Loop Control and More Control Functions

Objective

In this chapter, the students get to know the essential components and the demands on a block for continuously controlling process variables; they will also learn to set up and configure a temperature control by using the blocks CTRL_PID and PULSEGEN.

Theory in Short

In the process industry, certain process variables have to be kept to a certain value (***disturbance variable***), or process variables have to be set stability-oriented to specified setpoints (***response to setpoint changes***). To this end, control loops are used as shown in Figure 1.

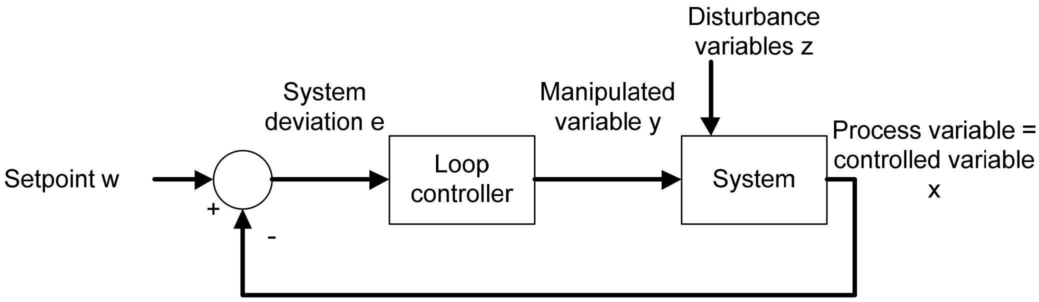


Figure 1: Control Loop

For our plant, the reactor temperature has to be set to a certain value for the reaction control to be in accordance with the specification. The disturbance variables are the ambient temperature and the materials that are used having different temperatures. For the temperature to be regulated, we first have to ascertain it by measurement. This measured value that corresponds to the ***actual value*** of the process variable is then compared with the desired value (***setpoint***). The difference between theactual value and the setpoint is called ***(system) deviation***.

If the system deviation is known, counter-measures can be derived. Regarding temperature regulation, the heater is switched on if the measured actual value is lower than the specified setpoint. For the process to handle this autonomously, a controller is needed. A controller that calculates the manipulated variable based only on the current deviation is called a proportional controller (P-controller for short).

In practice, controllers prevailed that can, with the aid of a few parameters, be used for a wide range of processes, the so-called ***PID controllers***.

The ***PCS7 Standard Library V71*** contains proven blocks that implement this function. Below, the block CTRL_PID is used.

THEORY

INTRODUCTION

The P-controller mentioned above is the simplest one. It works according to the principle: the larger the current deviation, that larger the manipulated value. That is, its behavior is derived directly from the current system deviation –which makes it fast and dynamically relatively favorable. However, certain disturbances are not completely compensated; i.e., a system deviation always remains.

Not every system tolerates a sustained system deviation. For that reason, additional steps have to be taken. One option is adding an integral component, which changes the P-controller into a PI controller. The effect of the integral component is this: a sustained system deviation is totaled. Thus, the manipulated value increases although the system deviation remains the same.

If abrupt disturbances occur in a system, they can be quickly counteracted with an additional differentiating component. The D-component calculates the manipulated variable using the time deviation of the system deviation. However, this behavior causes stochastic disturbances (noise). Here, an effective compromise has to be found. .

A combination of P, I and D components is referred to as a PID controller. In the process industry, 95% of applications are implemented with these controllers since the PID controller is set with only three parameters (gain, TN (reset time, integral action time) and TV (rate time, derivative time). These few parameters already allow for a good adaptation to numerous different dynamic processes.

However, setting the parameter presupposes knowledge of the system to be controlled. The knowledge about the system can be gained from experience, it can be ascertained experimentally, or it can be calculated by modeling the process. For a wide range of processes that are not dominated by delays and respond in a similar manner to positive as well as negative changes of manipulated variable interventions, it was possible to come up with different rules for controller adjustment suitable in practice. Examples are the rules for controller adjustment according to Chien, Hrones and Reswick [1], the method by Ziegler and Nichols [2] as well as the T-Sum Rule [3].

The process control system ***PCS7*** supports setting the parameters using a ***PID tuner***.

For the controller block CTRL_PID, the three parameters are GAIN, TN for the integral component, and TV for the differential component. In addition, there is the parameter TM_LAG that sets the delay time for the D-component. The time is specified in seconds. The input variable of the controller is the system deviation ER, and the output variable is the manipulated value LMN that is calculated according to the following formula:

$$LMN = GAIN \cdot \left(1 + \frac{1}{TN \cdot s} + \frac{TV \cdot s}{1 + TM\_LAG \cdot s} \right) \cdot ER\,.$$

INDUSTRIAL SUITABILITY OF CONTROLLERS

For a controller to work day by day and meeting industrial demands, additional functions have to be implemented. Primarily, these are:

- Bumpless changeover
- Anti reset windup (ARW)
- Support of different controlled system structures

The bumpless changeover is to prevent an abrupt change of the manipulated variable when switching between the manual and automatic mode, between internal and external setpoint selection, or when parameters are changed. A bumpless changeover between the manual and the automatic mode is required, for example, when a process in process engineering runs semi-automatically; i.e., when we perform the startup manually and then switch to the automatic mode for regular operation. In the manual mode, the operator specifies the manipulated variable directly, while in the automatic mode, the control algorithm calculates the manipulated variable.

The function anti-reset windup (ARW) is to prevent that the integral component (reset) of the manipulated variable continues to increase (wind up), since it is not possible to adjust a system deviation because of the manipulated variable restriction, for example.

Supporting the different control structures allows for optimizing the control system without having to replace the controller. In the chapter 'Expanded Control Structures', some of these control structures are explained in greater detail. With CTRL_PID from the ***PCS7 Standard Library V71***, the following control structures can be implemented:

- Fixed setpoint control
- Cascade control (single/multiple cascades)
- Ratio control
- Synchro control
- Blending control
- Feed forward injection of disturbance variable

With this block, the majority of control structures common in the process industry can be implemented. For connections going beyond thise -such as split range control, Smith-Prädikator control and override control- the ***PCS7 Advanced Process Library V71*** provides the continuous PID controller PIDConL.

EXPANDED CONTROL STRUCTURES

For some applications, single loop control loops are not sufficient so that expanded control structures have to be used to attain the desired objective.

If for a process variable, the response to setpoint changes and to the disturbance variable can not be optimized satisfactorily, a feed forward control /auxiliary variable control or a cascade control can be used.

If the disturbance variable can be measured and its point of attack is known, a compensation of the disturbance variable can be applied to the controller input or output. With the ***feed forward control***, the disturbance variable can be compensated completely so the controller can be set to the optimum command behavior (response to setpoint changes).

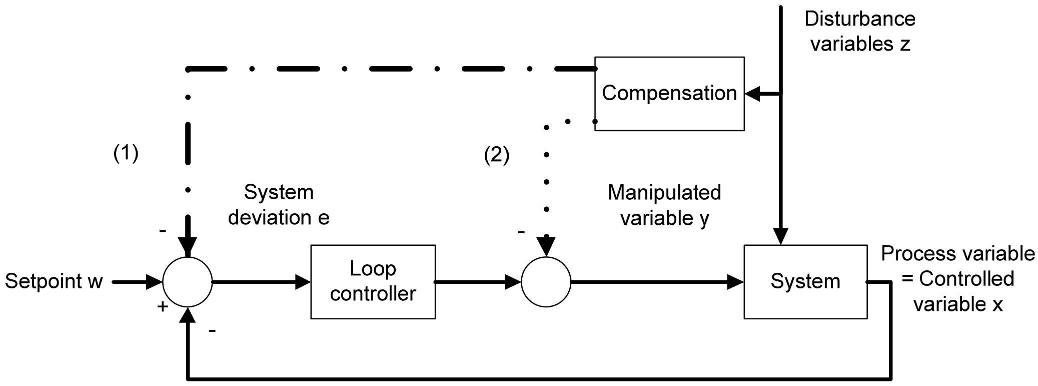


Figure 2: Feed forward control at the input (1) or the output (2) of the loop controller

If it is not possible to set the disturbance variable but another variable in the system, this auxiliary variable is connected with a controller to the controller input. Applying the ***auxiliary variable*** reduces the influence of the disturbance variable, but does not compensate it completely.

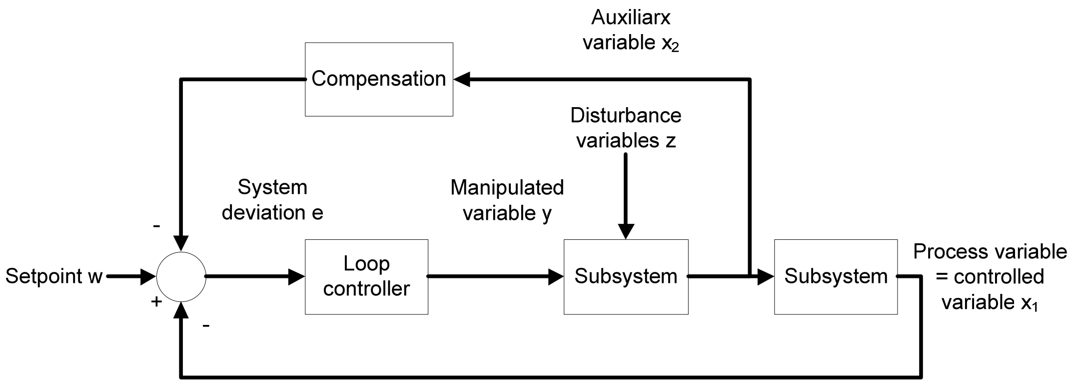


Figure 3: Compensation by auxiliary variables

If the injection is made at the controller input, compensation and controller are not independent of each other. That means: if the controller parameters are adjusted, the compensation has to be adjusted also.

If applying the disturbance and auxiliary variable is not sufficient, or if it is not possible to determine the point of attack of the disturbance variables with sufficient accuracy, or model the subsystems with sufficient accuracy, a two or multi-loop ***cascade control*** is used.

When designing the cascade control, it is assumed that the underlaid control loops (Controller 2 in Figure 4 -a socalled slave controller) respond faster than the overlaid control loops (Controller 1 in Figure 4 -a socalled master controller). Thus, the control is always optimized from the inside toward the outside.

The cascade control reduces the influence of the disturbance variable and makes the control faster. To use the cascade control, correspondingly measurable variable have to exist.

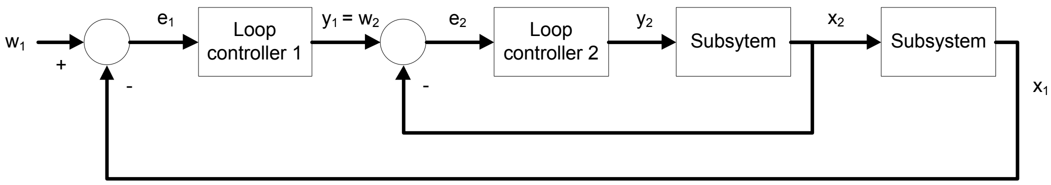


Figure 4: Cascade control with two loops

Ratio control is used if the process variable is determined in dependence of another variable; for example, the ratio control of two liquid streams that are to be blended; i.e., controlling the composition of the blend; or the ratio control of combustion gas and fresh air in a gas burner for optimum combustion. The setpoint of the process variable w_2 is calculated from the ratio r_w and the process variable x_1.

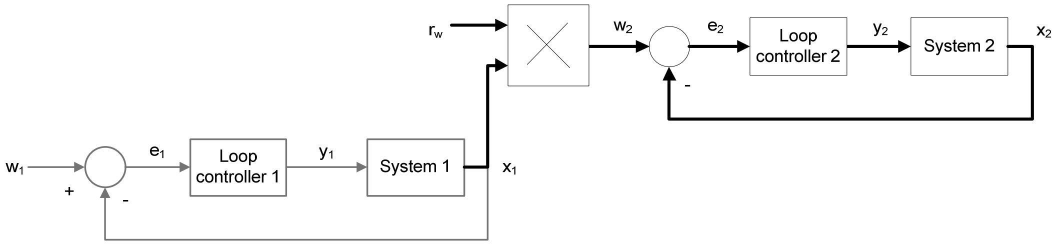


Figure 5: Ratio control

Connecting to Processes

The continuous output signal of the controller is not always read out at the process. This is particularly inadvisable for great forces or large streams. For that reason, binary connections are used. To this end, the analog signal is changed into a binary signal by means of ***pulse width modulation***. The ***PCS7*** provides the elementary block PULSEGEN [4] for this.

By modulating the pulse width, the function PULSEGEN transforms the input variable INV (= LMN manipulated value of the PID controller) into a pulse sequence of a constant period. It corresponds to the cycle time with which the input variable is updated, and has to be parameterized in PER_TM.

The pulse duration for each period is proportional to the input variable. Here, the cycle parameterized with PER_TM is not identical with the processing (machining) cycle of the function block PULSEGEN. As Figure 6 shows, a cycle PER_TM ② consists of several processing cycles ① of the function block PULSEGEN. The number of PULSEGEN calls for each PER_TM cycle represents here the measure for the accuracy of the pulse width modulation.

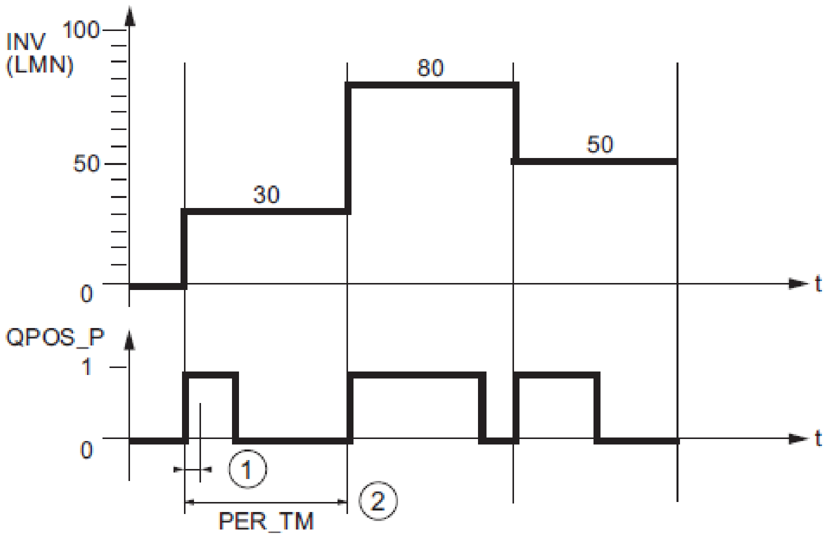


Figure 6: Time curve of input INV to output QPOS_P with PULSEGEN [4]

An input variable of 30% at 10 PULSEGEN calls for each PER_TM means the following:

– 1 at output QPOS for the initial three calls of the PULSEGEN (30% of 10 calls)

– 0 at output QPOS for seven additional calls of the PULSEGEN (70% of 10 calls)

At the start of each period, the pulse duration is recalculated. Through a sampling ratio of 1:10 (CTRL_PID calls to PULSEGEN calls), the manipulated value accuracy in this example is limited to 10%. Specified input values INV can be mapped only in the raster of 10% to a pulse length at output QPOS. Correspondingly, accuracy increases with the number of PULSEGEN calls for each CTRL PID call. If PULSEGEN is called 100 times and CTRL PID only once, the resolution is 1% of the manipulated variable range.

Note: You have to program the down scaling of the call frequency yourself.

LITERATURE

[1] Chien, Kun Li; Hrones, J. A.; Reswick, J. B. (1952): On the Automatic Control of Generalized Passive Systems. In: Transactions of the American Society of Mechanical Engineers, Vol. 74, Cambridge (Mass.), S. 175-185.

[2] Ziegler, J. G. and Nichols, N. B. (1942): Optimum settings for automatic controllers. In: Trans. ASME, 64, S. 759-768.

[3] Kuhn, U.: Eine praxisnahe Einstellregel für PID-Regler: Die T-Summen-Regel. Automatisierungstechnische Praxis, Nr. 5, 1995, S. 10-16.

(Practice-Oriented Controller Adjustment Rules for PID Controllers: The T-Sum Rule.Automation Engineering Practice, No. 5, 1995, Pages 10 to 16).

[4] SIEMENS (2009): Process Control System PCS 7: CFC Elementary Blocks.

Step by Step Instructions

Task

Corresponding to the information provided in the chapter 'Process Description', we are supplementing the CF charts in Chapter Functional Safety.

The following CFCs are created here:

– A1T2H003: Manual operation Educt B003 to Reactor R001

– A1T2H007: Manual operation Reactor R001 Stirring

– A1T2H008: Manual operation Reactor R001 Heating

– A1T1S003: Pump Discharge Educt Tank B003

– A1T1X006: Open/Close Valve Inlet Reactor R001 from Educt Tank B003

– A1T2X003: Open/Close inlet Reactor R001 from Educt Tank B003

– A1T2S001: Stirrer Reactor R001

– A1T2T001: Temperature Reactor R001

The following lock conditions have to be noted:

– Actuators are to be switched only if the main switch of the plant is switched on and the emergency OFF switch is unlocked.

– No container must overflow; i.e., there is either an encoder that signals the maximum level, or the maximum level (here: 1000ml) is known numerically and is evaluated by means of the measured level.

– No pump must draw in air; i.e., there is either an encoder that signals the minimum level, or the minimum level (here: 50ml) is known numerically and is evaluated by means of the measured level.

– A pump must not attempt to draw in liquid from a closed valve, or press liquid against a closed valve.

– The temperature in both reactors must not exceed 60°C.

– The heaters in both reactors must be started only if they are covered with liquid (here: a minimum of 200ml in the reactor).

– The stirrers in both reactors should be started only if they come into contact with a liquid (here: a minimum of 300ml in the reactor).

Objective

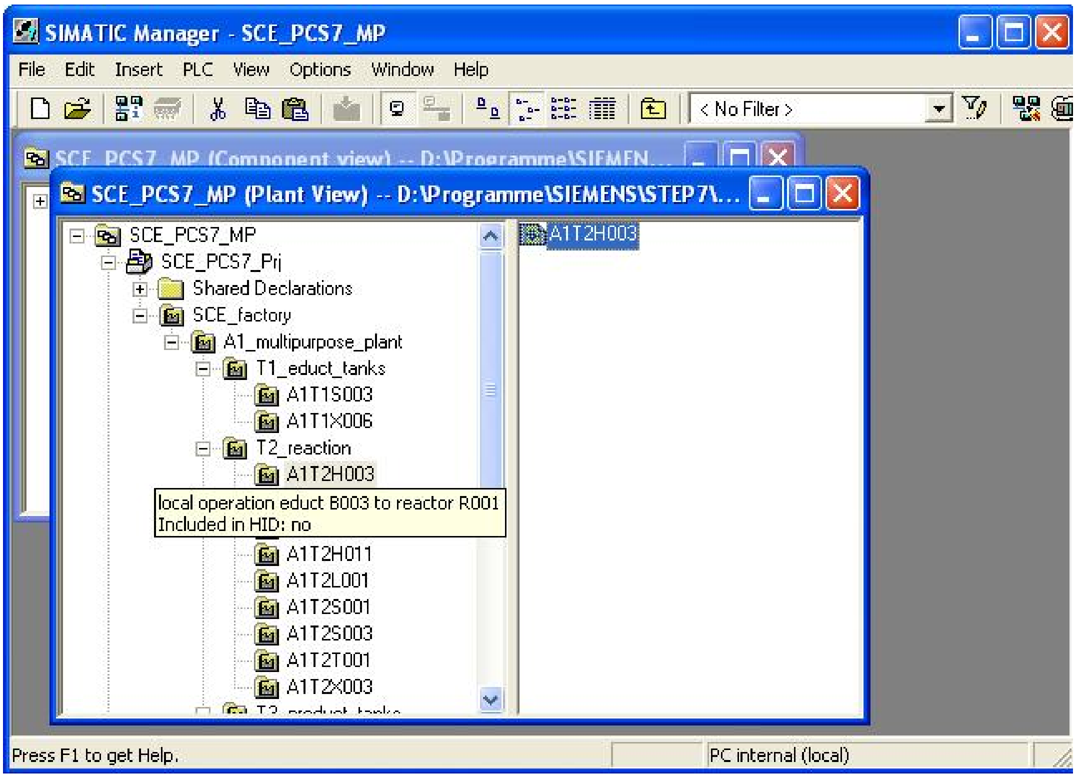
In this chapter, the student gains:

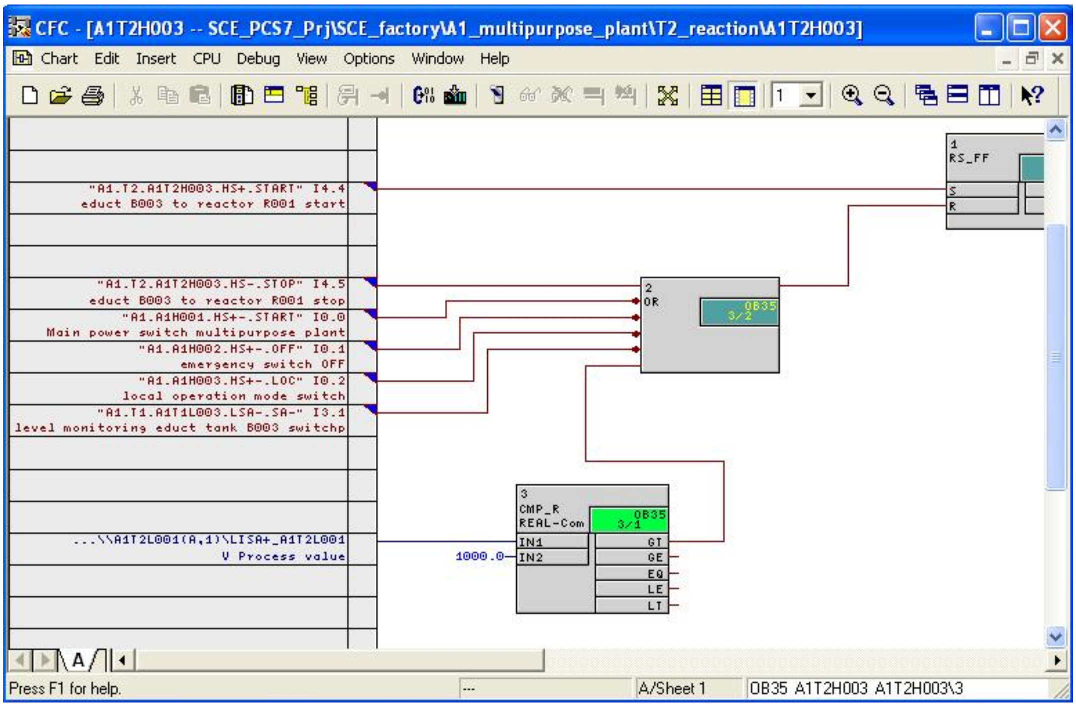
– Additional experience in programming with CFCs

– Knowledge of programming a continuous controller with pulse output and locks.

PROGRAMMING

1. To program the manual operation for filling Reactor R001from Educt Tank B003, we are setting up a new CFC in the ***SIMATIC Manager*** in the plant view for the EMSR location A1T2H003. Follow the screen shots below and the entries:





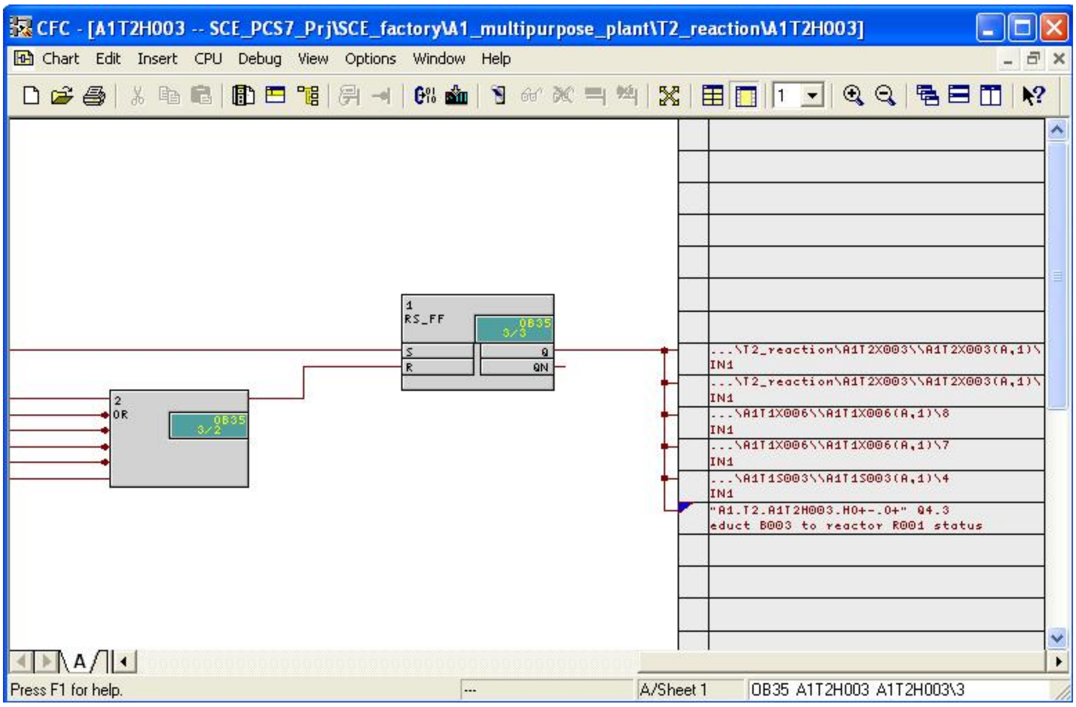


Table 1: New Blocks in Chart 'A1T2H003/Sheet1'

Block	**Catalog/Folder**	**Number of Connections**
RS_FF / RS FlipFlop	Blocks/FLIPFLOP	
OR / Or function	Blocks/BIT_LGC	6
CMP_R / Comparator for REAL values	Libraries/PCS 7 Library V71/ Blocks+Templates\Blocks/COMPARE	

Table 2: Input Connections in Chart 'A1T2H003/Sheet1'

Input	**Connected to**	**Inverted**
RS_FF.1.S	'A1.T2.A1T2H003.HS+.START' / I4.4 / Start Educt B003 to Reactor R001	no
OR.2.IN1	'A1.T2.A1T2H003.HS-.STOP' / I4.5 / Stop Educt B003 to Reactor R001	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 OFF	yes
OR.2.IN4	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	yes
OR.2.IN5	'A1.T1.A1T1L003.LSA-.SA-' / I3.1 / Level monitoring Educt Tank B003 Operating Point L	yes
CMP_R.3.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.3.IN2	1000.0	

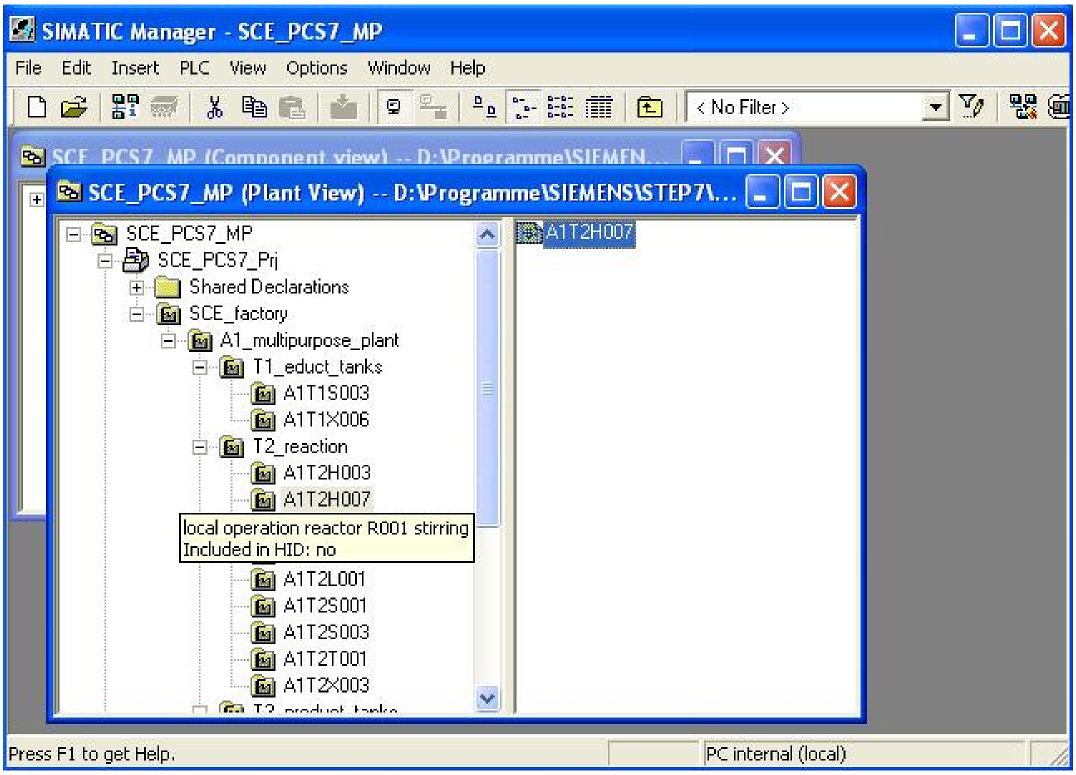
Table 3: Block Connections in Chart 'A1T2H003/Sheet1'

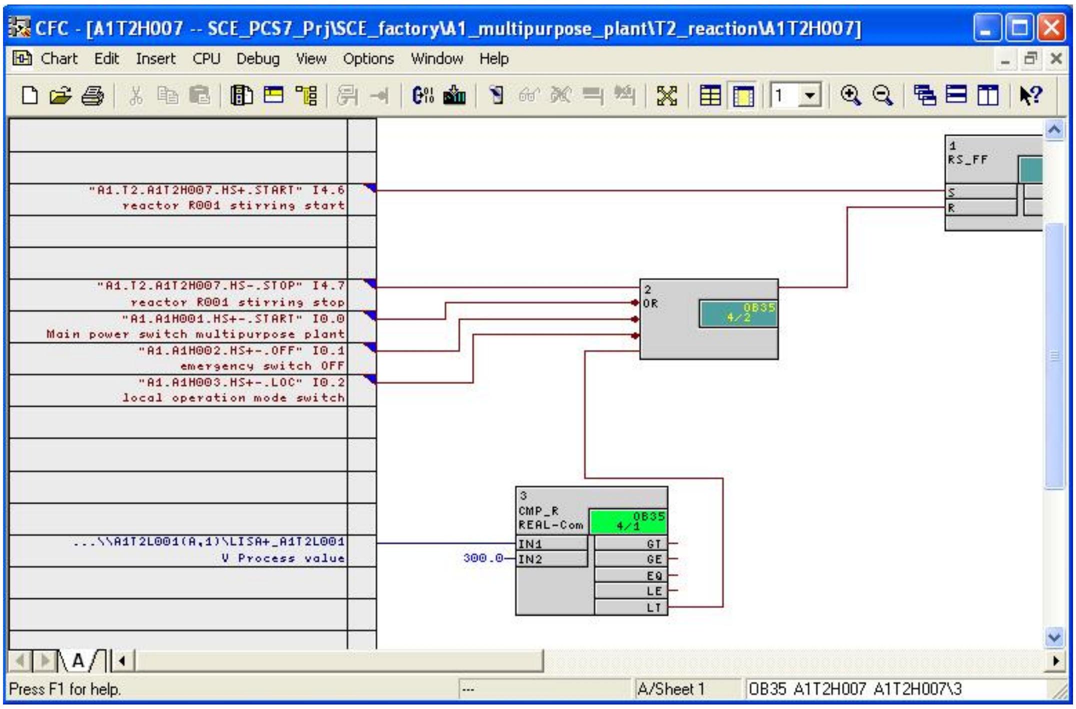
Input	Output	Inverted
RS_FF.1.R	OR.2.OUT	no
OR.2.IN6	CMP_R.3.GT	no

Table 4: Output Connections in Chart 'A1T2H0003/Sheet1'

Output	Connected to	Inverted
RS_FF.1.Q	'A1.T2.A1T2H003.HO+-.O-' / Q4.3 / Educt B003 to Reactor R001 Status value	no
RS_FF.1.Q	The other connections that are visible here are generated later from the charts A1T2X003, A1T1X006 and A1T1S003.	

2. To program the manual operation for the stirrer in Reactor R001, we are setting up a new CFC in the ***SIMATIC Manager*** in the plant view for the EMSR location A1T2H007. Follow the screen shots below and the entries:




CFC - [A1T2H007 -- SCE_PCS7_Prj\SCE_factory\A1_multipurpose_plant\T2_reaction\A1T2H007]
Chart Edit Insert CPU Debug View Options Window Help
"A1.T2.A1T2H007.HS+.START" I4.6
reactor R001 stirring start
"A1.T2.A1T2H007.HS-.STOP" I4.7
reactor R001 stirring stop
"A1.A1H001.HS+-.START" I0.0
Main power switch multipurpose plant
"A1.A1H002.HS+-.OFF" I0.1
emergency switch OFF
"A1.A1H003.HS+-.LOC" I0.2
local operation mode switch
...\\A1T2L001(A,1)\LISA+_A1T2L001
V Process value
1
RS_FF
S
R
2
OR
OB35
4/2
3
CMP_R
REAL-Com
OB35
4/1
IN1
IN2
300.0
GT
GE
EQ
LE
LT
Press F1 for help.
A/Sheet 1
OB35 A1T2H007 A1T2H007\3

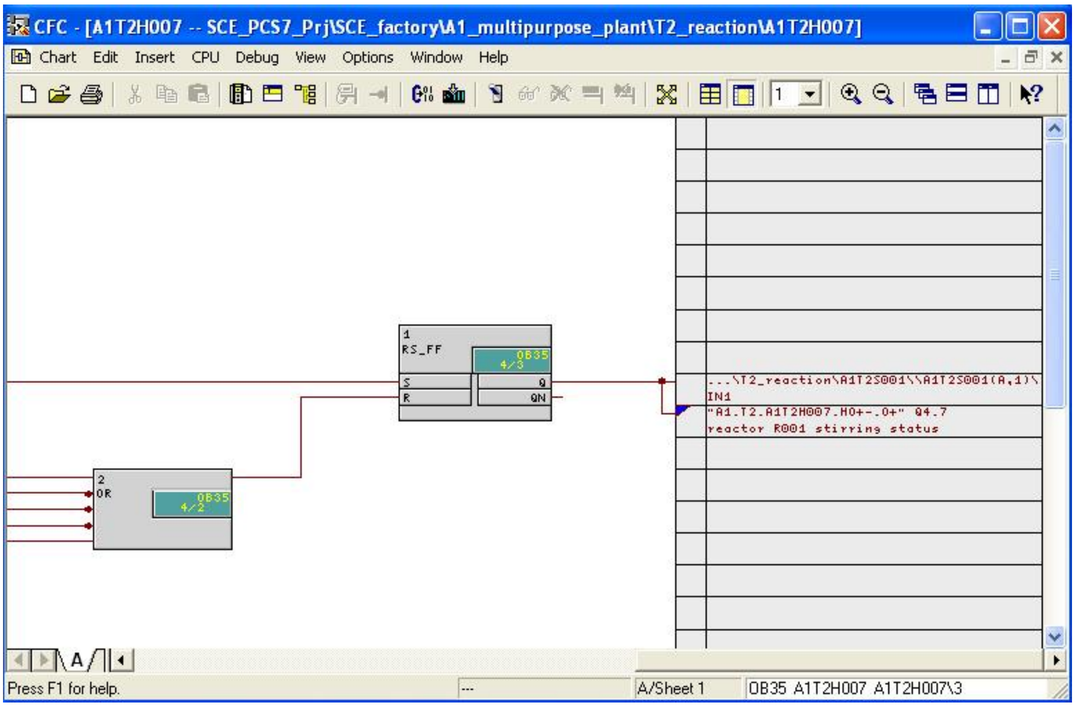

CFC - [A1T2H007 -- SCE_PCS7_Prj\SCE_factory\A1_multipurpose_plant\T2_reaction\A1T2H007]
Chart Edit Insert CPU Debug View Options Window Help
1
RS_FF
OB35
4/3
S
R
Q
QN
2
OR
OB35
4/2
...\T2_reaction\A1T2S001\\A1T2S001(A,1)\
IN1
"A1.T2.A1T2H007.H0+-.0+" Q4.7
reactor R001 stirring status
Press F1 for help.
A/Sheet 1
OB35 A1T2H007 A1T2H007\3

Table 5: New Blocks in Chart 'A1T2H007Sheet1'

Block	Catalog/Folder	Number of Connections
RS_FF / RS-FlipFlop	Blocks/FLIPFLOP	
OR / Or function	Blocks/BIT_LGC	5
CMP_R / Comparator for REAL values	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/COMPARE	

Table 6: Input Connections in Chart 'A1T2H007Sheet1'

Input	Connected to	Inverted
RS_FF.1.S	'A1.T2.A1T2H007.HS+.START' / I4.6 / Start stirring reactor R001	No
OR.2.IN1	'A1.T2.A1T2H007.HS-.STOP' / I4.7 / Stop stirring reactor R001	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 OFF	yes
OR.2.IN4	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	yes
CMP_R.3.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.3.IN2	300.0	

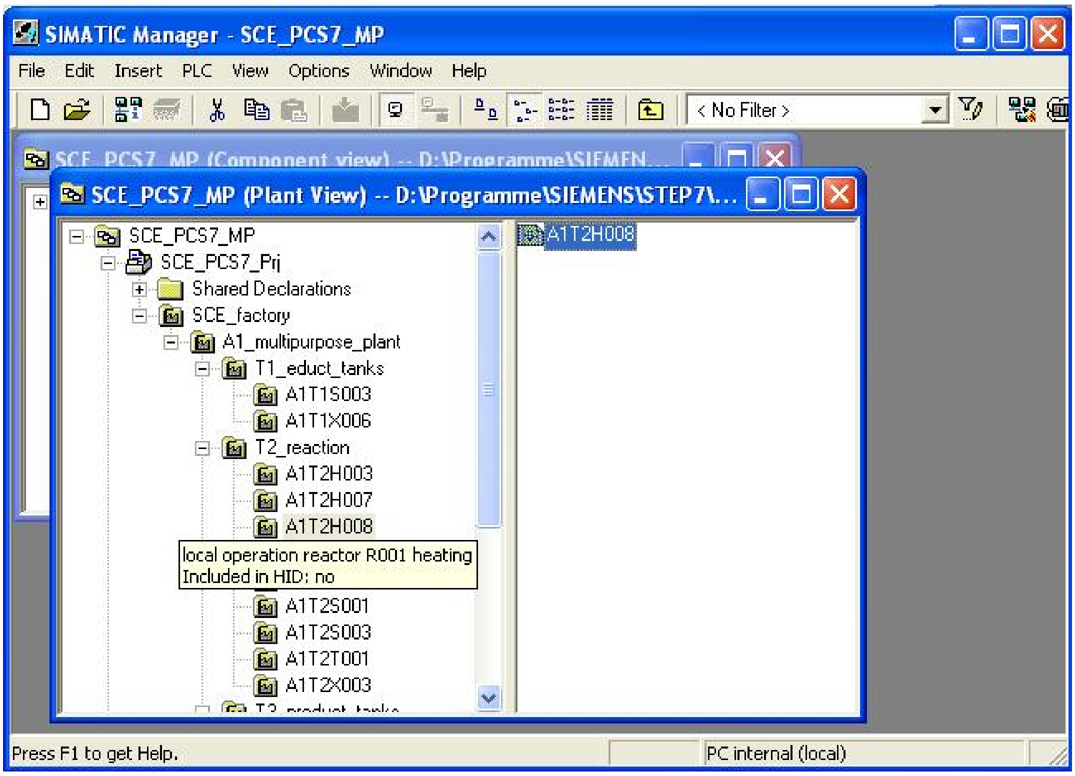
Table 7: Block Connections in Chart 'A1T2H007Sheet1'

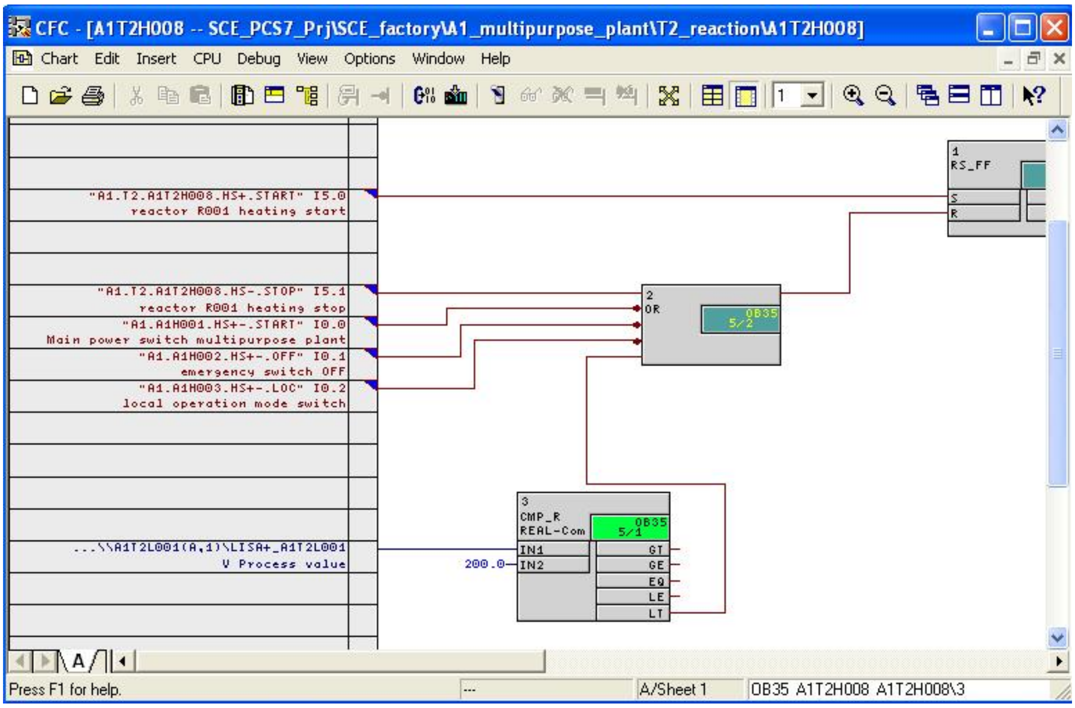
Input	Output	Inverted
RS_FF.1.R	OR.2.OUT	no
OR.2.IN5	CMP_R.3.LT	No

Table 8: Output Connections in Chart 'A1T2H0007Sheet1'

Output	Connection to	Inverted
RS_FF.1.Q	‚A1.T2.A1T2H007.HO+-.O-' / A4.7 / Reactor R001 stirring Status value	nein
RS_FF.1.Q	The second connection visible here will be generated later from chart A1T2S001.	

3. To program manual operation for heating in Reactor R001 we are setting up a new chart in the ***SIMATIC Manager*** in the plant view for the EMSR location A1T2H008. Follow the screen shots below and the entries:





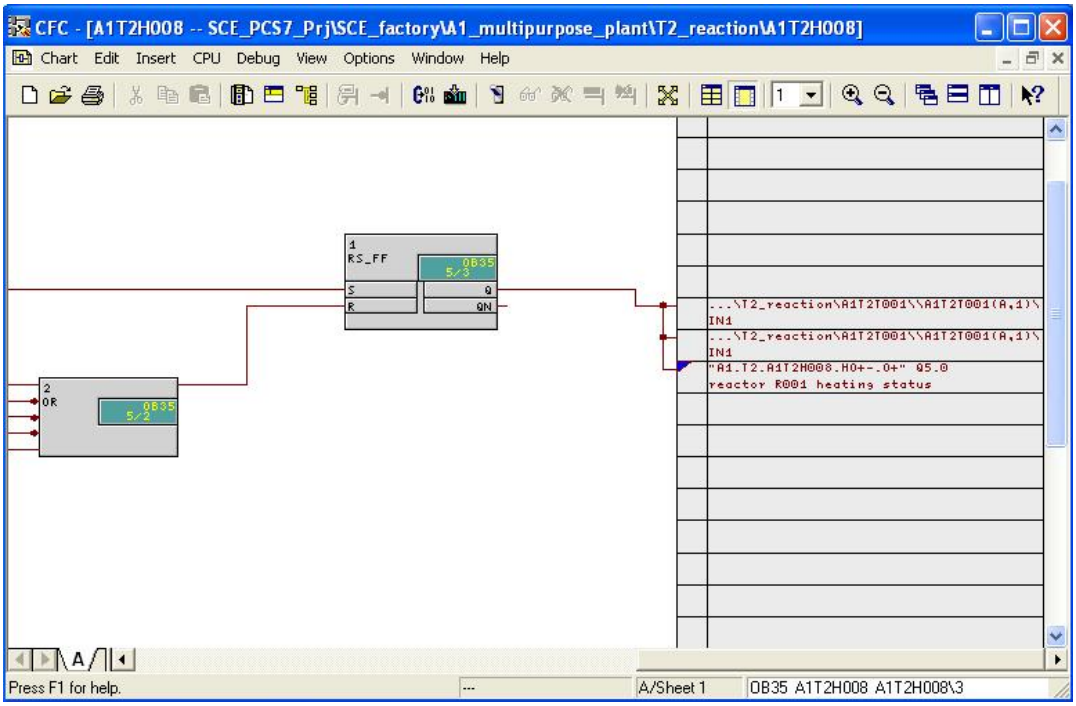


Table 9: New Blocks in Chart 'A1T2H008Sheet1'

Block	Catalog/Folder	Number of Connections
RS_FF / RS FlipFlop	Blocks/FLIPFLOP	
OR / Or function	Blocks/BIT_LGC	5
CMP_R / Comparator for REAL values	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/COMPARE	

Table 10: Input Connections in Chart 'A1T2H008Sheet1'

Input	Connection to	Inverted
RS_FF.1.S	'A1.T2.A1T2H008.HS+.START' / I5.0 / Start heating Reactor R001	no
OR.2.IN1	'A1.T2.A1T2H008.HS-.STOP' / I5.1 / Stop heating Reactor R001	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 OFF	yes
OR.2.IN4	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	yes
CMP_R.3.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.3.IN2	200.0	

Table 11: Block Connections in Chart 'A1T2H008Sheet1'

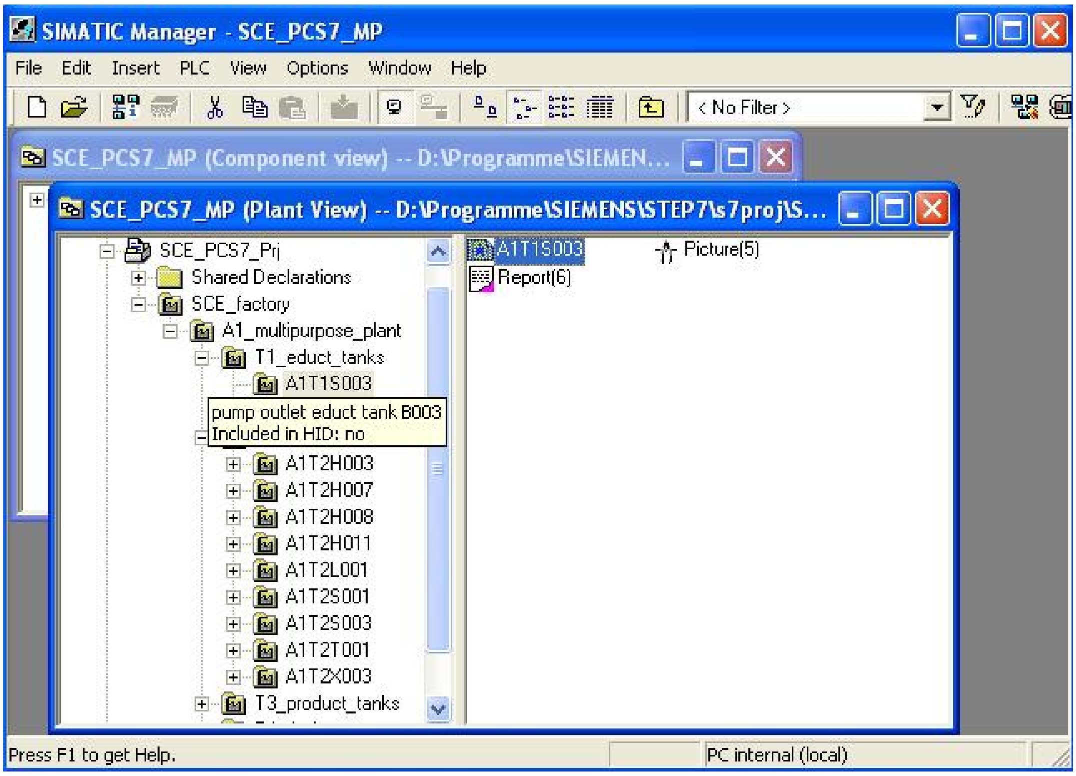
Input	Output	Inverted
RS_FF.1.R	OR.2.OUT	no
OR.2.IN5	CMP_R.3.LT	no

Table 12: Output Connections in Chart 'A1T2H0008Sheet1'

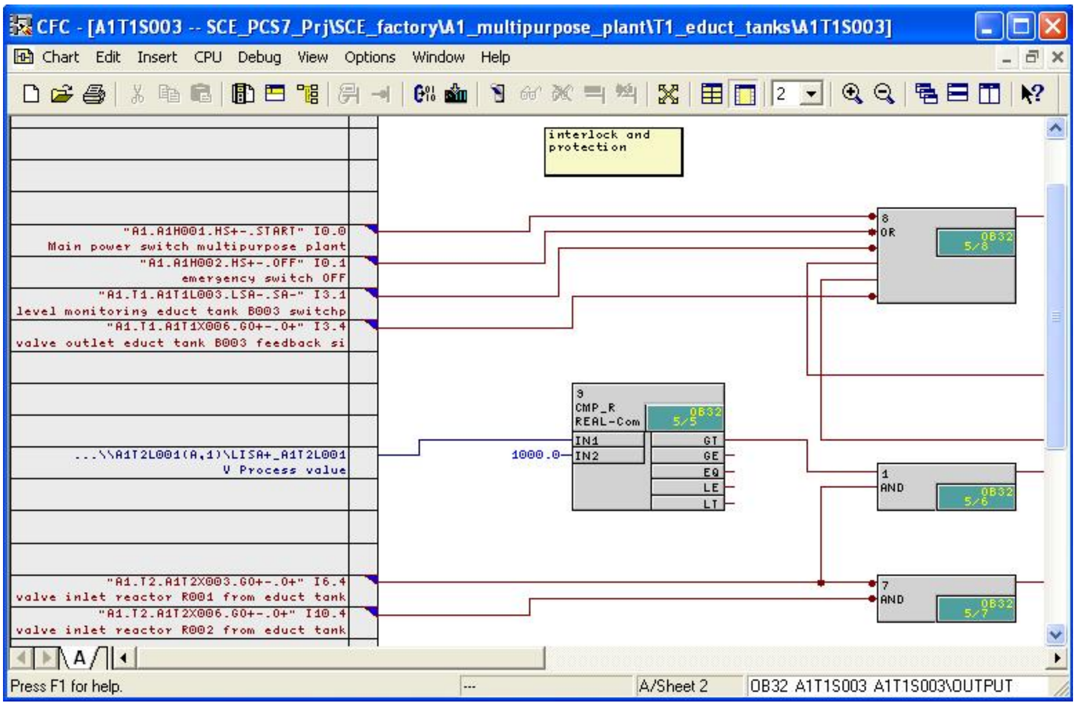
Output	Connection to	Inverted
RS_FF.1.Q	'A1.T2.A1T2H008.HO+-.O-' / Q5.0 / Heat Reactor R001 Status value	no
RS_FF.1.Q	The other two connections visible here are generated later from chart A1T2S001.	

4. Now we are gong to program the individual drive functions with locks for the 'Pump Outlet Educt Tank B003' in a CF chart with two sheets.

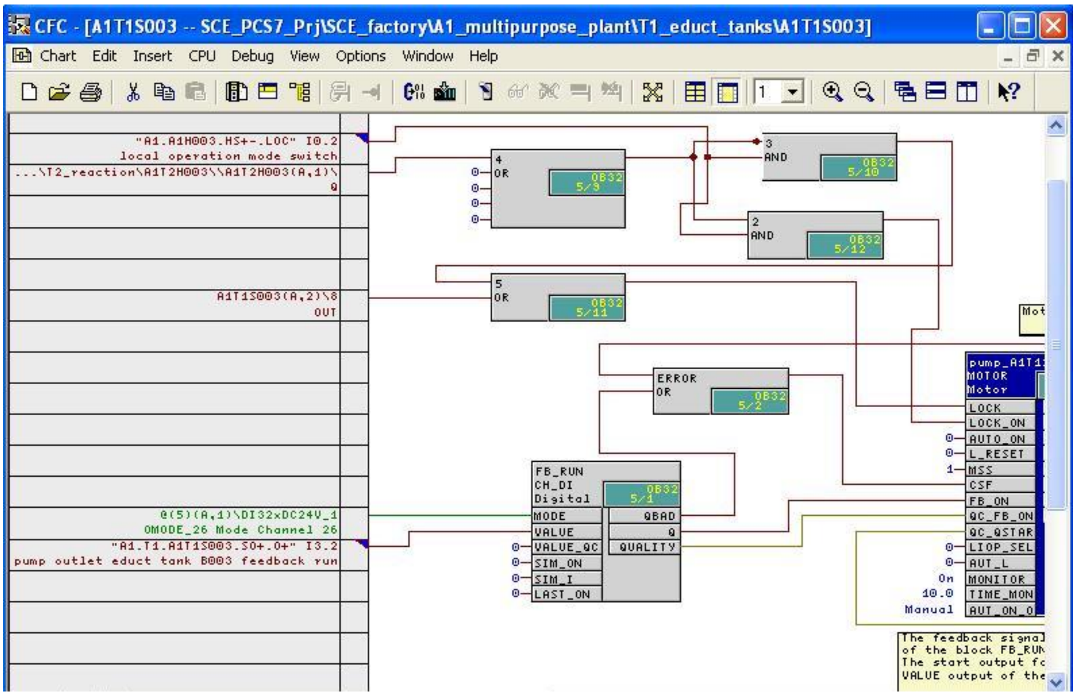
   To this end, in the ***SIMATIC Manager*** in the plant view in the folder 'MOTORS' we copy the process tag type 'MOTOR' from the master data library to the hierarchy folder for the EMSR location A1T1S003. Then, we make the other corrections and expansions in the CFC. Follow the screen shots and the entries below:



5. First we program in the second sheet of chart A1T1S003 the locks for 'Pump outlet Educt Tank B003'.



6. Then the other connections for 'Pump Outlet Educt Tank B003' in the first sheet of chart A1T1S003 follow.



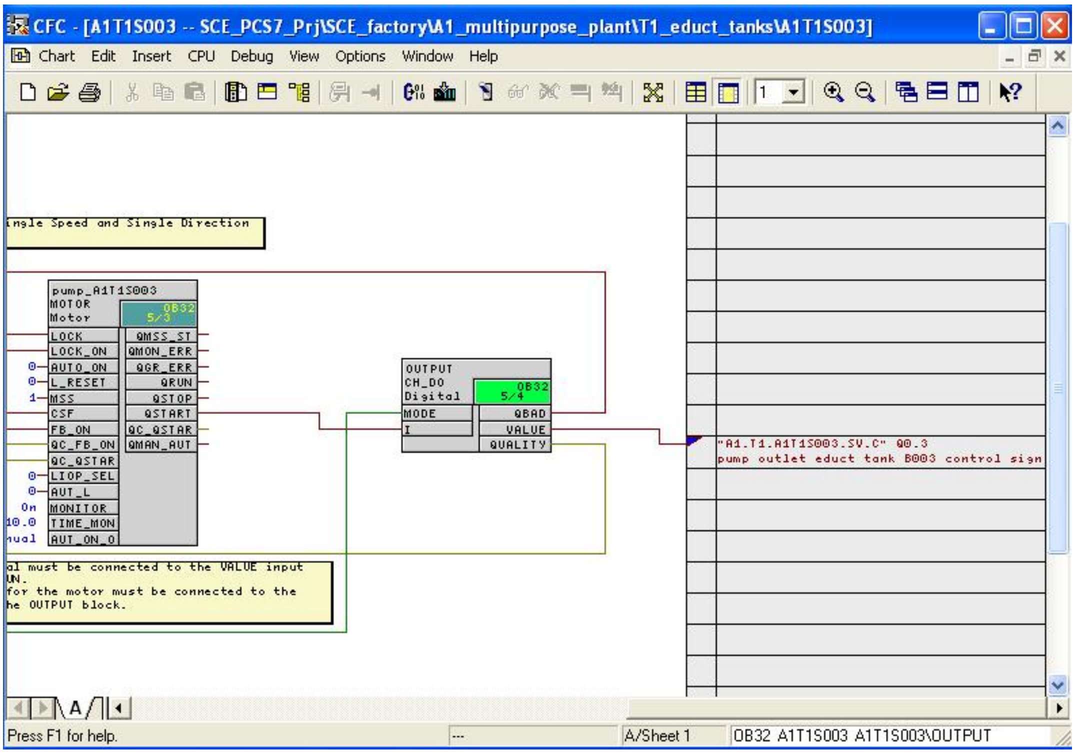

CFC - [A1T1S003 -- SCE_PCS7_Prj\SCE_factory\A1_multipurpose_plant\T1_educt_tanks\A1T1S003]
Chart Edit Insert CPU Debug View Options Window Help
ingle Speed and Single Direction
pump_A1T1S003
MOTOR
Motor
OB32
5/3
LOCK
LOCK_ON
AUTO_ON
L_RESET
MSS
CSF
FB_ON
QC_FB_ON
QC_QSTAR
LIOP_SEL
AUT_L
MONITOR
TIME_MON
AUT_ON_O
QMSS_ST
QMON_ERR
QGR_ERR
QRUN
QSTOP
QSTART
QC_QSTAR
QMAN_AUT
OUTPUT
CH_DO
Digital
OB32
5/4
MODE
I
QBAD
VALUE
QUALITY
"A1.T1.A1T1S003.SV.C" Q0.3
pump outlet educt tank B003 control sign
al must be connected to the VALUE input
ON.
for the motor must be connected to the
he OUTPUT block.
Press F1 for help.
A/Sheet 1
OB32 A1T1S003 A1T1S003\OUTPUT

Table 13: New Blocks in Chart 'A1T1S003/Sheet2'

Block	Catalog/Folder	Number of Connections
OR / Or function	Blocks/BIT_LGC	6
AND / And function	Blocks/BIT_LGC	2
AND / And function	Blocks/BIT_LGC	2
CMP_R / Comparator floating point numbers	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/COMPARE	

Table 14: Input Connections in Chart 'A1T1S003/Sheet2'

Input	Connection to	Inverted
OR.8.IN1	'A1.A1H001.HS+-.START' / I0.0 / Switch on multi-purpose plant	yes
OR.8.IN2	'A1.A1H002.HS+-.OFF' / I0.1 / Activate emergency OFF	yes
OR.8.IN3	'A1.T1.A1T1L003.LSA-.SA-' / I3.1 / Level monitoring Educt Tank B003 operating point L	yes
OR.8.IN6	'A1.T1.A1T1X006.GO+-.O+' / I3.4 / Open/close valve outlet Educt Tank B003 Feedback Open	yes
AND.1.IN2	'A1.T2.A1T2X003.GO+-.O+' / I6.4 / Open/close valve inlet Reactor R001 from Educt Tank B003 Feedback Open	no
AND.7.IN1	'A1.T2.A1T2X003.GO+-.O+' / I6.4 / Open/close valve inlet Reactor R001 from Educt Tank B003 Feedback Open	yes
AND.7.IN2	'A1.T3.A1T2X006.GO+-.O+' / I10.4 / Open/close valve inlet Reactor R002 from Educt Tank B003 Feedback Open	yes
CMP_R.9.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.9.IN2	1000.0	

Table 15: Block Connections in Chart 'A1T1S003/Sheet2'

Input	Output	Inverted
AND.1.IN1	CMP_R.9.GT	no
OR.8.IN4	AND.1.OUT	no
OR.1.IN5	AND.7.OUT	no

Table 16: New Blocks in Chart 'A1T1S003/Sheet1'

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

Table 17: Input Connections in Chart 'A1T1S003/Sheet1'

Input	Connection to	Inverted
AND.3.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
AND.2.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
OR.4.IN1	A1T2H003(A,1) / RS_FF.1.Q	no
OR.5.IN2	A1T1S003(A,2) / OR.8.OUT	no
CH_DI.FB_RUN.VALUE	'A1.T1.A1T1S003.SO+.O+' / I3.2 / Pump outlet Educt Tank B003 Feedback On	no
MOTOR.Pump_A1T2S003. MONITOR.	On	
MOTOR.Pump_A1T2S003. TIME_MON	10.0	

Table 18: Block Connections in Chart 'A1T1S003/Sheet1'

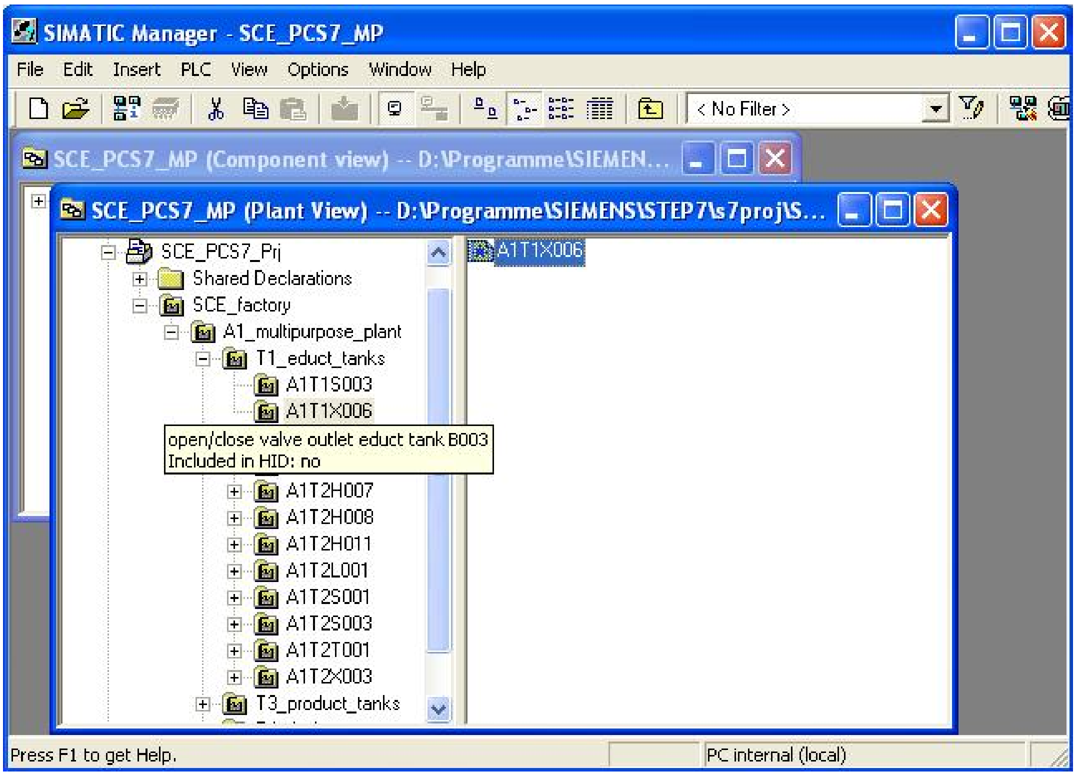
Input	Output	Inverted
AND.3.IN1	OR.4.OUT	yes
AND.2.IN1	OR.4.OUT	no
OR.5.IN1	AND.3.OUT	no
MOTOR.Pump_A1T1S003. LOCK	OR.5.OUT	no
MOTOR.Pump_A1T1S003. LOCK_ON	AND.2.OUT	no
MOTOR.Pump_A1T1S003. FB_ON	CH_DI.FB_RUN.Q	no
CH_DO.OUTPUT.I	MOTOR.Pump_A1T1S003.QSTART	no
MOTOR.Pump_A1T1S003. CSF	OR.ERROR.OUT	no
OR.ERROR.IN1	CH_DI.FB_RUN.QBAD	no
OR.ERROR.IN2	CH_DO.OUTPUT.QBAD	no

Table 19: Output Connections in Chart 'A1T1S003/Sheet1'

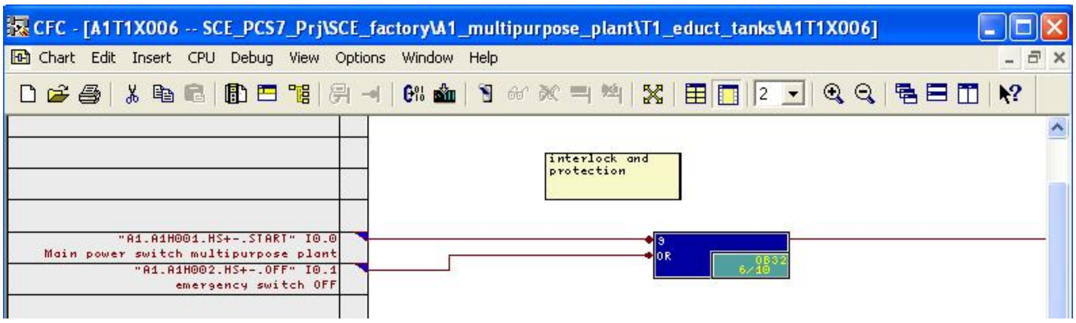
Output	Connection to	Inverted
CH_DO.OUTPUT.VALUE	'A1.T1.A1T1S003.SV.C' / Q0.3 / Pump outlet Educt Tank B003 Actuating signal	no

7. Next, we program the individual drive function with locks for the 'Open/Close valve outlet Educt Tank B003' in a CFC with 2 sheets.

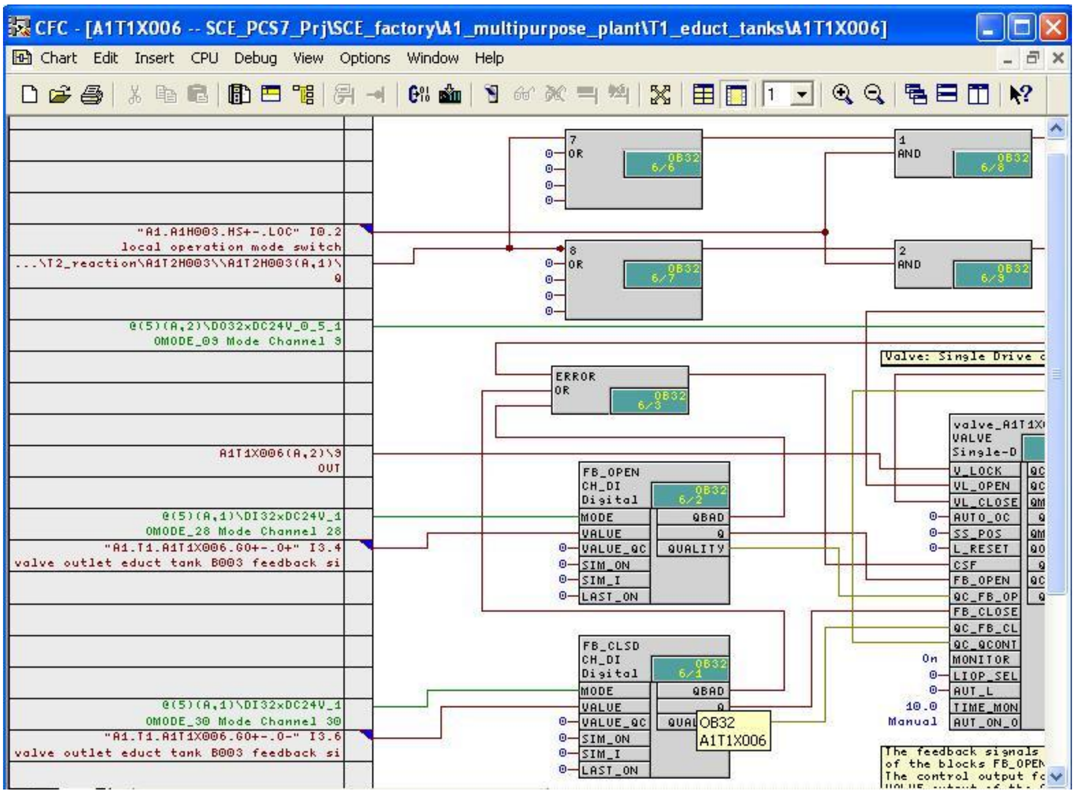
   To do this, we copy in the ***SIMATIC Manager*** in the plant view in the folder 'VALVES' the process tag type 'VALVE' from the master data library to the hierarchy folder for the EMSR location A1T1X006. Then, we make the other corrections and expansions in the CFC. Follow the screen shots below and the entries:



8. The lock for the 'Open/close valve outlet Educt Tank B003' is programmed in Sheet2 again.



9. All additional connections for the 'Open/close valve outlet Educt Tank B003' are made in Sheet 1.



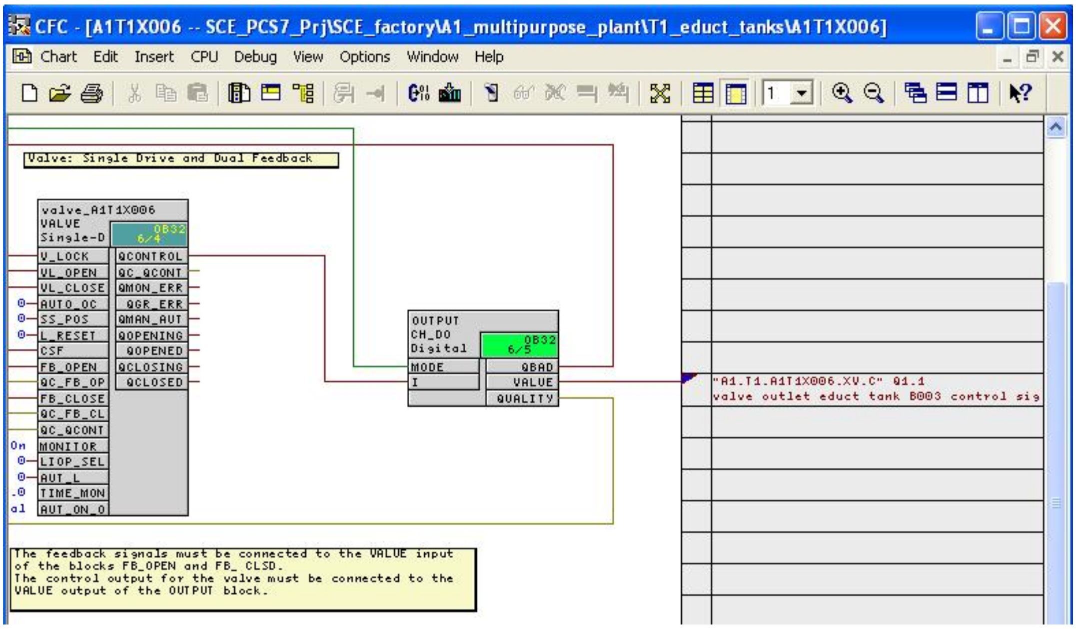


Table 20: New Blocks in Chart 'A1T1X006/Sheet2'

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

Table 21: Input Connections in Chart 'A1T1X006/Sheet2'

Input	Connection to	Inverted
OR.9.IN1	'A1.A1H001.HS+-.START' / I0.0 / Switch on multi-purpose plant	yes
OR.9.IN2	'A1.A1H002.HS+-.OFF' / I0.1 / Activate emergency OFF	yes

Table 22: Block Connections in Plan 'A1T1X006/Sheet2'

Input	Output	Inverted
None	None	no

Table 23: New Blocks in Chart 'A1T1X006/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	5

Table 24: Input Connections in Chart 'A1T1X006/Sheet1'

Input	Connection to	Inverted
AND.1.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
AND.2.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
OR.7.IN1	A1T2H003(A,1) / RS_FF.1.Q	no
OR.8.IN1	A1T2H003(A,1) / RS_FF.1.Q	yes
CH_DI.FB_OPEN.VALUE	'A1.T1.A1T1X006.GO+.O+' / I3.4 / Open/Close valve outlet Educt Tank B003 Feedback Open	no
CH_DI.FB_CLSD.VALUE	'A1.T1.A1T1X006.GO+.O+' / I3.6 / Open/close valve Educt Tank B003 Feedback Closed	no
VALVE.Ventil_A1T1X006.V_LOCK	A1T1X006(A,2) / OR.9.OUT	no
VALVE.Ventil_A1T1X006.MONITOR	On	
VALVE.Ventil_A1T1X006.TIME_MON	10.0	

Table 25: Block Connections in Chart 'A1T1X006/Sheet1'

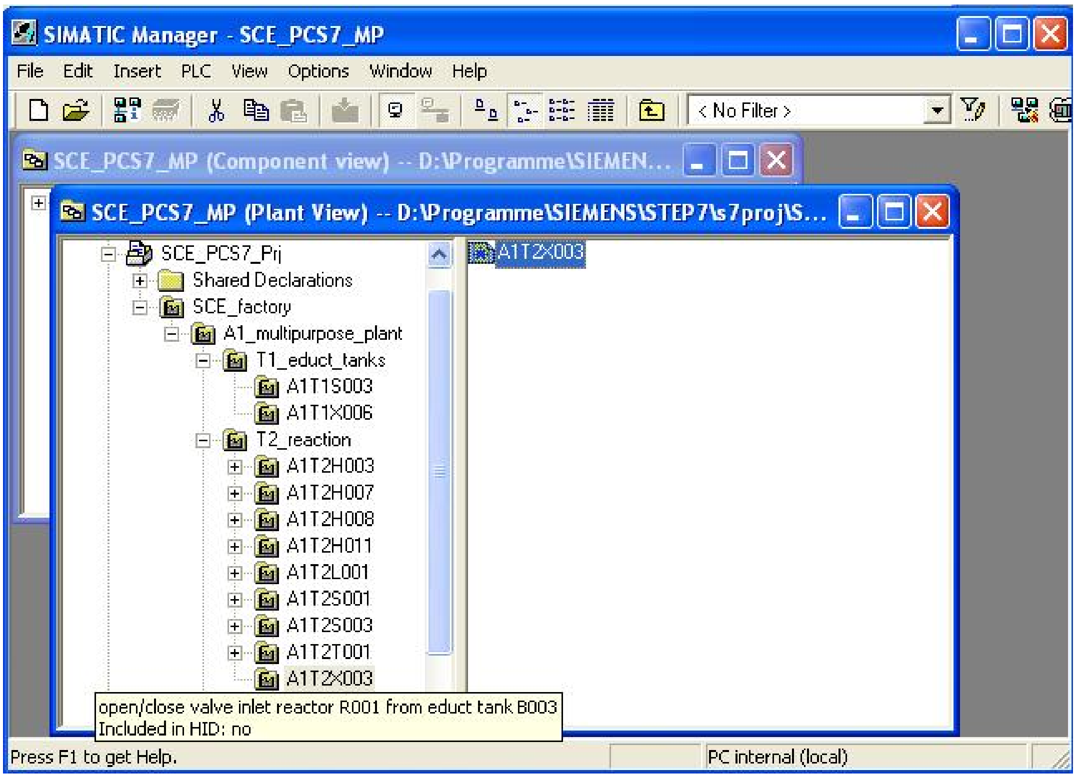
Input	Output	Inverted
AND.1.IN1	OR.7.OUT	no
AND.2.IN1	OR.8.OUT	no
VALVE.Ventil_A1T1X006.VL_OPEN	AND.1.OUT	no
VALVE.Ventil_A1T1X006.VL_CLOSE	AND.2.OUT	no
VALVE.Ventil_A1T1X006.FB_OPEN	CH_DI.FB_OPEN.Q	no
VALVE.Ventil_A1T1X006.FB_CLOSE	CH_DI.FB_CLSD.Q	no
CH_DO.OUTPUT.I	VALVE.Ventil_A1T1X006.QCONTROL	no
VALVE.Ventil_A1T1X006.CSF	OR.ERROR.OUT	no
OR.ERROR.IN1	CH_DO.OUTPUT.QBAD	no
OR.ERROR.IN2	CH_DI.FB_CLSD.QBAD	no
OR.ERROR.IN3	CH_DI.FB_OPEN.QBAD	no

Table 26: Output Connections in Chart 'A1T1X006/Sheet1'

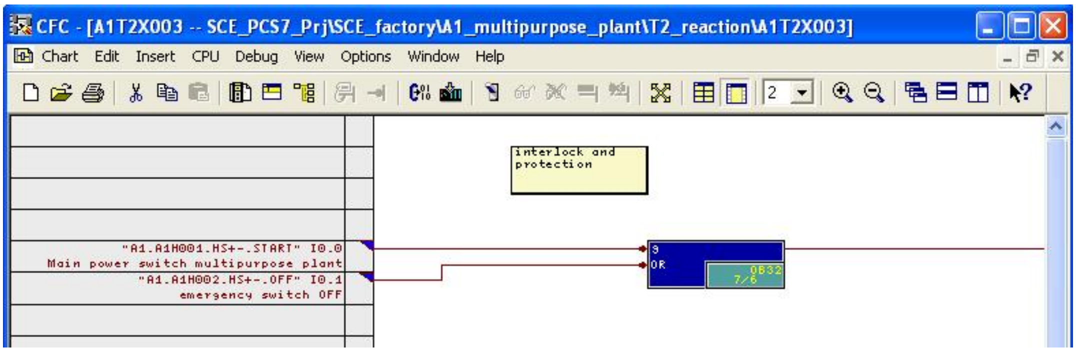
Output	Connection to	Inverted
CH_DO.OUTPUT.VALUE	'A1.T1.A1T1X006.XV.C' / Q1.1 / Open/Close valve outlet Educt Tank B003 Control signal	no

10. Now, we are going to also program in a CFC with 2 sheets the individual drive function with locks for the 'Open/Close Valve Inlet Reactor R001 from Educt Tank B003'.

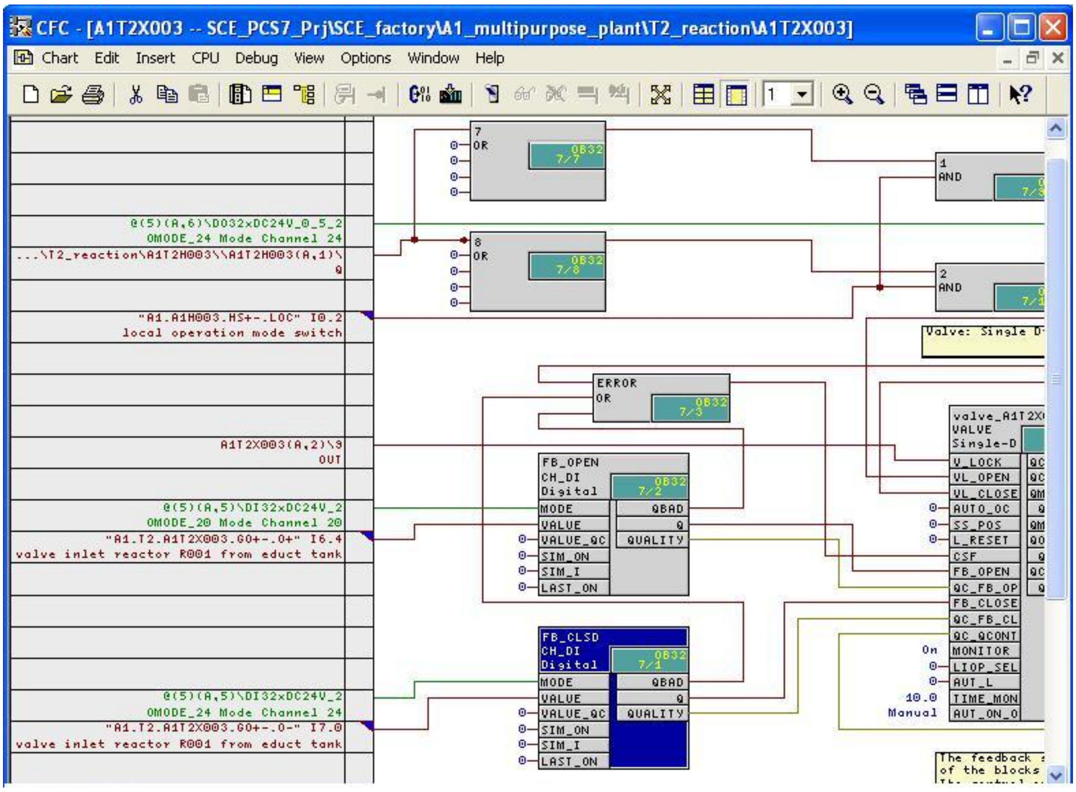
    To do this, we copy in the ***SIMATIC Manager*** in the plant view in the folder 'VALVES' the process tag type 'VALVE' from the master data library to the hierarchy folder for the EMSR location A1T2X003. We then make the other corrections and expansions in the CFC. Follow the screen shots below and the entries:



11. The lock for the 'Open/close valve inlet reactor R001 from educt tank B003' is programmed on Sheet2.



12. In Sheet1, the other connections for the 'Open/close valve inlet reactor R001 from educt tank B003' are set up.



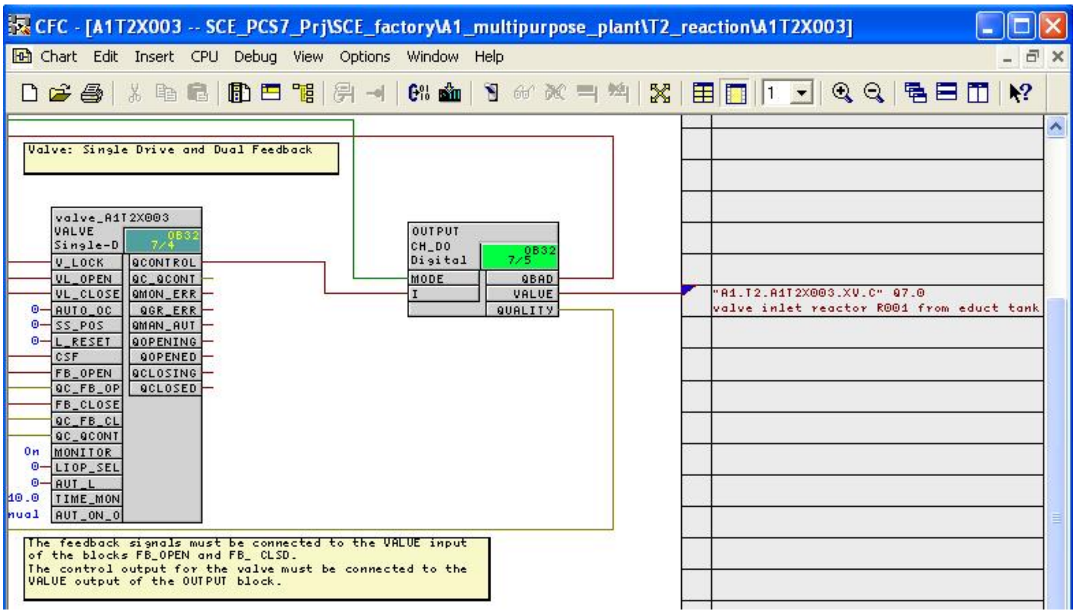


Table 27: New Blocks in Chart 'A1T2X003/Sheet2'

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

Table 28: Input Connections in Chart 'A1T2X003/Sheet2'

Input	Connection to	Inverted
OR.9.IN1	'A1.A1H001.HS+-.START' / I0.0 / Switch on multi-purpose plant	yes
OR.9.IN2	'A1.A1H002.HS+-.OFF' / I0.1 / Activate emergency OFF	yes

Table 29: Block Connections in Chart 'A1T2X003/Sheet2'

Input	Output	Inverted
None	None	no

Table 30: New Blocks in Chart 'A1T2X003/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	5

Table 31: Input Connections in Chart 'A1T2X003/Sheet1'

Input	Connection to	Inverted
AND.1.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
AND.2.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
OR.7.IN1	A1T2H003(A,1) / RS_FF.1.Q	no
OR.8.IN1	A1T2H003(A,1) / RS_FF.1.Q	yes
CH_DI.FB_OPEN.VALUE	'A1.T2.A1T2X003.GO+.O+' / I6.4 / Open/Close valve inlet reactor R001 from educt tank B003 Feedback Open	no
CH_DI.FB_CLSD.VALUE	'A1.T2.A1T2X003.GO+.O+' / I7.0 / Open/close valve inlet reactor R001 from educt tank B003 Feedback closed	no
VALVE.Ventil_A1T2X003.V_LOCK	A1T2X003(A,2) / OR.9.OUT	no
VALVE.Ventil_A1T2X003.MONITOR	On	
VALVE.Ventil_A1T2X003.TIME_MON	10.0	

Table 32: Block Connections in Chart 'A1T2X003/Sheet1'

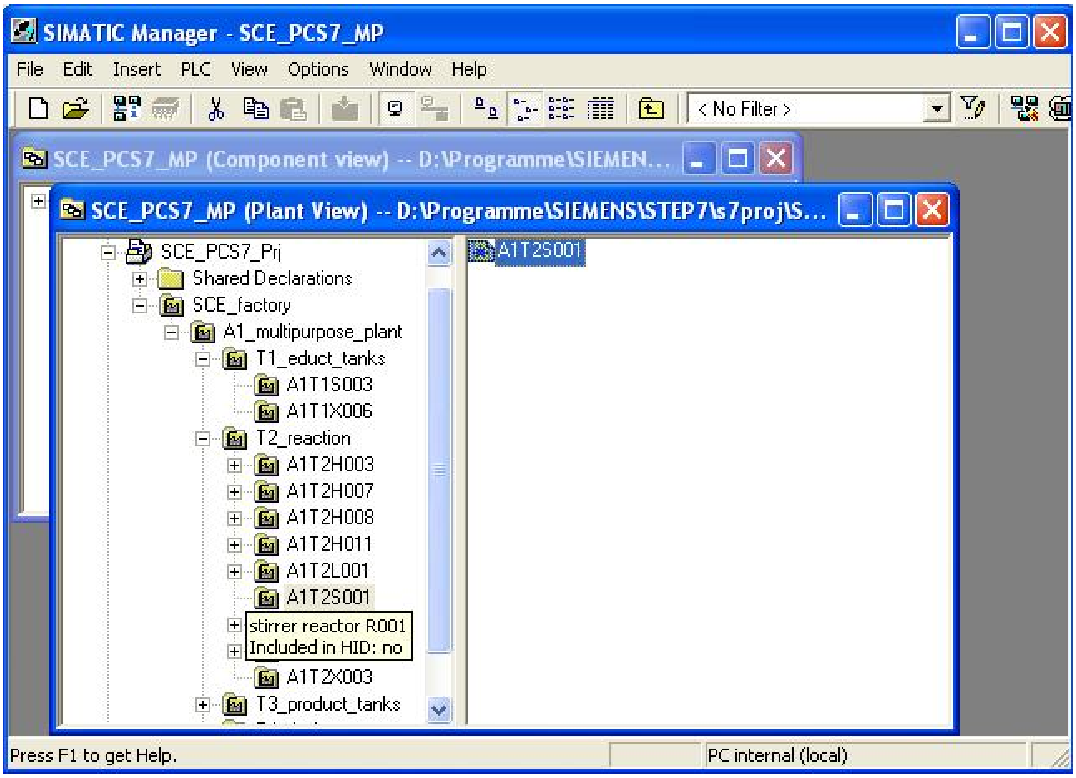
Input	Output	Inverted
AND.1.IN1	OR.7.OUT	yes
AND.2.IN1	OR.8.OUT	no
VALVE.Ventil_A1T2X003.VL_OPEN	AND.1.OUT	no
VALVE.Ventil_A1T2X003.VL_OPEN	AND.2.OUT	no
VALVE.Ventil_A1T2X003.FB_OPEN	CH_DI.FB_OPEN.Q	no
VALVE.Ventil_A1T2X003.FB_CLOSE	CH_DI.FB_CLSD.Q	no
CH_DO.OUTPUT.I	VALVE.Ventil_A1T2X003.QCONTROL	no
VALVE.Ventil_A1T2X003.CSF	OR.ERROR.OUT	no
OR.ERROR.IN1	CH_DO.OUTPUT.QBAD	no
OR.ERROR.IN2	CH_DI.FB_CLSD.QBAD	no
OR.ERROR.IN2	CH_DI.FB_OPEN.QBAD	no

Table 33: Output Connections in Chart 'A1T2X003/Sheet1'

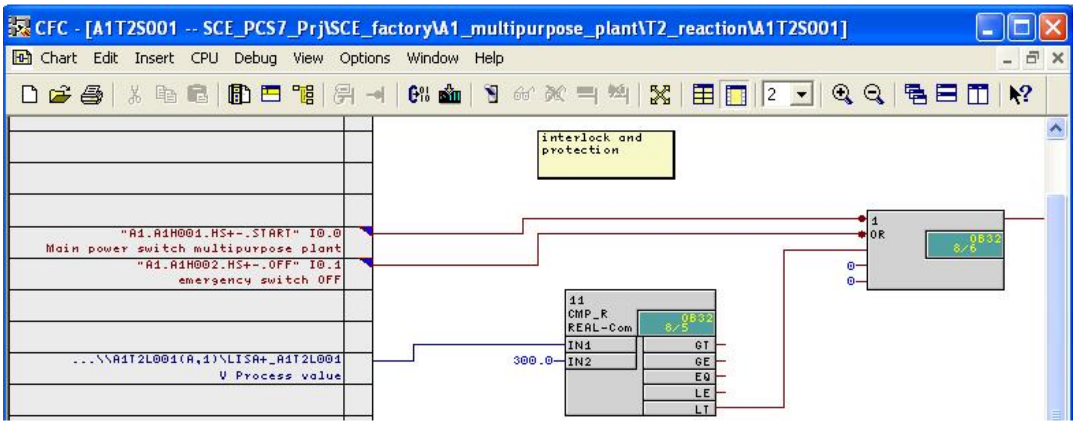
Output	Connection to	Inverted
CH_DO.OUTPUT.VALUE	'A1.T2.A1T2X003.XV.C' / Q7.0 / Open/close valve inlet reactor R001 from educt tank B003 Controlling signal	no

13. Now, we are going to program in a CFC with two sheets the individual drive functions for the stirrer 'Reactor R001'

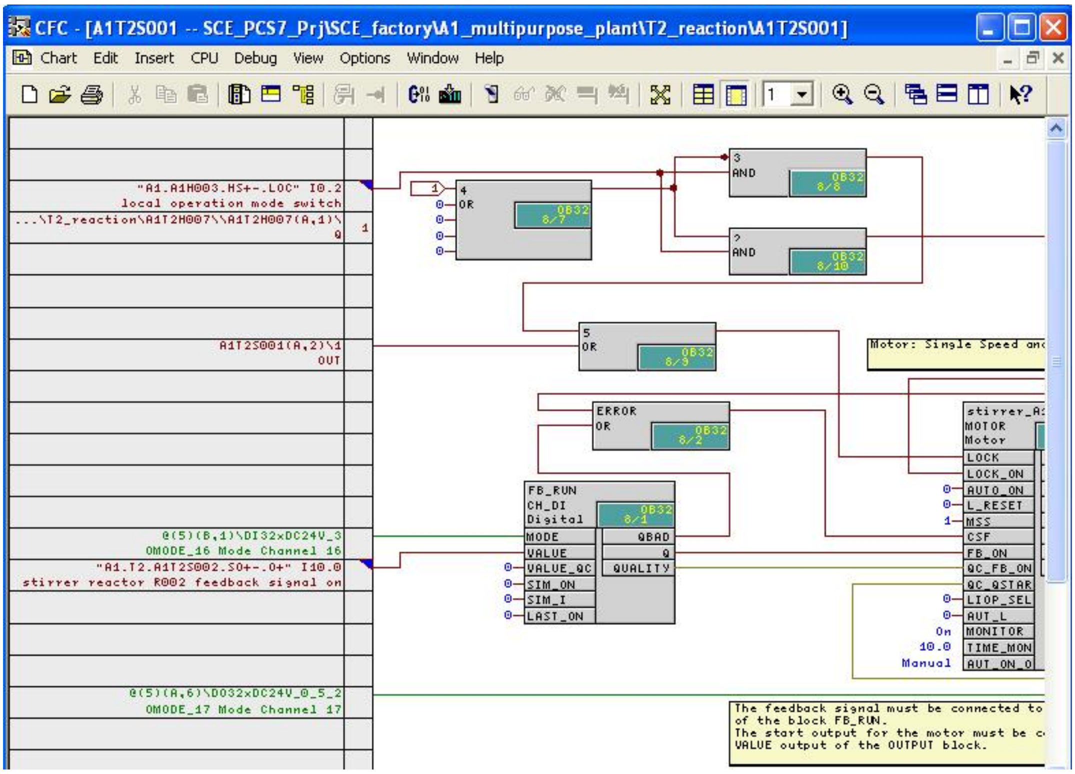
    To this end, we copy in the ***SIMATIC Manager*** in the plant view in the folder 'MOTORS' the process tag type 'MOTOR' from the master data library to the hierarchy folder for the EMSR location A1T2S001. Then, we make the other corrections and expansions in the CFC. Follow the screen shots that follow and the entries:



14. The locks for the 'Stirrer Reactor R001' are programmed in Sheet 2.



15. Other connections for the 'Stirrer Reactor R001' are programmed in Sheet 1.



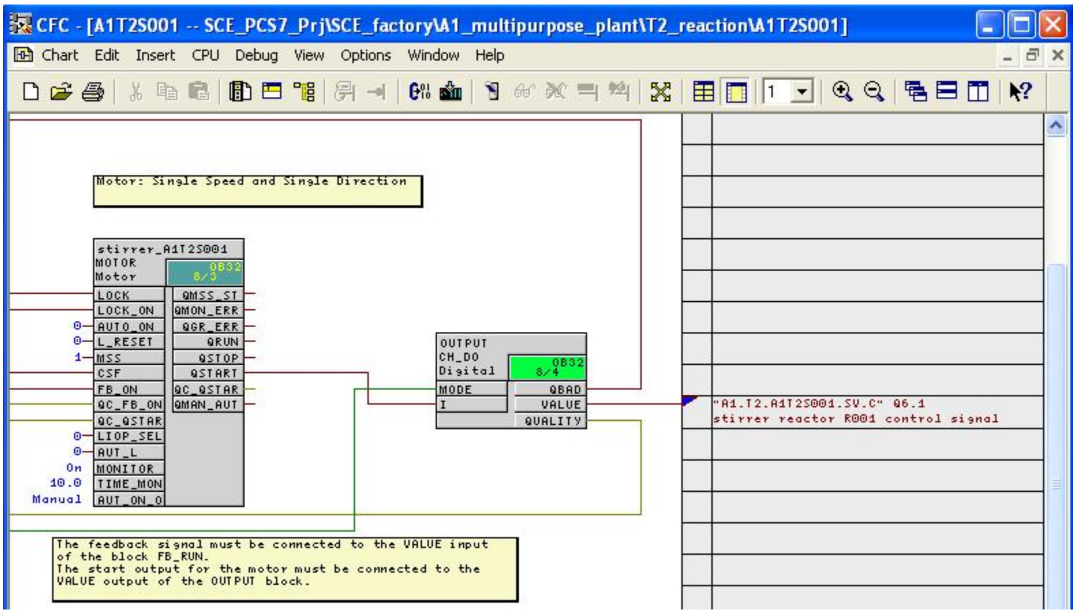


Table 34: New Blocks in Chart 'A1T2S001/Sheet2'

Block	Catalog/Folder	Number of Connections
OR / Or function	Blocks/BIT_LGC	5
CMP_R / Comparator Floating point numbers	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/COMPARE	

Table 35: Input Connections in Chart ‚A1T2S001/Sheet2’

Input	Connection 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 OFF	yes
CMP_R.11.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.11.IN2	300.0	

Table 36: Block Connections in Chart 'A1T2S001/Sheet2’

Input	Output	Inverted
OR.1.IN1	CMP_R.11.LT	no

Table 37: New Blocks in Chart 'A1T2S001/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 38: Input Connections in Chart 'A1T2S001/Sheet1'

Input	Connection to	Inverted
AND.3.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
AND.2.IN2	'A1.A1H003.HS+-.LOC' / I0.2 / Activate local operation	no
OR.4.IN1	A1T2H007(A,1) / RS_FF.1.Q	no
OR.5.IN2	A1T2S001(A,2) / OR.1.OUT	no
CH_DI.FB_RUN.VALUE	'A1.T2.A1T2S002.SO+.O+' / I10.0 / Stirrer reactor R001 Feedback On	
MOTOR.Stirrer_A1T2S001. MONITOR.	On	
MOTOR.Stirrer_A1T2S001. TIME_MON	10.0	

Table 39: Block Connections in Plan 'A1T2S001/Sheet1'

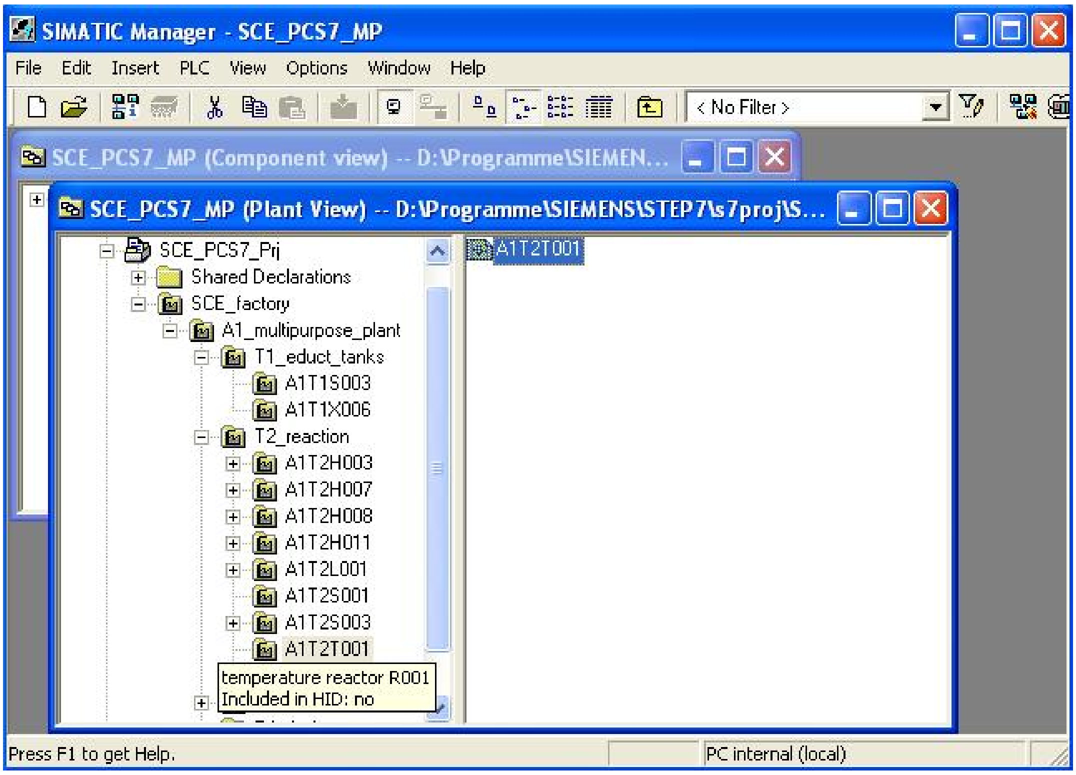
Input	Output	Inverted
AND.3.IN1	OR.4.OUT	yes
AND.2.IN1	OR.4.OUT	no
OR.5.IN1	AND.3.OUT	no
MOTOR.Stirrer_A1T2S001. LOCK	OR.5.OUT	no
MOTOR.Stirrer_A1T2S001. LOCK_ON	AND.2.OUT	no
MOTOR.Stiier_A1T2S001.F B_ON	CH_DI.FB_RUN.Q	no
CH_DO.OUTPUT.I	MOTOR.Stirrer_A1T2S001.QSTART	no
MOTOR.Stirrer_A1T2S001. CSF	OR.ERROR.OUT	no
OR.ERROR.IN1	CH_DO.OUTPUT.QBAD	no
OR.ERROR.IN2	CH_DI.FB_RUN.QBAD	no

Table 40: Output Connections in Chart 'A1T2S001/Sheet1'

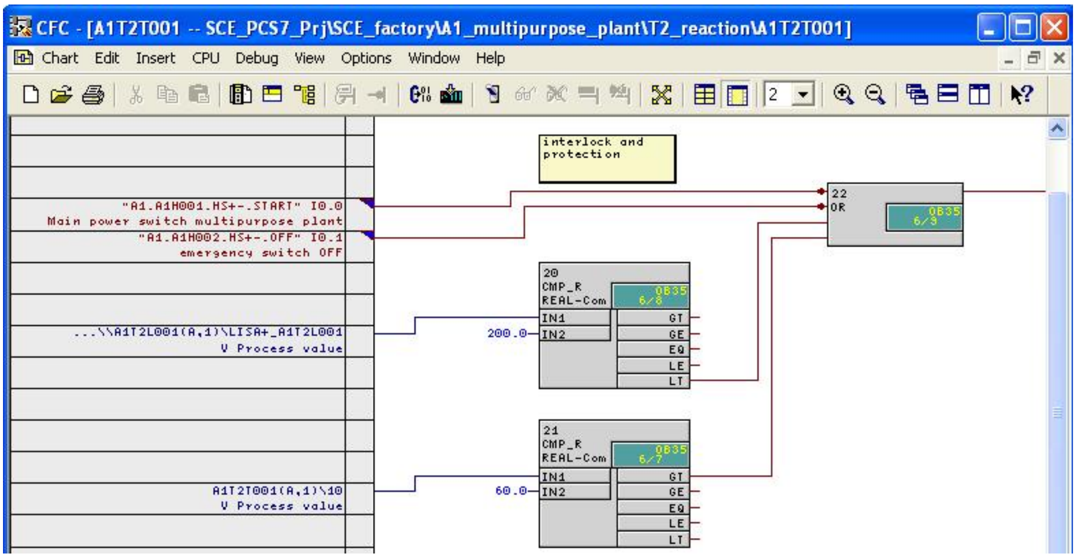
Output	Connection to	Inverted
CH_DO.OUTPUT.VALUE	'A1.T2.A1T2S001.SV.C' / Q6.1 / Stirrer reactor R001 Control signal	no

16. Now we are going to program in a CFC with 2 sheets the individual drive function with locks for 'Temperature Reactor R001'. We are going to use a continuous PID controller for this.

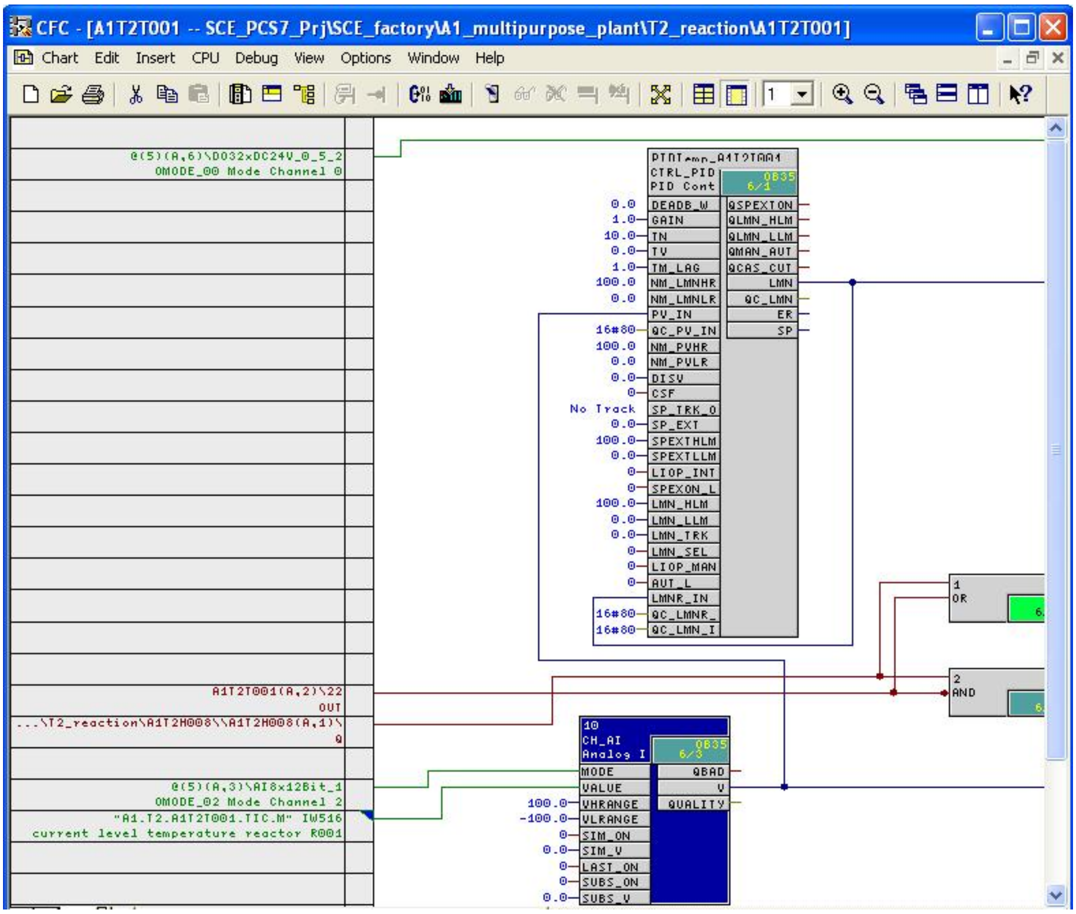
    First, in the ***SIMATIC Manager*** in the plant view we are setting up a new CFC for the EMSR location A1T2T001. Next, we make the other corrections and expansions in the CFC. Follow the screen shots below and entries:



17. The locks for 'Temperature Reactor R001' are programmed in Sheet2.



18. In Sheet 1 of chart A1T2T001, the other connections of 'Temperature Reactor R001' are programmed.



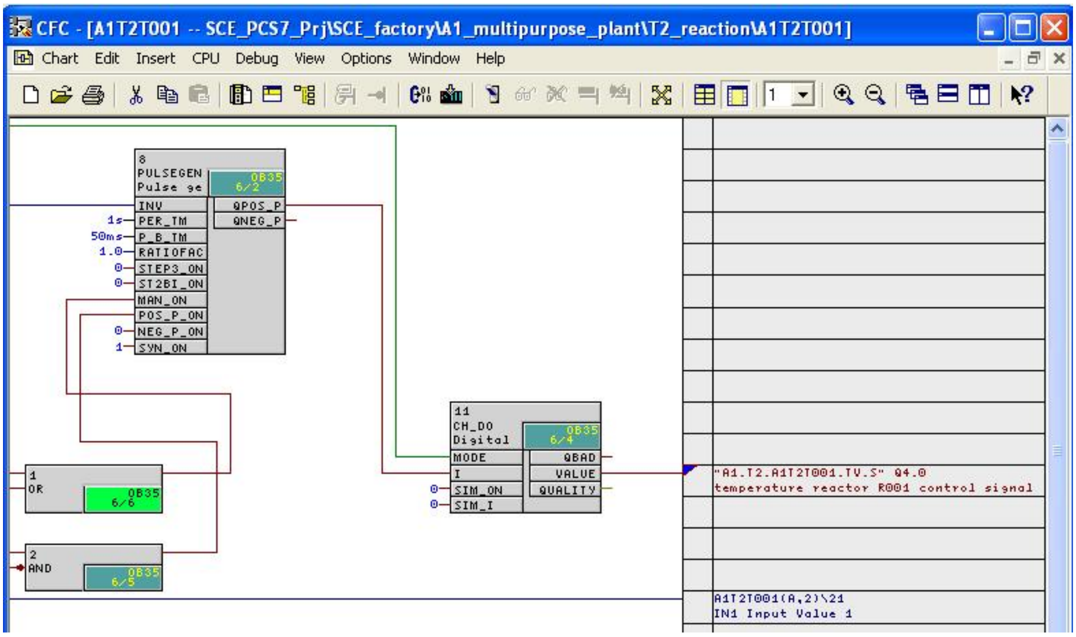


Table 41: New Blocks in Chart 'A1T2T001/Sheet2'

Block	Catalog/Folder	Number of Connections
OR / Or function	Blocks/BIT_LGC	4
CMP_R / Comparator floating point numbers	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/COMPARE	
CMP_R / Comparator floating point numbers	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/COMPARE	

Table 42: Input Connections in Chart 'A1T2T001/Sheet2'

Input	Connection to	Inverted
OR.22.IN1	'A1.A1H001.HS+-.START' / I0.0 / Switch on multi-purpose plant	yes
OR.22.IN2	'A1.A1H002.HS+-.OFF' / I0.1 / Activate emergency OFF	yes
CMP_R.20.IN1	A1T2L001(A,1) / CH_AI.LISA+_A1T2L001.V Process value	
CMP_R.20.IN2	200.0	
CMP_R.21.IN1	A1T2T001(A,1) / CH_AI.10.V Process value	
CMP_R.21.IN2	60.0	

Table 43: Block Connections in Chart 'A1T2T001/Sheet2'

Input	Output	Inverted
OR.22.IN3	CMP_R.20.LT	no
OR.22.IN4	CMP_R.21.GT	no

Table 44: New Blocks in Chart 'A1T2T001/Sheet1'

Block	Catalog/Folder	Number of Connections
CTRL_PID / Continuous PID controller	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/CONTROL	
PULSEGEN / Pulse width modulation for PID controller	Libraries/CFC Library /ELEM_400\Blocks/CONTROL	
CH_AI / Driver block Analog value input	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/DRIVER	
CH_DO / Driver block Binary output	Libraries/PCS 7 Library V71/Blocks+Templates\Blocks/DRIVER	
AND / And function	Blocks/BIT_LGC	2
OR / Or function	Blocks/BIT_LGC	2

Table 45: Input Connections in Chart 'A1T2T001/Sheet1'

Input	Connection to	Inverted
OR.1.IN1	A1T2H008(A,1) / RS_FF.1.Q	no
OR.1.IN2	A1T2T001(A,2) / OR.22.OUT	no
AND.2.IN1	A1T2H008(A,1) / RS_FF.1.Q	no
AND.2.IN2	A1T2T001(A,2) / OR.22.OUT	yes
CH_AI.10.VALUE	‚A1.T2.A1T2T001.TIC.M' / IW516 / Temperature actual value Reactor R001	
CH_AI.10.VHRANGE	100	
CH_AI.10.VLRANGE	-100	
PULSEGEN.8.STEP3_ON	0	

Table 46: Block Connections in Chart 'A1T2T001/Sheet1'

Input	Output	Inverted
CTRL_PID.PIDTemp_A1T2T001.PV_IN	CH_AI.10.V	
CTRL_PID.PIDTemp_A1T2T001.LMNR_IN	CTRL_PID.PIDTemp_A1T2T001.LMN	
PULSEGEN.8.INV	CTRL_PID.PIDTemp_A1T2T001.LMN	
PULSEGEN.8.MAN_ON	OR.1.OUT	no
PULSEGEN.8.POS_P_ON	AND.2.OUT	no
CH_DO.11.I	PULSEGEN.8.QPOS_P	no

Table 47: Output Connections in Chart 'A1T2T001/Sheet1'

Output	Connection to	Inverted
CH_DO.11.VALUE	‚A1.T2.A1T2T001.TV.S' / Q4.0 / Temperature Reactor R001 Control signal	no
CH_AI.10.V	A1T2T001(A,2) / CMP_R.21.IN1	

EXERCISES

We are going to apply what we learned in the theory chapters and the step by step instructions to the exercises. We are using the already existing multi-project from the step by step instructions (PCS7_SCE_0106_R1009.zip) and expand it.

The PID controller set up in the step by step instructions is to be expanded. First, we are going to implement the required settings for a setpoint/actual value follow up.

TASKS

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

1. A few inputs of the controller function block have to be wired as described in Table 48 below for the setpoint/actual value follow-up when the manual mode of the plant is activated.
2. The changeover from the automatic mode to the manual/follow-up mode takes place by means of the input 'AUT_L'. When this input is set to 0, it indicates the manual/follow-up mode; when set to 1, correspondingly to the automatic mode. There are two application cases where the follow-up mode is needed: Either the plant is in the manual mode, or it is in the automatic mode. However, there is no setpoint for the controller.

   Implement a combinational circuit that takes input 'AUT_L' to the correct state when the plant is put into the manual mode. Also, the combinational circuit is to have an additional unwired input that activates the follow up. This input is used later in the automatic mode.

Table 48: Inputs of Controller Function Block

Input	**Value**
LMN_SEL	0
LIOP_MAN_SEL	1
SPBUMPON	1
SPEXTSEL_OP	0