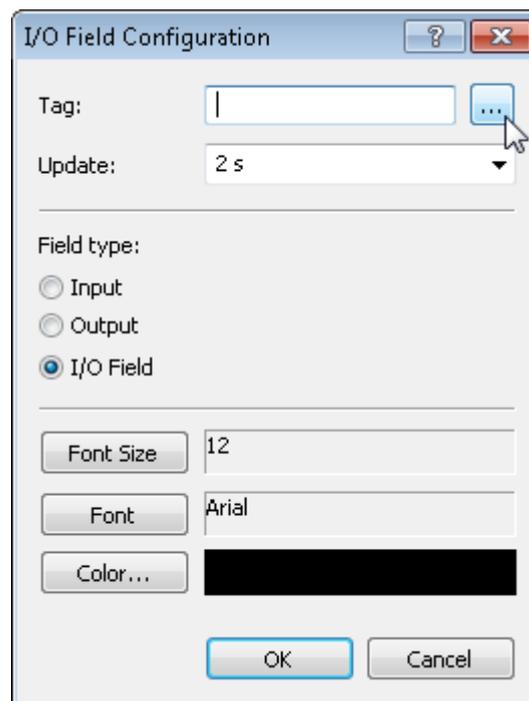
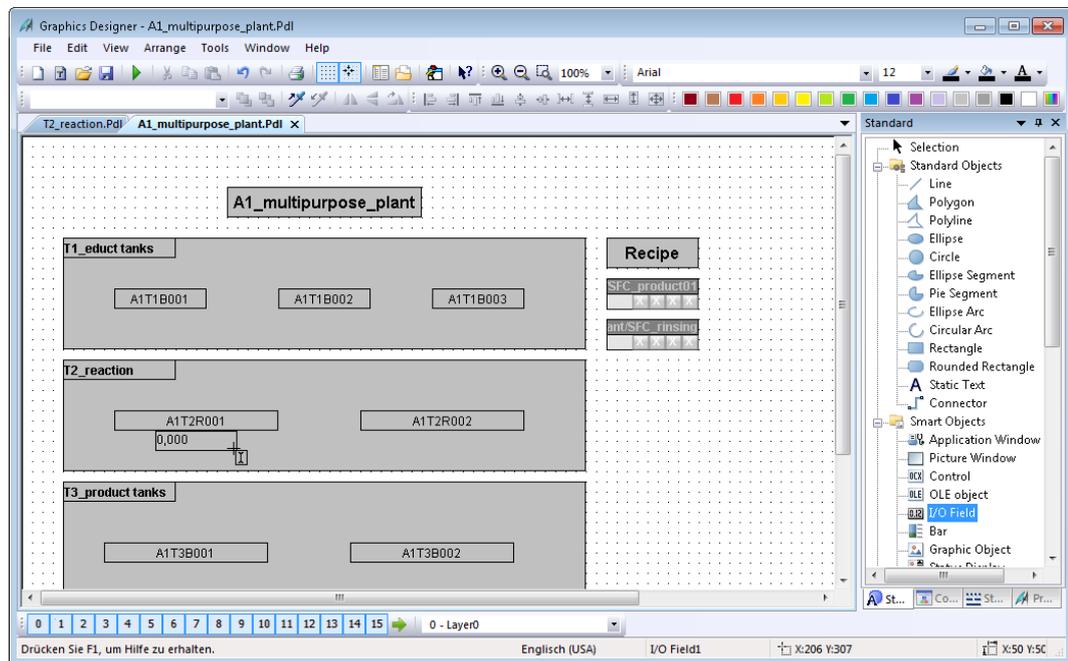


10. Now, in the Graphics Designer open the picture 'A1\_multipurpose\_plant.Pdl'.

(→ Graphics Designer → A1\_multipurpose\_plant.Pdl)

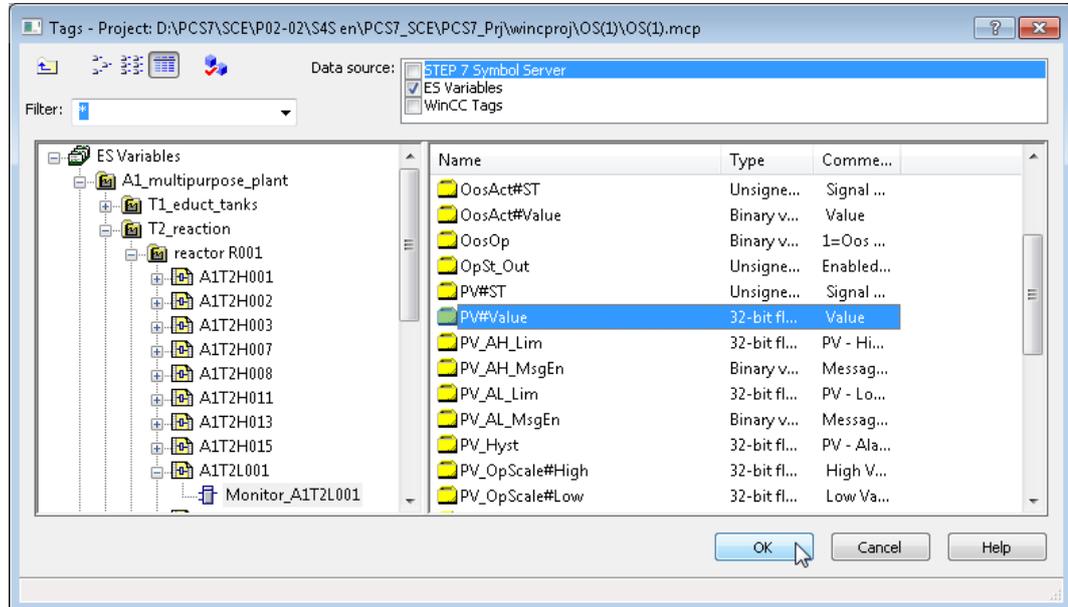


11. In this picture, drag an IO field from the smart object of the object palette to display the level of reactor A1T2R001. Then, open its variable selection. (→ Object palette → Smart Objects → I/O Field → )



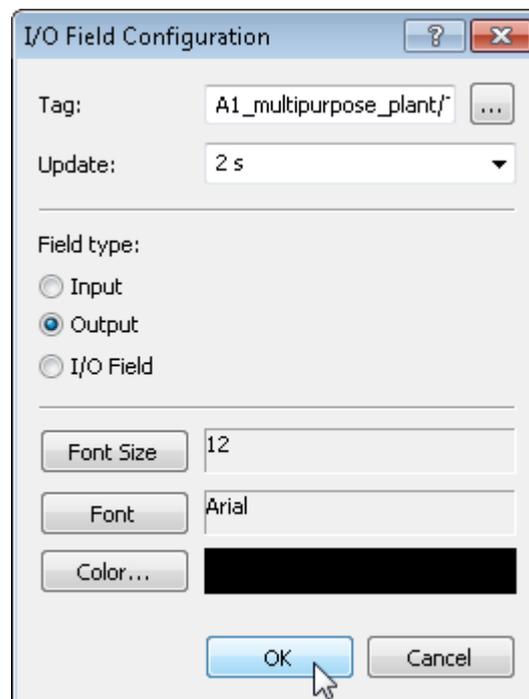
12. Within the variable selection, select as data source the ES variables. In the left window you will then see the hierarchy of your project. Here you can easily locate your MonAnS block. To display it in the IO field, select the connection 'PV#Value'.

(→ ES Variables → A1\_multipurpose\_plant → T2\_Reaction → Reactor R001 → A1T2L001 → 2 → PV#Value → OK)

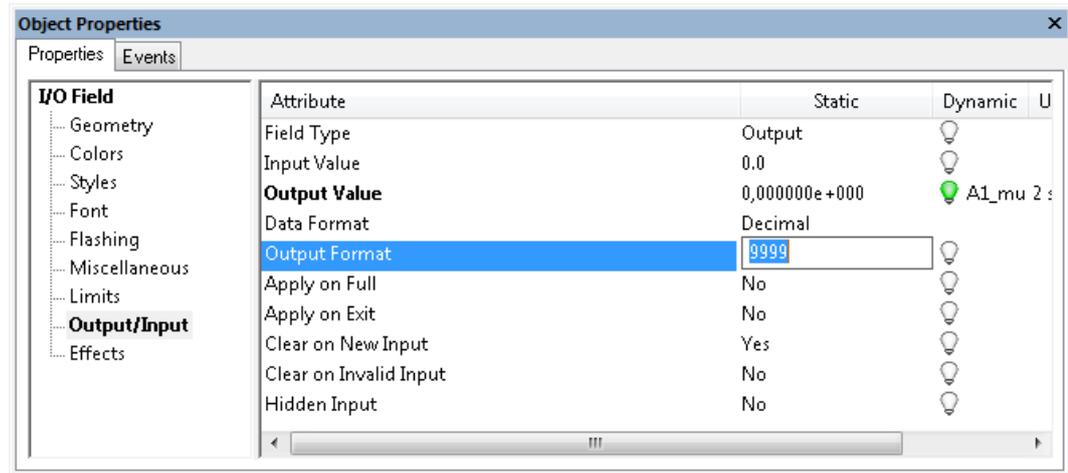


13. This variable is now displayed in the configuration dialog. After the following changes, accept this configuration.

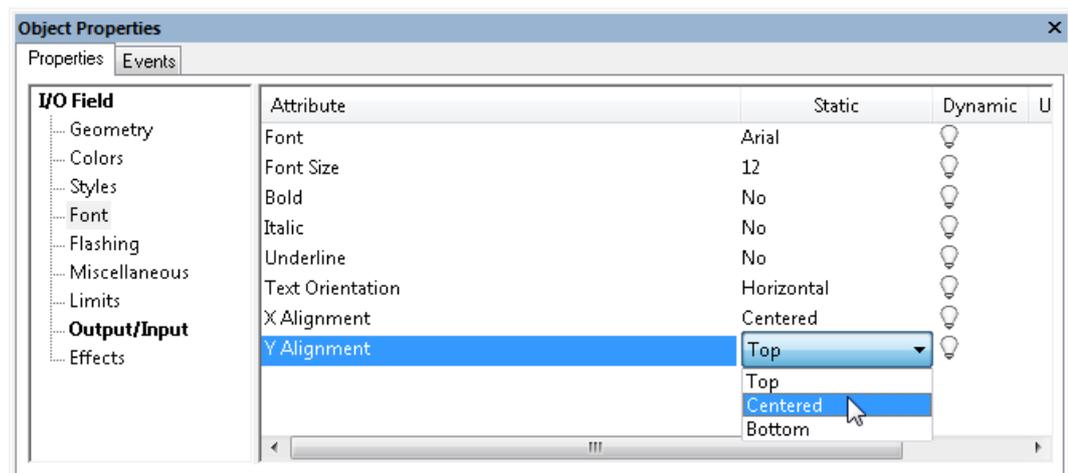
(→ Update: Upon change → Field type: Output → OK)



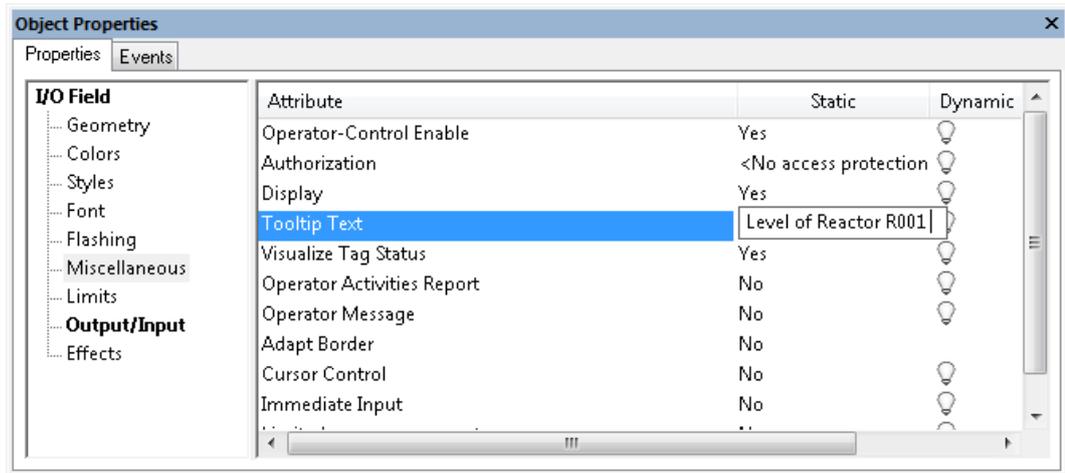
14. At Properties of the IO field, the output format is set for 4 digits before the comma (period) without decimal places. (→ Properties → Output/Input → Output Format → 9999 → OK)



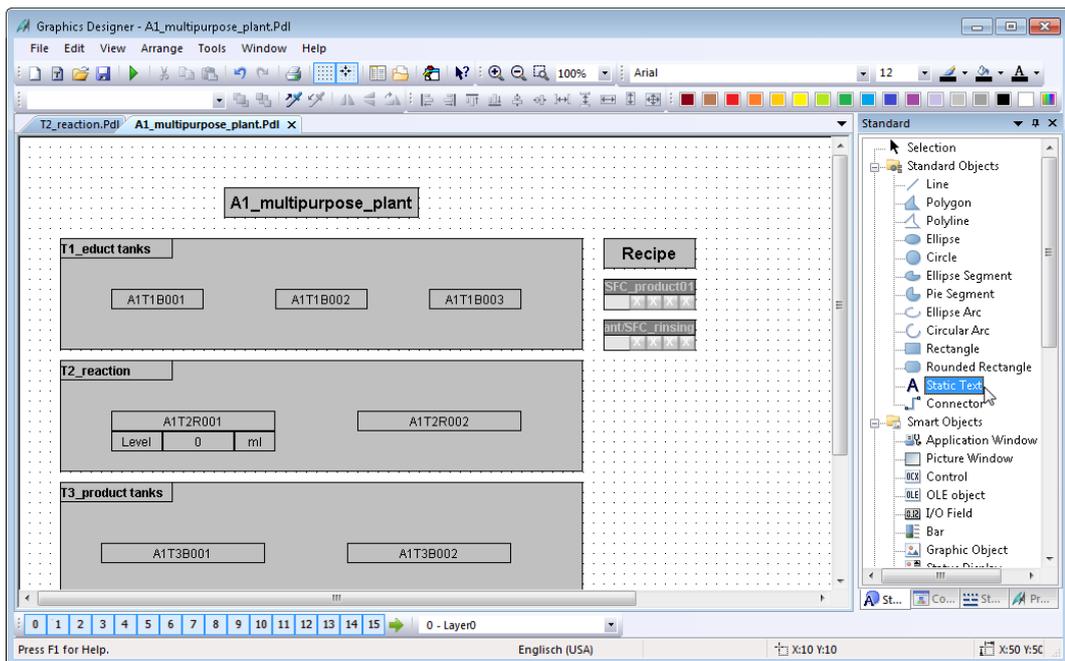
15. The following attributes for the font are selected. (→ Properties → Font → X-Alignment: Centered → Y-Alignment: centered)



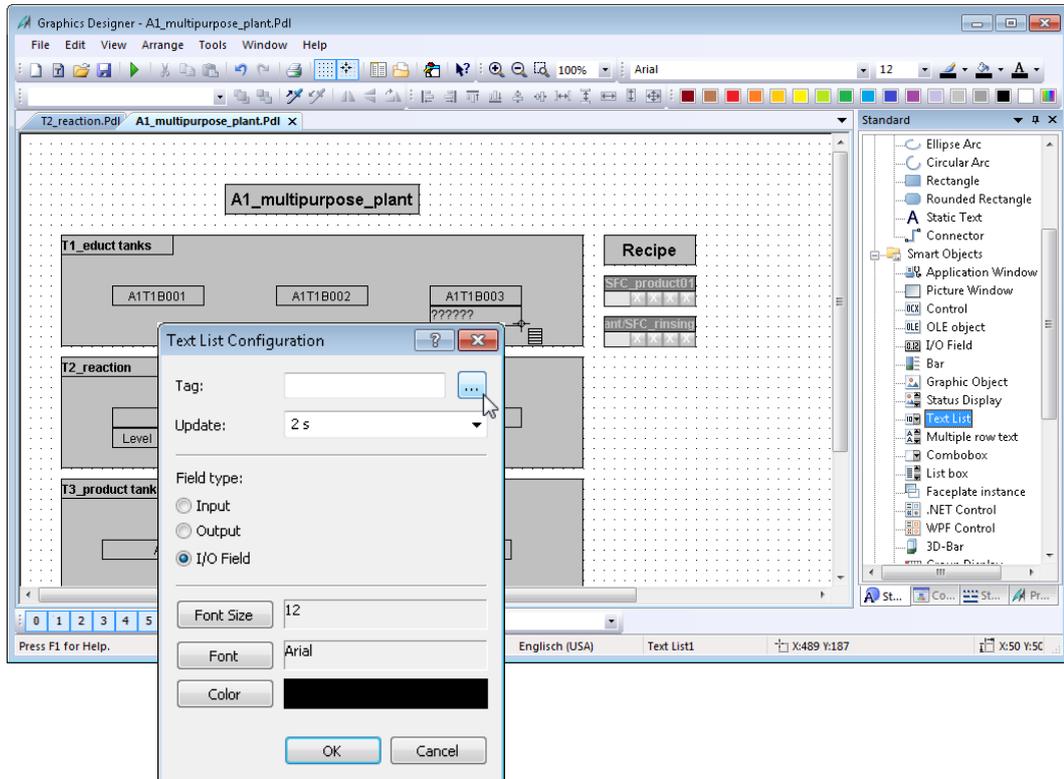
16. To better interpret the value in runtime, enter a tooltip text. (→ Properties → Miscellaneous → Tooltip Text: Level of Reactor R001 →  Close)



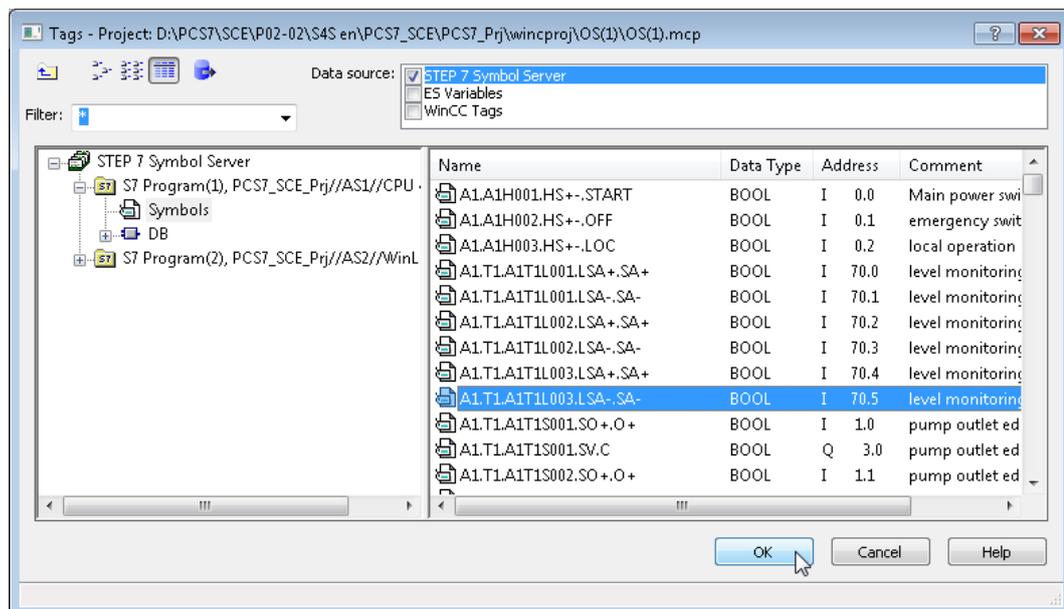
17. Next, position the I/O field under Reactor A1T2R001 and add two static texts 'Level' and 'ml'. (→ Object Palette → Static Text)



18. Below the educt tank A1T1B003, we want to display with a text list whether this tank is empty. After we dragged the display from the object palette into the picture, open its variable selection. (→ Standard Palette → Smart Objects → TextList → )

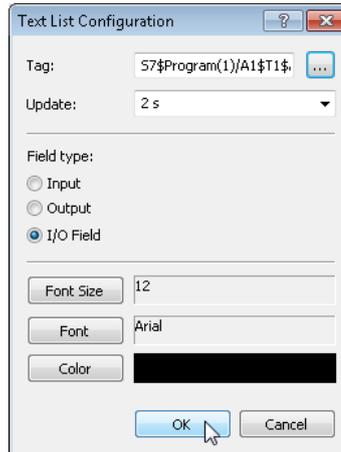


19. Within the variable selection, select this time as data source 'STEP 7 Symbol Server'. In the window to the left, the symbols of the S7 program are displayed. Select input. I70.5 'A1.T1.A1T1L003.LSA-.SA-'. (→ STEP 7 Symbol Server → S7 Program(1) → Symbols → I70.5 'A1.T1.A1T1L003.LSA-.SA-' → OK)



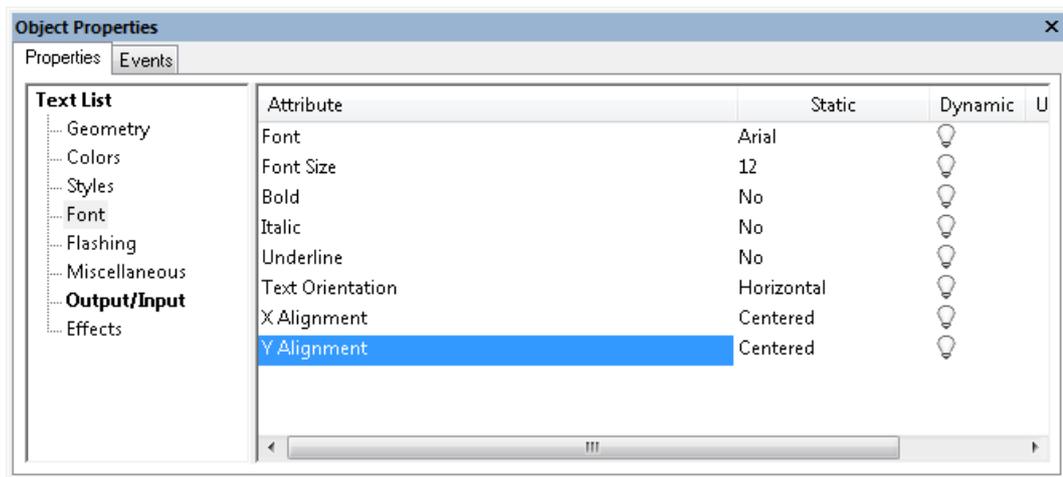
20. This variable is then displayed in the configuration dialog. After the following changes, accept the configuration.

(→ Update: Upon Change → Field Type: Output → OK)



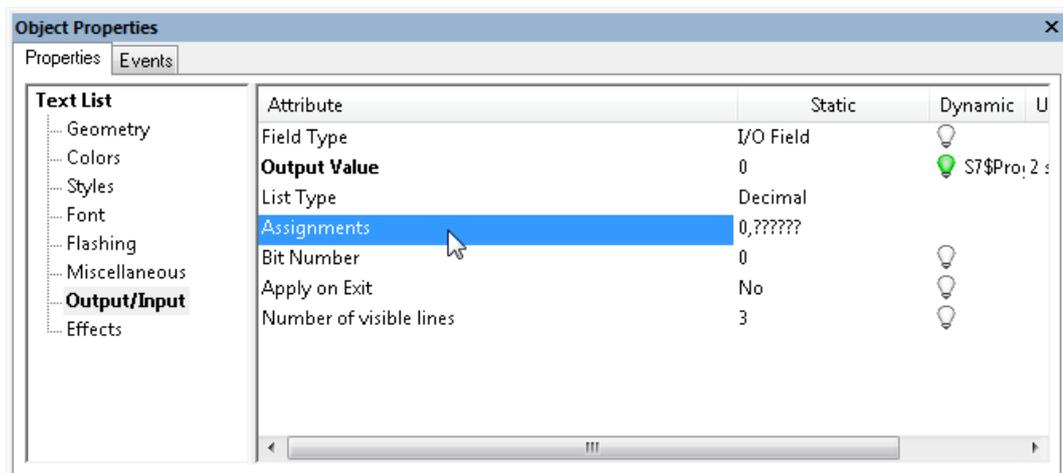
21. Next, we set the representation of the font in the properties of the text list.

(→ Properties → Font → X-Alignment: Centered → Y-Alignment: Centered)

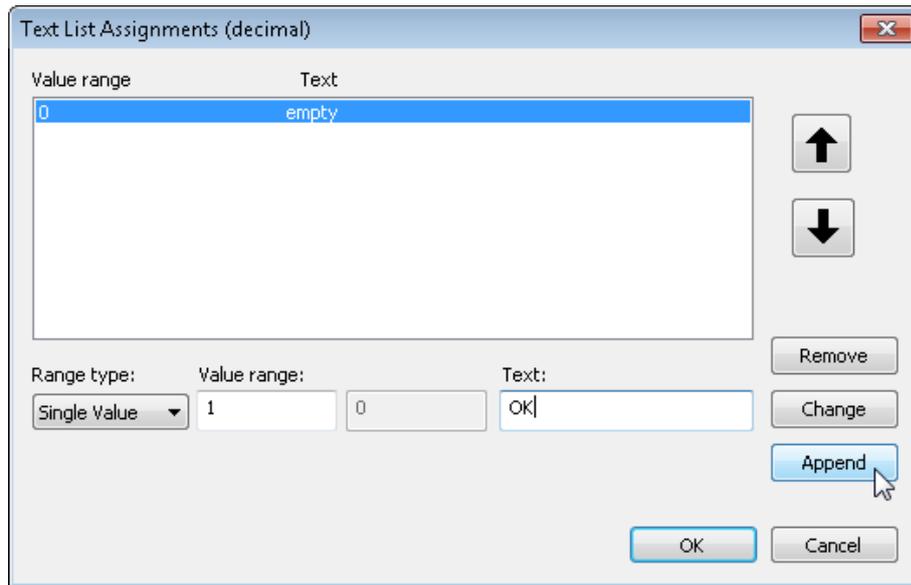


22. The texts are assigned to the values of the variable in the Properties also.

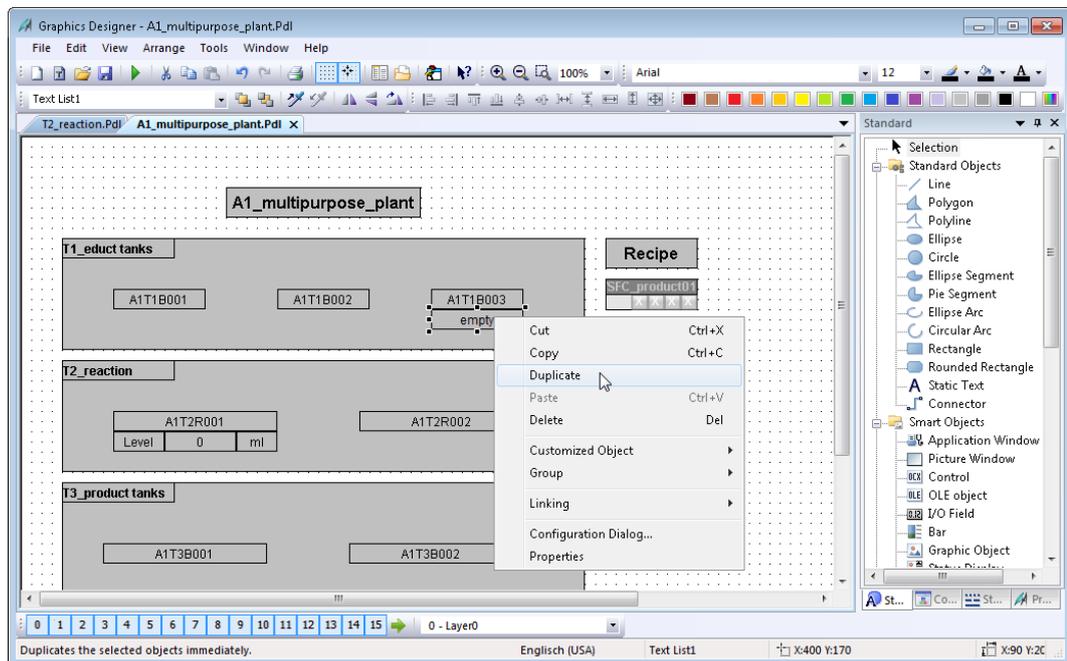
(→ Properties → Output/Input → Assignments)



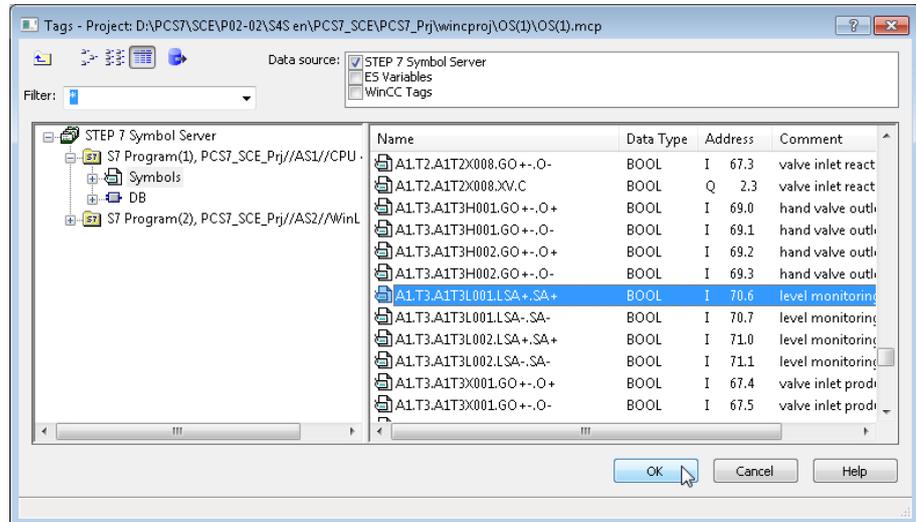
23. To the value 0 we assign the indication 'empty' and to the value 1 the indication 'OK'.  
 (→ Range type: Single Value → Value range: 0 → Text: empty → Change → Range type: Single Value → Value range: 1 → Text: OK → Append → OK → Close)



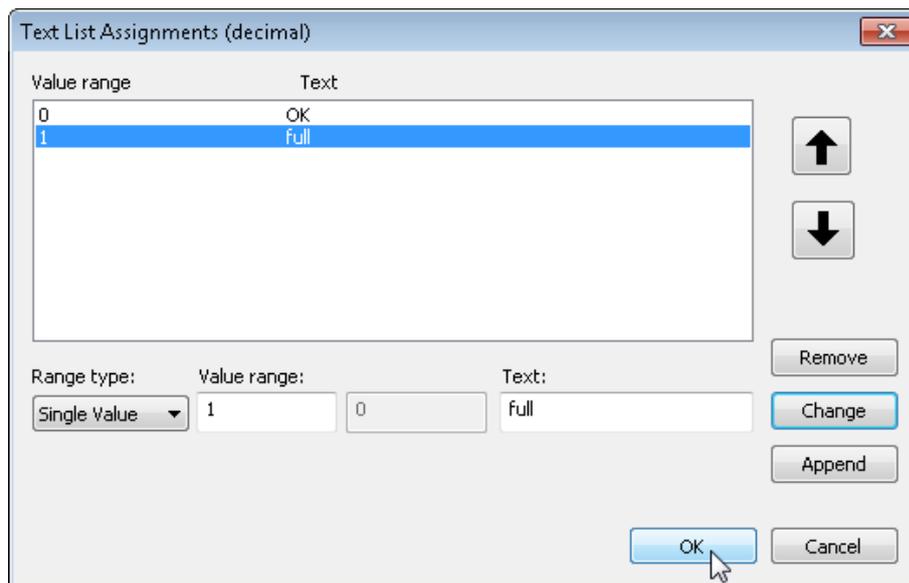
24. You will need exactly such a text list once more for a textual display below the product tank A1T3B001. It indicates whether the tank is full. The text list we already generated is now highlighted and duplicated. (→ Duplicate)



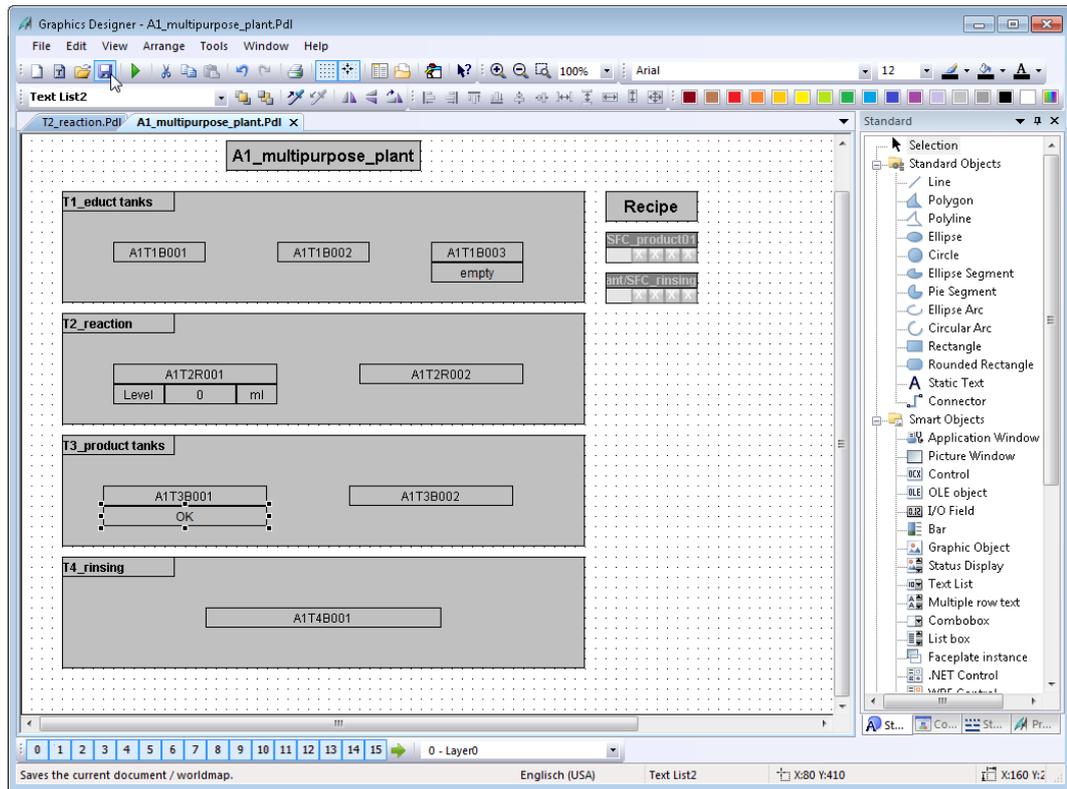
25. For product tank A1T3B001, select input I70.6 'A1.T3.A1T3L001.LSA+.SA+'. (→ STEP 7 Symbol Server → S7 Program(1) → Symbols → I70.6 'A1.T3.A1T3L001.LSA+.SA+' → OK)



26. Now, the assignment is changed for the attributes: To the value 0 we assign the indication 'OK' and to the value 1 the indication 'Full'. (→ Range type: Single Value → Value range: 0 → Text: OK → Change → Range type: Single Value → Value range: 1 → Text: Full → Change → OK)

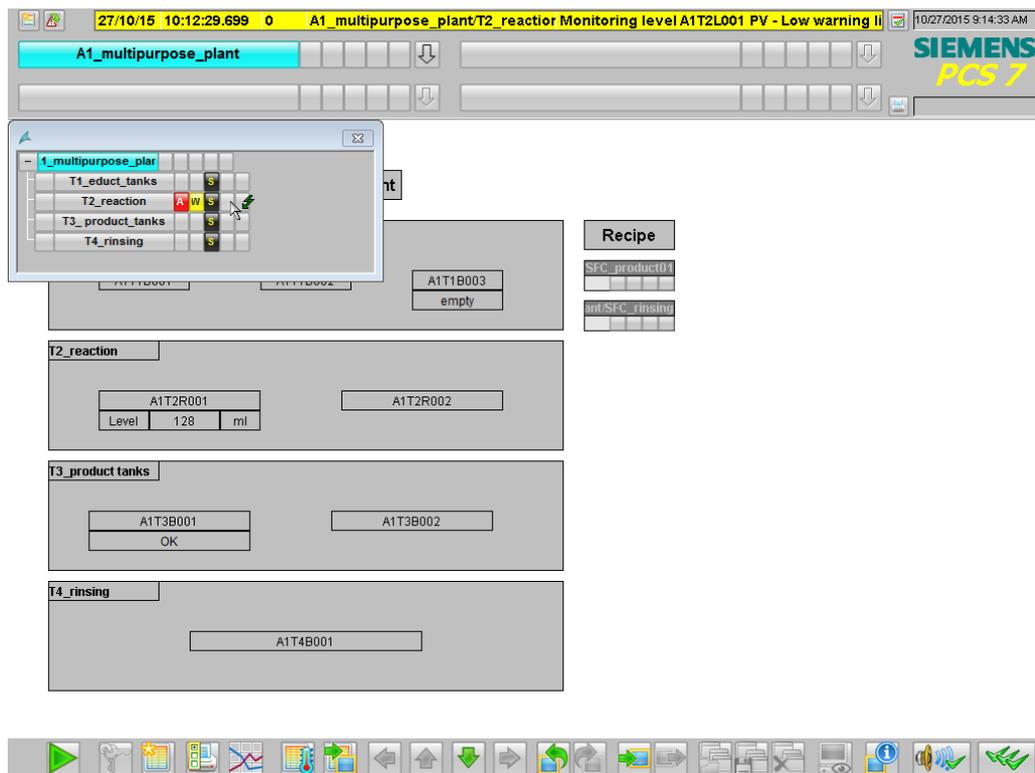


27. We now position the second text list below the product tank A1T3B001 and save the picture. (→ Save)

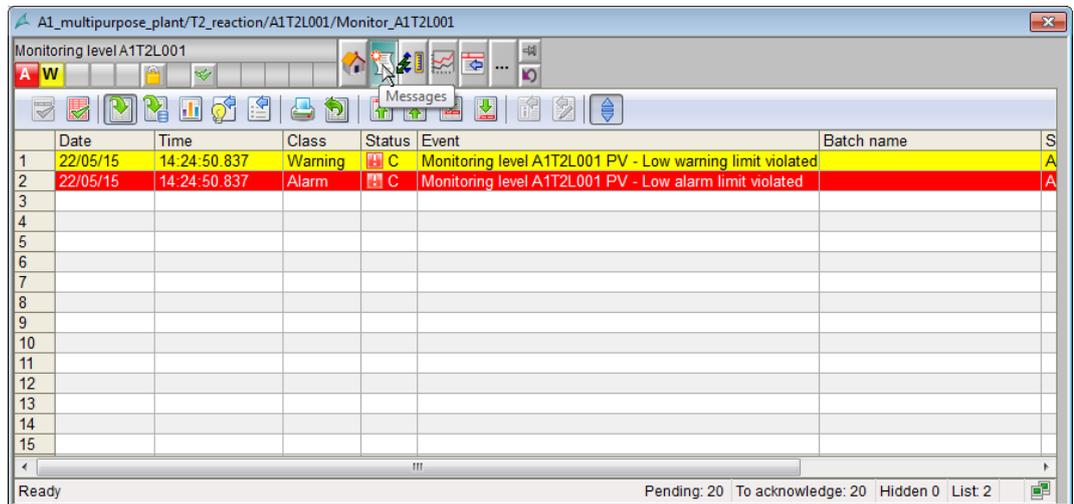
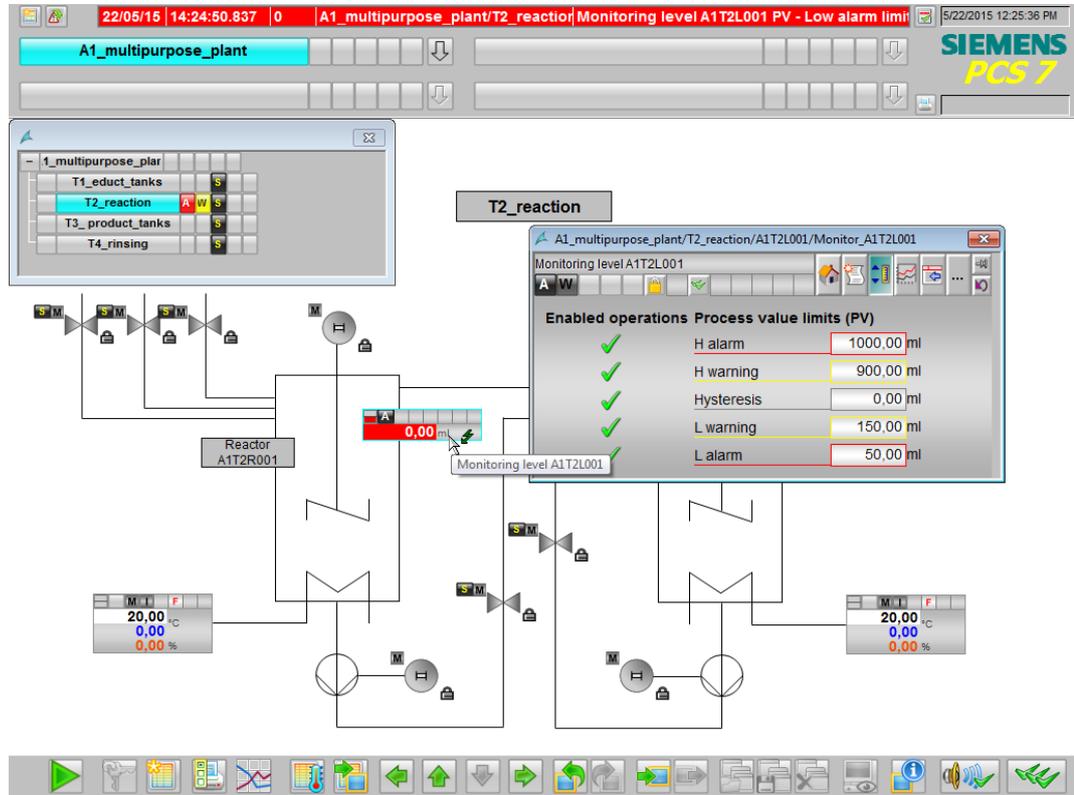


28. An illustration follows that shows how a warning is represented in runtime. In the message line, the warning or alarm that occurred last but has not been acknowledged appears. If the operator wants to switch directly to the picture where the alarm or warning was triggered, he can do this in two ways:

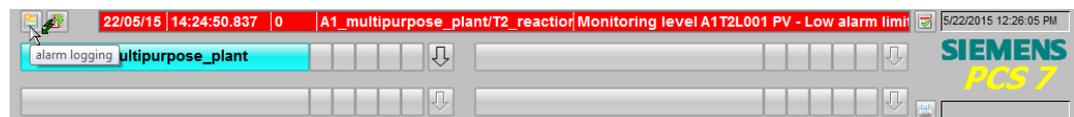
- With the button  Loop in Alarm in the message line
- By clicking on the fault indicator  in the display hierarchy (→  → )



29. In the picture 'T2\_Reaction', a display of the warning is provided in the block icon. By clicking on the block icon, additional information is displayed in the faceplate for the MonAnS block regarding the cause of the warning. Here, messages requiring acknowledgement can be acknowledged or limits can be adapted.



30. By clicking on the button Alarm System  in the message line, all queued messages are displayed. (→ )



## EXERCISES

In the exercises we apply what we learned in the Theory section and in the Step by Step Instructions. We are using and expanding on the existing multi-project from the step by step instructions (PCS7\_SCE\_0202\_R1504\_en.zip).

In the step by step instructions, only the alarm for a level of only one reactor was implemented. In the exercise, we are completing the alarm configuration of the level for reactor R002 as well as implementing the alarms and warnings of temperature overrange for both reactors. Finally, we are adding to the overview picture the missing displays.

## TASKS

The tasks below are based on the step by step instructions. For each task, the corresponding steps of the instructions can be used as an aid.

1. Implement alarm generation also for the level of the second reactor. To this end, insert the block MonAnS in the CFC A1T2L002, assign a name and parameters to it. After compilation, position the block in the operator screen 'T2\_Reaction.pdl'.
2. Next, in the overview display 'A1\_multipurpose\_plant.pdl' insert an I/O field for the level of reactor A1T2R002 and visualize the current reactor level.
3. Design alarms for the temperatures of the two reactors analogous to the levels. The temperature warning is to be triggered at 55°C and the alarm at 60°C. No additional monitor blocks are needed to design the temperature warnings and alarms since the PidConL blocks include this functionality. The connections for the warning limits and the alarm limits have the same name there as for the MonAnS blocks.
4. Now, set up I/O fields for the temperature below the level in the overflow display.
5. Finally, add the missing text lists for all elements.

