FUNCTIONAL SAFETY

OBJECTIVE

After working through this instruction module, the students know the basic requirements regarding functional safety. They also know the methods for identifying potential danger and for evaluating the risks resulting from it. They know methods and planning resources to secure plants by using the resources of process control engineering. They know the basic operations to lock controllers.

THEORY IN SHORT

In modern production plants, process variables are used to control technical processes and to make them safe. Based on the given technical boundary conditions, permissible and impermissible ranges are defined for these variables. The state of the entire plant results from the current values of all process values. The objective of plant safety is to prevent the plant entering an impermissible fault state. To this end, corresponding locking mechanisms are set up. The objective of lock states is preventing combinations, sequences, and time characteristics of signals that can lead to impermissible faults.

This can be done -among other things- with the resources of process control engineering through so-called protective devices. They prevent faults from occurring, or they limit the damage if a fault occurred despite the measures that were taken. To design suitable locking mechanisms, a protection concept has to be developed for the plant. This task requires exact knowledge of the boundary conditions regarding chemical, process and plant engineering. For that reason, the protection concept is developed by an interdisciplinary work group with the aid of HAZOP or PAAG analysis.

The technical implementation of the mechanisms in a process control system is to be designed as simple as possible; it has to have a direct effect, and it has to be easy to follow. In practice, recurring standard wiring is often resorted to for that reason. It can be arranged in 4 categories:

1) Combinational circuits are used to generate switching conditions by directly combining the corresponding process signals. To this end, the input signals are combined with the logical AND, OR and NOT operations. The status of the output signal of such a combination circuit can be determined anytime through the states of the input signals.
2) Prioritization circuits permit granting some signals priority over other signals. This is often necessary when selecting operating modes, and for start and stop functions. Prioritization circuits are often implemented with combination circuits.
3) Locking circuits prevent that signals that have the opposite effect can be set at the same time. If beyond this a certain sequence for several control signals is required, we refer to this as a sequence lock. Locks are implemented by using R-S storage elements that are interfaced.
4) Circuits with time response allow for the following: delayed switch-on and switch-off, defining a minimum and maximum execution time as well as the implementation of protection functions that require a certain response time. To implement such circuits different preassembled time blocks are available.

THEORY

PROCESS VARIABLES

Production plants manufacture material goods. To this end, they control and monitor material and power flows that can be described by physical variables such as volume, mass, temperature or volume stream. Based on process engineering and plant engineering boundary conditions, those physical variables are defined and specified that are relevant to the technical process and can be recorded by measurement. These variables are called process variables.

Process variables are used to control or secure technical processes. For each process variable, ranges are specified based on the chemical, process and plant engineering boundary conditions to which this process variable is applied (normal range) as well as ranges outside the normal range where, from the safety engineering aspect, no restrictions exist for continued operation (permissible fault range). If a process variable is located outside this range, undesired events that directly cause bodily harm or environmental damage have to be reckoned with (impermissible fault range).

The values of the process variables are recorded and evaluated by using resources from process control engineering: Three basic states are differentiated:

Normal state: The values of all process variables are in their respective normal range and the plant poses no danger in other respects either.
Permissible fault state: The values of one or several process variables are within the permissible fault range respectively, and the plant poses no danger in other respects either.
Impermissible fault state: The values of one or several process variables are in the respectively impermissible fault range, or the plant poses a danger in other respects.

Impermissible fault states exist if life and limb of employees are in jeopardy, the environment is damaged, technical devices are destroyed or production results are decreased. It suffices that a sufficiently high probability exists for one of these events to occur [1].

FUNCTIONAL SAFETY

In general, functional safety refers to securing the process engineering plant against faults. [1]. Regarding processes and states in process plants, certain events can occur that cause damage. The combination of the frequency with which damage occurs and its extent is referred to as the risk of the corresponding process or state. The objective of functional safety is to implement protective measures that decrease the existing risks to the extent that the remaining risk is below a limit risk that has to be defined; i.e., is justifiable [2].

The locking mechanisms discussed in the chapter 'Individual Drive Function' protect the plant or parts of a plant from faults caused by devices. That includes all those faults that are caused by a faulty function of the devices themselves, or by operating the device outside the permissible operating range (for example, a pump overheats because a dry run was not detected). These faults are device specific and can be detected regardless of the process engineering and plant engineering boundary conditions.

The locking mechanisms discussed can not of themselves protect a plant from faults caused by processes (for example, the dry run of a pump), since they depend on the process engineering and plant engineering events (for example, dropping below the minimum level of a tank causes the pump to run dry). For that reason, plants have to be made safe by implementing suitable, process related locks. These locks often utilize and expand the locking mechanisms of the individual drive functions (refer to chapter 'Individual Drive Function'). All types of the intended operation of the plants have to be taken into account.

Intended operation refers to the operation for which the plant is intended and configured according to its technical purpose [2]. Usually, this includes the following operating modes:

- Normal operation
- Startup and shutdown mode
- Commissioning and decommissioning
- Trial mode
- Inspection, maintenance and repair processes

First, an interdisciplinary work group develops a protection concept for the plant. The work group systematically identifies danger potentials and errors that can lead to danger. To this end, recognized methods for danger analysis are used; for example, the PAAG method [3].

Then, the risks have to be evaluated that result from the dangers that were recognized. Different methods for a graduated evaluation of the risk that is to be covered are available for this; for example, the ALARP method, the LOPA method or the risk graph method specified in [2]. If the initial risk of danger is larger than the limit risk specified, protective steps have to be taken to reduce the risk accordingly.

FUNCTIONAL SAFETY BY USING PROCESS CONTROL ENGINEERING RESOURCES

In principle, protective setups should preferably be used for functional safety that are not based on resources of process control engineering. However, this is often not possible, not sufficient, or the corresponding solution can not be implemented advisably under economic considerations because of the size and the complexity of the plant. In this case, protective functions are implemented with resources from process control engineering. For that reason, we differentiate two types of process control engineering systems:

Basic process control systems (BPCS) implement the automation functions and therefore serve the intended purpose of the plant in its normal range [2]. Basic process control systems respond when one or several process variables move outside the normal range. They indicate permissible faults or they automatically take steps to return the process variables to the normal range. From the view of functional safety, nothing is required of the BPCS.
Safety instrumented systems (SIS) are used to reduce risk when danger potentials are recognized. They can either prevent events or limit the damage. The objective of event-preventing PCT protective devices is to stop impermissible faults from occurring in the plant in the first place. They thus reduce the frequency of an undesired event occurring, and also the risk related to this event. The objective of damage limiting PCT protective devices, on the other hand, is to decrease the extent of the resulting damage if an undesired event already occurred, and lower the risk connected with it. Such protective devices are used only very rarely.

Figure 1 shows, in principle, the effect of PCT devices within the scope of functional safety. Curve Shape 1 shows a process variable that can't reach the impermissible fault because of the process. For that reason, a PCT monitoring device is sufficient here. In Curve Shape 2, however, exceeding the limit into the impermissible fault range is possible. However, since a non-PCT protective device is present, the PCT monitoring device is sufficient in this case also. In Curve Shape 3, such plant security is lacking. For that reason, an (event preventing) PCT protective device is used, in order to prevent that the process variable reaches the impermissible fault range.

For the process engineering systems of a plant, it has to be clearly defined whether an operating and monitoring function is to be implemented, or a protective function. This differentiation facilitates planning, setup and operation as well as later modifications of PCT equipment.

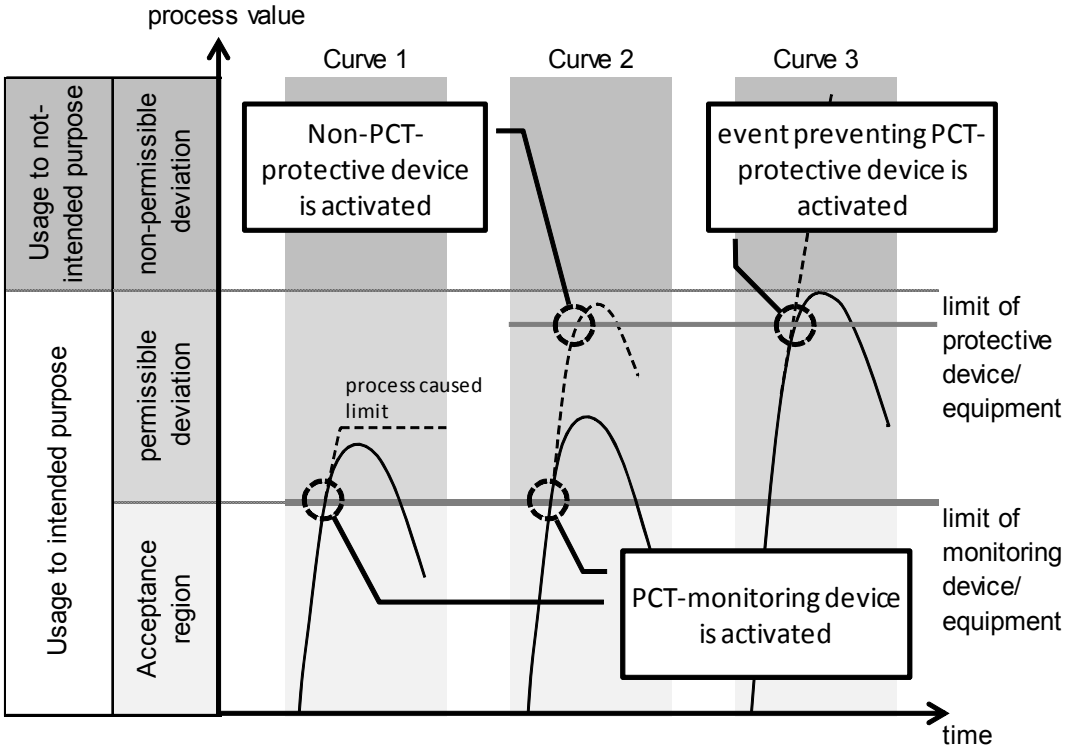


Figure 1: Mode of action of PCT equipment according to [2]

Since functions of PCT protective devices are requested only very rarely, operating devices sometimes -for economic reasons- utilize components of protective devices. In this case, signals that trigger the protective function must have priority over signals of the controlling and monitoring functions.

To implement protective functions, steps should be taken that they are as simple as possible, that they are easy to follow and that they have an immediate effect. It should be possible to record the process variables that are used directly and with simple and proven means. That is to say: the control design itself has to be of relatively low complexity.

STANDARD CIRCUITS FOR FUNCTIONAL SAFETY

The objective of protective devices that use process control engineering usually is to control certain combinations, sequences, time characteristics or the priority of signals in a way that impermissible process states are prevented. These functions are implemented with standard circuits that recur again and again. The most important standard circuits are discussed below.

Combinational Circuits

In many cases, certain control signals are permissible only if the process is in a certain state. This state can be described directly as the combination of the corresponding process signals. To link individual signals into a switching condition, simple combinational circuits can be used. They provide the option to specify the state of an output signal any time through the state of a set of input signals. To this end, the input signals are combined through the logic operations AND, OR and NOT. The combinational circuits themselves are stateless, i.e., they have no storage properties. The relationship between the input and the

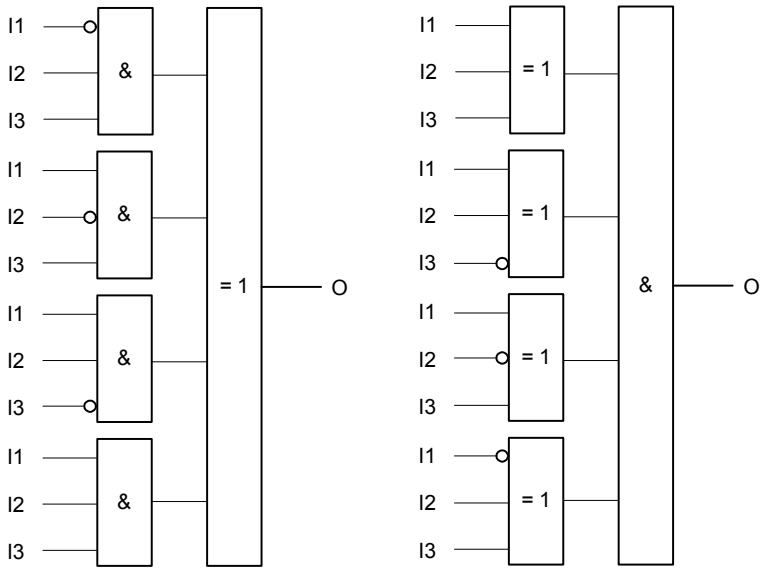
output signals can be described completely with a function table. The corresponding logic function can always be represented in (at least) two standardized forms.

Disjunctive normal form (DNF): In the case of this representation, first all combinations of the inputs are determined for which the output signal is to be set (i.e., all lines of the function table for which A = 1). These combinations are represented as AND operations of the input signals. The outputs of these AND operations are then combined by means of an OR operation. This sets the input as soon as one of the found combinations occurs.

Conjunctive normal form (CNF): In the case of this representation, first all combinations of the inputs are determined for which the output signal is to be set (i.e., all lines of the function table for which A = 0). These combinations are inverted and represented as OR operations of the input signals. The outputs of these OR operations are then combined by means of an AND operation. Inverting the found combinations causes the output to be set only if none of these combinations occurs.

Figure 2 shows an example of a function table with three input signals and the corresponding combinational circuits in the disjunctive and conjunctive normal form.

I1	I2	I3	O
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Function table Disjunctive normal form (DNF) Conjunctive normal form (CNF)

Figure 2: Structure of basic combinatorial logic circuit

Prioritization Circuits

Protective functions always have to have priority over control and monitoring functions. That is, here several control signal control the behavior of an actuating signal. For that reason, the control signals have to be prioritized accordingly. In most cases, prioritization is static and can therefore be implemented by using a combinational circuit.

Latching Circuits

It is not always possible to represent the conditions for an output state by the current state of the inputs alone. If, for example, output signal A is to be set by the input signal I1 and cleared by another input signal I2, this can no longer be represented combinationally. A has to remain set when I1 is cleared. Only when I2 is set is A to be cleared. This makes the effect of I2 dependent on whether I1 was set previously; i.e., by the current state Q of the system. This state has to be stored in the circuit. Such latching circuits are also called sequential circuits. Latching a state Q is implemented by using an R-S flip-flop.

As shown in Figure 3, such a circuit has two inputs, one input for setting (S) and one input for resetting (R) the output. Here it is important to define how the output is to be switched when both inputs are set. Depending on the implementation of the R-S flip-flop, either setting or resetting is dominant (refer to Figure 3).

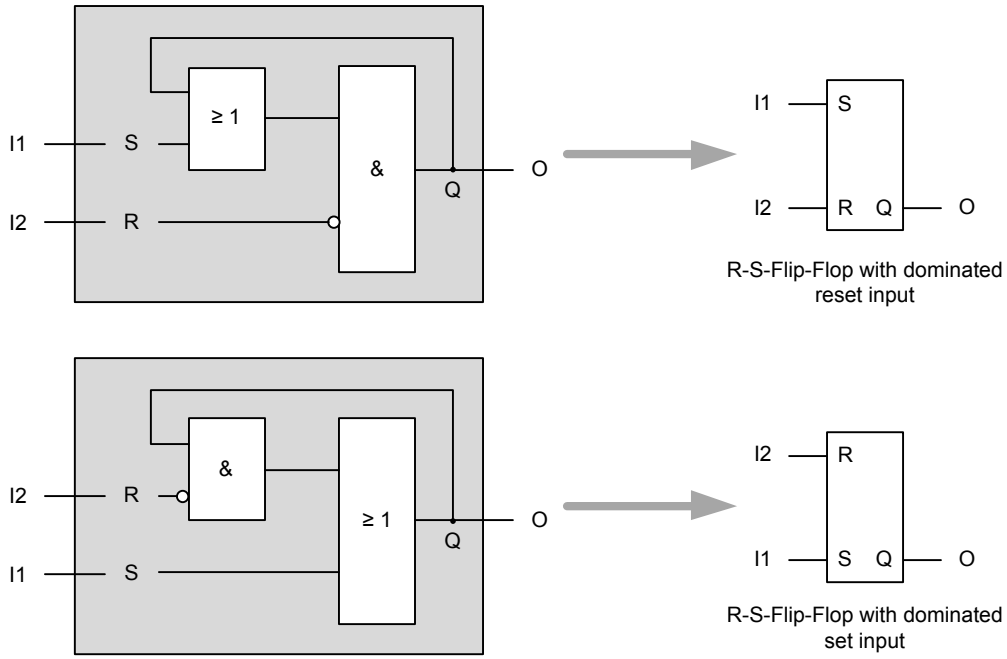


Figure 3: Design and function block of the R-S-Flip-Flop

Lock Circuits

Often, certain control signals must not be set at the same time. For example, an electrical motor with two speed directions must not be set to forward motion and backward motion at the same time. The two signals F (forward) and B (backward) must mutually lock each other.

A lock can be implemented with two R-S flip-flops that are coupled. In this case, two wiring options exist: the lock is set either by means of the set inputs, or the reset inputs. Both are shown in Figure 4. It should be noted that locking by means of the reset input works only if the reset input is dominant.

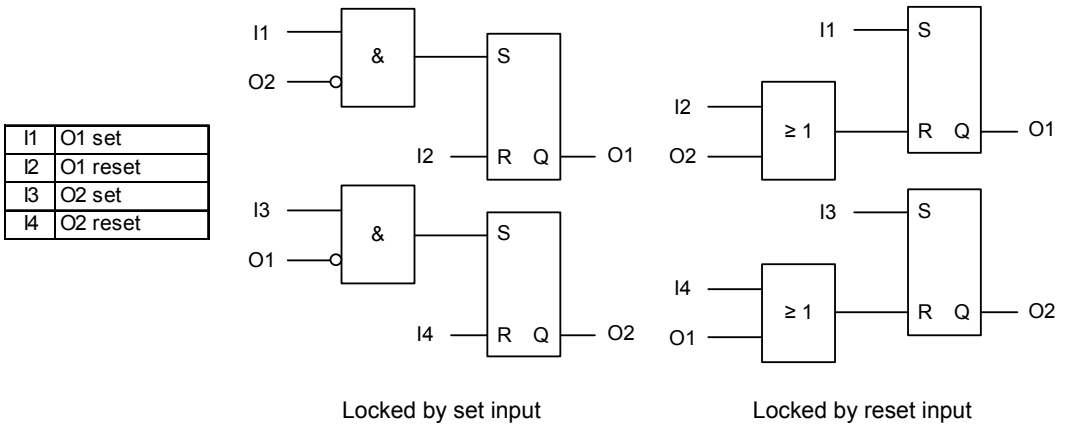


Figure 4: Interlock of two output signals

In some cases, the sequence in which certain control signals can be set has to be specified. In this case, a sequence lock has to be implemented. This also can be done by stringing together flip-flops. To do this, as many R-S flip-flops are needed as the number of steps to be coordinated. Figure 5 shows sequential locking for two signals.

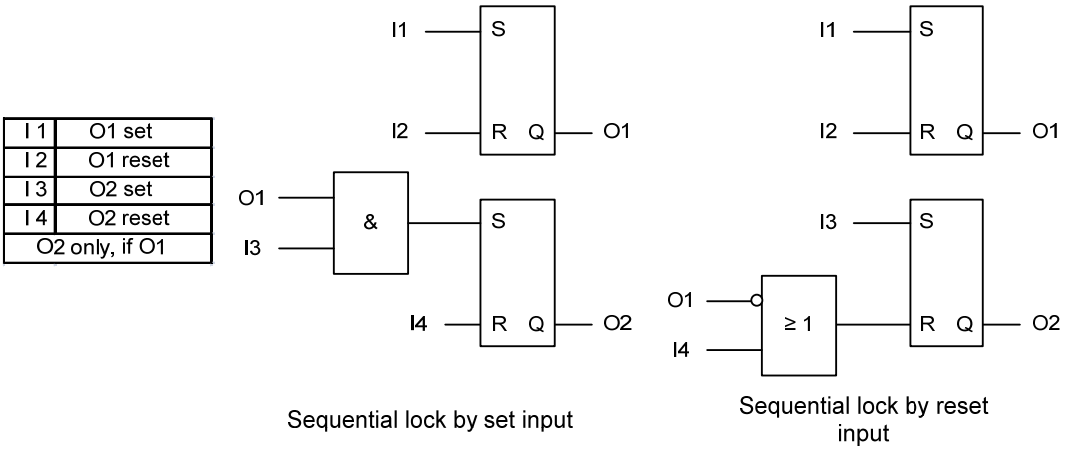


Figure 5: Sequential lock of two output signals

It has to be noted that with these circuits, only activation series are implemented, no sequences. Setting A2 does not cause A1 to be reset. When locking by means of the reset input, A2 is reset automatically when A1 is reset.

Circuits with Time Response

Circuits with time response also take into account the time since one or several events occurred. The principle is illustrated below using the ***two hand lock***. It is to prevent that workers are injured when operating a machine –a press, for example. To prevent that the worker still has one hand in the danger area of the machine, it can be activated only when two buttons are operated simultaneously. This task can also be solved by using a combinational circuit. To prevent, however, that one of the buttons is permanently set with adhesive tape, for example, it has to be ensured in addition that both buttons are pressed within a fixed time span. To this end, **impulse blocks** are used that -regardless of the length of the input signal set with respect to time- set the output signal for a specified duration, and then reset it automatically. Only a state change of the input (from reset to set) resets the output signal. Figure 6 shows the function symbol and the switching performance of an impulse block.

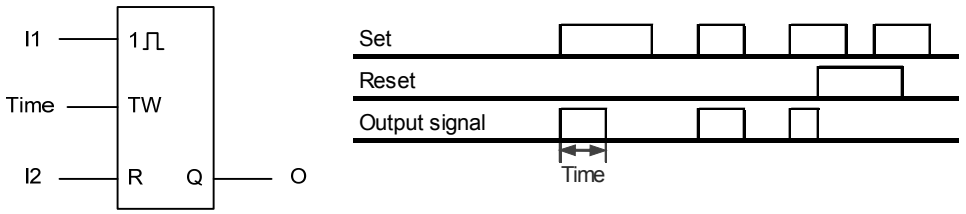


Figure 6: Function block and switching performance of an impulse block

The corresponding circuit for a two hand lock is shown in Figure 7. If one of the buttons is operated, output Q of the impulse block is set to the time T. If the second button is operated while Q is set, all conditions for the AND element are met, and output A is set. Then, the impulse block is jumpered by means of the OR operation with output A. .

I 1	Button left hand
I 2	Button right hand
T	Actuating time

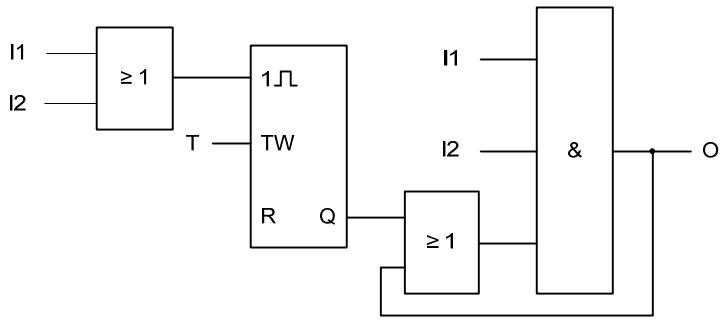


Figure 7: Two-hand lock when using an impulse block

Timers are also used for numerous protective functions in addition, for example for guard door controls where opened doors close again automatically after a specified time, or for motor startup controls, where after a futile start attempt, a rest period is forced for the drive to recover.

LITERATURE

[1] Strohrmann, G. (1983): Anlagensicherung mit Mitteln der MSR-Technik. Oldenbourg Verlag (Functional Safety Using MSR Technology, Oldenburg Publishers)

[2] VDI 2180 (Edition. 2007-04): (Securing Process Engineering Plants using Process Control Engineering).

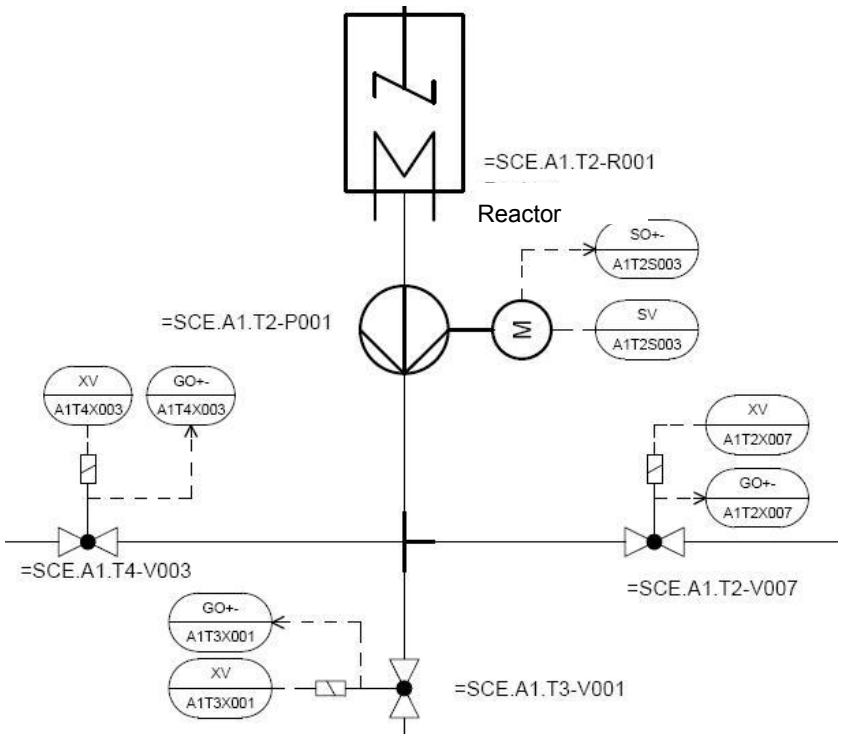
[3] DIN EN 61511 (Edition. 2005-05): Functional Safety Systems for the Process Industry)

STEP BY STEP INSTRUCTIONS

TASK

Corresponding to the information provided in the chapter 'Process Description', we are adding to the CF charts in chapter 'Individual Drive Function' the manual operation of the pump motor =SCE.A1.T2-P001. The following lock conditions have to be noted in this case:

- The pump motor must be switched on only when the main switch of the plant is switched on and the emergency stop switch is unlocked
- Product tank =SCE.A1.T3-B001 must not overflow. An encoder signals the maximum level.
- The pump must not draw in air. The minimum level (here: 50ml) in reactor =SCE.A1.T2-R002 is known numerically and is evaluated by means of the measured level.
- The pump must not press liquid against a closed valve. When the pump is switched on, always either valve =SCE.A1.T3-V001 or valve =SCE.A1.T2-V007 or valve=SCE.A1.T4-V003 has to be open.



Note: For the approach to the solution, please note the details in the theoretical part of the circuits to be latched.

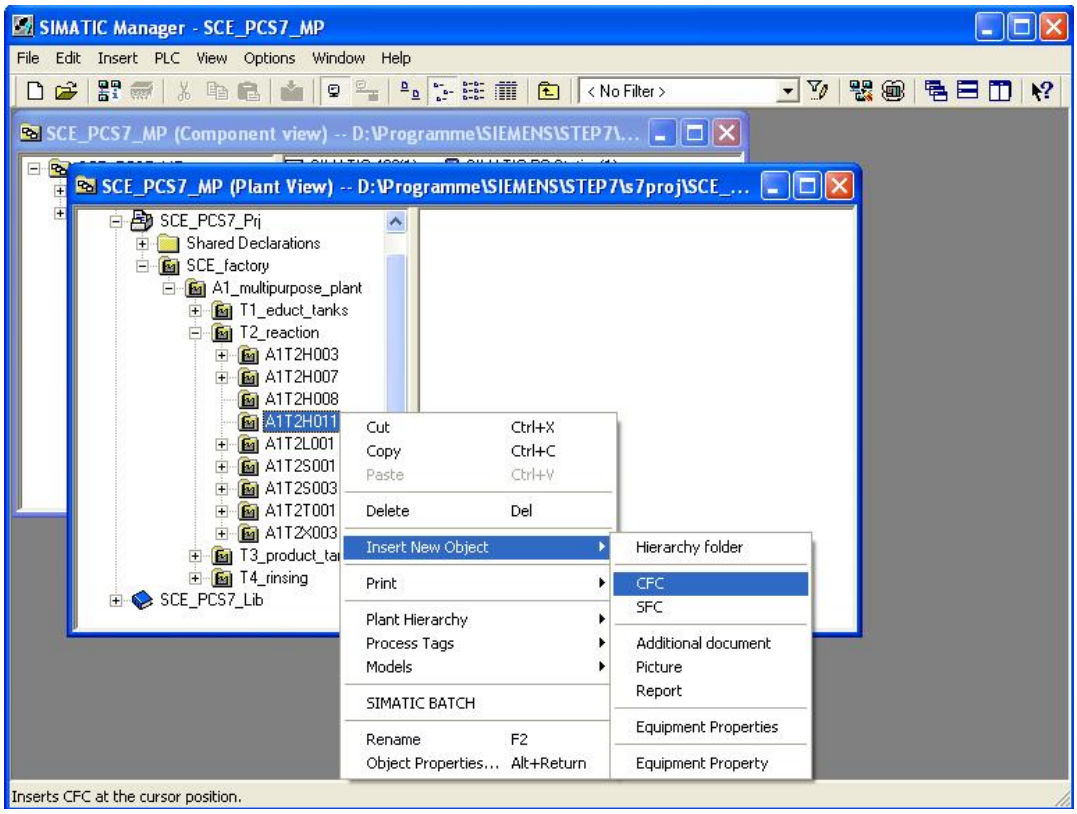
OBJECTIVE

In this chapter, the student will learn the following:

- Implementing expanded boundary conditions and manual operation
- Setting up wiring between CF charts
- Additional options for programming with CFC
- Utilizing additional sheets in the CF charts

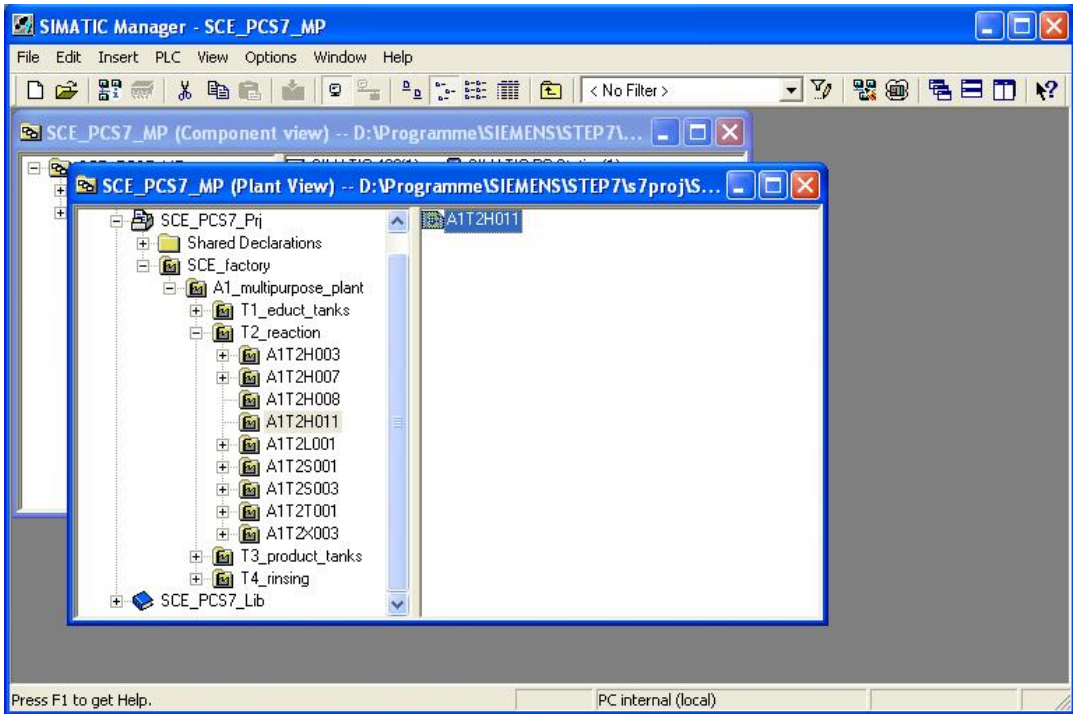
PROGRAMMING

1. To program draining reactor R001 manually, we are setting up a new CFC in the SIMATIC Manager in the plant hierarchy for the EMSR location A1T2H011. (→ SIMATIC Manager → View → Plant View → A1T2H011 → Insert New Object → CFC)



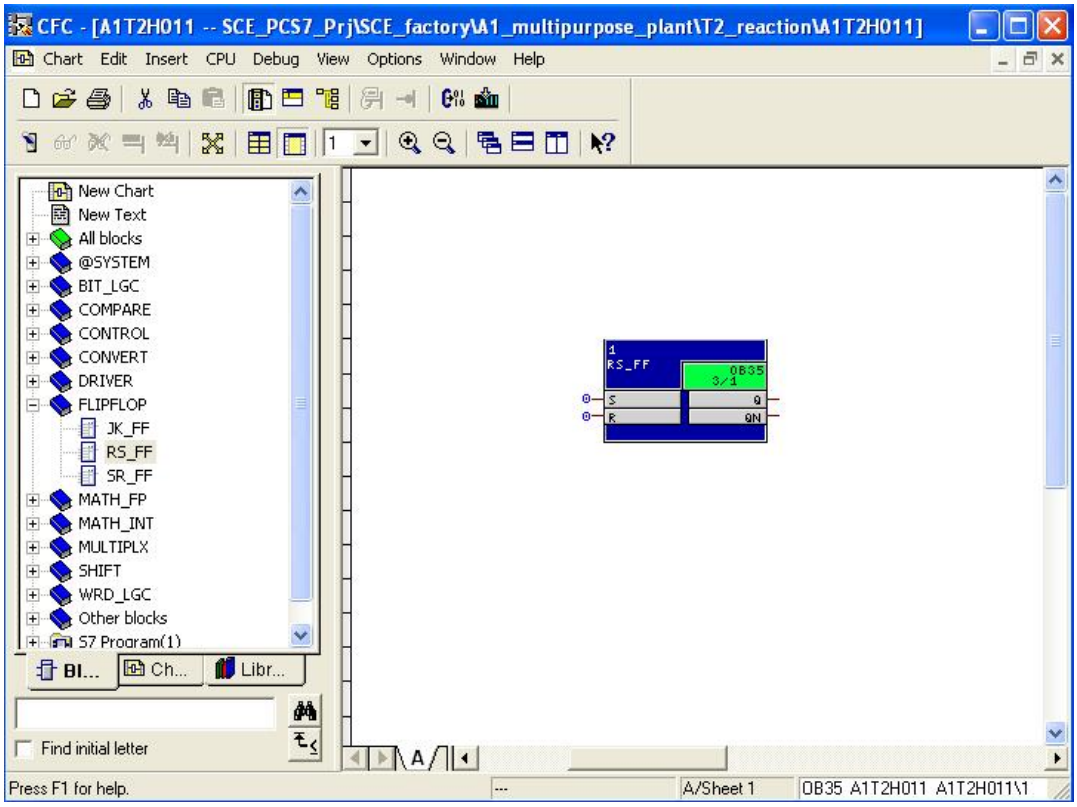
2. The chart is then renamed 'A1T2H011' and opened with a double click.

   (→ A1T2H011)



3. In the CFC editor, we then drag the block ’RS-FF’ from the folder ’FLIPFLOP’ in the ’block’ catalog to the first sheet of our chart. This provides us with a storage element where resetting or switching off is dominant.

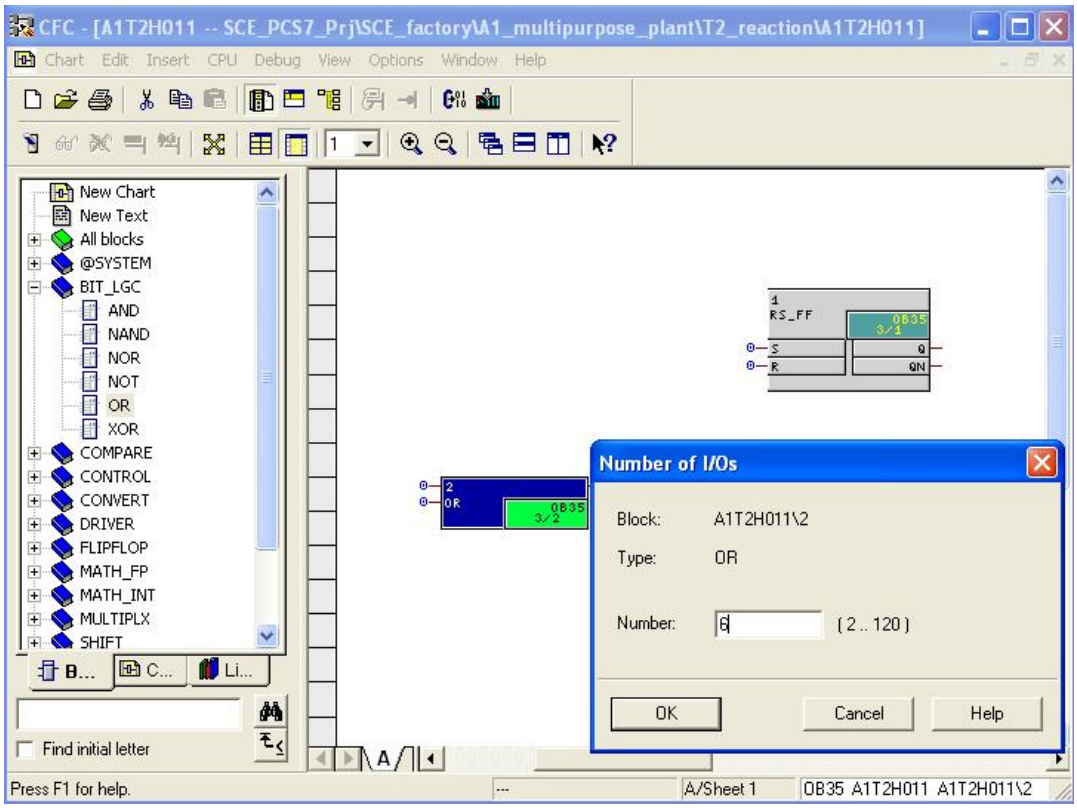
   (→ Blocks → FLIPFLOP → RS_FF)



Note: Additional information about the blocks that are used is provided in the detailed online help. To this end, highlight the corresponding block and press ’F1’ on the keyboard.

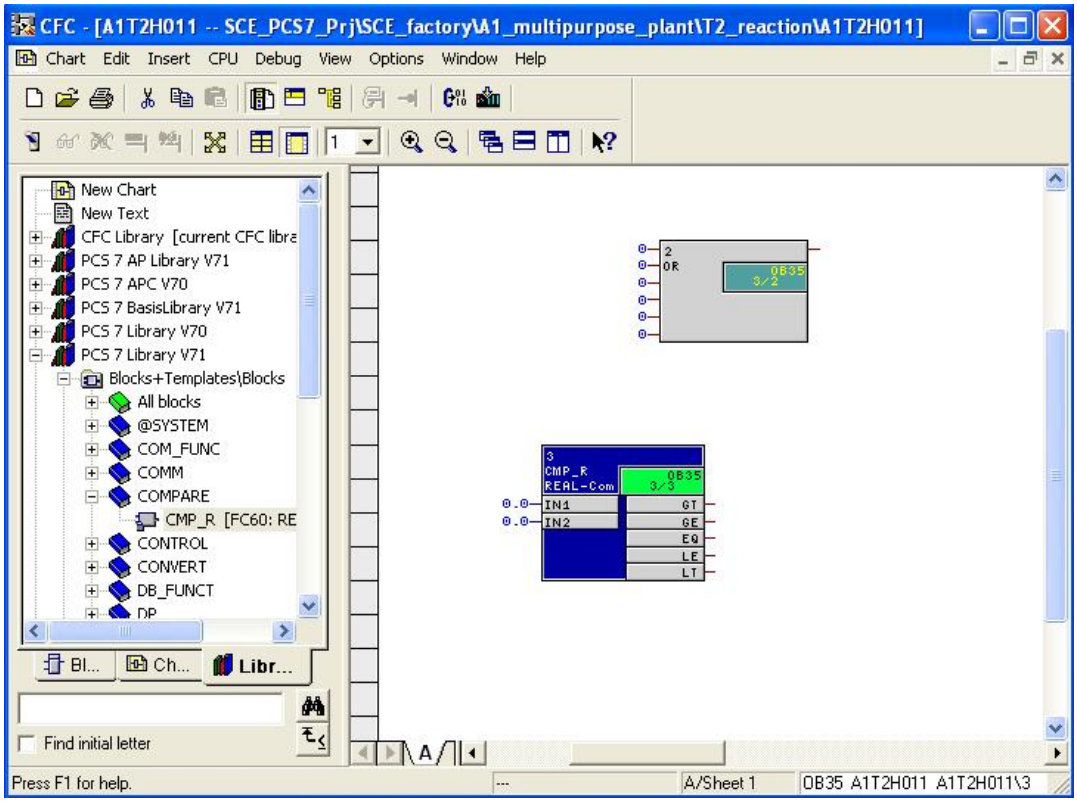
4. Next, we drag the block 'OR' from the folder 'BIT_LGC' into our chart. Then, we right-click on this block and increase the 'Number of I/Os' to 6.

   (→ BIT_LGC → OR → Number of I/Os → 6 → OK)

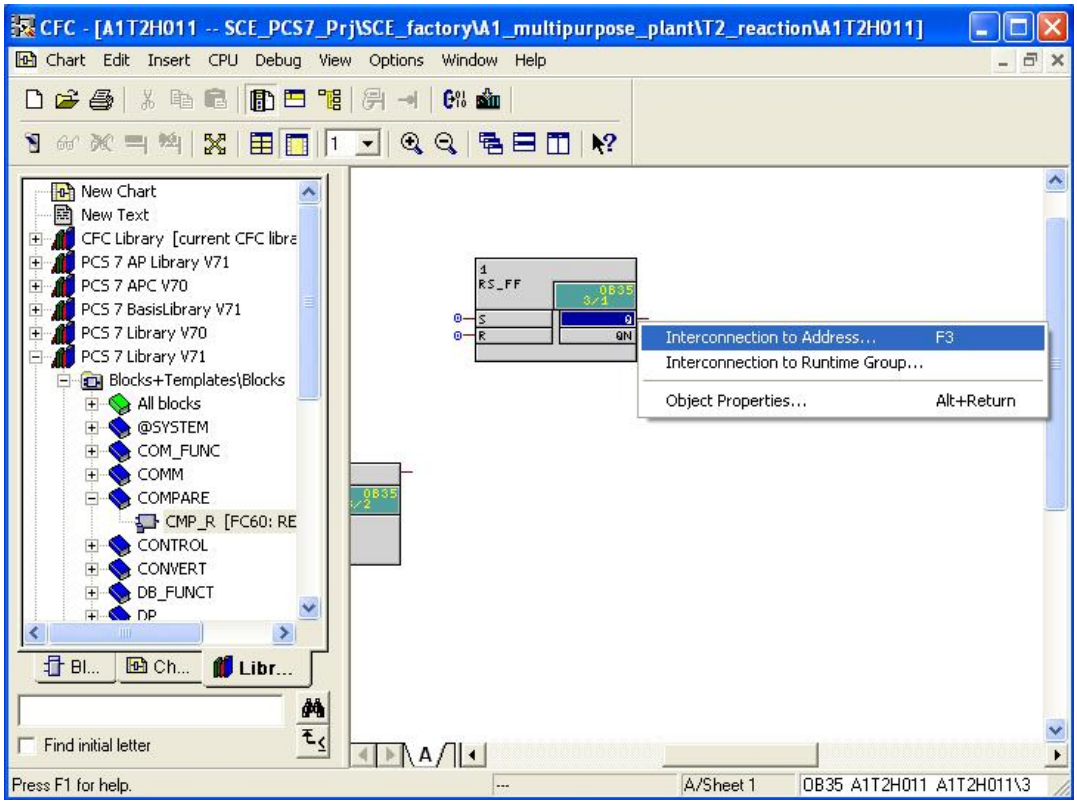


5. Now, we drag the block 'CMP_R' from the folder 'COMPARE' of the PCS7 library V71 in the 'Library catalog' into our chart. We need this block to take into account the level of reactor R001, existing as numerical value, for the lock.

   (→ Libraries → PCS7 Library V71 → Blocks+Templates\Blocks → COMPARE → CMP_R)

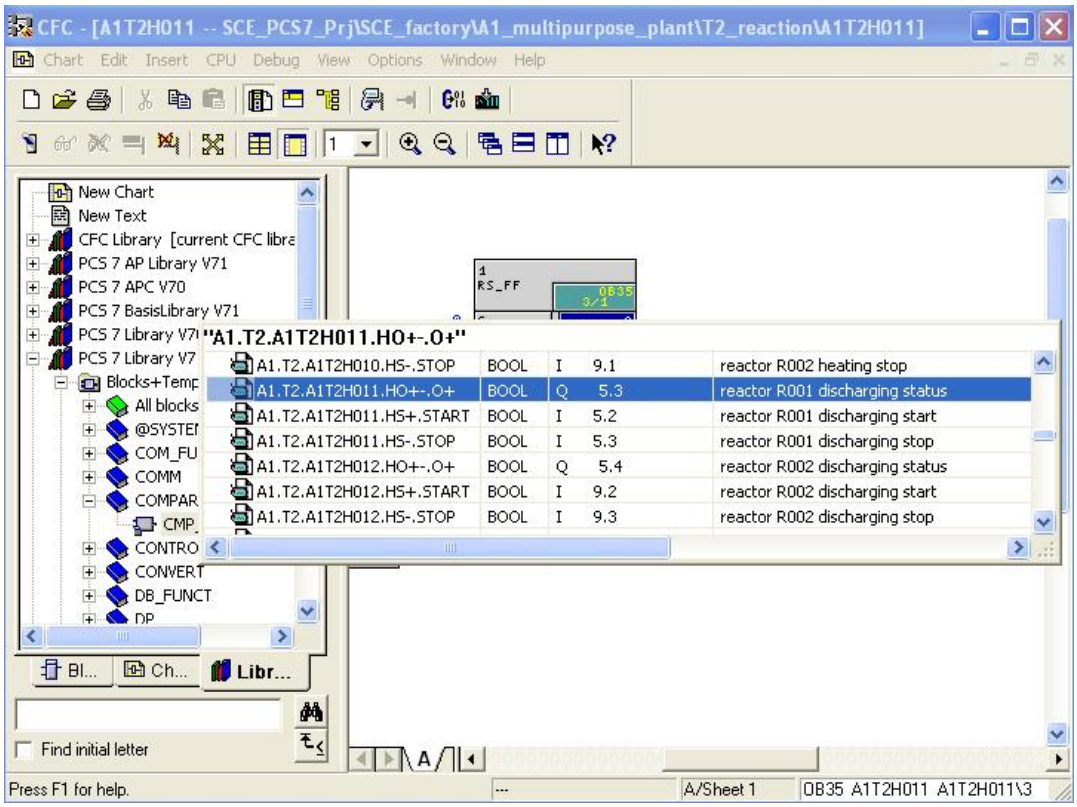


6. To display the status of the operator prompt, we are wiring output 'Q' of block 'RS-FF' with an operand from the symbol table. (→ RS_FF → Q → Interconnection to Address)



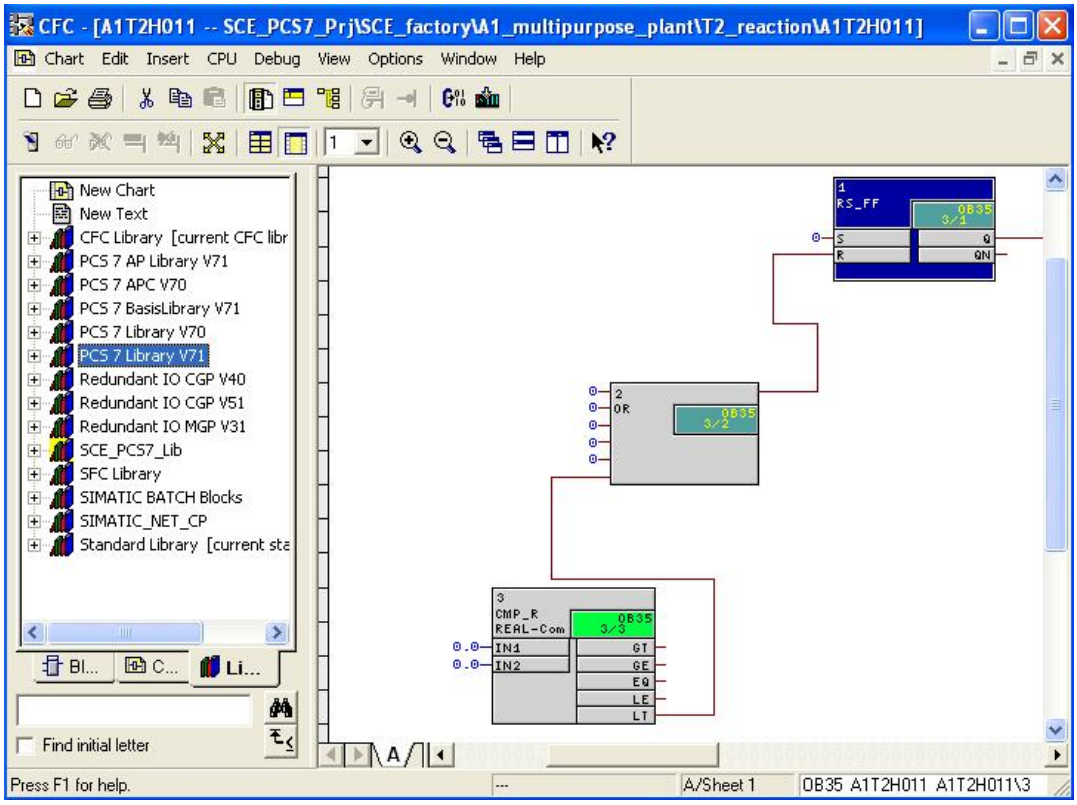
7. From the symbol table that is displayed, select output A5.3 'A1.T2.A1T2H011.H0+-.0+' to display the status of the operator prompt.

   (→ 'A1.T2.A1T2H011.H0+-.0+')



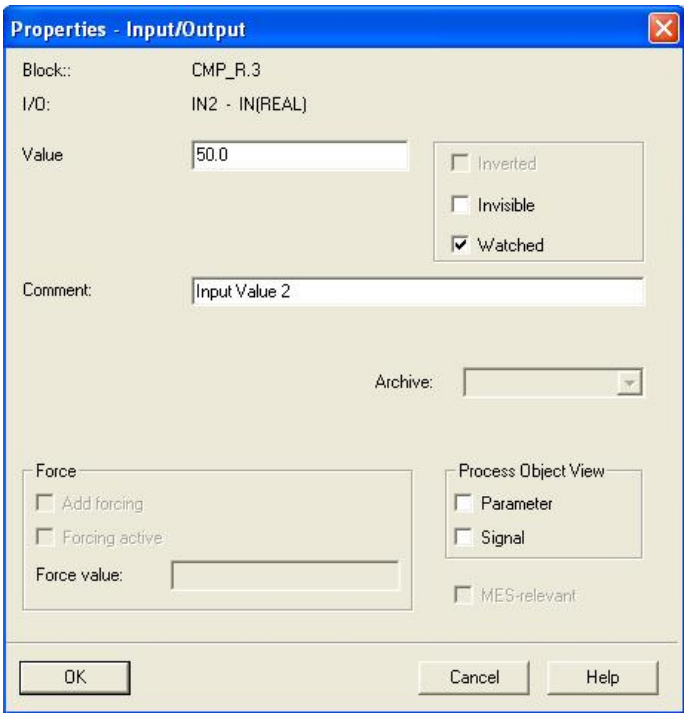
8. Then, we wire the blocks to each other (first wiring). To do this, simply click on the output of the 'OR' block and then on the input 'R' of the 'RS_FF' block. The lines that this wiring shows are drawn automatically and can not be changed in the CFC editor. Then output 'LT' of block 'CMP_R' is connected to an input of the 'OR' block.

   (→ OR → RS_FF → R → CMP_R → LT → OR)

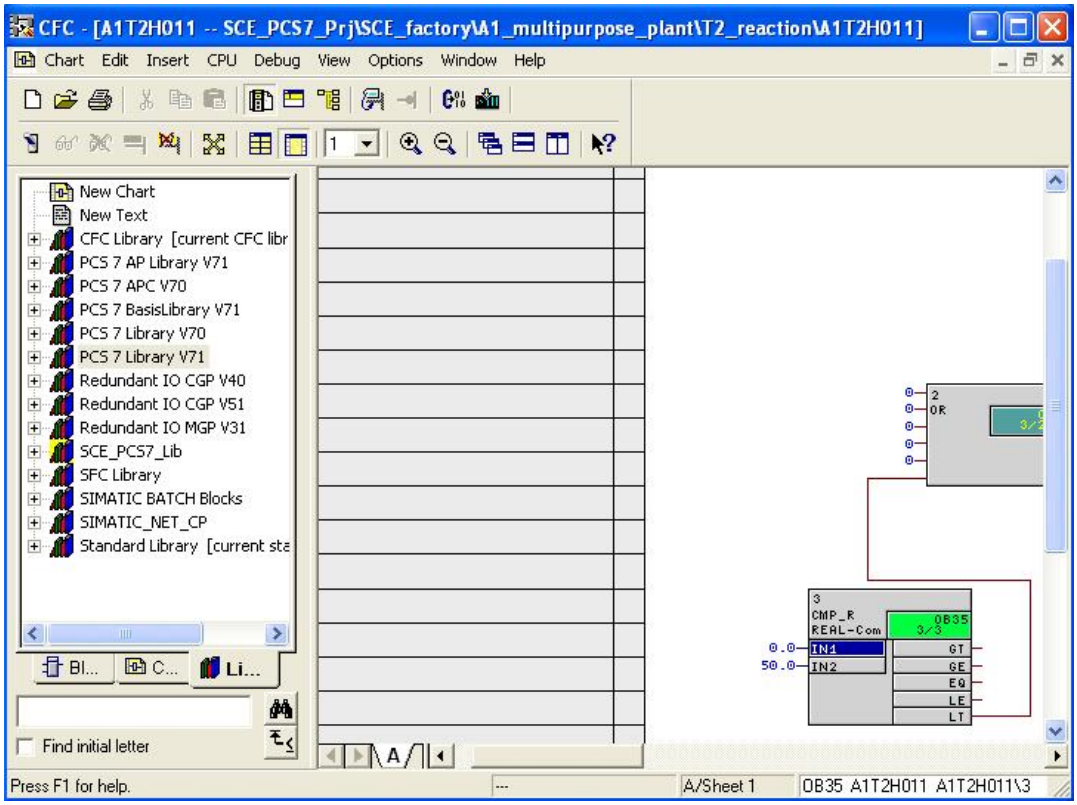


Note: Output 'LT' of block 'CMP_R' is in state 1 if 'IN1' is less than 'IN2'.

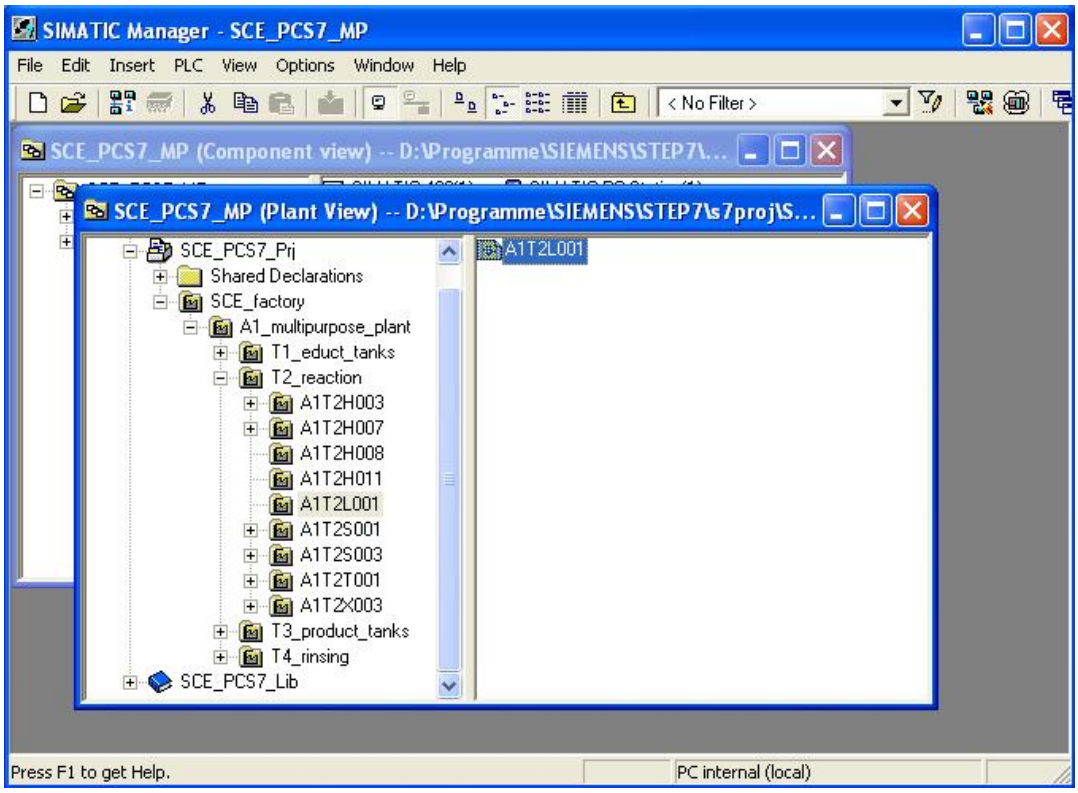
9. Next, the comparison value at input 'IN2' is set by opening here the properties with a double click. As value, enter 50.0 and accept this change with OK. (→ CMP_R → IN2 → 50.0 → OK)



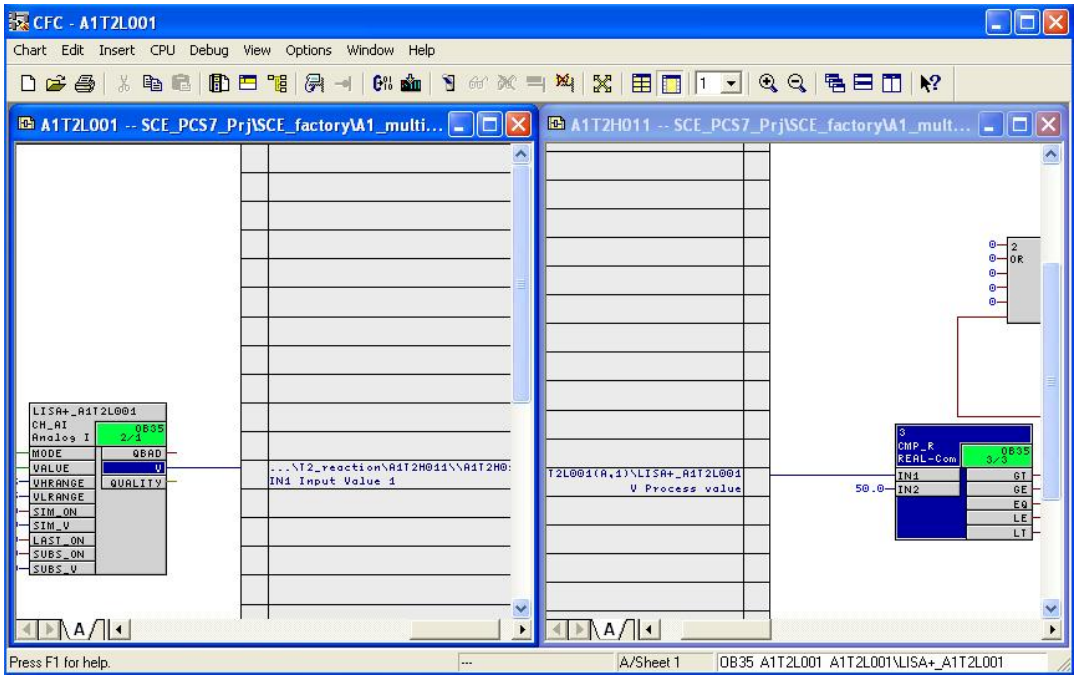
10. Now, we wire -chart-overreaching- input 'IN1' to the measured level of reactor =SCE.A1.T2.R001. To this end, we highlight 'IN1' at block 'CMP_R' (→ CMP_R → IN1)



11. Next, with a double click, we open CFC 'A1T2L001' in the plant view. (→ SIMATIC Manager → Plant View → A1T2L001)

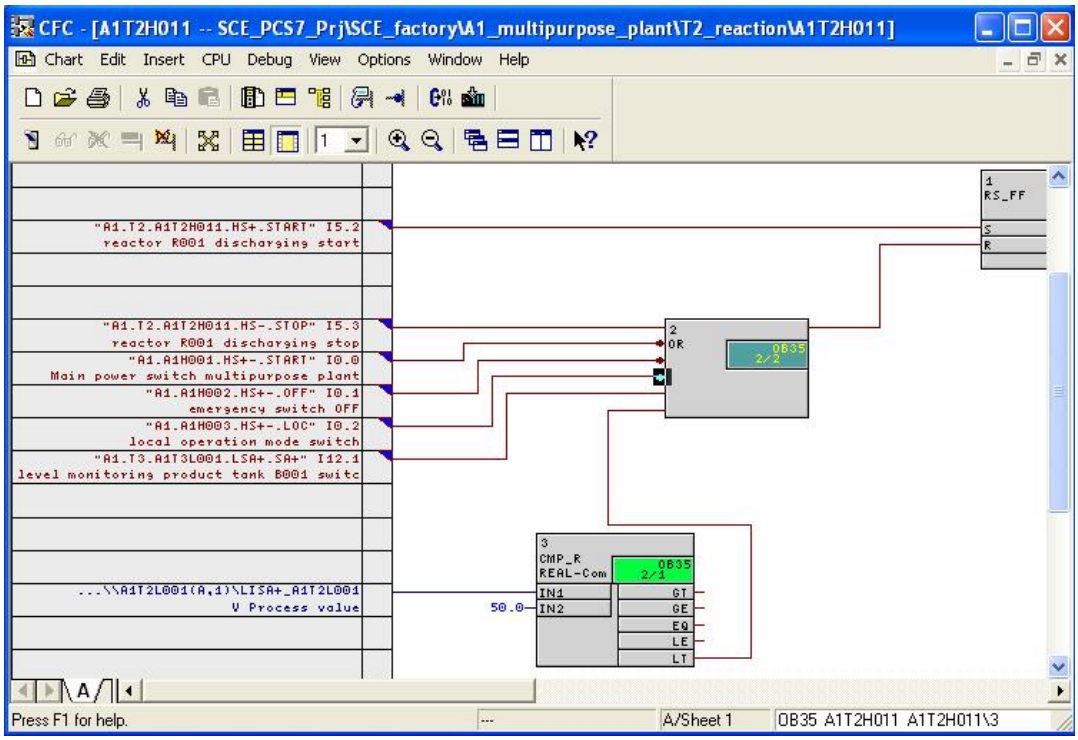


12. In the opened chart 'A1T2L001', we click on output 'V' at block 'CH_AI'. The chart-overreaching connection is set up and displayed for both charts at the margin bar. For chart 'A1T2L001', the wiring destination is displayed on the right. For chart 'A1T2H011', the wiring source is displayed on the left. (→ A1T2L001 → V)



Note: If you want to arrange two charts next to each other in a window as shown here, it is advisable to make the setting via the menu→ Window → Arrange → Vertically.

13. Additional input connections are set up just as already shown, corresponding to the task. The result of all wiring/value entries in chart 'A1T2H011/Sheet1' at this point of the configuration is shown here.



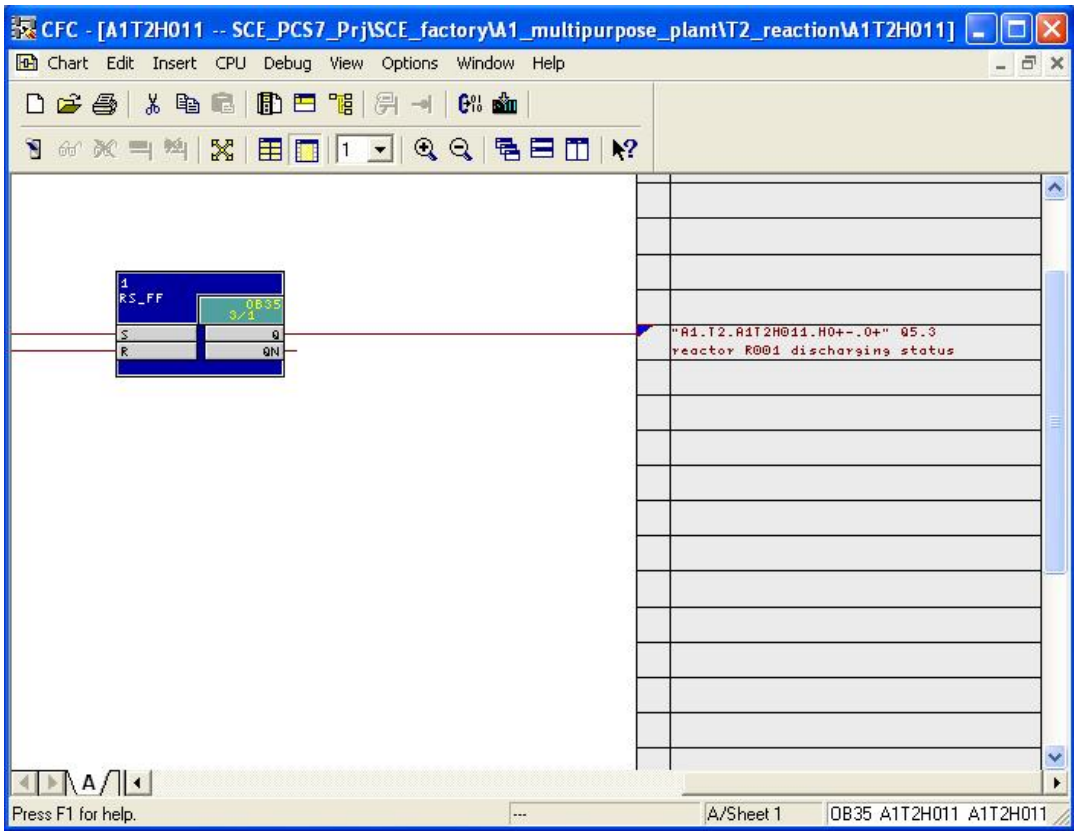


Table 1: Input Wiring in Chart 'A1T2H011/Seet1'

Input	Connected to	Inverted
RS_FF.1.S	'A1.T2.A1T2H011.HS+.START' / I5.2 / Reactor R001 Start discharging	no
OR.2.IN1	'A1.T2.A1T2H011.HS-.STOP' / I5.3 / Reactor R001 Stop discharging	no
OR.2.IN2	'A1.A1H001.HS+-.START' / I0.0 / Switch on multi purpose plant	yes
OR.2.IN3	'A1.A1H002.HS+-.OFF' / I0.1 / Activate Emergency	yes
OR.2.IN4	'A1.A1H003.HS+-.LOC' / I0.2 / Activate field operation	yes
OR.2.IN5	'A1.T3.A1T3L001.LSA+.SA+' / I12.1 / Level monitoring Product Tank B001 Operating Point H	no
CMP_R.3.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.3.IN2	50.0	

Note: 'A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value' means:

- Chart A1T2L001
- Subchart A
- Sheet 1
- Block CH_AI.LISA+_A1T2L001
- Connection V Process value

Table 2: Block Connections in Chart 'A1T2H011/Sheet1'

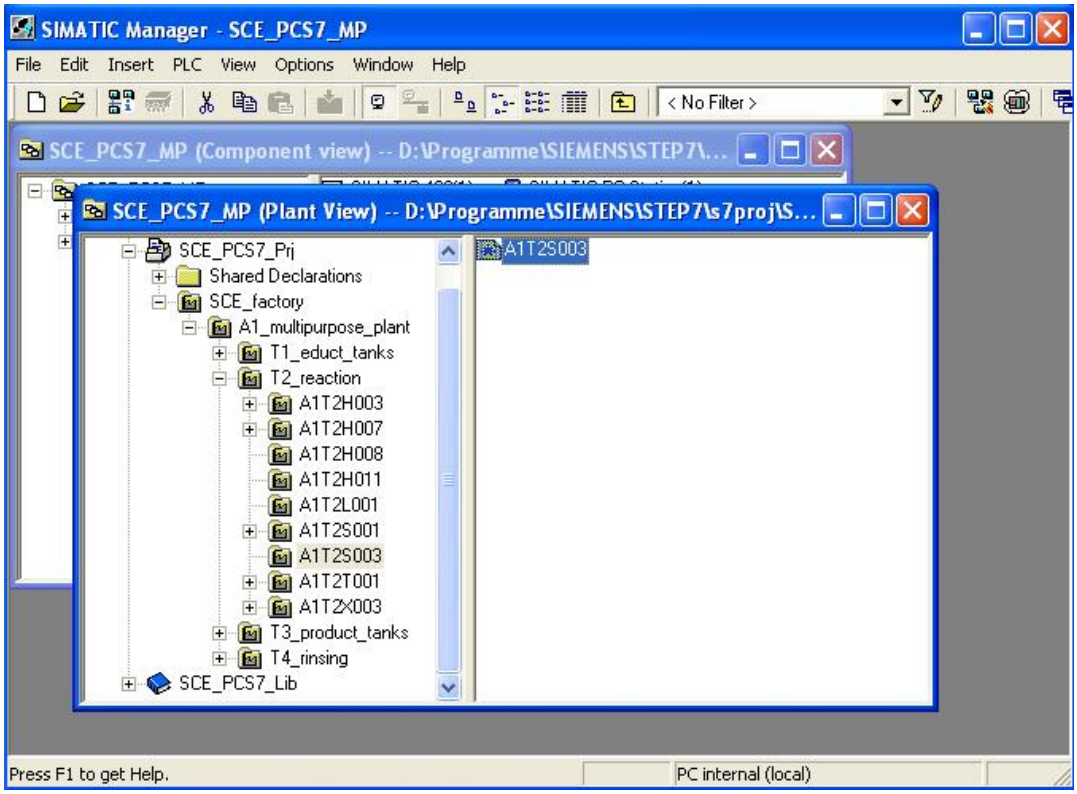
Input	Output	Inverted
RS_FF.1.R	OR.2.OUT	no
OR.2.IN6	CMP_R.3.LT	no

Table 3: Output Connections in Chart 'A1T2H011/Sheet1'

Output	Connection to	Inverted
RS_FF.1.Q	'A1.T2.A1T2H011.HO+-.0+' / O5.3 / Discharge Reactor R001 status value	no

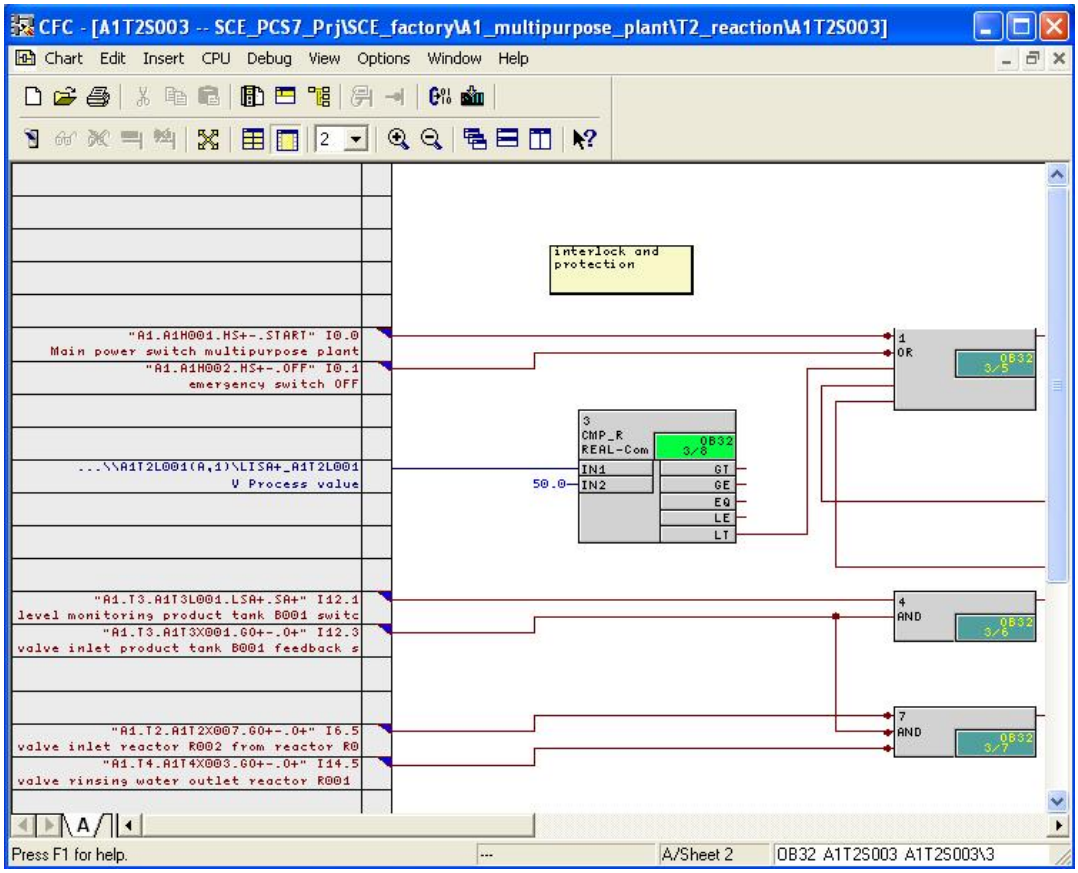
14. Now we generate the lock conditions for the pump 'Discharge Reactor R001'. In the plant view, we open CFC 'A1T2S003' with a double click.

    (→ SIMATIC Manager → Plant view → A1T2S003)



15. Now select Sheet 2, drag the selected blocks from the catalog to this sheet of chart 'A1T2S003' and set up the wiring shown here.

    The result of all wiring/value entries in chart 'A1T2S003/Sheet2' at this point of the configuration is shown here. This wiring relates exactly to the task here also. (→ Sheet 2 → …)



Note: Comment fields for documentation are added in the menu with (→ Insert → New Text).

Table 4 New Blocks in Chart 'A1T2S003/Sheet2'

Block	Catalog/Folder	Number of Connections
OR / Or function	Blocks/BIT_LGC	5
AND / And function	Blocks/BIT_LGC	2
AND / And function	Blocks/BIT_LGC	3
CMP_R / Comparator Floating point numbers	Libraries/PCS7 Library V71 / COMPARE	-

Table 5: Input Wiring in Chart 'A1T2S003/Sheet2'

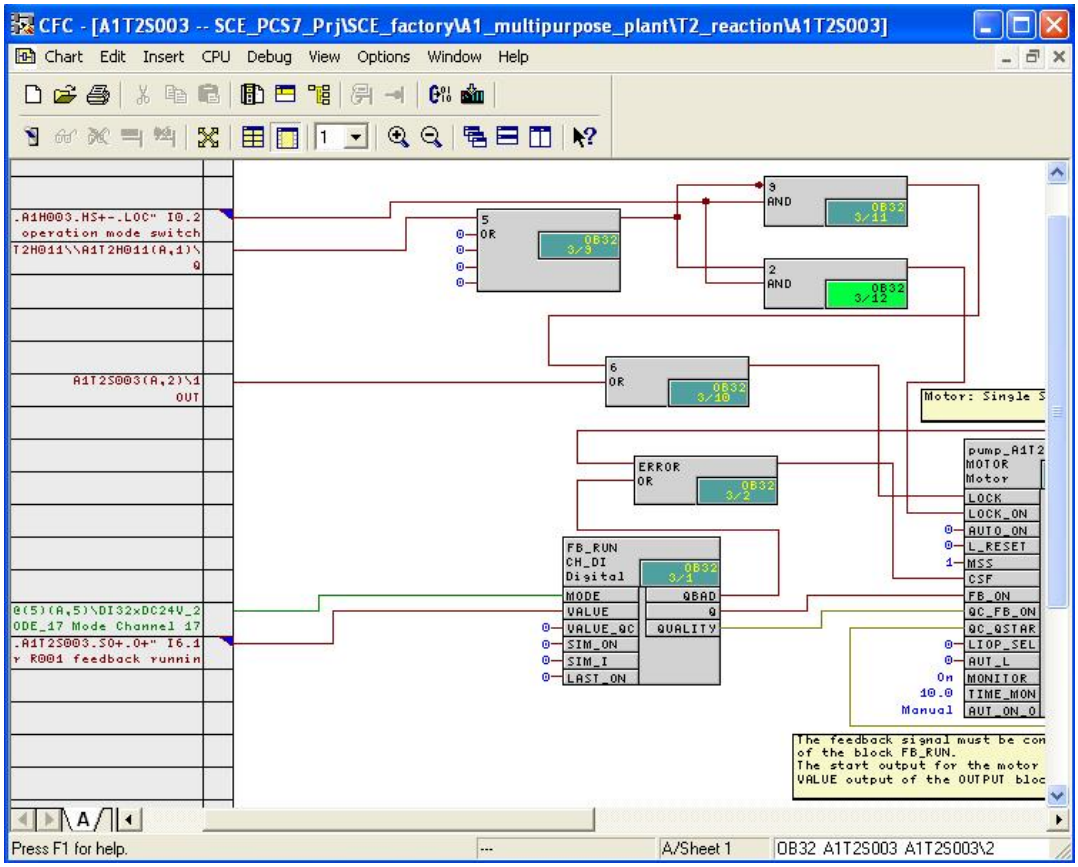
Input	Connected to	Inverted
OR.1.IN1	'A1.A1H001.HS+-.START' / I0.0 / Switch on multi purpose plant	yes
OR.1.IN2	'A1.A1H002.HS+-.OFF' / I0.1 / Activate Emergency	yes
AND.4.IN1	,A1.T3.A1T3L001.LSA+.SA+' / I12.1 / Level monitoring Product tank B001 Operating point H	no
AND.4.IN2	'A1.T3.A1T3X001.GO+-.O+' / I12.3 / Open/Closed valve inlet Product tank B001 Feedback open	no
AND.7.IN1	'A1.T2.A1T2X007.GO+-.O+' / I6.5 / Open/closed valve inlet Reactor R002 from Reactor R001 Feedback open	yes
AND.7.IN2	'A1.T3.A1T3X001.GO+-.O+' / I12.3 / Open/closed valve inlet Product tank B001 Feedback open	yes
AND.7.IN3	'A1.T4.A1T4X003.GO+-.O+' / I14.5 / Open/closed valve rinse water discharge Reactor R001 Feedback open	yes
CMP_R.3.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.3.IN2	,50.0	

Table 6: Block Connections in Chart 'A1T2S003/Sheet2'

Input	Output	Inverted
OR.1.IN3	CMP_R.8.LT	no
OR.1.IN4	AND.4.OUT	no
OR.1.IN5	AND.7.OUT	no

16. In Sheet 1 of Chart 'A1T2S003', we also add blocks and wiring for the lock conditions of pump 'Discharge Reactors R001'. (→ Sheet1 → …)

    The result of all wiring/value entries in chart 'A1T2S003/Sheet1' at this point of the configuration are shown here.



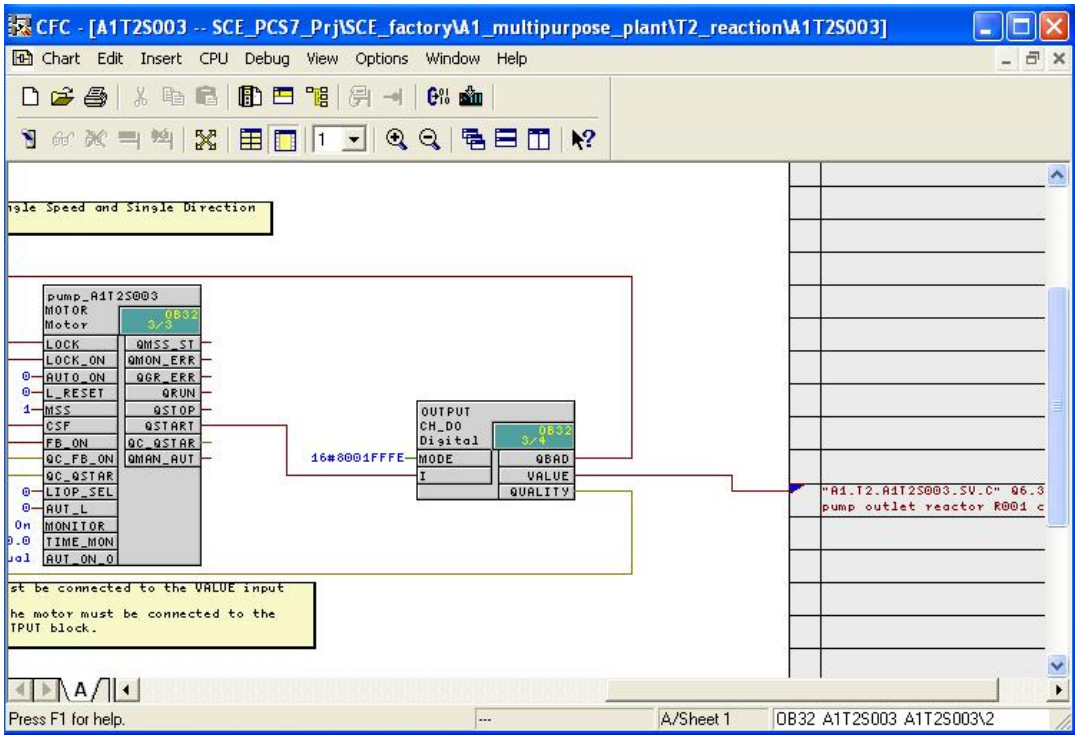


Table 7: New Block in Chart ’A1T2S003/Sheet1’

Block	Catalog/Folder	Number of Connections
AND / And function	Blocks/BIT_LGC	2
AND / And function	Blocks/BIT_LGC	2
OR / Or function	Blocks/BIT_LGC	5
OR / Or function	Blocks/BIT_LGC	2

Table 8: Input Wiring in Chart ’A1T2S003/Sheet1’

Input	Connection to	Inverted
AND.9.IN2	’A1.A1H003.HS+-.LOC’ / I0.2 / Activate field operation	no
AND.2.IN2	’A1.A1H003.HS+-.LOC’ / I0.2 / Activate field operation	no
OR.5.IN1	A1T2H011(A,1) / RS_FF.1.Q	no
OR.6.IN2	A1T2S003(A,2) / OR.1.OUT	no
CH_DI.FB_RUN.VALUE	’A1.T2.A1T2S003.SO+.O+’ / I6.1 / Pump Discharge Reactor R001 Feedback on	no
MOTOR.Pump_A1T2S003. MONITOR.	On	
MOTOR.Pump_A1T2S003. TIME_MON	10.0	

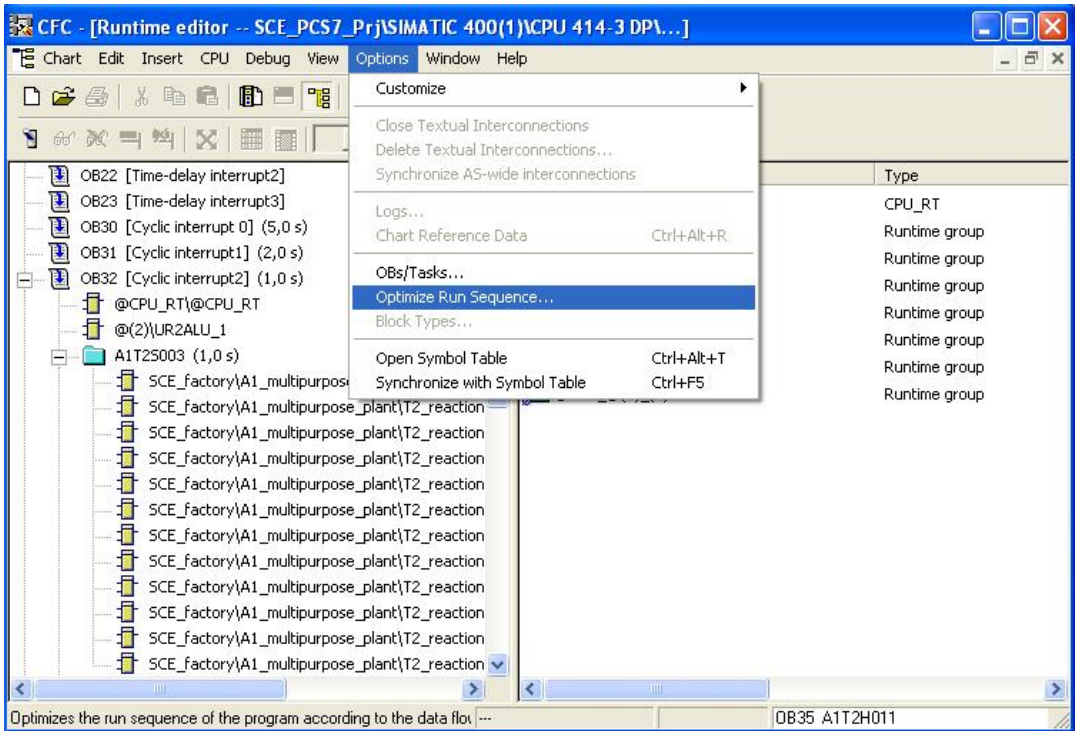
Table 9: Block Connections in Chart ‘A1T2S003/Sheet1’

Input	Output	Inverted
AND.9.IN1	OR.5.OUT	yes
AND.2.IN1	OR.5.OUT	no
OR.6.IN1	AND.3.OUT	no
MOTOR.Pumpe_A1T2S003 .LOCK	OR.6.OUT	no
MOTOR.Pumpe_A1T2S003 .LOCK_ON	AND.2.OUT	no
MOTOR.Pumpe_A1T2S003 .FB_ON	CH_DI.FB_RUN.Q	no
MOTOR.Pumpe_A1T2S003 .CSF	OR.ERROR.OUT	no
OR.ERROR.IN1	CH_DO.OUTPUT.QBAD	no
OR.ERROR.IN2	CH_DI.FB_RUN.QBAD	no
CH_DO.OUTPUT.I	MOTOR.pump_A1T2S003.QSTART	no

Table 10: Output Connections in Chart ’A1T2S003/Sheet1’

Output	Connection to	Inverted
CH_DO.OUTPUT.VALUE	‚A1.T2.A1T2S003.SV.C’ / Q6.3 / Pump Discharge Reactor R001 Actuating signal	no

17. All blocks used in the charts are integrated when inserted in the run sequence of the entire program. By clicking on the symbol ' ' this run sequence can be displayed. To improve the data flow within the overall program, it is recommended to optimize the sequence after the program is generated. This is done in the menu under 'Options', 'Optimize Run Sequence'. (→ → Options → Optimize Run Sequence)



EXERCISES

We are going to apply what we learned in the theoretical part and the step by step instructions to the exercises. We are going to use and expand the already existing multi-project provided in the step by step instructions (PCS7_SCE_0105_R1009.zip).

The protective steps implemented for the pump in the step by step instructions are to be applied also to the valve. Its controller was implemented in the previous exercise.

TASKS

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

1. With respect to functional safety, analyze which functions are needed for the valve. Create a switching table using the signals in Table 11 and combine these signals in a way that the valve is taken to a safe state if needed.
2. Open CFC A1T3X001 and switch to a blank sheet. Implement a combinational circuit that implements the switching table in Task 1, and connect the necessary inputs. Connect the output of the combinational circuit with the 'V_LOCK' input of the valve block. The valve will now receive information as to when it is to enter the rest position. We still have to set whether the safe state is the open or the closed valve. This is done by means of the input 'SS_POS' that has to have the value '0' for the valve to be closed in the rest position.
3. There are two more inputs that can influence the valve's position: 'VL_CLOSE' and 'VL_OPEN'. These inputs have a lower priority than 'V_LOCK'. How these two inputs are set is decided as follows: 1) by the connection 'A1T2H011(A,1) / RS_FF.1.Q' that is generated in CFC A1T2H011 and indicates whether Reactor R001 is discharging at the moment, und 2) by the signal 'A1.A1H003.HS+-.LOC' that indicates whether field operation is activated. If it is a field operation and Reactor R001 is discharging, the valve should be open (set 'VL_OPEN' of the valve block to TRUE). If it is a field operation without Reactor R001 being discharged, the valve should be closed (set 'VL_CLOSE' of the valve block to TRUE). Implement a combinational circuit that meets these conditions.

Table 11: Signals for locking the valve

Signal	Comment
A1.A1H001.HS+-.START	Main power switch multipurpose plant
A1.A1H002.HS+-.OFF	Emergency switch OFF