

Learn-/Training Document

Siemens Automation Cooperates with Education (SCE) | As of Version V9 SP1

PA Module P01-08 SIMATIC PCS 7 – Sequential control systems

siemens.com/sce



Matching SCE Trainer Packages for this Learn-/Training Document

- SIMATIC PCS 7 Software Package V9.0 (set of 3) Order No.: 6ES7650-0XX58-0YS5
- SIMATIC PCS 7 Software Package V9.0 (set of 6) Order No.: 6ES7650-0XX58-2YS5
- SIMATIC PCS 7 Software Upgrade Packages (set of 3) Order No.: 6ES7650-0XX58-0YE5 (V8.x→ V9.0)
- SIMIT Simulation Platform with Dongle V10 (contains SIMIT S & CTE, FLOWNET, CONTEC libraries) 2500 simulation tags Order No.: 6DL8913-0AK00-0AS5
- Upgrade SIMIT Simulation Platform V10 (contains SIMIT S & CTE, FLOWNET, CONTEC libraries) from V8.x/V9.x Order No.: 6DL8913-0AK00-0AS6
- Demo Version SIMIT Simulation Platform V10
 Download
- SIMATIC PCS 7 AS RTX Box (PROFIBUS) only in combination with ET 200M for RTX Order No.: 6ES7654-0UE23-0XS1
- ET 200M for RTX Box (PROFIBUS) only in combination with PCS 7 AS RTX Box Order No.: 6ES7153-2BA10-4AB1

Note that these trainer packages are replaced with successor packages when necessary. An overview of the currently available SCE packages is available at: <u>siemens.com/sce/tp</u>

Continued training

For regional Siemens SCE continued training, get in touch with your regional SCE contact siemens.com/sce/contact

Additional information regarding SCE

siemens.com/sce

Information regarding use

The SCE Learn-/Training Document for the integrated automation solution Totally Integrated Automation (TIA) was prepared for the program "Siemens Automation Cooperates with Education (SCE)" specifically for training purposes for public educational facilities and R&D institutions. Siemens does not guarantee the contents.

This document is to be used only for initial training on Siemens products/systems, which means it can be copied in whole or part and given to those being trained for use within the scope of their training. Circulation or copying this Learn-/Training Document and sharing its content is permitted within public training and advanced training facilities for training purposes.

Exceptions require written consent from the Siemens. Send all related requests to <u>scesupportfinder.i-ia@siemens.com</u>.

Offenders will be held liable. All rights including translation are reserved, particularly if a patent is granted or a utility model or design is registered.

Use for industrial customer courses is explicitly not permitted. We do not consent to commercial use of the Learn-/Training Document.

We wish to thank the TU Dresden, particularly Prof. Dr.-Ing. Leon Urbas and the Michael Dziallas Engineering Corporation and all other involved persons for their support during the preparation of this Learn-/Training Document.

Table of contents

1	Go	al	6
2	Pre	erequisite	6
3	Re	quired hardware and software	7
4	Th	eory	8
	4.1	Theory in brief	8
	4.2	Continuous and sequential control systems	9
	4.3	Structure of sequential function charts	9
	4.4	Design of sequential control systems	12
	4.5	Interaction of sequential control and logic control systems	13
	4.6	Protective functions and operating modes in sequential control systems	14
	4.7	Sequential control systems in PCS 7	15
	4.8	References	16
5	Та	sk	17
6	Pla	anning	
7	Lea	arning objective	19
8	Str	uctured step-by-step instructions	20
	8.1	Creating and configuring an SFC	20
	8.2	Editing the sequential function chart	23
	8.3	Editing properties of steps and transitions	28
	8.4	Editing steps and transitions	
	8.4	.1 Transition: Init_OK	
	8.4	.2 Step: EductB003toR001	
	8.4	Transition: L001 >= 350 ml	44
	8.4	.4 Step: Heating25°CStir	45
	8.4	.5 Transition: T001 >= 25°C	46
	8.4	.6 Step: Wait	47
	8.4	.7 Step: EduktB002inR002	47
	8.4	.8 Transition: L002 >= 200 ml	49
	8.4	.9 Step: EductB001toR002	49

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

	8.4.	10	Transition: L002 >= 350 ml	51
	8.4.	11	Step: Stir	51
	8.4.	12	Transition: Parallel_OK	53
	8.4.	13	Step: R002toR001	55
	8.4.	14	Transition: L002 <= 50 ml	56
	8.4.	15	Step: Heating28°C	57
	8.4.	16	Transition: T001 >= 28°C	58
	8.4.	17	Step: R001toProdB001	58
	8.4.	18	Transition: L001 <= 50 ml	60
	8.4.	19	Step: END	60
	8.5	Con	npiling and downloading objects	64
	8.6	Test	ting the SFC	69
	8.7	Che	cklist – step-by-step instruction	77
9	Exe	rcises	S	78
	9.1	Tasl	k	78
	9.2	Che	cklist – exercise	78
10	Add	litiona	al information	79

Sequential control systems

1 Goal

The students can successfully implement sequential control systems using sequential function charts. They understand the structure and mode of functioning of sequential function charts and are familiar with corresponding design methods. Their knowledge regarding operating modes and protective measures will be expanded for sequential control systems. The students understand the interaction between the programs of basic automation and sequential control systems. They know how to create sequential controls in **PCS 7**.

2 Prerequisite

This chapter builds on chapter 'Functional safety'. To implement this chapter, you can use an existing project from the previous chapter or the archived project 'p01-07-exercise-r1905-en.zip' provided by SCE. The download of the project(s) is stored on the SCE Internet for the respective module.

The (optional) simulation for the SIMIT program can be retrieved from the file 'p01-04-plantsimv10-r1905-en.simarc'. It can be run in demo mode.

3 Required hardware and software

- 1 Engineering station: Requirements include hardware and operating system (for further information, see Readme on the PCS 7 installation DVD)
- 2 SIMATIC PCS 7 software V9 SP1 or higher
 - Installed program packages (contained in SIMATIC PCS 7 Software Trainer Package):
 - Engineering \rightarrow PCS 7 Engineering
 - Engineering \rightarrow BATCH Engineering
 - Runtime \rightarrow Single Station \rightarrow OS Single Station
 - Runtime \rightarrow Single Station \rightarrow BATCH Single Station
 - Options \rightarrow SIMATIC Logon
 - Options \rightarrow S7-PLCSIM V5.4 SP8
- 3 Demo Version SIMIT Simulation Platform V10



3 SIMIT V10 or higher

4 Theory

4.1 Theory in brief

Sequential control systems allow for time-discrete or event-discrete execution of sequential and parallel processes. They are used to coordinate various continuous functions as well as to control complex process sequences. Dependent on defined states or events, changes of operating state and changes of state are produced in existing logic control systems and the desired sequential flow is thus implemented. Sequential control systems are implemented with one or more *sequential function charts*.

A sequential function chart is an alternating stringing together of *steps*, which trigger certain actions and *transitions*, which initiate the change from one step to another as soon as the corresponding *step enabling condition* is met. Each sequential function chart has exactly one *start step* and one *end step* as well as any number of intermediate steps that are each interconnected by arrows and interposed transitions. The charts may also generate feedback through loops within the sequential function chart. Likewise, they can contain simultaneous or alternative branches. However, in this case the design must ensure that the sequence does not contain uncertain or inaccessible parts.

Formal design methods using *state diagrams* or *Petri nets* are particularly suitable for designing a sequential control system. State diagrams are easy to learn, allow for automatic error diagnostics and can be easily converted to many existing programming languages for sequential control systems. However, the design of parallel structures is not possible because state diagrams have only exactly one active state.

Petri nets are considerably more complex and pose a greater mathematical challenge. However, all structures that are permitted in sequential control systems can be modeled and extensively analyzed. Thus, necessary properties of the control system can be verified formally. Petri nets also permit easy implementation in sequential control systems.

Parameter assignment and activation of lower-level logic control systems by sequential control systems takes place by the setting of corresponding global control signals. The effect of these control signals can be temporary or permanent, direct or delayed. Like logic control systems, sequential control systems must support various operating modes, in which manual control of transitions and temporary or permanent interruption of process sequences, in particular, must be possible. In addition, process-specific protective functions are implemented through sequential control systems.

Sequential control systems are implemented in *PCS* **7** with **sequential function charts (SFC)**. SFCs provide efficient operating mode management, high controllability through multiple switching modes and extensive parameter assignment capability through various execution options. In *PCS* **7**, SFCs and CFCs interact and are linked by means of process values and control values. The interaction behavior can also be controlled in detail.

4.2 Continuous and sequential control systems

Within the scope of basic automation, various logic control systems are developed that each implements a limited, clearly defined function. The functions continuously process input signals and generate corresponding output signals. Activation and parameter assignment of the functions is possible through various control signals. To implement complex process sequences, such as manufacturing specifications for products (*recipes*), the various functions must be coordinated and activated at the right time with the right parameters. This task can be realized using sequential control systems.

Sequential control systems enable step-by-step, event-discrete processing of sequential and parallel processes using **sequential function charts** (also called **sequencers**). They produce changes of operating state and changes of state dependent on defined states or events in the existing logic control systems and thus implement the desired sequential flow. Sequencers function charts are also called **Sequential function charts**.

4.3 Structure of sequential function charts

A sequential function chart is an alternating sequence of *steps* and *transitions*. The individual steps activate certain actions in each case. Transitions control the change from one step to another.

The first step of a sequential function chart is called the *start step*. It is the explicit entry point into the sequence and is always executed for that reason. The last step of a sequential function chart is correspondingly called the *end step*. It is the only step in the sequence that does not have a follow-on transition. After the end step is executed, the sequential function chart is terminated (completed) or the execution restarts. The latter case is also referred to as sequence loop.

Steps and transitions are interconnected by directed graphs. A step can be connected to several follow-on transitions; the reverse is also possible. A transition is enabled when all upstream steps are active and the step enabling condition is met. In this case, first the directly preceding steps are deactivated and then the direct follow-on steps are activated.

The simplest form of a sequential function chart is the unbranched sequence. Each step is followed by exactly one transition, which in turn is followed by exactly one follow-on step. Thus, a purely sequential process sequence is realized. Figure 1 shows the corresponding graphic basic elements.



Figure 1: Basic elements of a sequential function chart

Loops within the sequential function chart occur when a cyclical execution within the sequence is possible through the stringing together of multiple steps. The sequence loop is a special case of a loop in which all steps are run through cyclically.

Jumps are another option for structuring sequential function charts. When a jump label is reached, processing is continued with the step to which the jump label points. Jumps within the sequential function chart can also produce loops. Because this type of structuring is difficult to understand, jumps should be avoided if possible.

From the process view it is often necessary to react differently to various events during program runtime. In this case, a step has several alternative follow-on steps. This structure is called an *alternative branch*. The step is connected to each possible follow-on step via a separate transition. To ensure that no more than one of these transitions will be enabled at a given time (and the branches are truly alternative), the transitions should be mutually locked or explicitly prioritized. Otherwise, the transitions are evaluated from left to right in most control systems and the first transition whose step enabling condition is met is enabled.

Figure 2 shows the general structure of an alternative branch with two branches. It is represented by bordering horizontal single lines with protruding ends. As you can see, alternative branches always start and end with transitions.



Figure 2: Alternative and simultaneous branches in sequential function charts

Another frequent requirement is that a step be followed by several follow-on steps that are to be processed simultaneously. In this case, the output step has exactly one transition that activates several follow-on steps simultaneously. This structure is called a *simultaneous branch*. The follow-on steps of the individual branches are then processed independent of one another and then joined again. All branches end in a joint transition. Only when all branches are processed completely and the step enabling condition of the follow-on transition is met can the joint follow-on steps be activated.

The sequence of a simultaneous branch with two branches is also shown in Figure 2. The branches are represented by bordering horizontal double lines with protruding ends. As you can see, simultaneous branches always begin and end with actions.

A special control problem is the possibility of creating faulty sequential function charts through the unfavorable use of jumps and branches. Three possible cases have to be distinguished.

- Uncertain sequence: A sequential function chart contains a structure whose accessibility is not ensured through the defined sequential flow.
- Partial deadlock: A sequential function chart contains an inner loop that is no longer exited. Although the steps within this loop can be executed, the steps outside the loop cannot. Portions of the sequential function chart are thus not accessible.
- Total deadlock: A sequential function chart contains a structure for which there is no permissible step enabling condition. In this case the sequential function chart remains permanently in one state and all other steps are inaccessible.

Such structures are not permitted in sequential function charts and have to be ruled out with corresponding formal design methods. Figure 3 shows an example of two sequential function charts with illegal structures.

In the left sequence, it cannot be ensured that step S6 is accessible because, when transition t3 is enabled, the alternative branch after step S3 will prevent the simultaneous branch in transition t4 from being rejoined again. For this reason, the sequence is uncertain. The right sequence, on the other hand, is executed exactly once and then remains in step S4. Because step S2 is not active in this state, the simultaneous branch in transition t3 can no longer be joined. A total deadlock results and step S5 is not accessible.



Uncertain structure

Illegal structure

Figure 3: Uncertain and illegal structures in sequential function charts

4.4 Design of sequential control systems

Numerous formal design methods exist for sequential control systems. In practice, however, the state diagram and Petri net models have proven valuable, in particular.

A state diagram is a connected, directed graph. States are represented as circles and state transitions as arrows that connect exactly two states to one another. In a state diagram, exactly one state is always active at a time. The states can be linked to certain actions. There is the option to assign a certain sequential flow to these actions.

They can be executed once on entering the state or exiting the state, or cyclically as long as the state is active. State transitions can be subject to transition conditions.

State diagrams can be structured hierarchically and linked to each other. State diagrams are considered easy to learn and enable automatic error diagnostics, for example, through pair, time or state monitoring. These diagrams can be easily converted to many existing programming languages for sequential control systems.

Petri nets are particularly suitable for modeling concurrent processes. A Petri net consists of places and transitions that are connected to each other through arrows. This also results in a directed graph. A place is represented with a circle and a transition with a rectangle (often reduced to a bar). Active places are indicated with tokens; they are represented with a dot within the circle for the corresponding place.

In contrast to function diagrams, the state in a Petri net is determined by the number of active places in the entire network. The dynamics of the system is modeled through the movement of the tokens within the network. The meaning of the places and transitions for the modeled process (i.e. the *semantics* of the Petri net) is not defined and has to be specified depending on the application. Petri nets whose semantics have been specified are called *interpreted Petri nets* (*IPN*). For the control design, *signal interpreted Petri nets* (*SIPN*) are generally usable.

Petri nets can be extensively analytically tested. They also allow easy conversion to existing programming languages for sequential control systems. There are numerous expansions for Petri nets that are optimized for specific applications or provide for more exact process modeling. For this reason, Petri nets can become rather complex, which makes them more demanding as a design method. Nevertheless, based on their structural similarity to sequential function charts and their ability to model parallel processes, Petri nets offer clear advantages.

The design method that is used depends ultimately on the requirements of the design task as well on the preference of the developer. Additional information is provided in the pertinent technical literature.

4.5 Interaction of sequential control and logic control systems

As described above, each step in the sequential function chart can be assigned certain actions. In general, these actions consist of the parameter assignment and activation of logic control systems for which corresponding control signals are suited.

Process and control signals that are used by sequential function charts must be declared globally so that they are equally available to the programs of the sequential control and logic control systems. The signals are usually listed in a symbol table. Control signals generally act as long as the corresponding step is active. For implementing more complex function sequences, however, it is also possible to vary the processing of a control signal itself (latching or non-latching, time delayed or limited).

Usually, process-specific functions are implemented with sequential control systems while logic control systems implement all device-specific functions.

4.6 Protective functions and operating modes in sequential control systems

As in the case of single control functions, adequate protective functions and operating modes have to be implemented for sequential control systems. Sequential control systems must be manually operable even under fault conditions. Corresponding operating modes must be provided in the control system for this.

- Automatic mode: The action of the sequential function chart is executed when the upstream transition is enabled.
- Manual mode: The action of the sequential function chart is initiated by the operator even if the upstream transition is not enabled.
- Mixed mode: The action of the sequential function chart is executed when the upstream transition is enabled or the operator has initiated it. Alternatively, both the initiation by the operator and the enable of the upstream transition can be required.

Using manual mode prevents permanent blocking of the sequential control system as a consequence of a fault. The mixed mode allows for manual interruption of the process for testing or commissioning purposes. The step enabling conditions of all transitions of the sequential control system must be expanded accordingly.

Sequential function charts must be able to respond to faults in the controlled devices. This requires continuous fault monitoring. This monitoring serves to detect and reports faults in the controlled devices. It enables automated safeguarding of the plant by automatically stopping the sequential function chart in the event of a fault. In addition, it has to be possible for the operator to stop and abort a sequential function chart when there is a fault.

In both cases, corresponding protective functions must be activated in order to bring the plant to a safe state. If the sequence is stopped, it is necessary to ensure a safe, process-compatible resumption of the sequence, even after an extended interruption. Process-specific protective functions are implemented in sequential control systems, such as sequential interlocking of several devices in the event of a fault state in the process.

4.7 Sequential control systems in PCS 7

Sequential control systems are implemented in *PCS* **7** with *sequential function charts (SFC)*. They contain the sequencers and define their sequence topology, the conditions for the transitions and the actions of the steps. The start conditions and sequence properties can be defined and prioritized separately for each sequencer.

In addition, pre-processing and post-processing steps can be defined that are executed once prior to or after execution of the sequencer.

Operating modes and switching modes

The behavior of a sequential control system in *PCS* **7** depends on the selected operating mode, the specified switching mode, its current operating state and the execution options. For sequential control systems, two different operating modes can be selected.

- Auto: The program controls the sequence.
- Manual: The operator controls the sequence with commands or by changing the execution options.

In manual mode, the commands Start, Stop, Hold, Complete, Abort, Resume, Restart, Reset and Error are available to the operator for operating the sequential control system manually. The behavior of a sequential function chart when advancing from active steps to follow-on steps can be controlled through different switching modes, depending on the selected operating mode.

- Switching mode T: The sequential control system runs process-controlled, which means automatically. When a transition is enabled, the predecessor steps are deactivated and follow-on steps are activated. (T = Transactions)
- Switching mode O: The sequential control system runs operator-controlled, which means manually. The transition is enabled by an operator command. Each follow-on transition of an active step automatically sets an operator prompt. (O = Operator)
- Switching mode T or O: The sequential control system runs process-controlled or operator-controlled. The transition can be enabled either through an operator command or through a fulfilled step enabling condition.
- Switching mode T and O: The sequential control system runs process-controlled and operator-controlled. The transition is only enabled when an operator command has been issued and the step enabling condition is met.
- Switching mode T/T and O: In this switching mode, it can be specified individually for each step whether the sequential control system runs process-controlled or operatorcontrolled. As a result, hold points can be defined in the sequential control system in test mode. (T/T = Test Transactions)

In *Auto mode*, only the switching modes *T* and *T/T* and *O* can be selected. The operating mode of the sequential control system shows the current state in the sequence and the resulting operating behavior. A corresponding operating state logic defines the possible state, the permissible transitions between the states and the transition conditions for a state change. *PCS* **7** defines separate operating state logic for sequential control systems and sequential function charts. It is possible to have sequential function charts run dependent on the state of the sequential control system.

Execution options

By using execution options, the runtime behavior of a sequential control system can be controlled. For example, it can be specified whether a sequential control system is executed once or cyclically (*cyclic operation* option) or whether the actions of the active step will actually be executed (*command output* option). In addition, a time monitoring can be activated for the individual steps in a sequential function chart, which reports a step error if the time is exceeded (*time monitoring* option).

Interaction behavior

In **PCS 7**, CFCs and SFCs interact by means of process values and control values. These values are linked to each other using the desired signals, either from the global symbol table or by specifying the absolute signal address. It is possible to control the processing of the control signals using the features of the SFC. In the **SFC Library**, **PCS 7** provides pre-assembled sequential function charts for various standard scenarios. These templates can be used and adapted to current projects.

4.8 References

- [1] Seitz, M. (2008): Speicherprogrammierbare Steuerungen. Hanser Fachbuchverlag
- [2] Wellenreuther, G. and Zastrow, D (2002): Automatisieren mit SPS: Theorie und Praxis. Vieweg+Teubner
- Uhlig, R (2005): SPS Modellbasierter Steuerungsentwurf f
 ür die Praxis: Modellierungsmethoden aus der Informatik in der Automatisierungstechnik. Oldenbourg Industrieverlag
- SIEMENS (2014): Process Control System PCS 7: SFC for SIMATIC S7 (V9.0).
 A5E41356233-AB (<u>support.industry.siemens.com/cs/ww/en/view/109755020</u>)

5 Task

In accordance with the recipe in chapter 'Process Description', a sequential function chart is to be created and programmed.

- First, 350 ml is to be drained from educt tank =SCE.A1.T1-B003 to reactor =SCE.A1.T2-R001 and at the same time 200 ml is to be drained from educt tank =SCE.A1.T1-B002 to reactor =SCE.A1.T2-R002.
- 2. When the filling of reactor =SCE.A1.T2-R001 is finished, the filled liquid is to be heated to 25 °C with the stirrer switched on.
- 3. When the filling of reactor =SCE.A1.T2-R002 is finished, 150 ml of educt A from educt tank =SCE.A1.T1-B001 is to be added to reactor =SCE.A1.T2-R002. When this is complete, the stirrer of reactor =SCE.A1.T2-R002 is to be switched on 10 s later.
- 4. When the temperature of the liquid in reactor =SCE.A1.T2-R001 has reached 25 °C, the mixture from reactor =SCE.A1.T2-R002 is to be pumped to reactor =SCE.A1.T2-R001.
- 5. The mixture in reactor =SCE.A1.T2-R001 is now to be heated to 28 °C and then drained to product tank =SCE.A1.T3-B001.

6 Planning

All the needed actuators and signals are already implemented and interlocked according to the safety requirements. They only still have to be appropriately linked to the sequential control system.

The sequential function chart named in the task description only then has to be converted into steps and transitions. The following specifics are known:

- Task 1 executes parallel steps (both reactors can operate independent of each other)
- Task 3 names a time condition
- Task 4 merges the two parallel processing steps (in Reactor R001)

The following tables give an overview of the utilized actuators and must be initialized like this:

Block	SP_LiOp	SP_ExtLi	SP_IntLi
	.Value	.Value	.Value
A1T2T001\control_A1T2T001	TRUE	TRUE	FALSE

Block	ModLiOp .Value	AutModLi .Value	ManModLi .Value
A1T1X006\valve_A1T1X006	TRUE	TRUE	FALSE
A1T1S003\pump_A1T1S003	TRUE	TRUE	FALSE
A1T2X003\valve_A1T2X003	TRUE	TRUE	FALSE
A1T2S001\stirrer_A1T2S001	TRUE	TRUE	FALSE
A1T2T001\control_A1T2T001	TRUE	TRUE	FALSE
A1T2S004\pump_A1T2S004	TRUE	TRUE	FALSE
A1T2X008\valve_A1T2X008	TRUE	TRUE	FALSE
A1T2S003\pump_A1T2S003	TRUE	TRUE	FALSE
A1T3X001\valve_A1T3X001	TRUE	TRUE	FALSE
A1T1X005\valve_A1T1X005	TRUE	TRUE	FALSE
A1T1S002\pump_A1T1S002	TRUE	TRUE	FALSE

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

Block	ModLiOp .Value	AutModLi .Value	ManModLi .Value
A1T2X005\valve_A1T2X005	TRUE	TRUE	FALSE
A1T1X004\valve_A1T1X004	TRUE	TRUE	FALSE
A1T1S001\pump_A1T1S001	TRUE	TRUE	FALSE
A1T2X004\valve_A1T2X004	TRUE	TRUE	FALSE
A1T2S002\stirrer_A1T2S002	TRUE	TRUE	FALSE

7 Learning objective

In this chapter, students learn the following:

- Creating and editing SFCs
- Establishing links between SFCs and CFCs
- Establishing links between SFCs and the addresses from the symbol table
- Testing of SFC programs

8 Structured step-by-step instructions

8.1 Creating and configuring an SFC

1. To start, create a new SFC in the plant view in folder 'A1_multipurpose_plant'. (\rightarrow A1_multipurpose_plant \rightarrow Insert New Object \rightarrow SFC)

SIMATIC Manager - SCE_PCS7_1	1P window Help				<u>_ </u>
			< No Filter >		
SCE_PCS7_MP (Plant View) C	:\Program Files (x86)\	\STEP7 \S7Proj	SCE_PCS7_MP\SCE_	_mp	
SCE_PCS7_MP 	A1H001	En 12_reaction	n 🛛 🖌 🛐 I 3_product Mathematika Mathema	_tanks [፼] I4_rinsing -∱- Picture(2)	
⊡ 📴 T1_educt_tanks	Cut Ctr	·l+X			
🕀 🖻 educt_tank B(Copy Ctr	·I+C			
Eren educt_tank Bl Eren educt_tank Bl Eren educt_tank Bl	Paste Ctr Delete Del	1+V I			
⊡ eactor R001	Insert New Object	Hi	erarchy Folder		
⊡ ⊡ T3_product_tanks	Access Protection	► Ci	=c		
	Print	, SF	C		
⊡… T4_rinsing ⊕… rinsing_tank B ⊡ ∳ SCE_PCS7_Lib	Charts Plant Hierarchy Process Tags	Pi Pi Re	ditional Document cture eport		
Generations	Models	► Ec	uipment Properties		
eductTank	Rename F2 Object Properties Alt	+Return Ed	uipment Property		
Process tag types					
Inserts SFC at the cursor position.					1.

2. Then open the object properties of the SFC. (\rightarrow SFC(1) \rightarrow Object Properties)

	SIMATIC Manager - SCE_PC57_MP					<u> </u>
Fi	le Edit Insert PLC View Options Wi	ndow Help				
	🛛 🗅 🌽 🎛 🐖 👗 🖻 🛍 🕍		🗄 🏢 🔁 < N	o Filter >	🖸 🏏 號 🎟 🖷	🗏 🔲 😽
	SCE_PC57_MP (Plant View) C:\Prog	gram Files (x86)\\	STEP7\S7Proj\SCE	PCS7_MP\SCEMP		
	⊡- SCE_PCS7_MP	Mathematical T1_educt_tanks Mathematical Mathematical	In T2_reaction Im A1H002	ia T3_product_tanks Ist A1H003	T4_rinsing	
ш	🗄 🧰 Shared Declarations	-A- Picture(2)	(100)	(HOL	Open Object	Ctrl+Alt+O
ш	🖻 🛅 A1_multipurpose_plant					
ш	🖻 🙆 T1_educt_tanks				Open External View	
ш	🕀 🚘 educt_tank B001				Cut	Ctrl+X
ш	😟 🚘 educt_tank B002				Copy	Ctrl+C
ш	🖻 🔁 educt_tank B003				Paste	Ctrl+V
ш	⊡ 🙆 T2_reaction					
ш	teactor HUU1				Delete	Del
ш	H eactor RUU2				PLC	•
ш	Brief Product tank 8001					
ш	product_tank B001				Access Protection	• •
ш	E M T4 rinsing				Print	•
ш	⊡ 💼 insing tank B001					
ш	🖻 🚫 SCE_PCS7_Lib				Charts	•
ш	🗄 💼 Shared Declarations				Plant Hierarchy	•
ш	🖻 🚰 Models				Rename	F2
ш	🕀 🍖 eductTank				Object Properties	Alt+Return
ш	🕀 🎯 product_tank					
ш	庄 🎡 reactor					
	Emergence Process tag types					
Dis	splays properties of the selected object for ed	liting.				

 Next, under General, the name is changed to 'SFC_Product01' and a comment as well as the author is entered. (→ General → SFC_Product01)

Properties SFC chart		×		
General AS Operating Parameters OS Version				
Name:	SFC_Product01			
Project path:	SCE_PCS7_Prj\AS1\CPU 414-3 DP\S7 Program(1)\Charts			
Technological path:	SCE_PCS7_Prj\A1_multipurpose_plant			
Storage location of project:	C:\Program Files (x86)\SIEMENS\STEP7\S7Proj\SCE_PCS7_MP\S	č		
Author:	plt-admin	1		
Date created:	05/02/2019 02:47:19			
Last modified:	05/02/2019 02:47:19			
Comment:	SFC for produce of product 1			
Write-protected				
ОК	Cancel Help			

 Set the operating parameters as follows; they can also be changed later in online mode. (→ AS Operating Parameters)

Properties SFC chart 🔀					
General AS Operating Parameters OS Version					
Defaults Step control mode:	Operating mode:				
Command output Cyclic operation	SFC startup after CPU restart				
Start options Autostart Use default operating parameters when SF	C chart starts				
ОК	Cancel Help				

5. On the OS tab, it is important that the check box be selected so that the SFC will be available later for visualization. Accept all parameters in the "Version" tab with OK. (\rightarrow OS \rightarrow Transfer chart to OS for visualization \rightarrow Version \rightarrow OK)

Properties SFC chart	Properties SFC chart	X
General AS Operating Parameters OS Version	General AS Operating Parameters OS Version	
	Version: 0.0001	
	Data version: V9.0 SP1	
Create block icon:		
✓ Transfer chart to OS for visualization		
OK Cancel Help	Cancel H	leip

Note:

In the "Create block icon" input field, you can specify the block icon in WinCC that is to be displayed for this block. You can thus select different variants for the same block type, if present. Leaving the field blank results in the standard display.

8.2 Editing the sequential function chart

 Now, double-click the sequential function chart 'SFC_Product01' in SIMATIC Manager to open it. (→ SFC_Product01)



2. You can now build the sequential control system in the SFC Editor with the following toolbar buttons.



3. For this task, you will need additional steps and transitions. To insert both, select the button and select the location where you want to insert them. ($\rightarrow +$)

SFC - SFC_Product01	
SFC Edit Insert CPU Debug View Options Window Help	
_ D 😂 🎒 X 🖻 💼 🎞 🗉 6% 🚵 省 📲 🔀 X 🗖 - 🔆 Q Q 🖷 🚍	🖿 N?
▶ 中 韓 母 异 abl	
SFC_Insert Step + Transition 7_Prj\A1_multipurpose_plant	
START 1 END	
	•
Switches to the mode for inserting steps + transitions.	

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

SFC - SFC_Product01	
SFC Edit Insert CPU Debug View Options Window Help	
_ D 🛩 🚳 % 🖻 💼 🎞 6% 🏜 🕲 🦉 🔀 🖾 🐼 - 🔆 🔍 🔍 🖷 🚍	🖽 k?
▶ 中 韓 母 ֆ abl	
SFC_Product01 SCE_PCS7_Prj\A1_multipurpose_plant	
	•
Press F1 for help.	

Note:

- The numbering of steps and transition has no bearing on the order of execution of the sequential function chart.
- 4. After three steps and transitions have been inserted in this way, click the $\frac{1}{100}$ icon to insert a simultaneous branch. Then select the location where you want to insert it. ($\rightarrow \frac{1}{100}$)



For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

SFC - SFC_Product01	
SFC Edit Insert CPU Debug View Options Window Help	
D 😅 🛃 ち 🖻 🔳 🔲 💷 🕅 🏜 🕅 📲 🞇 🔀 🗟 🔍 🤤 🚭	🖿 🖿 💦
▶ 中 韓 母 ֆ abl	
🕸 SFC_Product01 SCE_PCS7_Prj\A1_multipurpose_plant	
START	
· · · · · · · · · · · · · · · · · · ·	
4	
3	
5	
END	
Press F1 for help.	

5. Additional steps and transitions are then also to be inserted in the simultaneous branch. Therefore, change back to the + button and insert the additional steps and transitions. (\rightarrow

Therefore, change back to the \top button and insert the additional steps and transitions. (\rightarrow



For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

6. Then click the **h** icon to edit normally. (\rightarrow **h**) SFC - SFC_Product01 SFC Edit Insert CPU Debug View Options Window Help D 😅 🎒 | X 🖻 💼 | 🗖 📰 | 68 🎪 | 🦹 | 🎬 🔀 | 🖾 🐼 - 🏡 | Q, Q, | 🖷 🖃 🔟 | 🎌 中幸 다 다 타 abl k Selection On Jct01 -- SCE_PCS7_Prj\A1_multipurpose_plant <u>- 0 ×</u> ٠ START 6 7 8 9 8 10 11 3 2 4 3 5 END • Switches from the insert to the selection mode.

8.3 Editing properties of steps and transitions

1. The procedure for changing properties of a step is shown next. To do this, right-click on the step and select Object Properties. (\rightarrow Step 6 \rightarrow Object Properties)

SFC - SFC_Product01		
SFC Edit Insert CPU Debug View O	options Window Help	
🗋 🖆 🎒 👗 🖻 💼 🗖 🖬	n 🤹 🔊 🐮 🔀 🖾 🗖 - 🖗 🖉	R, Q, 🖷 🖿 🖿 M?
እ 字 藝 🗘 🛱 🛼 abl		
SFC_Product01 SCE_PCS7_Prj\A1	L_multipurpose_plant	-OX
		▲
	START	
	Cut Ctrl+X	
	6 Delete Del	
	Copy Object Properties Ctrl+S	hift+C
	8 Paste Object Properties Ctrl+5	hift+V
	Object Properties Alt+Re Select Jump Destination	eturn
	Go To Jump Destination	
	10 11	
	5	
Press F1 for help.		//.

2. In the object properties, each step is to be assigned a name and a comment for better clarity. (\rightarrow Name: EductB003toR001 \rightarrow Comment: educt tank B003 to reactor R001 \rightarrow Close)

Properties - 6 SCE_	PCS7_Prj\A1_multipurpose_plant\\SFC_Product01	×
General Initialization	Processing Termination	
Name:	EductB003toR001 Number: 6 Confirmation	
- Run times		
Minimum:	Maximum:	
Comment:	educt tank B003 to reactor R001	
OS comment:		
Acknowledgment information:		
Close App	bly ← ↑ ↓ → Print Browse Groto Help	

3. Confirm the prompt to save changes with "Yes". (\rightarrow Yes)

Assign pa	Assign parameters to step/transition (253:10030)			
<u> </u>	The properties of step 6 have been changed! Do you want to save?			
(Yes	No	Cancel		

Similar to the properties of the steps, the properties of the transitions can also be changed.
 To do so, right-click the transition and select Object Properties. (→ 1 → Object Properties)



 Here you also change the name and the comment first. (→ Name: Init_OK → Comment: all initial conditions are fulfilled → Close)

Properties - 1 SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01				
General Condition OS	Comment			
Name:	Init_OK Number: 1			
Comment:	all initial conditions are complied	A V		
Close Apply	← ↑ ↓ → Print Browse Gio to	Help		

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

6. Save the change. (\rightarrow Yes)



Repeat the previous steps until the SFC looks like this. It is important to also enter a minimum run time of 10 seconds for the 'Stir' step. (→ Run times Minimum: T#10s)



For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved. p01-08-sequential-function-chart-v9-tud-0719-en.docx

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

Properties - 11 9	5CE_PC57_Prj\A1_multipur	pose_plant\\SFC_Product01	×
General Initialization	on Processing Termination		
Name:	Stir	Number: 11 Confirmation	
Run times			٦
Minimum:	T#10s	Maximum:	
Comment:	Stir for min 10 s	×	
OS comment:		×	
Acknowledgmer information:	nt	×	
Close /		Print Browse Gio to Help	
SFC - SFC Product01	1	-	
SFC Edit Insert CPU	Debug View Options Window H	elp	
	💼 🎞 🗉 0% 🏜 🕲 📲	X X A + Q Q = = □ N	
▶ 中 韓 다 타	‡ _∓ abl		
	;		
SFC_Product01 S	SCE_PCS7_Prj\A1_multipurpose_p	lant[
SFC_Product01 5	SCE_PCS7_Prj\A1_multipurpose_p	lant[
SFC_Product01 5	SCE_PCS7_Prj\A1_multipurpose_p	al conditions are complied	
Base SFC_Product01 S	START	al conditions are complied	
EductB003toR001	educt tank B003 to reactor R001	al conditions are complied	
EductB003toR001	START	al conditions are complied EductB002toR002 educt tank B002 to reactor R00	
EductB003toR001	educt tank B003 to reactor R001 el R001 min 350 ml	al conditions are complied EductB002toR002 educt tank B002 to reactor R00 U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to reactor R00	J× ▲
EductB003toR001 L001 >= 350 ml leve Heating25°CStir T001 >= 25 °C tem	educt tank B003 to reactor R001 el R001 min 350 ml heating reactor R001 to 25 °C	al conditions are complied EductB002toR002 educt tank B002 to reactor R00 U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to reactor R00 U002 >= 350 ml level R002 min 350 ml	12 12
EductB003toR001 L001 >= 350 ml leve Heating25°CStir T001 >= 25 °C tem Wait	educt tank B003 to reactor R001 el R001 min 350 ml heating reactor R001 to 25 °C auting for parallel process	al conditions are complied EductB002toR002 educt tank B002 to reactor R00 U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to reactor R00 U002 >= 350 ml level R002 min 350 ml Stir stir for min 10 s	12 12
SFC_Product01 S EductB003toR001 L001 >= 350 ml Heating25°CStir T001 >= 25 °C Wait	educt tank B003 to reactor R001 el R001 min 350 ml heating reactor R001 to 25 °C sperature R001 min 25 °C	al conditions are complied EductB002toR002 educt tank B002 to reactor R00 U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to reactor R00 U002 >= 350 ml level R002 min 350 ml Stir stir for min 10 s Stir [11] T#10s stir for min 10 s	JZ 12
SFC_Product01 S EductB003toR001 L001 >= 350 ml Heating25°CStir T001 >= 25 °C Wait	educt tank B003 to reactor R001 el R001 min 350 ml heating reactor R001 to 25 °C nperature R001 min 25 °C waiting for parallel process	al conditions are complied EductB002toR002 educt tank B002 to reactor R00 U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to reactor R00 U002 >= 350 ml level R002 min 350 ml U002 >= 350 ml level R002 min 10 s Stir stir for min 10 s Stir for min 10 s parallel branches finish	J∑ ▲

For unrestricted use in educational / R&D institutions. $\textcircled{\mbox{\sc only}}$ Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

8.4 Editing steps and transitions

8.4.1 Transition: Init_OK

 The actual function of the sequential function chart is implemented in this section. The 'START' step should not contain any instructions. Therefore, start by double-clicking on the transition 'Init_OK'. (→ Init_OK)



2. Select the 'Condition' tab and add the initialization conditions by clicking on 'Browse'. (\rightarrow Condition \rightarrow Browse)

Properties - In	it_OK SCE_PCS7_Prj\A1_multipurpose_p	lant\\SFC_Product01	×
General Cond	lition OS Comment		
1	f(x)	f(x)	
2	f(x)	f(x)	
3	f(x)	f(x) &	
4	f(x) 💌	f(x)	
5	<u>f(x)</u>	f(x)	
6	f(x) 💌	f(x)	^{&}
7	f(x)	f(x)	8
8	f(x)	f(x) &	
9	f(x) 💌	f(x)	
10	<u>f(x)</u>	f(x)	J I
~ 1			
Close	ApplyT	Browse Go to	Help

3. A window opens for adding I/Os or symbols.

rowse - SCE_PCS7_Prj\A1_multipurpose_plant						
Plant View Component View Runtime Groups S	ymbols					
A1_multipurpose_plant\\			I∕Os <filtere< td=""><td>ed></td><td></td><td></td></filtere<>	ed>		
 A1_multipurpose_plant A1H001 A1H002 A1H003 SFC_Product01 T1_educt_tanks T2_reaction T3_product_tanks T4_rinsing 		Data t		CFC i	SFC a	
Close Apply Filter		Back	Up	date	Help	

4. Now, select the 'Symbols' tab, select the symbol of the main power switch 'A1.A1H001.HS+-.START' and click 'Apply'. The symbol is entered on the left side of the first condition. (\rightarrow Symbols \rightarrow A1.A1H001.HS+-.START \rightarrow Apply)

Plant View Component View	Runtime G	roups Sy	mbols					
		Syn	nbols					
Symbol 🛛	Address	Data	Comment					
A1.A1H001.HS+START A1.A1H002.HS+OFF	I 0.0 I 0.1	BOOL BOOL	Main power emergency s	switch multipurpose plant witch OFF				
1.A1H003.HS+.LOC 1.T1.A1T1L001.LSA+.SA+	I 0.2 I 70.0	BOOL	locia operationali lovel monitor	on mode switch ing educt tank B001 switchpo	bint high			
(1.T1.A1T1L002.LSA+.SA+ (1.T1.A1T1L002.LSA+.SA+ (1.T1.A1T1L002.LSASA-	I 70.2 I 70.3	BOOL	le Properti	ies - Init_OK SCE_PCS	7_Prj\A1_m	ultipurpose_plant\\SFC_I	Product01	
1.T1.A1T1L003.LSA+.SA+ 1.T1.A1T1L003.LSASA-	I 70.4 I 70.5	BOOL BOOL	le Genera	OS Comment	ffer -	-	f(x)	
A1.T1.A1T1S001.SV.C A1.T1.A1T1S002.SO+.O+	Q 3.0	BOOL BOOL	pi 2		f(x) -		f(x)	
1.T1.A1T1S002.SV.C 1.T1.A1T1S003.SO+.O+	Q 3.1	BOOL	pi 3		f(x)		<u>f(x)</u> &	
Close	Apply	Filter.	5		f(x)		f(x)	
			6		f(x)	-	<u>f(x)</u>	&
			7		f(x)	<u>- </u> -	f(x) f(x) 8	&
			9		f(x)	-	f(x)	
			10		f(x)		f(x)	Ī
				se Apply (+ I I ⇒ I	Brint Browse	Gete	Help

5. Next, enter '1' or 'TRUE' on the right side of the first condition, so that the next steps will be processed only when the plant is switched on. Apply this value. ($\rightarrow 1 \rightarrow Apply$)

Properties - Init_OK SCE_PCS7_	_Prj\A1_multipurpose_plant\\SFC_Pro	duct01 🗵
General Condition OS Comment		
1		ffeel
2		- (w)
2		- (w) e
5		- (w)
6	<u>f(x)</u>	<u>f(x)</u>
7	<u>f(x)</u>	&
8	<u>f(x)</u>	<u>f(x)</u> &
9	<u>f(x)</u>	<u>f(x)</u>
10	f(x)	<u>f(x)</u>
	(. L. L	
Close Apply + 1	Print Browse	Gio to Help

 Now add conditions so that the EMERGENCY STOP is unlocked and the local operation is deactivated. Close the dialog. (→ A1.A1H002.HS+-.OFF → 1 → A1.A1H003.HS+-.LOC → 0 → Close)

Properties - Init_OK SCE_PCS7_	Prj\A1_multipurpose_plant\\SFC_Pro	duct01 🗵
General Condition OS Comment		
1 "A1.A1H001.HS+START"	f(x) = TRUE	f(g)
2 "A1.A1H002.HS+OFF"	f(x) = TRUE	f(x)
3 "A1.A1H003.HS+LOC"	f(x) = 💌 FALSE	1 8 - 1
4	f(x)	f(x)
5	f(x)	f(x)
6	f(x)	f(x) &
7	f(x)	f(x) &
8	f(x)	<u>f(x)</u> &
9	f(x)	f(x)
10	f(x)	f(x)
Close Apply + 1	↓ → Print Browse	Go to Help

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

8.4.2 Step: EductB003toR001

1. Next, open the 'EductB003toR001' step. (→ EductB003toR001)



2. Select the 'Initialization' tab and click on 'Browse'. (\rightarrow Initialization \rightarrow Browse)

Properties	- EductB003toR001 SCE_PCS7_Pr	j∖A	1_multipurpose_plant\\SFC_Produ	ct01 🗵
General	Initialization Processing Termination			
1	7	:=		f(x) 🔺
2	ī	:=		f(x)
3	ī	:=		f(x)
4	7	:=		f(x)
5	7	:=		f(x)
6	7	:=		f(x)
7 🔽	ī	:=		f(x)
8	7	:=		f(x)
9	Ī	:=		f(x)
10 🔽	7	:=		f(x) 👻
Close		Prir	t Go to	Help

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.
In the 'Plant View' tab of the selection window select the valve block 'Valve_A1T1X006' in CFC 'A1T1X006'. (→ A1_multipurpose_plant → T1_educt_tanks → educt_tank B003 → A1T1X006 → valve_A1T1X006)

Browse - SCE_PCS7_Prj\A1_multipurpose_	plar	nt				×		
Plant View Component View Runtime Groups	s	ymbols						
educt_tank B003\\A1T1X006\valve_A1T1X0	I/Os <filtered></filtered>							
⊡ 🙆 A1_multipurpose_plant		Name 🛆	Data type	I	CFC i	SFC 🔺		
		AcAcquireId	DWORD	IN				
± - 🛺 A1H002		AcPriority	BYTE	IN				
A1H003		AcquireId	DWORD	IN				
SEC Product01		AcRequest	BYTE	IN				
		AcSafeNotM	DWORD	IN				
En Bardust task 2001		AcSafeOrMsk	DWORD	IN				
		AcWaitTmMax	REAL	IN				
E educt_tank BUU2		AutModLi	STRUCT	IN				
educt_tank B003		AutModOp	BOOL	IN				
🕀 🕀 🕀 🕀		BatchEn	BOOL	IN				
🕀 🕀 🕀 🕀 🕀		BatchiD	DWORD CTDINC[22]	IN				
🖻 🙀 A1T1X006		Batchivame Dre Dret	STRING[32]	IN				
Error		Close Aut	STRUCT	IN				
		CloseForce	STRUCT	IN				
		CloseLocal	STRUCT	IN	~			
		CloseMan	BOOL	IN	^			
		CSE	STRUCT	IN	x			
		CtrlChnST	STRUCT	IN	x			
		EN	BOOL	IN				
Output		EnAcquire	BOOL	IN				
Permit		EventTsIn	ANY	IN				
Protect		ExtMsg1	STRUCT	IN				
valve_A1T1X006		ExtMsg2	STRUCT	IN		-		
E T2_reaction	-	•	·					
Close Apply F	ilter	Bac	k Up	date		Help		

4. First, set the 'ModLiOp' input to '1', so that the valve can still only be controlled via interconnections or SFC. Because the 'ModLiOp' input is of data type 'STRUCT', you must first right-click to open the shortcut menu and click 'Open Structure' there. (→ ModLiOp → Open Structure)

Plant View Component View Runtime Groups Symbols educt_tank B003\\A1T1X006\valve_A1T1X006. I/Os <filtered> Image: All-multipurpose_plant Image: All-multipurpose_plant Image: All-multipurpose_plant Image: All-multipurpose_plant Image: All-multipurpose_plant</filtered>
educt_tank B003\\A1T1X006\valve_A1T1X006. I/Os <filtered> A1_multipurpose_plant A1H001 A1H001 A1H002 A1H002 EdVa108 ANY A1H003 FaultExt STRUCT SFC_Product01 FbkOpen STRUCT IN educt_tank B001 Feature STRUCT IN educt_tank B002 STRUCT IN Intl educt_tank B003 CocalOp BOOL IN educt_tank B003 STRUCT IN Intlock educt_tank B003 STRUCT IN IN intlock STRUCT IN IN intlock STRUCT IN IN intocalio STRUCT IN<</filtered>
Image: Second structure Name Data type I CFC i SFC Image: Second structure A1H001 Image: Second structure
A1H001 ExtVa108 ANY IN A1H002 A1H003 FaultExt STRUCT IN SFC_Product01 FbkOpen STRUCT IN x B Educt_tanks Feature STRUCT IN x B Educt_tank B001 Intl_En BOOL IN x B Educt_tank B002 Intlock STRUCT IN x B Educt_tank B003 Intlock STRUCT IN x Calli STRUCT IN x Intlock IN B Educt_tank B003 Intlock STRUCT IN x Calli STRUCT IN x Intlock IN Intlock IN B A1T1L003 Intlock STRUCT IN IN IN IN IN B A1T1X006 Interlock Montor Open Chart IN IN IN B Error FbkOpen Montor Open Chart IN IN IN Interlock MonTiDyr <td< th=""></td<>
Image: Althom of the second struct of the
A1H003 SFC_Product01 Image: SFC_Product02 Image: SFC_Product03 Image: SFC_Product03 I
SFC_Product01 Image: SFC_Product01 Image: SFC_Product01 Image: SFC_Product_tanks
Feature STRUCT IN Image: Struct stark sector of the start sta
Feature2 STRUCT IN Intl_En BOOL IN Intl_En BOOL IN Intl_En BOOL IN Intl_Calli STRUCT IN Intl_Calli STRUCT IN Intl_En BOOL IN Intlock STRUCT IN Intelock ManModUi STRUCT IN ManModOp BOOL IN MondLiOp STRUCT IN MonSafef Show Block Intelock
Intl_En BOOL IN Intl_En BOOL IN Intlock STRUCT IN Intelock ManModOp BOOL IN Intelock Monitor Open Chart IN Intelock MonTiDyr Show Block Intelock
Image: educt_tank budz Imtock STRUCT IN x Imtock ManModLi STRUCT IN Imtock Monitor Open Chart Imtock
Encore Enco
ATTILUU3 LocalOp DOOL IN LocalSetting INT IN LocalSetting INT IN ManModLi STRUCT IN ManModOp BOOL IN Montor Open Chart Interlock MonTiDyr Show Block
ATTISU03 ATTISU03 ATTISU03 ATTISU03 ATTISU03 ATTISU03 ATTISU03 ManModLi STRUCT IN ManModOp BOOL IN ModLiOp STRUCT IN MonModOp BOOL IN MonMod Struct IN Mon Struct IN Mon Mod Struct IN Mon Mod Struct IN Mon Struct I
ATTIX006 ManModOp BOOL IN FokClose FokOpen FokOpen Interlock MonTiDyr Show I/O
Error ModLiOp STRUCT IN FbkClose Monitor Open Chart - FbkOpen MonSafeF Show Block Interlock MonTiDyr Show I/O
FbkClose Monitor Open Chart FbkOpen MonSafef Show Block Interlock MonTiDyr Show I/O
FbkOpen MonSafeF Show Block Interlock MonTiDyr Show I/O
Interlock Mon TiDyr Show I/O
OR_Local MonTiSta Apply I/O
OR_OOS MS_RelO
Output MsgEvid1 Open Structure
Pemit MsgLock
Protect NoFokciose BOOL IN
valve A1T1X006 Occupied BOOL IN
Close Apply Filter Back Update Help

5. The structure dialog opens and you select 'Value' of data type BOOL. With Apply, your selection is applied on the left side of the first instruction. (\rightarrow Value \rightarrow Apply)

Structu	re - ModLiO)p			×
Name	Data type	1/0	CFC interconnection	SFC access (writing)	Comment
Value	BOOL	IN			Value
ST	BYTE	IN			Signal Status
	lose			Back	Help

 Now, enter "1" on the right side of the first instruction. This sets the 'ModLiOp' input to SFC mode. With 'Apply', "1" is automatically replaced with "TRUE". (→ 1 → Apply)

Propert	ies	- EductB003toR001 SCE_PCS7_Prj	i\A	1_multipurpose_plant\\SFC_Product	01 🗵
Genera	al	nitialization Processing Termination			
1	M	1X006\valve_A1T1X006.ModLiOp.Value	:=	1	
2	₽		:=		(x)
3	∣⊡		:=		(x)
4	∣⊡		:=		(x)
5	₽		:=		(x)
6	₽		:=		(x)
7	₽		:=		(x)
8	₽		:=		(x)
9	∣⊡		:=		(x)
10	₽		:=		(x) 👻
Clo	se		Prir	nt Gio to H	lelp

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

7. Then add inputs 'AutModLi' = '1' and 'ManModLi' = '0', so that the valve is set to automatic mode. (\rightarrow AutModLi \rightarrow 1 \rightarrow ManModLi \rightarrow 0 \rightarrow Apply)

Pro	operti	ies	- EductB003toR001 SCE_PCS7_Prj	\A	1_multipurpose_plant\\SFC_Product	01 🗵
	Genera		nitialization Processing Termination			
	1		1X006\valve_A1T1X006.ModLiOp.Value	:=	TRUE	
	2		1X006\valve_A1T1X006.AutModLi.Value	:=	TRUE	
	3		X006\valve_A1T1X006.ManModLi.Value	:=	FALSE	(x)
	4			:=	f	(x)
	5			:=	f	(x)
	6			:=	f	(x)
	7			:=	f	(x)
	8			:=	f	(x)
	9			:=	f	(x)
	10			:=	f	(x) 👻
	Clo	se		Prir	nt Gro to H	lelp

8. The same must now be done for Pump A1T1S003 and Valve A1T2X003 because they are also involved in filling Reactor R001 from Educt tank B003. Then, change to the 'Processing' tab. (A1T1S003 → ModLiOp.Value = 1 → AutModLi.Value = 1 → ManModLi.Value = 0 → Apply → A1T2X003 → ... → Apply → Processing)

Propert	ies	 EductB003toR001 SCE_PCS7_Prj 	\A	1_multipurpose_plant\\SFC_Produc	t01 🗵
Genera	al	nitialization Processing Termination			
	1				
1	M	1X006\valve_A1T1X006.ModLiOp.Value	:=	TRUE	
2	☑	A1T1X006\valve_A1T1X006.AutModLi.Vi	:=	TRUE	f(x)
3	∣⊡	A1T1X006\valve_A1T1X006.ManModLi.\	:=	FALSE	f(x)
4	∣⊡	1S003\pump_A1T1S003.ModLiOp.Value	:=	TRUE	f(x)
5]⊡	1S003\pump_A1T1S003.AutModLi.Value	:=	TRUE	f(x)
6	∣⊡	S003\pump_A1T1S003.ManModLi.Value	:=	FALSE	f(x)
7	∣⊡	2X003\valve_A1T2X003.ModLiOp.Value	:=	TRUE	f(x)
8	∣⊡	2X003\valve_A1T2X003.AutModLi.Value	:=	TRUE	f(x)
9	∣⊡	:X003\valve_A1T2X003.ManModLi.Value	:=	FALSE	f(x)
10	∣⊡		:=		f(x) 👻
Clo	se		Pri	nt Browse Gio to	Help

Properties - EductB003toR001 SCE_PC	57_Prj\A1_multipurpose_plant\\SFC_Pro	duct01 🗵
General Initialization Processing Terminat	ion	
1 1	:=	
2 🔽	:=	f(x)
3 🔽	:=	f(x)
4	:=	f(x)
5 🔽	:=	f(x)
6 🔽	:=	f(x)
7	:=	f(x)
8 🔽	:=	f(x)
9 🔽	:=	f(x)
	:=	f(x) 👻
Close Apply 🔶 🕇 🦊 •	Print Browse Go to	Help

9. The instructions for opening the valves and starting the pump are entered in the 'Processing' tab. For the valves, set inputs 'OpenAut.Value' = '1' and 'CloseAut.Value' = '0'. For the pump, use inputs 'StartAut.Value' = '1' and 'StopAut.Value' = '0'. (A1T1X006 → ... → A1T1S003 → ... → A1T2X003 → ... → Apply → Close)

Propert	ies	 EductB003toR001 SCE_PCS7_Prj 	i\A	1_multipurpose_plant\\SFC_Produc	t01 🗵
Genera	al I	Initialization Processing Termination			
1	⊡	[1X006\valve_A1T1X006.OpenAut.Value	:=	TRUE	f(x)
2		F1X006\valve_A1T1X006.CloseAut.Value	:=	FALSE	f(x)
3		T1S003\pump_A1T1S003.StartAut.Value	:=	TRUE	f(x)
4		T1S003\pump_A1T1S003.StopAut.Value	:=	FALSE	f(x)
5	⊡	F2X003\valve_A1T2X003.OpenAut.Value	:=	TRUE	f(x)
6	⊡	F2X003\valve_A1T2X003.CloseAut.Value	:=	FALSE	f(x)
7			:=		f(x)
8			:=		f(x)
9	∣⊡		:=		f(x)
10	⊡		:=		f(x) 👻
Clos	se		Pri	nt Gro to	Help

For unrestricted use in educational / R&D institutions. $\ensuremath{\textcircled{O}}$ Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

10. The instructions that are to be executed when completing the step are then entered in 'Termination'. You must close the valves and the pump here. The valves and the pump could also be reset back to manual mode and operator mode at this point. However, it is recommended that this be saved for the 'END' step. Proceed by copying the instruction from 'Processing' to 'Termination' and then inverting the values ('TRUE' -> 'FALSE' and vice versa). To copy and paste, select the numbers in front of the instructions and open the shortcut menu.

Proper	ties - EductB003to	oR001 :	SCE_PCS7_Prj 1	j\A1_mult	ipurpose_plant\	\SFC_Pro	oduct01	×				
Gener	al Initialization Pro	ocessing	Termination									
	Г1X006\valve_A	A1T1X006	OpenAut.Value	:= TRUE			f(x)					
2	✓ [1X006\valve_/	A1T1X006	CloseAut.Value	:= FALSE			f(x)					
3	▼ T1S003\pump_	A1T1S003	StartAut.Value	:= Tf Pro	perties - EductB	003toR0	01 SCI	:_PCS7_P	rj\A1_multip	urpose_plant	:\\SFC_Proc	iuct01 🖄
4	T1S003\oumo	A1T1S003	StopAut.Value	:= F/ G	eneral Initialization	Proces	sing Ter	mination				
5		Cun+2	nAut.Value	= <u>T</u> f	1				.=			f(x)
6	Cut	Ctrl+X	eAut.Value	:= F/	2 Undo	Ct	rl+Z					
7	Paste	Ctrl+V		:= 🗌 👘	3 Cut	Ct	rl+X					
8	Delete	Ctrl+D		:= 🗌 👘	4 Copy	ct	rl+⊂	1				f(x)
9	Select All	Ctrl+A		:= 🗌 👘	5 Delete	Ct	rl+V					f(x)
10	Insert Empty Lin	e Ctri			_	-			- 11		न	f(x)
	Copy Action(s)	Ctri Pro	operties - Edu	ictB003to	R001 SCE_PCS	7_Prj\A1	l_multip	urpose_p	lant\\SFC_Pr	oduct01	<u>ــــــــــــــــــــــــــــــــــــ</u>	f(x)
Cl	Paste Action(s)	Ctr 🚺	General Initializ	ation Proc	cessing Terminatio	n					. —	
			1 🔽 [1X00	06\valve_A	1T1X006.OpenAut.	Value :=	FALSE			f(x) 🔺		f(x)
			2 V [1X00	06\valve_A	1T1X006.CloseAut.	Value :=	TRUE					
			3 VT1S0)03\pump_A	1T1S003.StartAut.	Value 🚦	FALSE			f(x)		
			4 V T1S0	03\pump A	TT1S003.StopAut.	Value -	TRUE			f(s)	Gete	Help
			5 V [2X00	03\valve A	1T2X003.OpenAut.	Value 🚦	FALSE			f(s)		
			6 V [2X00	 D3\valve_A	1T2X003.CloseAut.	Value -	TRUE			f(s)		
			7 17	_						f(x)		
						-1				f(x)		
			9 7			— Ci				f(x)		
						-1				f(x) = 1		
						1				<u></u>		
			Close	Apply] ← ★ ↓ →	Prin	E	Browse.	Go to	Help		
											1	

11. Then, close the properties dialog for the 'EductB003toR001' step. The SFC Editor shows the 'Init_OK' transition and the 'EductB003toR001' and 'Stir' steps grayed out because instructions already exist there. (→ Close)

Propert	ies	EductB003toR001 SCE_PCS7_Pr	j\A1_multipurpose_plant\\SFC_Produ	ct01 🖄
Genera	al I	nitialization Processing Termination		
1	ান	[1X006\valve_A1T1X006.OpenAut.Value	FALSE	figi 🔺
2		F1X006\valve_A1T1X006.CloseAut.Value	·= TRUE	
3		T1S003\pump_A1T1S003.StartAut.Value	·= FALSE	f(s)
4		T1S003\pump_A1T1S003.StopAut.Value	·= TRUE	f(s)
5		F2X003\valve_A1T2X003.OpenAut.Value	;= FALSE	f(x)
6		F2X003\valve_A1T2X003.CloseAut.Value	:= TRUE	f(x)
7			:=	f(x)
8			:=	f(x)
9			:=	f(x)
10			:=	f(x) 👻
Clo	se		Print Browse Go to	Help
SFC - S	FC_	Product01 ert CPU Debug View Options Window He	p	
	e	- X 🖻 💼 🗖 📼 🕼 🏜 🕅 📲 🗄	* X ⊠⊲⊲ = = = ×	
	蕐	다 卧 ţabl		
SFC_	Prod	uct01 SCE_PCS7_Prj\A1_multipurpose_pl	ant	
Sta SFC_	Prod	uct01 SCE_PCS7_Prj\A1_multipurpose_pl	nt	
Sta SFC_I	Prod	uct01 SCE_PCS7_Prj\A1_multipurpose_pl	int	
Ba SFC_	Prod	uct01 SCE_PCS7_Prj\A1_multipurpose_pl	nt	
Sh SFC_I	Prod	uct01 SCE_PCS7_Prj\A1_multipurpose_pla	ont	
Ba SFC_I	Prod	start SCE_PCS7_Prj\A1_multipurpose_pla	conditions are complied	
	Prod	START	conditions are complied	
Educt	Prod	START	conditions are complied	LUX
Educt	Prod B003	START	conditions are complied EductB002toR002 educt tank B002 to rea	×
Educt	B003	START Init_OK all initia	conditions are complied EductB002toR002 educt tank B002 to rea	×
Educt	B003	START Init_OK all initia stoR001 educt tank B003 to reactor R001 so ml level R001 min 350 ml	conditions are complied EductB002toR002 educt tank B002 to rear	ctor R002
Educt Heat	Prod B003	START Int_OK all initia stoR001 educt tank B003 to reactor R001 so mi level R001 min 350 ml so CStir heating reactor R001 to 25 °C	conditions are complied EductB002toR002 educt tank B002 to read L002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to read	ctor R002
Educt	Prod B003	START Int_OK all initia STARO1 educt tank B003 to reactor R001 Socstir heating reactor R001 to 25 °C	conditions are complied EductB002toR002 educt tank B002 to read L002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to read	ctor R002
Educt Heat	Prod B003	START START Int_OK all initia toR001 educt tank B003 to reactor R001 S0 ml level R001 min 350 ml S°CStir heating reactor R001 to 25 °C temperature R001 min 25 °C	conditions are complied EductB002toR002 educt tank B002 to read L002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to read L002 >= 350 ml level R002 min 350 ml	ctor R002
Educt	Prod B003 11>= : 101>= 01>= Wai	START START Int_OK all initia START Int_OK all initia STOR001 educt tank B003 to reactor R001 SO ml level R001 min 350 ml So CStir heating reactor R001 to 25 °C So c temperature R001 min 25 °C	conditions are complied EductB002toR002 educt tank B002 to read U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to read U002 >= 350 ml level R002 min 350 ml Stir stir for min 10 s	ctor R002
Educt	Prod B003 11>= 1 11>= 1 01>= Wa	START START Int_OK all initia STOR001 educt tank B003 to reactor R001 so ml level R001 min 350 ml S°CStir heating reactor R001 to 25 °C t waiting for parallel process	conditions are complied EductB002toR002 educt tank B002 to read L002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to read L002 >= 350 ml level R002 min 350 ml Stir stir for min 10 s	ctor R002
Educt	Prod B003 11>= ; 11>= ; 11>= ; Wai	START START Init_OK all initia STOR01 educt tank B003 to reactor R001 so ml level R001 min 350 ml S°CStir heating reactor R001 to 25 °C S°C temperature R001 min 25 °C t waiting for parallel process RUN /	conditions are complied EductB002toR002 educt tank B002 to real U002 >= 200 ml level R002 min 200 ml EductB001toR002 educt tank B001 to real U002 >= 350 ml level R002 min 350 ml Stir stir for min 10 s	ctor R002

For unrestricted use in educational / R&D institutions. $\textcircled{\mbox{$\odot$}}$ Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

8.4.3 Transition: L001 >= 350 ml

Now, open the 'L001 >= 350 ml' transition. Enter the condition that the level of Reactor R001 is greater than or equal to 350 ml. (→ L001 >= 350 ml → Condition → Browse → ...Reactor R001\\A1T2L001\Level_A1T2L001.PV_Out → right-click → Open Structure → Value → >= → 350 → Apply → Close)

Properties - L001 >= 350 ml SCE_	PCS7	7_Prj\	A1_multipurpose_plant\\	SFC_	Product01	×
General Condition OS Comment						
			[1		1
evel_A1T2L001.PV_Out.Value	f(x) >	= 🔳	350.0	$f(\mathbf{x})$		
2	f(x)			f(x)		
3	f(x) <			f(x)	&	
4	f(x) <	=		f(x)		
5	f(x)	=		f(x)		
6	f(x)	-		f(x)	_ ^{&} -	
7	f(x)	•		f(x)		8
8	f(x)	•		f(x)	&	
9	f(x)	•		f(x)		
10	f(x)	•		f(x)		Ŧ
Close Apply + 1	+	→	Print Browse	Got	to H	lelp

8.4.4 Step: Heating25°CStir

 In the "Heating25°CStir" step, again add inputs 'ModLiOp', 'AutModLi' and 'ManModLi' for 'stirrer_A1T2S001' and 'control_A1T2T001' in the 'Initialization' tab. For the control, switch the setpoint setting to SFC mode 'SP_LiOp' = '1' and to external setpoint setting 'SP_ExtLi' = '1' and 'SP_IntLi' = '0'. (→ Heating25°CStir → 'Initialization' → ...)

Propert	ies	 Heating25°CStir SCE_PCS7_Prj\A 	1_	_multipurpose_plant\\SFC_Product	01 🗵
Genera	al	nitialization Processing Temination			1
1		F2S001\stimer_A1T2S001.ModLiOp.Value ;	:=	TRUE	f(x)
2		2S001\stirrer_A1T2S001.AutModLi.Value :	:=	TRUE	f(x)
3	∣⊡	2S001\stirrer_A1T2S001.ManModLi.Value :	:=	FALSE	$f(\mathbf{x})$
4	∣⊡	2T001\control_A1T2T001.ModLiOp.Value :	:=	TRUE	$f(\mathbf{x})$
5	⊽	T001\control_A1T2T001.AutModLi.Value	:=	TRUE	f(x)
6	∣⊡	001\control_A1T2T001.ManModLi.Value :	:=	FALSE	f(x)
7	∣⊡	2T001\control_A1T2T001.SP_LiOp.Value :	:=	TRUE	f(x)
8	∣⊡	2T001\control_A1T2T001.SP_ExtLi.Value :	:=	TRUE	f(x)
9]⊡	2T001\control_A1T2T001.SP_IntLi.Value :	:=	FALSE	f(x)
10	∣⊽		:=		f(x) 👻
Clo	se		^o rir	t Gro to	Help

2. Then switch to the 'Processing' tab and add the I/Os and values shown. These properties are used to start the stirrer and assign the setpoint 25 °C to the control.

Propert	ies	- Heating25°CStir SCE_PCS7_Prj\/	A1 _	_multipurpose_plant\\SFC_Product01	1 🗵
Genera	al I	nitialization Processing Termination			
	ı — 1				
1		T2S001\stirrer_A1T2S001.StartAut.Value	:=	TRUE	f(x)
2		T2S001\stirrer_A1T2S001.StopAut.Value	:=	FALSE	f(x)
3		T2T001\control_A1T2T001.SP_Ext.Value	:=	25.0	f(x)
4			:=	f	f(x)
5			:=	f	f(x)
6			:=	f	f(x)
7			:=	f	(x)
8			:=	f	f(x)
9			:=	f	f(x)
10			:=	f	(x) 🖵
Clo	se		Prir	nt Browse Gio to H	Help

3. In the 'Termination' tab, stop the stirrer and set the setpoint to 0 °C. Then close the dialog.

Propert	ies	 Heating25°CStir SCE_PCS7_Prj\/ 	A1	_multipurpose_plant\\SFC_Product0	1 🗵
Genera	al	nitialization Processing Termination			
1		T2S001\stimer_A1T2S001.StartAut.Value	:=	FALSE	f(x)
2		T2S001\stirrer_A1T2S001.StopAut.Value	:=	TRUE	f(x)
3		T2T001\control_A1T2T001.SP_Ext.Value	:=	0.0	f(x)
4			:=		f(x)
5			:=		f(x)
6			:=		f(x)
7			:=		f(x)
8			:=		f(x)
9			:=		f(x)
10			:=		f(x) 🖵
Clo	se		Prir	nt Gio to	Help

8.4.5 Transition: T001 >= 25°C

1. Now you set the properties for the 'T001 >= 25°C' transition. For this you need the measured temperature. (\rightarrow T001 >= 25°C \rightarrow Condition \rightarrow ...\T2_Reaction\reactor R001\\A1T2T001\ In_A1T2T001 \rightarrow PV_Out \rightarrow Value \rightarrow Apply \rightarrow >= \rightarrow 25.0 \rightarrow Apply \rightarrow Close)

Properties - T001 >= 25 °C SCE_P	CS7_Prj\A1_multipurpose_plant\\SF	C_Product01
General Condition OS Comment		
1 1\ln_A1T2T001.PV_Out.Value	f(x) >= 🔻 25.0	f(x)
2	(k) T	f(x)
3	(x) T	f(x) &
4	× · · · · · · · · · · · · · · · · · · ·	f(x)
5	(x) •	f(x)
6	(x) v	f(x) & -
7	× · · · · · · · · · · · · · · · · · · ·	f(x) 9
8	(x) T	f(x) &
9	(x) •	f(x)
10	(x) •	f(x) I
Close Apply + 1	↓ → Print Browse	Gio to Help

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

8.4.6 Step: Wait

1. The 'Initialization', 'Processing' and 'Termination' tabs remain empty in the 'Wait' step. This is indicated by the tabs not being highlighted.

Propert	ies - Wait SCE_PCS7_Prj\A1_multipu	rpose_plant\\SFC_Product01	×
Genera	Initialization Processing Termination		
1		:=	f(x)
2		:=	f(x)
3		(=	f(x)
4		:=	f(x)
5		:=	f(x)
6		:=	f(x)
7		:=	f(x)
8	v	:=	f(x)
9	v	:=	f(x)
10		:=	f(x) 👻
Clo	se Apply ← ↑ ↓ →	Print Browse Go to	Help

8.4.7 Step: EduktB002inR002

 Now complete the simultaneous branch. Start with the 'EductB002toR002' step and utilize the figures below. (→ EductB002toR002)

Propert	ies	 EductB002toR002 SCE_PCS7_Prj 	\A	1_multipurpose_plant\\SFC_Produc	t01 🗵
Genera	al	nitialization Processing Termination			
1		1X005\valve_A1T1X005.ModLiOp.Value	:=	TRUE	f(x)
2		1X005\valve_A1T1X005.AutModLi.Value	:=	TRUE	f(x)
3		X005\valve_A1T1X005.ManModLi.Value	:=	FALSE	f(x)
4		1S002\pump_A1T1S002.ModLiOp.Value	:=	TRUE	f(x)
5		1S002\pump_A1T1S002.AutModLi.Value	:=	TRUE	f(x)
6		S002\pump_A1T1S002.ManModLi.Value	:=	FALSE	f(x)
7		2X005\valve_A1T2X005.ModLiOp.Value	:=	TRUE	f(x)
8		2X005\valve_A1T2X005.AutModLi.Value	:=	TRUE	f(x)
9		X005\valve_A1T2X005.ManModLi.Value	:=	FALSE	f(x)
10			:=		f(x) 👻
Clo	se		Pri	nt Go to	Help

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

Propert	ties	 EductB002toR002 SCE_PCS7_Prj 	i\A	1_multipurpose_plant\\SFC_Produc	:t01 🗵
Gener	al	nitialization Processing Termination			
1	되	F1X005\valve_A1T1X005.OpenAut.Value	:=	TRUE	f(x)
2	P	F1X005\valve_A1T1X005.CloseAut.Value	:=	FALSE	f(x)
3		T1S002\pump_A1T1S002.StartAut.Value	:=	TRUE	f(x)
4	∣⊡	T1S002\pump_A1T1S002.StopAut.Value	:=	FALSE	$f(\mathbf{x})$
5	₽	F2X005\valve_A1T2X005.OpenAut.Value	:=	TRUE	f(x)
6		F2X005\valve_A1T2X005.CloseAut.Value	:=	FALSE	f(x)
7			:=		f(x)
8			:=		f(x)
9			:=		f(x)
10	∣⊡		:=		f(x) 👻
Clo	ose		Pri	nt Gio to	Help

Propert	es - EductB002toR002 SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Produ	ct01 🗵
Genera	I Initialization Processing Termination	
1	F1X005\valve_A1T1X005.OpenAut.Value := FALSE	f(x) 🔺
2	T1X005\valve_A1T1X005.CloseAut.Value := TRUE	f(x)
3	T1S002\pump_A1T1S002.StartAut.Value := FALSE	f(x)
4	T1S002\pump_A1T1S002.StopAut.Value := TRUE	f(x)
5	C2X005\valve_A1T2X005.OpenAut.Value := FALSE	f(x)
6	C2X005\valve_A1T2X005.CloseAut.Value := TRUE	f(x)
7	=:	f(x)
8	=:	f(x)
9	=	f(x)
10	=:	f(x) 👻
Clo	se Apply ← ↑ ↓ → Print Browse Go to	Help

8.4.8 Transition: L002 >= 200 ml

1. The 'L002 >= 200ml' transition then looks like this. (\rightarrow L002 >= 200 ml)

Properties - L002 >= 200 ml SCE_PC57_Prj\A1_multipurpose_plant\\!	5FC_Prod	luct01 🗵
General Condition OS Comment		
1 evel_A1T2L002.PV_Out.Value f(x) >= 200.0	f(x)	
2 f(x) -	f(x)	
3 f(x) -	f(x) &	-
4 f(x) -	f(x)	
5 f(x) -	f(x)	
6 f(x) 💌	f(x)	& -
7 f(x) V	f(x)	8
8 f(x) V	f(x) &	
9 f(x) v	f(x)	
10 f(x) 🔽	f(x)	Ŧ
Close Apply ← ↑ ↓ → Print Browse	Gio to	Help

8.4.9 Step: EductB001toR002

1. You must make the following interconnections in the 'EductB001toR002' step.

Propert	ies	5- EductB001toR002 SCE_PCS7_Prj\A1_multipurpose_plant\\S	FC_Product01	×
Genera	al	Initialization Processing Termination		
1	M	1X004\valve_A111X004.ModLiOp.Value := IRUE		
2		1X004\valve_A1T1X004.AutModLi.Value := TRUE	f(x)	1
3		X004\valve_A1T1X004.ManModLi.Value := FALSE	f(x)	
4		TS001\pump_A1T1S001.ModLiOp.Value := TRUE	f(x)	
5		1S001\pump_A1T1S001.AutModLi.Value := TRUE	$f(\mathbf{x})$	
6		S001\pump_A1T1S001.ManModLi.Value := FALSE	$f(\mathbf{x})$	
7		2X004\valve_A1T2X004.ModLiOp.Value := TRUE	$f(\mathbf{x})$	
8		2X004\valve_A1T2X004.AutModLi.Value := TRUE	$f(\mathbf{x})$	
9		/X004\valve_A1T2X004.ManModLi.Value := FALSE	$f(\mathbf{x})$	
10		:=	f(x) 👻	
Clo	se	Apply ← ↑ ↓ → Print Browse G	io to Help	

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

Propert	ies	- EductB001toR002 SCE_PC57_Prj	i\A	1_multipurpose_plant\\SFC_Produc	t01 🗵
Gener	al	nitialization Processing Termination			
1		F1X004\valve_A1T1X004.OpenAut.Value	:=	TRUE	f(x)
2		F1X004\valve_A1T1X004.CloseAut.Value	:=	FALSE	f(x)
3	₽	T1S001\pump_A1T1S001.StartAut.Value	:=	TRUE	$f(\mathbf{x})$
4		T1S001\pump_A1T1S001.StopAut.Value	:=	FALSE	f(x)
5	₽	F2X004\valve_A1T2X004.OpenAut.Value	:=	TRUE	f(x)
6		F2X004\valve_A1T2X004.CloseAut.Value	:=	FALSE	f(x)
7			:=		f(x)
8			:=		f(x)
9			:=		f(x)
10			:=		f(x) 👻
Clo	se		Pri	nt Gio to	Help

Propert	ies	 EductB001toR002 SCE_PCS7_Prj 	i\A	1_multipurpose_plant\\SFC_Produ	ct01	×
Genera	al I	Initialization Processing Termination				
						1
1	☑	F1X004\valve_A1T1X004.OpenAut.Value	:=	FALSE	f(x) 🔺	
2		F1X004\valve_A1T1X004.CloseAut.Value	:=	TRUE	f(g)	
3		T1S001\pump_A1T1S001.StartAut.Value	:=	FALSE	$f(\mathbf{x})$	
4		T1S001\pump_A1T1S001.StopAut.Value	:=	TRUE	$f(\mathbf{x})$	
5		F2X004\valve_A1T2X004.OpenAut.Value	:=	FALSE	$f(\mathbf{x})$	
6	⊡	F2X004\valve_A1T2X004.CloseAut.Value	:=	TRUE	$f(\mathbf{x})$	
7			:=		f(x)	
8			:=		f(x)	
9			:=		f(x)	
10			:=		f(x) 👻	
Clo	se		Pri	nt Gio to	Help	

8.4.10 Transition: L002 >= 350 ml

1. The 'L002 >= 350 ml' transition then looks like this. (\rightarrow L002 >= 350 ml)

Properties - L002 >= 350 ml SCE_PCS7_Prj\A1_multipurpose_plant\\	SFC_P	roduct01 🛛 🗵
General Condition OS Comment		
1 evel_A1T2L002.PV_Out.Value f(x) >= 350.0	f(x)	1
2 f(x) V	f(x)	
3 f(x) -	f(x)	& _
4 f(x) •	f(x)	
5 f(x) V	f(x)	
6 f(x) v	f(x)	&
7 f(x) 💌	f(x)	8
8 f(x) V	f(x)	& .
9 f(x) V	f(x)	
10 f(x) V	f(x)	↓ I
Close Apply ← ↑ ↓ → Print Browse	Gio to	Help

8.4.11 Step: Stir

1. The 'Stir' step has a minimum run time of 10 seconds. You already set this parameter back at the beginning. You must now initialize, start and then stop stirrer_A1T2S002.

Properties - Stir SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01					
General	Initialization Processing Termination				
1	✓ [C2S002\stirrer_A1T2S002.ModLiOp.Value]	:=	JTRUE	<u>f(x)</u>	
2	2S002\stimer_A1T2S002.AutModLi.Value	:=	TRUE	f(x)	
3	2S002\stirrer_A1T2S002.ManModLi.Value	:=	FALSE	f(x)	
4	v	:=		f(x)	
5	v	:=		f(x)	
6	v	:=		f(x)	
7	v	:=		f(x)	
8	v	:=		f(x)	
9	v	:=		f(x)	
10	v	:=		f(x) 👻	
Close		Prir	it Browse Go to	Help	

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

Properties - Stir SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01						
Gener	al	nitialization Processing Termination				
1	اها	T2S002\stimer_A1T2S002_Start Art_Value			10 M	
-		T25002 diller_ATT25002.5tatAdt.Value	.=			
<u></u>]™ I⊏l	123002\stiffer_A1125002.StopAut.value	:=	FALSE	(8)	
3			:=		T(X)	
4	M		:=		<u>f(x)</u>	
5			:=		f(x)	
6			:=		f(x)	
7			:=		f(x)	
8			:=		f(x)	
9			:=		f(x)	
10			:=		f(x) 🖵	
Clo	se		Prir	nt Go to	Help	

Propert	Properties - Stir SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01						
Genera	al 🛛	nitialization Processing Termination					
1		T2S002\stimer_A1T2S002.StartAut.Value	:=	FALSE f(x)			
2		T2S002\stirrer_A1T2S002.StopAut.Value	:=	TRUE			
3			:=	f(x)			
4			:=	f(x)			
5			:=	f(x)			
6			:=	f(x)			
7			:=	f(x)			
8			:=	f(x)			
9			:=	f(x)			
10			:=	f(x)	-		
	_						
Clo	se		Pri	nt Browse Gio to Hel	<u>ه</u>		

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

8.4.12 Transition: Parallel_OK

1. Parameter assignment of the simultaneous branch is now complete. The 'Parallel_OK' transition remains blank. This means that as soon as the 'Wait' and 'Stir' steps are processed, the 'R002toR001' step becomes active.

Properties - Parallel_OK SCE_PCS7_	Prj\A1_multipurpose_plant\\SFC_I	Produc	t01 🗵
General Condition OS Comment			
		e 1	1 1
		T(X)	
2 f(x)		f(x)	
3 f(x)		f(x) &	
4 f(x)		f(x)	
5 f(x)		f(x)	
6 (f(x)		f(x)	&
7 f(x)		f(x)	8
8 f(x)		f(x) &	
9 f(x)		f(x)	
10 f(x)		f(x)	I I
Close Apply 🗲 🕇 🕹	Print Browse	Gio to	Help

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved. p01-08-sequential-function-chart-v9-tud-0719-en.docx

2. The sequential control system looks like this.



8.4.13 Step: R002toR001

1. Next, interconnect the 'R002toR001' step.

Properties - R002toR001 SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01						×
Gener	al	Initialization Processing Termination				
1		2S004\pump_A1T2S004.ModLiOp.Value	:=	TRUE	f(x)	
2		2S004\pump_A1T2S004.AutModLi.Value	:=	TRUE	f(x)	-
3	∣⊡	S004\pump_A1T2S004.ManModLi.Value	:=	FALSE	$f(\mathbf{x})$	
4	∣⊡	2X008\valve_A1T2X008.ModLiOp.Value	:=	TRUE	$f(\mathbf{x})$	
5		2X008\valve_A1T2X008.AutModLi.Value	:=	TRUE	$f(\mathbf{x})$	
6		X008\valve_A1T2X008.ManModLi.Value	:=	FALSE	$f(\mathbf{x})$	
7			:=		f(x)	
8	∣⊡		:=		f(x)	
9			:=		f(x)	
10			:=		f(x) 👻	
Clo	ose		Pri	nt Gro to	Help	

Properties - R002toR001 SCE_PC57_Prj\A1_multipurpose_plant\\SFC_Product01						
Genera	al	Initialization Processing Termination				
1	Ī⊾	T2S004\pump_A1T2S004.StartAut.Value	:=	TRUE	$f(\mathbf{x})$	≜∥∣
2	∣⊡	T2S004\pump_A1T2S004.StopAut.Value	:=	FALSE	$f(\mathbf{x})$	
3	∣⊡	F2X008\valve_A1T2X008.OpenAut.Value	:=	TRUE	$f(\mathbf{x})$	
4	∣⊡	F2X008\valve_A1T2X008.CloseAut.Value	:=	FALSE	$f(\mathbf{x})$	
5			:=		f(x)	
6	∣⊡		:=		f(x)	
7	∣⊡		:=		f(x)	
8	∣⊾		:=		f(x)	
9	∣⊡		:=		f(x)	
10	∣⊡		:=		f(x)	-
Clo	se		Pri	nt Go to	Help	

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

Properties - R002toR001 SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01					
Gener	ral	Initialization Processing Termination			
1		T2S004\pump_A1T2S004.StartAut.Value	:=	FALSE	<u>f(8)</u>
2	<u> </u>	T2S004\pump_A1T2S004.StopAut.Value	:=	TRUE	f(x)
3		[2X008\valve_A1T2X008.OpenAut.Value	:=	FALSE	f(x)
4		C2X008\valve_A1T2X008.CloseAut.Value	:=	TRUE	f(x)
5			:=		f(x)
6			:=		f(x)
7	∎		:=		f(x)
8			:=		f(x)
9	<u> </u>		:=		f(x)
10	_⊡		:=		f(x) 👻
Cl	ose		Pri	nt Go to	Help

8.4.14 Transition: L002 <= 50 ml

1. The 'L002 <= 50 ml' transition must be interconnected as follows.

Properties - L002 <= 50 ml SCE_PCS7_Prj\A1_multipurpose_plant\\S	FC_P	roduct01
General Condition OS Comment		
1 evel_A1T2L002.PV_Out.Value f(g) <= ▼ 50.0	$f(\mathbf{x})$	
2 f(x) v	f(x)	
3 f(x) V	f(x)	&
4 f(x)	f(x)	
5 f(x) -	f(x)	
6 f(x) -	f(x)	— & ⊢
7 f(x) V	f(x)	
8 f(x) V	f(x)	& 8
9 f(x) v	f(x)	
10 f(x) -	f(x)	I I
Close Apply ← ↑ → Print Browse	Got	to Help

8.4.15 Step: Heating28°C

In the 'Heating28°C' step, the control is activated again. Because it is already set to SFC mode and automatic mode, only the setpoint has to be specified. You set this back to 0°C on the Termination tab.

Propertie	es - Heating28°C SCE_PCS7_Prj\A1_	m	ıltipurpose	_plant\\S	FC_Product01	×
General	Initialization Processing Termination					
1	T2T001\control_A1T2T001.SP_Ext.Value	:=	28.0			f(8)
2		:=				f(x)
3		:=				f(x)
4		:=				f(x)
5		:=				f(x)
6		:=				f(x)
7		:=				f(x)
8		:=				f(x)
9		:=				f(x)
10		:=				f(x) 👻
Clos		Pri	nt Bi	rowse	Go to	Help

Propert	ies -	Heating28°C SCE_PCS7_Prj\A1_	m	ultipurpose_plant\\SFC_Product01	×
Genera	al li	nitialization Processing Termination			
		T2T001\control_A1T2T001.SP_Ext.Value	:=	0.0	f(x)
2]⊡[:=		f(x)
3	9		:=		f(x)
4	9		:=		f(x)
5]可[:=		f(x)
6	9		:=		f(x)
7	9		:=		f(x)
8	9		:=		f(x)
9]可[:=		f(x)
10	<u> </u> []		:=		f(x) 👻
Clo	se		Prit	nt Gio to	Help

For unrestricted use in educational / R&D institutions. $\ensuremath{\mathbb{C}}$ Siemens 2020. All rights reserved.

8.4.16 Transition: T001 >= 28°C

1. The condition in the 'T001 >= 28° C' transition now looks like this.

Properties - T001 >= 28 °C SCE_PCS7_Prj\A1_multipurpose_plant\\S	FC_Product01
General Condition OS Comment	
1 1\ln_A1T2T001.PV_Out.Value f(g) >= 28.0	f(x)
2 f(x) •	f(x)
3 f(x) V	f(x) &
4 f(x) -	f(x)
5 f(x) 🔽	f(x)
6 f(x) T	f(x) & -
7 f(x) V	f(x) &
8 f(x) V	f(x) &
9 f(x) V	f(x)
10 f(x) 🔽	f(x)
Close Apply ← ↑ ↓ → Print Browse	Go to Help

8.4.17 Step: R001toProdB001

1. The last step "R001toProdB001" of the recipe fills the content of Reactor R001 into the connected Product tank B001. The interconnections are shown below.

Properties	- R001toProdB001 SCE_PCS7_Prj	A1_multipurpose_plant\\SFC_Produc	:t01 🗵
General	Initialization Processing Termination		
1	2S003\pump_A1T2S003.ModLiOp.Value	:= TRUE	f(x)
2 🔽	2S003\pump_A1T2S003.AutModLi.Value	:= TRUE	f(x)
3 🔽	S003\pump_A1T2S003.ManModLi.Value	:= FALSE	f(x)
4	3X001\valve_A1T3X001.ModLiOp.Value	:= TRUE	f(x)
5 🔽	3X001\valve_A1T3X001.AutModLi.Value	:= TRUE	f(x)
6	X001\valve_A1T3X001.ManModLi.Value	:= FALSE	f(x)
7 🔽		:=	f(x)
8		:=	f(x)
9 🔽		:=	f(x)
10 🔽		:=	f(x) 👻
Close		Print Gio to	Help

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

Propert	ies	- R001toProdB001 SCE_PCS7_Prj\	A1	_multipurpose_plant\\SFC_Product	01 🗵
Gener	al	nitialization Processing Termination			
1	না	T2S003\pump_A1T2S003.StartAut.Value	·	TRUE	ffgi 🔺
2		T2S003\pump_A1T2S003.StopAut.Value	:=	FALSE	
3	🗹	r3X001\valve_A1T3X001.OpenAut.Value	:=	TRUE	f(x)
4	<u> </u>	F3X001\valve_A1T3X001.CloseAut.Value	:=	FALSE	f(x)
5			:=		f(x)
6	∣⊡		:=		f(x)
7	∣⊡		:=		f(x)
8	∣⊡		:=		f(x)
9	∣⊡		:=		f(x)
10	∣⊡		:=		f(x) 👻
Clo	se		Prir	nt Browse Gio to	Help

Propert	ies	 R001toProdB001 SCE_PCS7_Prj¹ 	A1	_multipurpose_plant\\SFC_Product	01	×
Genera	al İ	nitialization Processing Termination				
1	☑	T2S003\pump_A1T2S003.StartAut.Value	:=	FALSE	$f(\mathbf{x})$	
2		T2S003\pump_A1T2S003.StopAut.Value	:=	TRUE	f(x)	
3		F3X001\valve_A1T3X001.OpenAut.Value	:=	FALSE	$f(\mathbf{x})$	
4		F3X001\valve_A1T3X001.CloseAut.Value	:=	TRUE	$f(\mathbf{x})$	
5			:=		f(x)	
6	₽		:=		f(x)	
7	₽		:=		f(x)	
8			:=		f(x)	
9	₽		:=		f(x)	
10			:=		f(x)	-
			_			
Clo	se		Pri	nt Browse Gio to	Help	

8.4.18 Transition: L001 <= 50 ml

 The 'L001 <= 50 ml' transition is the last transition of the recipe. It can be enabled when Reactor R001 is empty (<= 50 ml).

Properties - L001 <= 50 ml SCE_PCS7_Prj\A1_multipur	pose_plant\\SFC_Product0	1 🗵
General Condition OS Comment		
1 evel A1T2L001.PV Out Value f(x) <= 50.0	ffg]	
2 f(x) V	f(x)	
3 f(x) -	f(x) &	1
4 f(x) 💌	f(x)	
5 f(x) v	f(x)	
6 f(x) 💌	&	
7 f(x) V	f(x)	8
8 f(x) v	f(x) &	
9 f(x) -		
10 f(x) V	f(x)	Ŧ
	- 1 - 1	
Close Apply ← ↑ → Print	Browse Go to	Help

8.4.19 Step: END

 In the 'END' step, automatic mode needs to be switched off and manual mode to be switched on again for every valve, pump, stirrer and controller used. (→ step 56). The internal setpoint setting must also be set again for the control. (→ 'Initialization')

Pro	perti	es	- END SCE_PCS7_Prj\A1_multipur	po	se_plant\\SFC_Product01	×
G	ienera		nitialization Processing Termination			
	1	~	:T001\control_A1T2T001.AutModLi.Value	:=	FALSE	f(x)
	2	~	001\control_A1T2T001.ManModLi.Value	:=	TRUE	f(x)
	3	~	2T001\control_A1T2T001.SP_ExtLi.Value	:=	FALSE	f(x)
	4	~	2T001\control_A1T2T001.SP_IntLi.Value	:=	TRUE	f(x)
	5	☑		:=		f(x)
	6	☑		:=		f(x)
	7	☑		:=		f(x)
	8	☑		:=		f(x)
	9	☑		:=		f(x)
	10	☑		:=		f(x) 👻
	Clos	se		Pri	nt Gio to	Help

Block	AutModLi	ManModLi	SP_ExtLi	SP_IntLi
	.Value	.Value	.Value	.Value
A1T2T001\control_A1T2T001	FALSE	TRUE	FALSE	TRUE

2. Then, all utilized pumps, valves, stirrers and controls are set back to operator mode. ('ModLiOp' = '0'). (\rightarrow 'Termination' - 1)

Pro	Properties - END SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01					×
G	ienera		nitialization Processing Termination			
	1	☑	1X006\valve_A1T1X006.ModLiOp.Value	:=	FALSE	f(8)
	2	☑	1S003\pump_A1T1S003.ModLiOp.Value	:=	FALSE	f(x)
	3	☑	2X003\valve_A1T2X003.ModLiOp.Value	:=	FALSE	$f(\mathbf{x})$
	4	☑	F2S001\stirrer_A1T2S001.ModLiOp.Value	:=	FALSE	$f(\mathbf{x})$
	5	☑	2T001\control_A1T2T001.ModLiOp.Value	:=	FALSE	$f(\mathbf{x})$
	6	☑	2S004\pump_A1T2S004.ModLiOp.Value	:=	FALSE	f(x)
	7	☑	2X008\valve_A1T2X008.ModLiOp.Value	:=	FALSE	$f(\mathbf{x})$
	8	☑	2S003\pump_A1T2S003.ModLiOp.Value	:=	FALSE	f(x)
	9	•	3X001\valve_A1T3X001.ModLiOp.Value	:=	FALSE	$f(\mathbf{x})$
	10	☑	1X005\valve_A1T1X005.ModLiOp.Value	:=	FALSE	f(x) 👻
				_		
	Clos	se		Pri	nt Gio to	Help

 $(\rightarrow$ 'Termination' - 2)

Propert	ies - END SCE_PCS7_Prj\A1_multipurpose_plant\\SFC_Product01	×
Genera	al Initialization Processing Termination	
11	IS002\pump_A1T1S002.ModLiOp.Value := FALSE	f(x)
12	2X005\valve_A1T2X005.ModLiOp.Value := FALSE	f(x)
13	TX004\valve_A1T1X004.ModLiOp.Value := FALSE	f(x)
14	IS001\pump_A1T1S001.ModLiOp.Value := FALSE	f(x)
15	ZX004\valve_A1T2X004.ModLiOp.Value := FALSE	f(x)
16	I Γ2S002\stirrer_A1T2S002.ModLiOp.Value := FALSE	f(x)
17	ZT001\control_A1T2T001.SP_LiOp.Value := FALSE	f(x)
18	I =	f(x)
19		f(x)
20	=	f(x) 👻
Clo	se Apply ← ↑ ↓ → Print Browse Go to	Help

Block	ModLiOp .Value	SP_LiOp .Value
A1T1X006\valve_A1T1X006	FALSE	
A1T1S003\pump_A1T1S003	FALSE	
A1T2X003\valve_A1T2X003	FALSE	
A1T2S001\stirrer_A1T2S001	FALSE	
A1T2T001\control_A1T2T001	FALSE	FALSE
A1T2S004\pump_A1T2S004	FALSE	
A1T2X008\valve_A1T2X008	FALSE	
A1T2S003\pump_A1T2S003	FALSE	
A1T3X001\valve_A1T3X001	FALSE	
A1T1X005\valve_A1T1X005	FALSE	
A1T1S002\pump_A1T1S002	FALSE	
A1T2X005\valve_A1T2X005	FALSE	
A1T1X004\valve_A1T1X004	FALSE	
A1T1S001\pump_A1T1S001	FALSE	
A1T2X004\valve_A1T2X004	FALSE	
A1T2S002\stirrer_A1T2S002	FALSE	

For unrestricted use in educational / R&D institutions. $\textcircled{\mbox{\sc only}}$ Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA

SFC - SFC_Product01
SFC Edit Insert CPU Debug View Options Window Help
▶ 中韓 ↔ ḥ ħ ab
SFC_Product01 SCE_PC57_Prj\A1_multipurpose_plant
START
Init_OK all initial conditions are complied
EductB003toR001 educt tank B003 to reactor R001 EductB002toR002 educt tank B002 to reactor R002
L001 >= 350 ml level R001 min 350 ml L002 >= 200 ml level R002 min 200 ml
Heating25°CStir heating reactor R001 to 25 °C EductB001toR002 educt tank B001 to reactor R002
T001 >= 25 °C temperature R001 min 25 °C L002 >= 350 ml level R002 min 350 ml
Wait waiting for parallel process Stir stir for min 10 s
Perild OV if both parallel brancher finish
in bour parallel branches finisition
R002toR001 reactor R002 to reactor R001
L002 <= 50 ml reactor R002 empty
Heating28°C heating reactor R001 to 28 °C
T001 >= 28 °C temperature R001 min 28 °C
RUUITOP/OdBUUI reactor R001 to product tank
1001 d= 50 mL reactor P001 empty or produ
END
Press F1 for help. OB35

For unrestricted use in educational / R&D institutions. $\textcircled{\mbox{\sc os}}$ Siemens 2020. All rights reserved.

p01-08-sequential-function-chart-v9-tud-0719-en.docx

8.5 Compiling and downloading objects

 After all steps and transitions of the SFC are complete, you can compile and download the project, as you have already learned. (* SCE_PCS7_Prj * PLC * Compile and Download Objects...)



 Prior to compiling and downloading, open the settings for compiling and downloading the charts. (→ Charts → Edit)

Compile and Download Objects				_ 🗆 🗙
Selection table:				
Objects	Status	Operating mode	Compile	Download
□-==				
⊟- <mark>∭</mark> AS1			×	
Du Hardware	undefined		4	v
🖃 – 🦉 CPU 414-3 DP			4	×
Blocks				
p Charts	undefined		✓	v
Connections	undefined		×	\checkmark
Settings for compilation/download Edit Test Update Update Upon opening Upon of Upon opening Upon of Compile only OD onto load if compilation error is detected	g Mode	logSelec Igle Object All S	t objects	Deselect All
Devices connected to an enterprise network or directly to the against unauthorized access, e.g. by use of firewalls and netw For more information about industrial security, please visit: http://www.siemens.com/industrialsecurity	internet must be a vork segmentation	appropriately protected h		
Start Close				Help

 Here it is important to select 'Entire program' as the "Scope' and 'Download mode' on the "Compile Charts as Program" and "S7 Download" tabs, respectively. (→ Compile Charts as Program → Scope: Entire program → S7 Download → Download mode: Entire program → OK)

ompile Program / Download to Target System 🔀	Compile Program / Download to Target System
Compile Charts as Program S7 Download	Compile Charts as Program S7 Download
CPU: CPU 414-3 DP Program name: AS1\CPU 414-3 DP\S7 Program(1)\	CPU: CPU 414-3 DP Program name: AS1\CPU 414-3 DP\S7 Program(1)
Scope © Entire program © Changes only	Download mode © Entire program © Changes only
Generate module drivers Block Driver Settings Generate SCL source	 To test CPU (entire program) Include user data blocks
	Show Changes Before downloading the entire program, the CPU is set to STOP and all blocks are deleted. Do you want to download the S7 program?
	Read the notes in the online help about possible effects Auto-archiving Browse for Version Project
OK Abbrechen Hilfe	Archive project after successful download

4. Confirm the warning with "OK". $(\rightarrow OK)$

Download	(244:4028)
<u>^</u>	Note that the blocks in the CPU will be deleted if you download later (e.g. in the Simatic Manager with "CPU->Compile and Download Objects"). Also note that the operator input from the OS or test mode will be lost if the set values values are not read back into the offline program.
OK	

)biects	Status	Operating mode	Compile	Downloa
- En SCE PCS7 Pri	Claudo			
Dun Hardware	undefined			
		STOP		
Blocks				
Charts	undefined			
Connections	undefined			V
 ∓⊒0S				
iettings for compilation/download Update	Dperating Mode	w log-	Select objects	Deselect Al
iettings for compilation/download Update Edit Test Status I Upon opening T Compile only Do not load if compilation error is	Operating Mode	w log ingle Object All	Select objects Select All	Deselect Al Check proje

5. Now you can start compiling and downloading. (\rightarrow Start)

6. Carefully read all warnings that follow and confirm them. (\rightarrow OK \rightarrow Yes)

Compile and download objects (3280:826)						
Downloading program changes during operation can, in the case of malfunctions or program errors, cause serious damage to personnel and equipment! Make sure also that downloading to the individual CPUs is not done simultaneously after compilation. Make sure that no dangerous situations could occur before executing this function!						
ОК		Cancel				

Compile and download objects (3280:822)						
<u>^</u>	If you want to download changes online, please make sure that the prerequisites have been met (e.g. correct settings selected, no previous complete compilation from the OS). A complete download is only possible if the PLCs are not in RUN. Do you want to continue?					
Ye	s No					

7. The log should contain no errors and at most only warnings. To see details for a warning, view the log of the single object. ($\rightarrow \times$)



8.6 Testing the SFC

1. You can now set PLCSIM to RUN-P mode. (\rightarrow PLCSIM \rightarrow RUN-P)

S7-PLCSIM1 AS1\CPU 414-3 DP
File Edit View Insert PLC Execute Tools Window Help
🗋 🖙 🖬 🖨 (PLCSIM(TCP/IP) 💽 🕺 🖻 🖷 🖷 🖷 🖛 🕺 🕅 🏗 🛅 🛅 🛅 酒 酒 🎘 🖄 很
<u> </u>
SF BUN-P IB 1 Bits QB 3 Bits
DC RUN 7654 3210 7654 3210
Press F1 to get Help. CPU/CP: MPI=2 DP=2 IP=192.1///
S7-PLCSIM1 AS1\CPU 414-3 DP
File Edit View Insert PLC Execute Tools Window Help
L 🚰 🖬 🖞 PLCSIM(TCP/IP) 🔄 👗 🏝 🖷 🖷 🖷 🖷 🗮 🖓 🕺 🗑 🐻 📅 📅 🛅 🗃 🗃 🗃 🗃
▶
SF RUN-P IB 1 Bits V QB 3 Bits V
DC RUN 76543210 76543210
Press F1 to get Help. CPU/CP: MPI=2 DP=2 IP=192.1/

2. Double-click the sequential function chart in the plant hierarchy to open it. (\rightarrow SFC_Product01)

SIMATIC Manager - SCE_PCS7_MP	
File Edit Insert PLC View Options Window Help	
🔄 🗅 🖆 🔡 🐖 👗 🛍 🛍 🔍 🖕 🖭 🔛 🏥 💼 🤁 <no filter=""> 💽 🍞 🔡 🎯 🖷 🚍 🗂</no>	N?
SCE_PCS7_MP (Plant View) C:\Program Files (x86)\\STEP7\S7Proj\SCE_PCS7_MP\SCE_MP	괴죄
SCE_PCS7_MP Image: T1_educt_tanks Image: T2_reaction Image: T3_product_tanks Image: T4_rinsing Image: T1_educt_tanks Image: T1_ed	
Press F1 to get Help. PLCSIM.TCPIP.1	

3. In order to observe the sequence, test mode must be activated \mathbb{I} . (\rightarrow Test Mode on/off \mathbb{I})



4. The simulation must be reset, the main switch and Emergency Stop selected and local operation deselected.

Prozessbild	_ 🖸 🗗 🗙
님 📑 🗠 😋 🛄 123 100% 💌 🔍 🎕 Tahoma 💌 12	■ F K U 炒工/工 A工 三工 田工 間口 A工 国工 出工
	Vor-Ort Bedienpanel / Field Operator Control Panel
	Hauptschalter NOTAUS LOKAL Main Switch Emergency Local
T1.B001 T1.B002 T1.B003	T2.R001 T2.R002
	START STOP STATUS START STOP STATUS
⋧⊶∙∙ ⋧⊶∙∙	Von / From T1.B001
\bigcirc \bigcirc \bigcirc \bigcirc \bigcirc \bigcirc	Von / From T1.B002
	Von / From T1.B003
	Rühren / Stir
⋨⋼⋼● ⋨⋼∊● ⋨⋼⋼●	Heizen / Heat
M •••	Entleeren / Discharge
	Spülen / Clean
T2.R001 T2.R002	Umfüllen / Decant
	Legende Signalanzeigen / Key for Indicators
	Ventil: Stellsignal / Geöffnet / Geschlossen
	Valve: Control Signal / Open / Closed
	Motor / Pumper Stellisignal / Laurneloung Motor / Pump : Control Signal / Running Indication
	Heizung: Stellsignal Heater: Control Signal
T3.B001 T3.B002	
T4.B001	Simulationssteuerung / Simulation Control Panel
\rightarrow	Befüllen/Charge T1.B001 🔤 🔤 RESET 📃 RESET 50%
	Befüllen/Charge T1.B002 💻 📄 Entleeren/Discharge T3.B001 📃
	Befüllen/Charge T1.B003 💻 🔤 Entleeren/Discharge T3.B002 📃

5. You can now start testing the SFC. (\rightarrow Start)

SFC - SFC	_Produ	ct01							
SFC Edit In	isert C	PU Debug	View Opti	ons Window	Help				
🗅 🚔 🎒	*	6 6 5	E 6 %	👛 🛐 🕫	8 🔀	\$\$ <mark>\$</mark> \$	€	9, 17, 18, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19	
▶ 幸 幸		화 🛼 abl							
SFC_Pro	duct01	SCE_PCS	7_Prj\A1_	multipurpose	_plant	ONLINE			
			ſ	START	7				
				Init_OK	all initial cor	ditions are compl			
								=	
	Edu	ctB003toR001	educt tank BO	13 to reactor R		EductB002t	oR002	educt tank 8002 to reactor R0	
			J				_		
	U	001 >= 350 ml e	vel R001 min 350	l ml		L002 >= 200	0 ml	evel R002 min 200 ml	
	Hea	tina25°CStir	heating reacto	r R001 to 25 °C		EductB001t	oR002	educt tank 8001 to reactor R0	
			J -			T			
	RUN		[
Idle							T		
MANUAL									
MANUAL		Start		Hold		Resume		Command Output	
AUTO	\times	Ab Trigg	er start pro	cessing. lete		Stop		Cyclic Operation	
<u>_</u>	5	Restart		Reset	2	Error		Time Monitoring	
r									RUN(//
6. Der flow of the sequential function chart is then visible. Active and previously processed steps and transitions are marked.



Learn-/Training Document | PA Module P01-08, Edition 02/2020 | Digital Industries, FA



7. Current conditions and values can be displayed by double-clicking on or opening individual steps or transitions.

OS comments (initialization)		OS comn	nents (processing)	OS comments(closing)	
General	Initiali	zation	Processing	Termination	
1	p_A1T2S004.St	artAut.Value ;=	TRUE	f(g) 1	
20	p_A1T2S004.St	opAut.Value :=	FALSE	f(x) 0	
3 1	A1T2X008.Op	enAut.Value :=	TRUE	f(g) 1	
40	A1T2X008.Clo	seAut.Value :=	FALSE	f(x) 0	
5		:=		f(x)	
6		:=	:	f(x)	
7		:=	:	f(x)	
8		:=	:	f(x)	
9		:=	:	f(x)	
10		(=	:	f(g)	
Close	⊢ †	↓ →		Gio to Help	

Properties - L	ent Cond. OS Comment	SCE_PCS7_Prj\A:	L_multipurpose_pland.after Error	ant\\SFC_Produc 🖄
200.0155	(x) J02.PV_Out.Value	<= 50.0	f(x) 50.0	-
	f(x)		f(x)	
	f(x)		f(x)	& -
	f(x)		f(x)	
	f(x)		f(x)	
	f(x)		f(x)	&
	f(x)		f(x)	8
	f(x)		f(x)	
	f(x)		f(x)	
	f(x)		f(x)	I I
Close	<u>←</u> <u>+</u>	↓ →		Gio to Help



8. In the 'R001toProdB001' state, the SFC and the simulation look like this.

8.7 Checklist – step-by-step instruction

The following checklist helps students to independently check whether all steps of the step-bystep instruction have been carefully completed and enables them to successfully complete the module on their own.

No.	Description	Checked
1	SFC_Product01 created and configured	
2	Sequential function chart created	
3	All steps and transitions named and commented	
4	All steps and transitions (except Wait) edited (grayed out)	
5	END step (Initialization) contains manual setpoint setting and operation of temperature control (SP_ExtLi, SP_IntLi, ManModLi, AutModLi)	
6	END step (Termination) contains enable of operator mode of all 16 utilized blocks (one ModLiOp each)	
7	END step (Termination) contains enable of setpoint setting of temperature control (SP_LiOp)	
8	Successfully tested	
9	Project successfully archived	

Table 1: Checklist for step-by-step instructions

9 Exercises

In the exercises, you apply what you learned in the theory section and in the step-by-step instructions. The existing multiproject from the step-by-step instructions (p01-08-project-r1905en.zip) is to be used and expanded for this. The download of the project is stored as zip file "Projects" on the SCE Internet for the respective module.

The purpose of this exercise is to implement an additional recipe that allows cleaning of the reactors. The following task suggests a possible concept.

9.1 Task

In the 'A1_multipurpose_plant' chart folder, create the SFC 'SFC_Rinse' that cleans Reactors R001 and R002 with rinsing water. The cleaning is to consist of the following steps:

- Filling the reactors (up to 500 ml) with rinsing water
- Stirring the rinsing water (for 20 seconds) in the reactors
- Draining the rinsing water to the product tanks.

Design the rinsing operation in such a way that both reactors are cleaned at the same time. Check whether both reactors are empty (< 50 ml) before rinsing starts.

9.2 Checklist – exercise

The following checklist helps students to independently check whether all steps of the exercise have been carefully completed and enables them to successfully complete the module on their own.

No.	Description	Checked
1	SFC_Rinse created and configured	
2	All steps and transitions named and commented	
3	Parallel processing is available	
4	Start transition is available	
5	END step (Termination) contains enable of operator mode of all utilized blocks (one ModLiOp each)	
6	Successfully tested	
7	Project successfully archived	

Table 2: Checklist for exercises

For unrestricted use in educational / R&D institutions. © Siemens 2020. All rights reserved.

10 Additional information

More information for further practice and consolidation is available as orientation, for example: Getting Started, videos, tutorials, apps, manuals, programming guidelines and trial software/ firmware, under the following link:

siemens.com/sce/pcs7

Preview "Additional information"

Getting Started, Videos, Tutorials, Apps, Manuals, Trial-SW/Firmware

- SIMATIC PCS 7 Overview
- SIMATIC PCS 7 Videos
- > Getting Started
- > Application Examples
- > Download Software/Firmware
- SIMATIC PCS 7 Website
- SIMATIC S7-400 Website

Further Information

Siemens Automation Cooperates with Education siemens.com/sce

Siemens SIMATIC PCS 7 siemens.com/pcs7

SCE Learn-/Training Documents siemens.com/sce/documents

SCE Trainer Packages siemens.com/sce/tp

SCE Contact Partners siemens.com/sce/contact

Digital Enterprise siemens.com/digital-enterprise

Industrie 4.0 siemens.com/future-of-manufacturing

Totally Integrated Automation (TIA) siemens.com/tia

TIA Portal siemens.com/tia-portal

SIMATIC Controller siemens.com/controller

SIMATIC Technical Documentation siemens.com/simatic-docu

Industry Online Support support.industry.siemens.com

Product catalogue and online ordering system Industry Mall **mall.industry.siemens.com**

Siemens Digital Industries, FA P.O. Box 4848 90026 Nuremberg Germany

Subject to change and errors © Siemens 2020

siemens.com/sce