



SIEMENS



Learn-/Training Document

Siemens Automation Cooperates with Education
(SCE) | As of Version V15.1

TIA Portal Module 052-100
Sequencer Programming
with GRAPH and SIMATIC S7

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

SIEMENS

Global Industry
Partner of
WorldSkills
International



Matching SCE trainer packages for this Learn-/Training Document

SIMATIC Controller

- **SIMATIC ET 200SP Open Controller CPU 1515SP PC2 F with WinCC RT Advanced 512 PTs**
Order no.: 6ES7677-2SB42-4AB1
- **SIMATIC ET 200SP Distributed Controller CPU 1512SP F-1 PN Safety**
Order no.: 6ES7512-1SK00-4AB2
- **SIMATIC CPU 1516F PN/DP Safety with Software**
Order no.: 6ES7516-3FN00-4AB2
- **SIMATIC S7 CPU 1516-3 PN/DP with Software**
Order no.: 6ES7516-3AN00-4AB3
- **SIMATIC CPU 1512C PN with Software and PM 1507**
Order no.: 6ES7512-1CK00-4AB1
- **SIMATIC CPU 1512C PN with Software, PM 1507 and CP 1542-5 (CP PROFIBUS)**
Order no.: 6ES7512-1CK00-4AB2
- **SIMATIC CPU 1512C PN with Software**
Order no.: 6ES7512-1CK00-4AB6
- **SIMATIC CPU 1512C PN-1 without Power Supply / with CP for PROFIBUS DP**
Order no.: 6ES7512-1CK00-4AB7
- **SIMATIC S7-1200 Basic Controller, CPU 1215C; DC/DC/DC**
Order no.: 6ES7215-1AG40-4AB1
- **SIMATIC S7-1200 Basic Controller, CPU 1215C; AC/DC/RELAY**
Order no.: 6ES7215-1BG40-4AB1
- **SIMATIC S7-1200 Basic Controller, CPU 1215C; DC/DC/RELAY**
Order no.: 6ES7215-1HG40-4AB1

SIMATIC STEP 7 Software for Training

- **SIMATIC STEP 7 Professional V15.1 - Single License**
Order no.: 6ES7822-1AA05-4YA5
- **SIMATIC STEP 7 Professional V15.1 - Classroom License for 6 Users**
Order no.: 6ES7822-1BA05-4YA5
- **SIMATIC STEP 7 Professional V15.1 - Upgrade License for 6 Users**
Order no.: 6ES7822-1AA05-4YE5
- **SIMATIC STEP 7 Professional V15.1 - Student License for 20 Users**
Order no.: 6ES7822-1AC05-4YA5

Please note that these trainer packages are replaced with successor packages when necessary.
An overview of the currently available SCE packages is provided under: [siemens.com/sce/tp](https://www.siemens.com/sce/tp)

Training courses

For regional Siemens SCE training courses, contact your regional SCE representative:

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

Additional information regarding SCE

[siemens.com/sce](https://www.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 and R&D institutions. Siemens does not guarantee the contents.

This document is only to be used for initial training on Siemens products/systems. This means it can be copied in whole or in part and given to trainees/students for use within the scope of their training/course of study. Disseminating or duplicating this document and sharing its content is permitted within public training and advanced training facilities for training purposes or as part of a course of study.

Exceptions require written consent from Siemens. Send all related requests to

scsupportfinder.i-ia@siemens.com.

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 expressly prohibited. We do not consent to commercial use of the training documents.

We wish to thank the TU Dresden, the Michael Dziallas Engineering company and all other involved persons for the support in the preparation of this SCE Learn-/Training Document.

Table of Content

1	Goal	6
2	Requirement	6
3	Required hardware and software	7
4	Theory.....	8
4.1	Notes on S7-GRAPH programming language	8
4.2	GRAFCET according to EN 60848.....	8
5	Task	9
5.1	Description of control task	9
5.2	Technology diagram	9
5.3	Switching on	10
5.4	Operating mode selection	10
5.5	EMERGENCY STOP	10
5.6	Manual mode	10
5.7	Automatic mode.....	11
5.8	Indicator lights	13
5.9	Reference list.....	14
6	Planning	16
6.1	Sequence diagram of the sorting station.....	17
7	Structured step-by-step instructions	21
7.1	Retrieving an existing project	21
7.2	Importing "Tag table sorting station"	22
7.3	Creating function block FB50 "AUTOMATIC_MODE"	24
7.4	Block properties of FB50 "AUTOMATIC_MODE"	26
7.5	Specifying the interface of FB50 "AUTOMATIC_MODE"	27
7.6	Structure of the sequencer	29
7.7	Programming of FB50 "AUTOMATIC_MODE"	31
7.8	Programming the organization block OB1	48
7.9	Result in the LAD programming language	53
7.10	Saving and compiling the program.....	54

7.11	Downloading the program	55
7.12	Monitoring program blocks	56
7.13	Sequencer in test mode	60
7.14	Synchronization of the sequencer	61
7.15	Creating function block FB30 "SIGNAL_LAMPS"	63
7.16	Specifying the interface of FB30 "SIGNAL_LAMPS"	64
7.17	Programming of FB30 "SIGNAL_LAMPS"	66
7.18	Creating function block FB20 "CLOCK_PULSE"	73
7.19	Specifying the interface of FB20 "CLOCK_PULSE"	74
7.20	Programming of FB20 "CLOCK_PULSE"	75
7.21	General notes on use of events	78
7.22	Creating function block FB10 "RELEASE"	83
7.23	Specifying the interface of FB10 "RELEASE"	84
7.24	Programming of FB10 "RELEASE"	86
7.25	Creating function block FB40 "OPERATING_MODES"	90
7.26	Specifying the interface of FB40 "OPERATING_MODES"	91
7.27	Programming of FB40 "OPERATING_MODES"	92
7.28	Archiving the project	103
7.29	Checklist – step-by-step instructions	104
8	Exercise	105
8.1	Task – Exercise	105
8.2	Planning	105
8.3	Checklist – Exercise	105
9	Additional information	106

Basics of GRAPH Programming

1 Goal

In this chapter, you will learn how to program a sequence control with the graphic programming tool S7-GRAPH and about the basic elements of a control program written with GRAFCET.

The module shows the procedure in the following steps using the sorting station as an example.

- Presentation of the task with the motion sequences and switching states.
- Division of the sequence diagram into multiple sequencers.
- Graphical representation of the sequence diagram in multiple GRAFCETs.
- Creation of a control program according to the produced GRAFCETs of the sorting station, which is implemented using the S7-GRAPH programming language.

The functioning of the created program will be checked using the testing and diagnostics functions of S7-GRAPH.

2 Requirement

This chapter builds on the hardware configuration of a SIMATIC S7. However, any hardware configurations that have digital input and output boards can be used. You can use the following project for this chapter, for example:

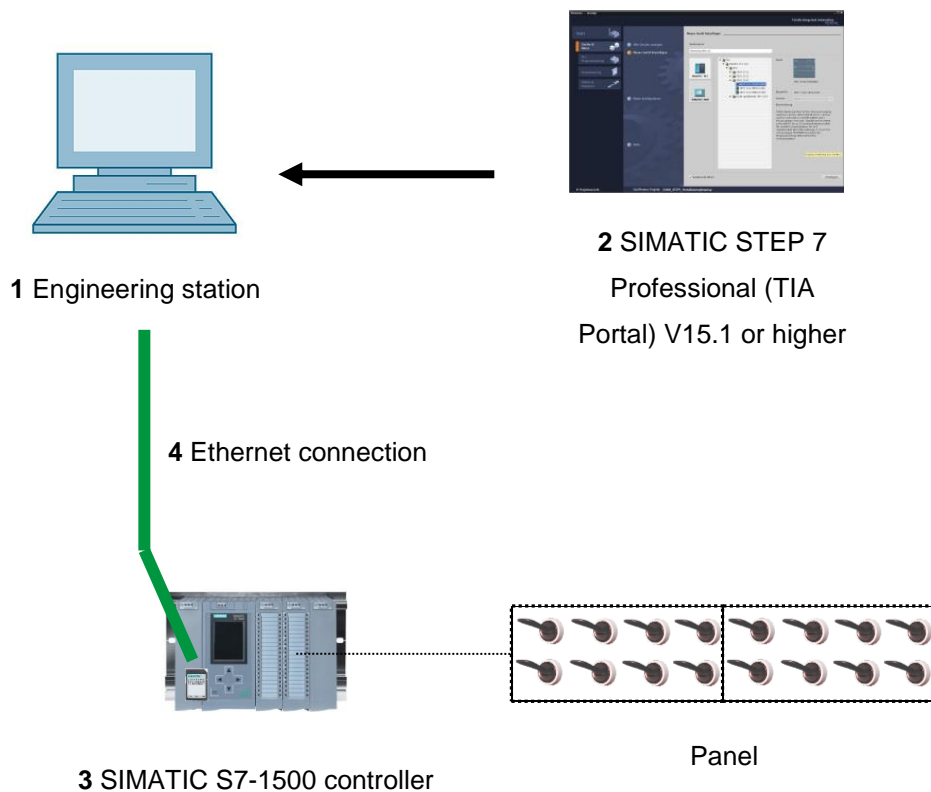
sce-012-101-hardware-config-s7-1516fzap15_1

3 Required hardware and software

- 1 Engineering station: Hardware and operating system requirements apply (for additional information, see Readme on the TIA Portal Installation DVDs)
- 2 SIMATIC STEP 7 Professional software in TIA Portal – V15.1 or higher
- 3 SIMATIC S7 controller, e.g. CPU 1516F-3 PN/DP – Firmware V2.0 or higher with memory card and 16DI/16DO and 2AI/1AO

Please note: The digital inputs should be fed out to a panel.

- 4 Ethernet connection between engineering station and controller



4 Theory

4.1 Notes on S7-GRAPH programming language

S7-GRAPH

- Has been certified according to IEC 61131-3 and PLCopen Base Level since November 2001.
- Siemens is thus the first manufacturer to receive the PLCopen certificate for the S7-GRAPH (SFC - Sequential Function Chart) sequencer programming.
- Programmers that use S7-GRAPH create their programs strictly according to the international standard IEC 61131-3.



Data formats, language elements and graphic representation conform to IEC 61131-3 throughout.

- The S7-GRAPH programming language extends the range of functions of STEP 7 to include a graphic programming option for sequence controls.
- With S7-GRAPH, you can program sequence controls clearly and quickly. The process is broken down into individual steps and the sequence is graphically represented.
- The actions to be executed are defined within the individual steps.
- The step-enabling conditions for advancing to the next steps (transitions) can be created in the LAD or FBD programming language.

4.2 GRAFCET according to EN 60848

GRAFCET is a process-oriented representation of a control task, independent of its implementation, e.g. the equipment used. GRAFCET facilitates the interaction of various specialties, such as mechanics, pneumatics, hydraulics, process engineering, electrics, electronics, etc. A control task is represented with its essential properties in a basic structure (step field) and with the details required for the respective application in a detailed structure (command field) in a clear way.

5 Task

5.1 Description of control task

The automated sorting station (see Figure 1) is used to separate plastic and metal components. A component is fed to the conveyor via a chute. The conveyor starts as soon as the component has been detected. If a component made of metal is on the conveyor, it is detected, transported up to the height of the metal magazine and pushed by a cylinder into the metal magazine. If no metal is detected, the component is made of plastic. The plastic component is transported to the end of the belt, where it falls into the plastic magazine. As soon as a component is sorted, the next component can be fed.

5.2 Technology diagram

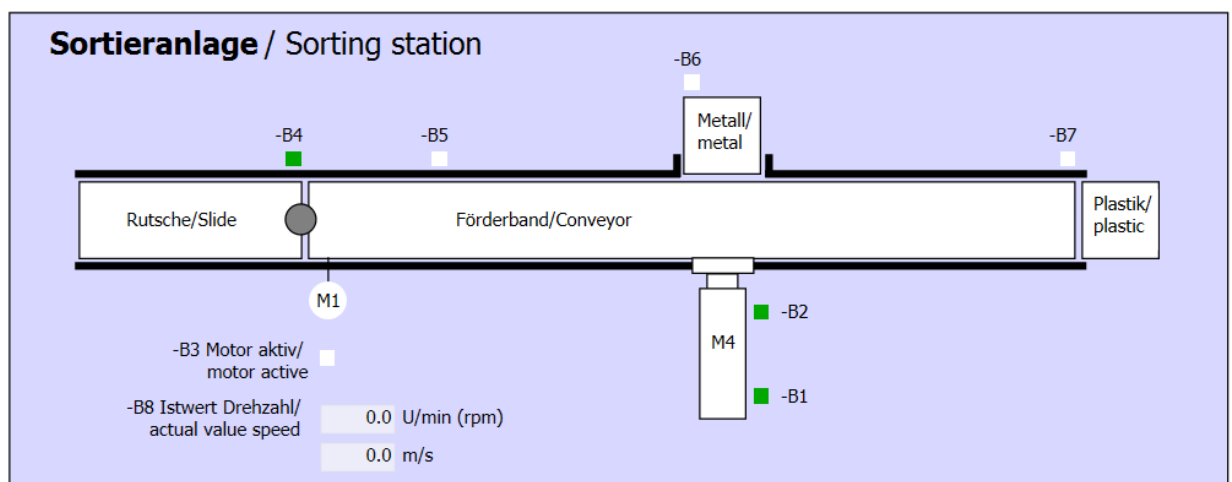


Figure 1: Technology diagram

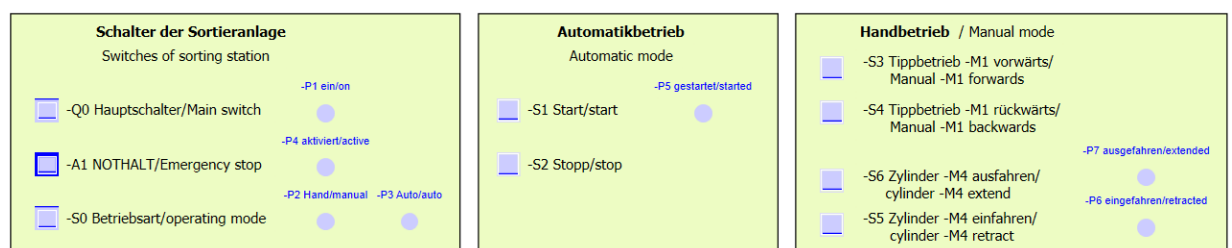


Figure 2: Operator panel

5.3 Switching on

The station is switched on with the main switch -Q0. Relay -K0 (main switch "ON") is energized and provides the supply voltage for the sensors and actuators.

This operating state is indicated by indicator light -P1 (main switch on)

5.4 Operating mode selection

Once the station has been switched on, two operating modes are possible: manual mode or automatic mode. The operating mode is selected using switch -S0.

The selected operating mode is indicated by indicator lights -P2 (manual mode) and -P3 (automatic mode).

5.5 EMERGENCY STOP

In the absence of feedback from the EMERGENCY STOP (-A1), all drives must be stopped immediately.

When the initial state of the pushbutton and the feedback of the EMERGENCY STOP function is present again, the release takes place and the sorting station can be put back into operation.

Activation of the EMERGENCY STOP is indicated by indicator light -P4 (EMERGENCY STOP activated).

5.6 Manual mode

The station is set up in manual mode.

Retracting and extending the cylinder

After pushbutton -S6 (cylinder -M4 extend) is pressed, cylinder -M4 is extended.

After pushbutton -S5 (cylinder -M4 retract) is pressed, the cylinder is retracted.

The extension and retraction of the cylinder continue only as long as the pushbutton is pressed, and the end position has not yet been reached.

When the two pushbuttons are pressed simultaneously, no motion should take place.

Conveyor motor in jog mode

With pushbutton -S3 (manual mode conveyor -M1 forwards), motor -Q1 (conveyor motor -M1 forwards fixed speed) is moved forward in jog mode. With pushbutton -S4 (manual mode conveyor -M1 backwards), motor -Q2 (conveyor motor -M1 backwards fixed speed) is moved backward in jog mode. When the two pushbuttons are pressed simultaneously, no motion should take place.

For safety reasons, only the preset speed may be used here. Output -Q3 (conveyor motor -M1 variable speed) must therefore be deactivated.

Initial state

When the station starts or after the EMERGENCY STOP is tripped, the station must be moved in manual mode to a defined operating state (initial state). In the initial state, the conveyor is empty and stopped and the cylinder is retracted.

5.7 Automatic mode

In automatic mode, the station handles the process automatically.

Starting and stopping

If the station is in the initial state, automatic mode starts when pushbutton -S1 (automatic start) is pressed. When pushbutton -S2 (automatic stop) is pressed, automatic mode is stopped again as soon as the initial state has been reached.

If EMERGENCY STOP has been tripped or the operating mode has been changed, automatic mode is stopped immediately (without return to the initial state).

The current state is indicated by indicator light -P6 (automatic mode started).

Automatic mode

If light sensor -B4 (part at slide) detects a component, the conveyor motor starts. The component slides onto the transport conveyor and is further conveyed.

If inductive sensor -B5 detected a metal component, this is transported up to light sensor -B6 (part in front of cylinder -M4). The conveyor is then switched off. As soon as -B3 (sensor motor -M1 active) no longer supplies a signal, the cylinder control (see below) is activated and moves the component into the metal magazine. As soon as the cylinder is retracted again, the sorting station is back in the initial state.

If a metal component was not detected by sensor -B5, the component is interpreted as a plastic component and is transported to the end of the conveyor. It is detected there by light sensor -B7 and conveyed after a delay time into the plastic magazine at the end of the conveyor.

Cylinder control

If a metal component reaches light sensor -B6 (part in front of cylinder -M4) and the conveyor has been stopped, cylinder -M4 moves to the front-end position -B2 (cylinder -M4 extended), thereby pushing the metal component from the conveyor into the metal magazine. Cylinder -M4 then moves to the rear-end position -B1 (cylinder -M4 retracted).

Speed control (conveyor speed)

In automatic mode, the motor can be moved at a fixed or variable speed.

Fixed speed requires signal "1" at -Q1 "Conveyor motor -M1 forwards fixed speed" or -Q2 "Conveyor motor -M1 backwards fixed speed". For variable speed, -Q3 "Conveyor motor -M1 variable speed" must be activated and a "manipulated value for motor speed" (analog value +/-10 V corresponds to +/- 50 rpm or 10 m/s) must be specified at -U1. Signal "1" must not be present at -Q1 "Conveyor motor -M1 forwards fixed speed" or -Q2 "Conveyor motor -M1 backwards fixed speed". Otherwise, -U1 has no effect on the speed of the conveyor.

5.8 Indicator lights

As soon as relay -K0 (main switch "ON") becomes energized, indicator light -P1 (main switch on) lights up.

If switch -S0 (mode selector manual/automatic) is set to Manual, the indicator light -P2 (manual mode) lights up. If switch -S0 is set to Automatic, the indicator light -P3 (automatic mode) lights up.

If the EMERGENCY STOP function has tripped, -P4 (EMERGENCY STOP activated) lights up.

If automatic mode has been selected and the station is in the initial state, -P5 (automatic mode started) flashes to signal that automatic mode can be started. As soon as automatic mode has been started, -P5 lights up.

Indicator light -P6 (cylinder -M4 retracted) lights up as soon as end position sensor -B1 (cylinder -M4 retracted) has been reached. Indicator light -P7 (cylinder -M4 extended) lights up as soon as cylinder -M4 has reached the front-end position sensor -B2 (cylinder -M4 extended). Indicator lights -P6 and -P7 are not lit if the cylinder is located in neither of the two end positions.

5.9 Reference list

The following signals are required as global operands for this task.

DI	Type	Identifier	Function	NC/NO
I 0.0	BOOL	-A1	return signal emergency stop ok	NC
I 0.1	BOOL	-K0	main switch „ON“	NO
I 0.2	BOOL	-S0	mode selector manual(0) / automatic(1)	man = 0 auto = 1
I 0.3	BOOL	-S1	pushbutton automatic start	NO
I 0.4	BOOL	-S2	pushbutton automatic stop	NC
I 0.5	BOOL	-B1	sensor cylinder -M4 retracted	NO
I 0.6	BOOL	-B2	sensor cylinder -M4 extended	NC
I 0.7	BOOL	-B3	sensor motor -M1 active (pulse signal for positioning)	NO
I 1.0	BOOL	-B4	sensor part at slide	NO
I 1.1	BOOL	-B5	sensor metal part	NO
I 1.2	BOOL	-B6	sensor part in front of cylinder -M4	NO
I 1.3	BOOL	-B7	sensor part at end of conveyor	NO
I 1.4	BOOL	-S3	pushbutton manual mode conveyor –M1	NO
I 1.5	BOOL	-S4	pushbutton manual mode conveyor –M1	NO
I 1.6	BOOL	-S5	pushbutton manual mode cylinder -M4 retract	NO
I 1.7	BOOL	-S6	pushbutton manual mode cylinder -M4 extend	NO

DO	Type	Identifier	Function	
Q 0.0	BOOL	-Q1	conveyor motor -M1 forwards fixed speed	
Q 0.1	BOOL	-Q2	conveyor motor -M1 backwards fixed speed	
Q 0.2	BOOL	-Q3	conveyor motor -M1 variable speed	
Q 0.3	BOOL	-M2	cylinder -M4 retract	
Q 0.4	BOOL	-M3	cylinder -M4 extend	
Q 0.5	BOOL	-P1	display „main switch on“	
Q 0.6	BOOL	-P2	display „manual mode“	
Q 0.7	BOOL	-P3	display „automatic mode“	
Q 1.0	BOOL	-P4	display „emergency stop activated“	
Q 1.1	BOOL	-P5	display „automatic mode started“	
Q 1.2	BOOL	-P6	display cylinder -M4 „retracted“	
Q 1.3	BOOL	-P7	display cylinder -M4 „extended“	

Legend for reference list

DE Digital Input

AE Analog Input

E Input

NC Normally Closed

NO Normally Open

DO Digitaler Output

AO Analog Input

O Output

6 Planning

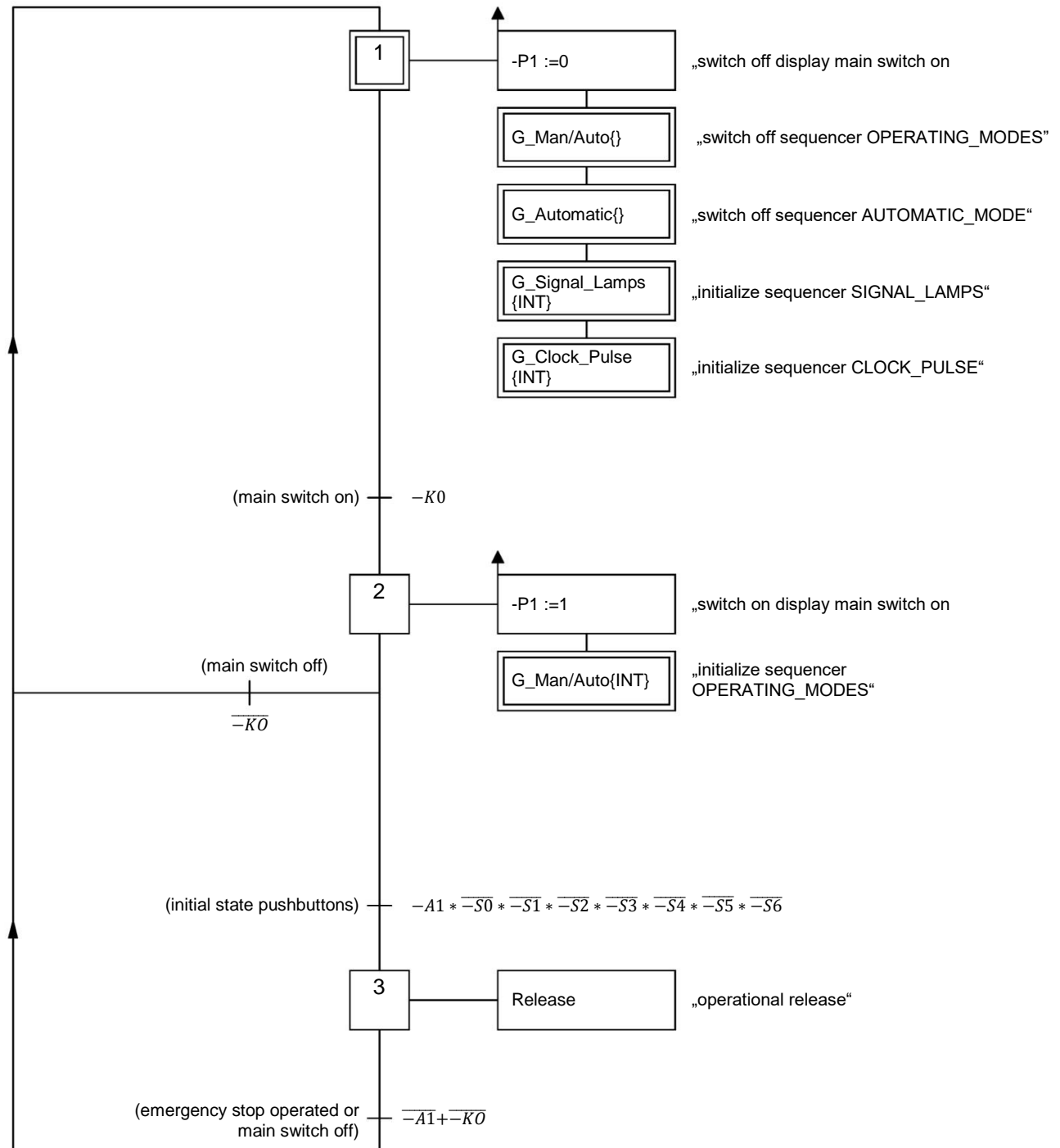
In order to represent the individual requirements of the task, the sequence diagram of the sorting station is divided into five subareas.

- The conditions for the release of the controller are defined in the first part of the sequence diagram.
- A clock pulse at 1 Hz is generated in the second part of the sequence diagram.
- The third part of the sequence diagram shows the control of the signal lamps.
- The fourth part of the sequence diagram describes the selection of operating modes and manual mode.
- The automatic mode of the sorting station is presented in the fifth part of the sequence diagram.

These five subareas are described in the following GRAFCETS.

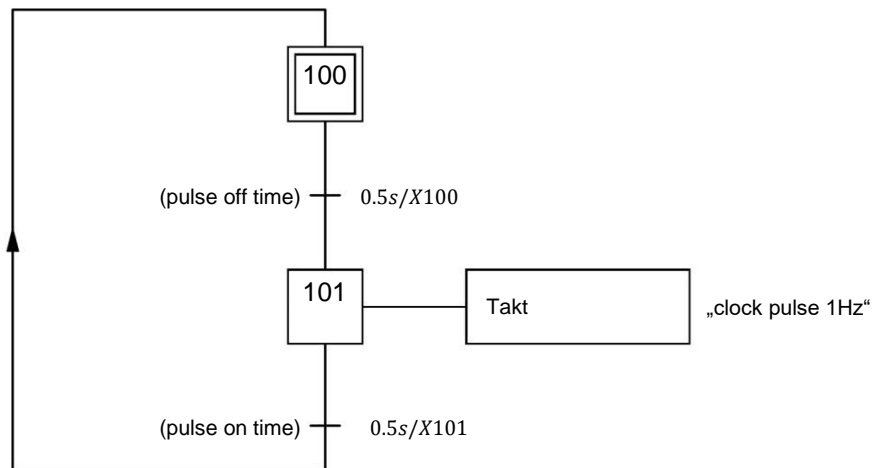
6.1 Sequence diagram of the sorting station

GRAFCET RELEASE

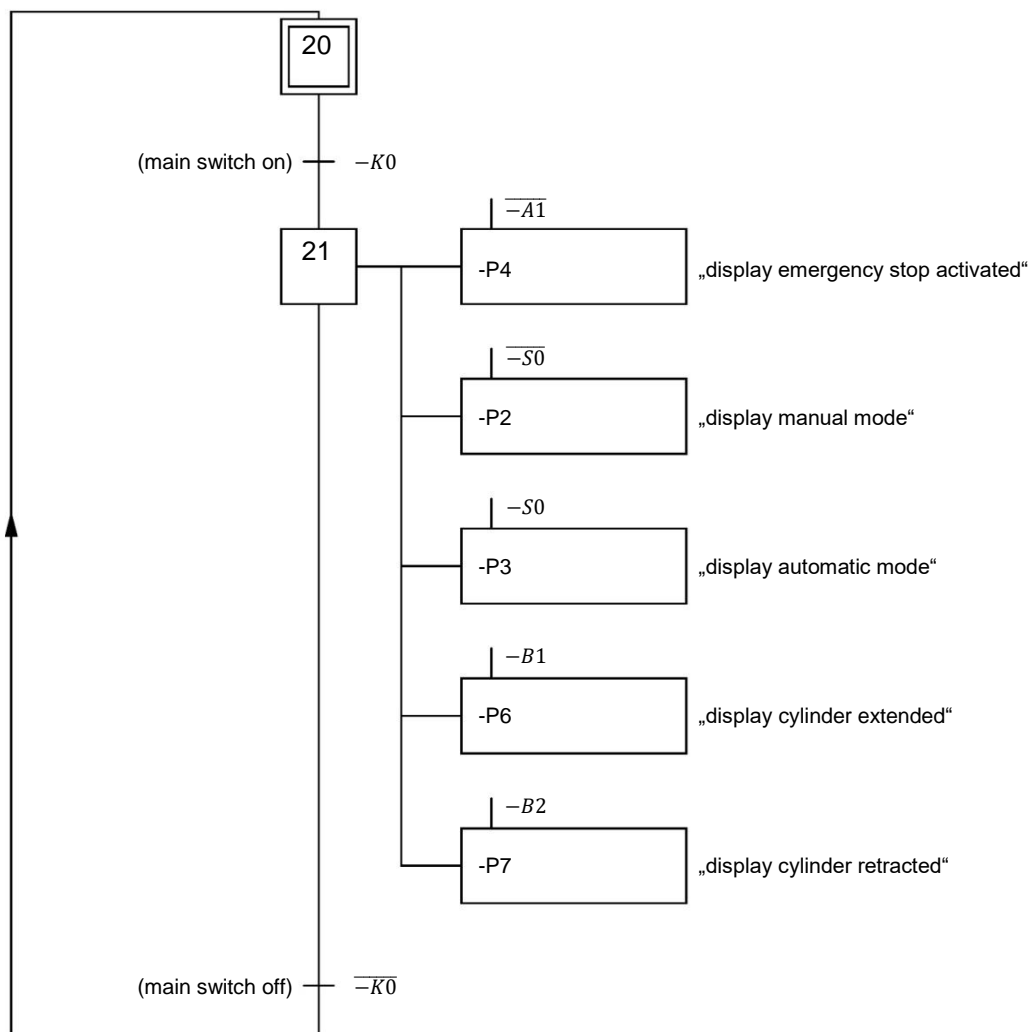


The conditions for switching the station on and off, the operational release and the EMERGENCY STOP function are represented in this GRAFCET.

GRAFCET CLOCK_PULSE

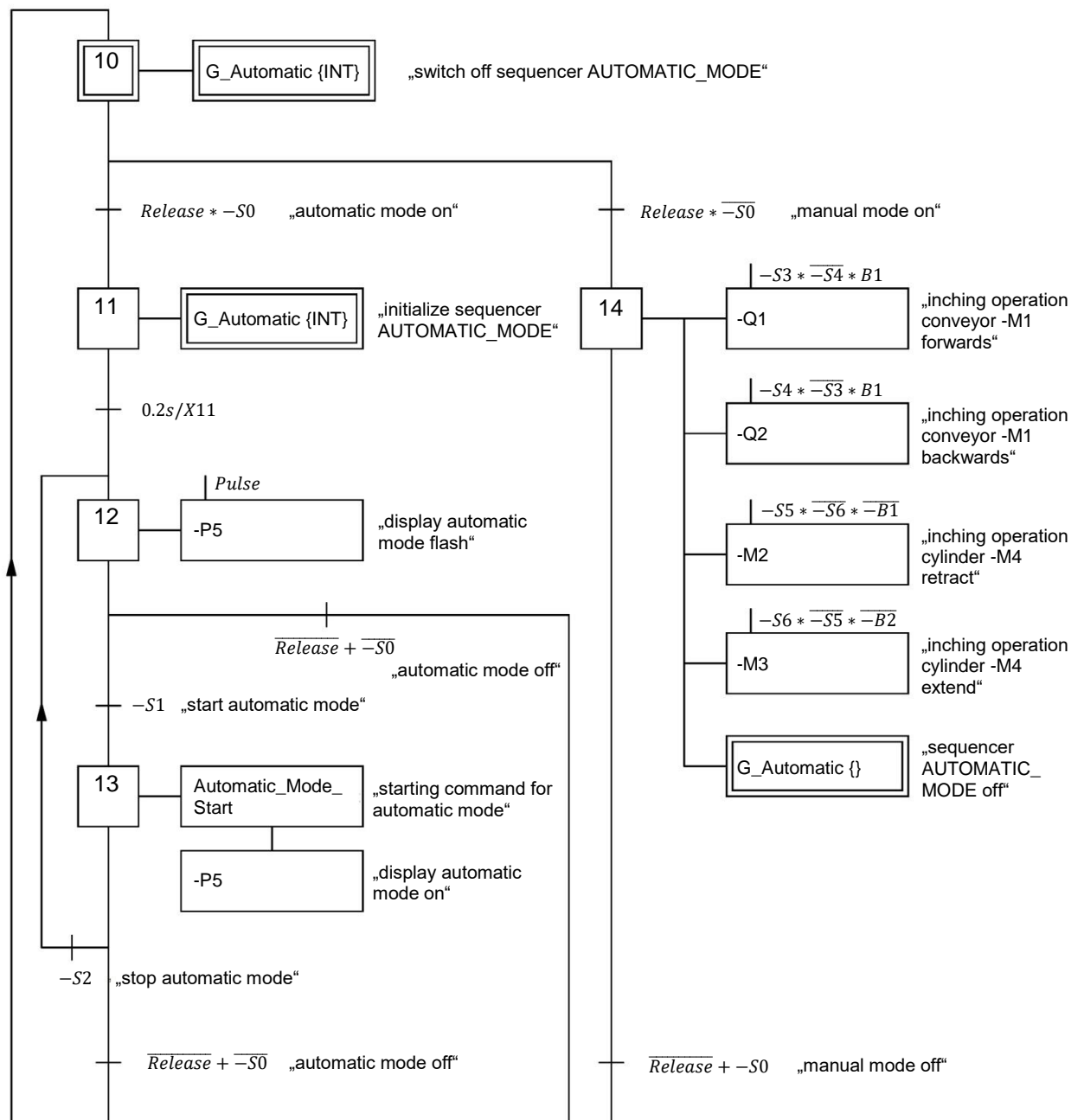


GRAFCET SIGNAL_LAMPS



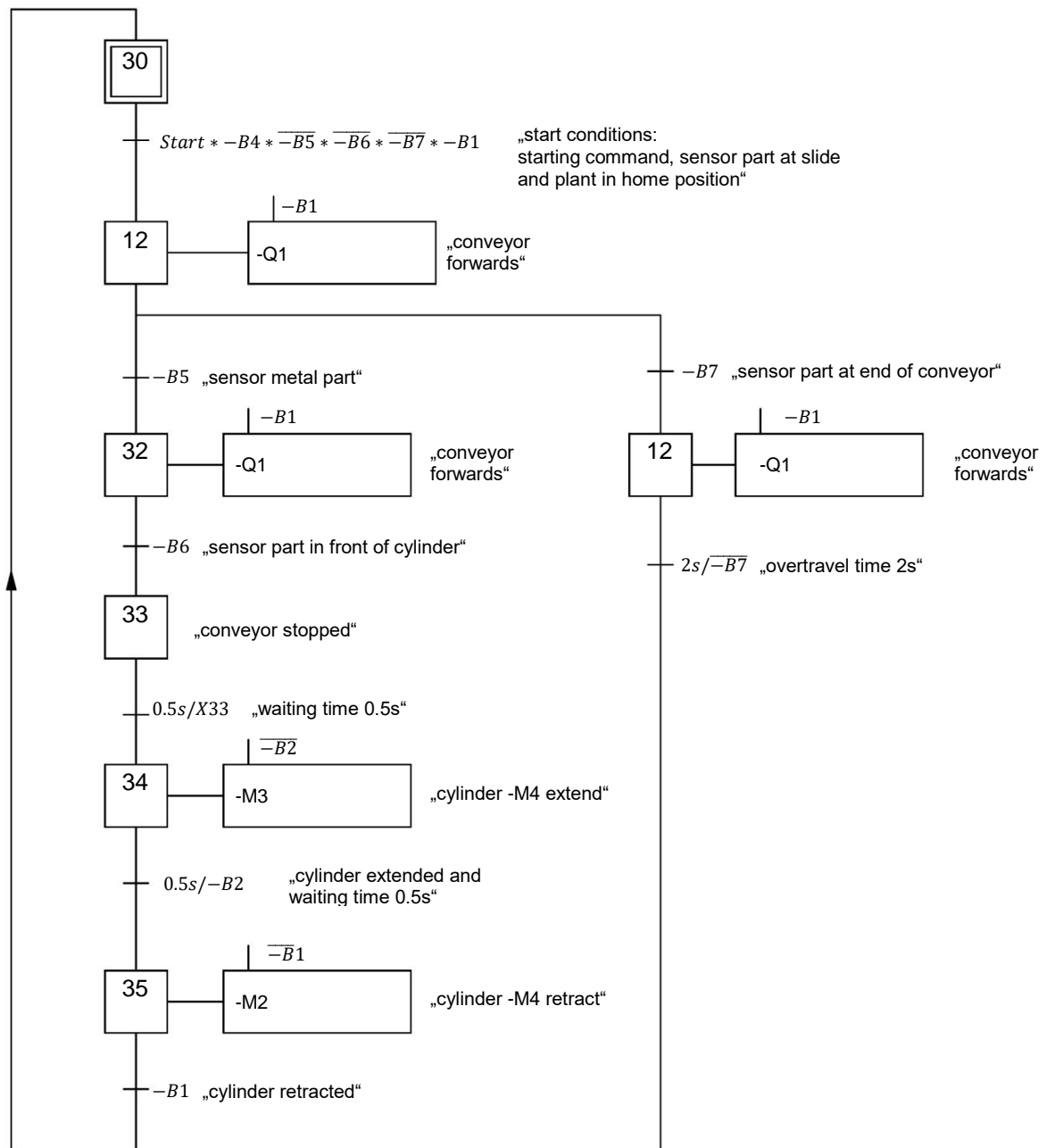
Only after switching on the main switch are the signal lamps for EMERGENCY STOP, the operating mode and the cylinder position activated.

GRAFCET OPERATING_MODES



The conditions for the operating mode selection and for starting of automatic mode and manual mode are represented in this GRAFCET.

GRAFCET AUTOMATIC_MODE



This GRAFCET shows the automatic mode of the sorting station.

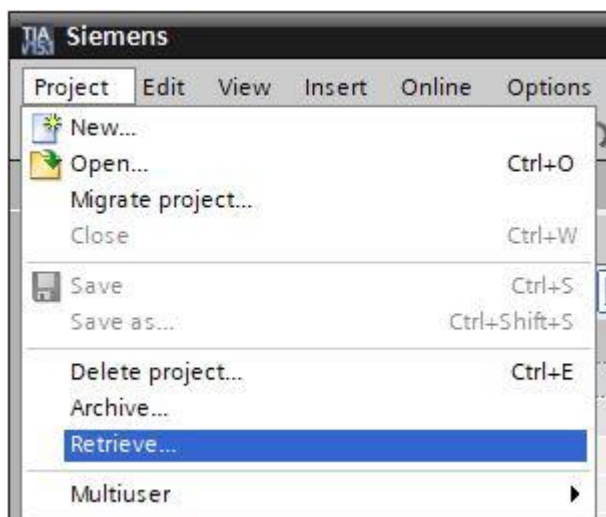
7 Structured step-by-step instructions

You can find instructions on how to carry out planning below. If you already have a good understanding of everything, it will be sufficient to focus on the numbered steps. Otherwise, simply follow the detailed steps in the instructions.

7.1 Retrieving an existing project

→ Before we can start programming the required GRAPH function blocks (FB) of the sorting station, we need a project with a hardware configuration (e.g. sce-012-101-hardware-config-s7-1516f....zap15_1). To retrieve an existing project that has been archived, you must select the relevant archive with → Project → Retrieve in the project view. Confirm your selection with "Open".

(→ Project → Retrieve → Select a *.zap15_1 archive ... → Open)

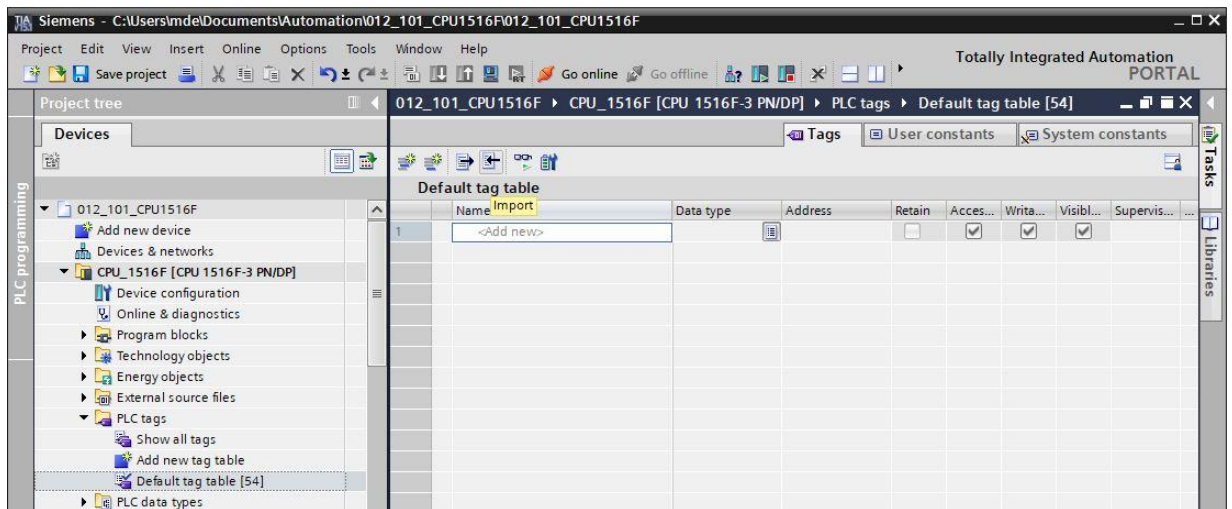



→ The next step is to select the target directory where the retrieved project will be stored. Confirm your selection with "OK". (→ Target directory → OK)

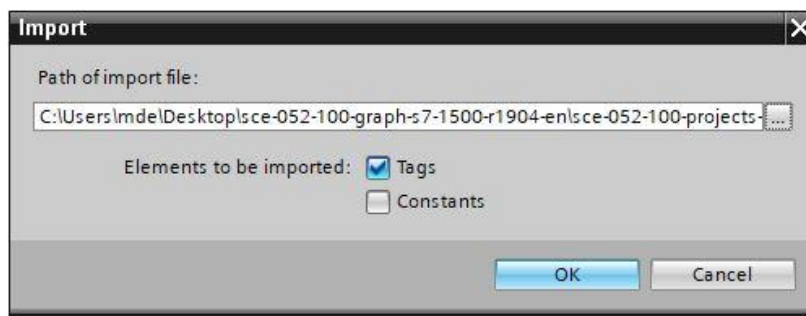
7.2 Importing "Tag table sorting station"

→ To insert an existing symbol table, open the default tag table and then click the "Import" button.

(→ Import )



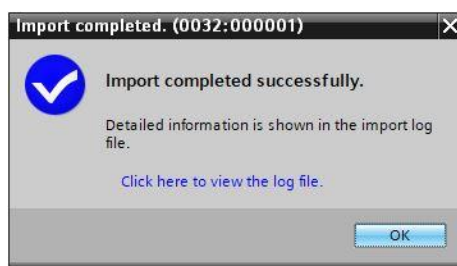
→ Use the  button to choose the location of the import file.



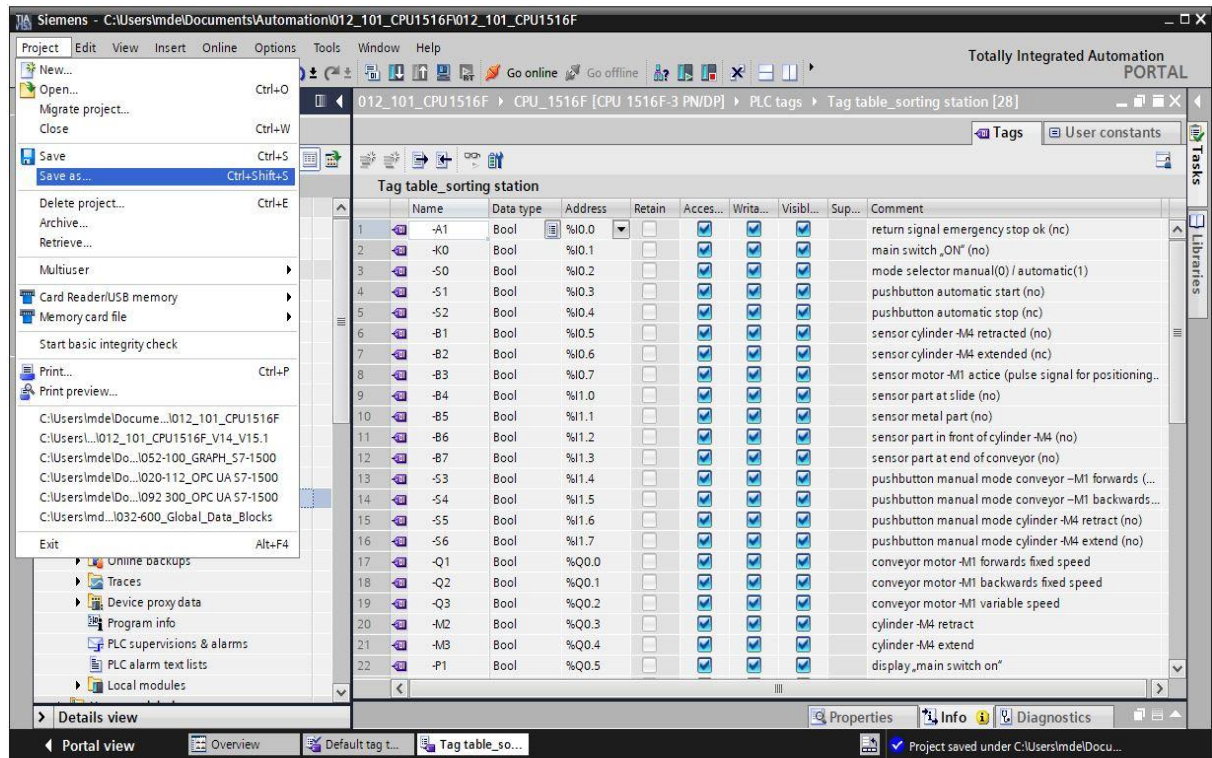
→ Select the desired symbol table (e.g. in .xlsx format) and confirm the selection with "Open".

(→sce-020-100-tag-table-sorting-station... → Open → OK → OK)

→ When the import is finished, you will see a confirmation window and have an opportunity to view the log file for the import. Click → OK.




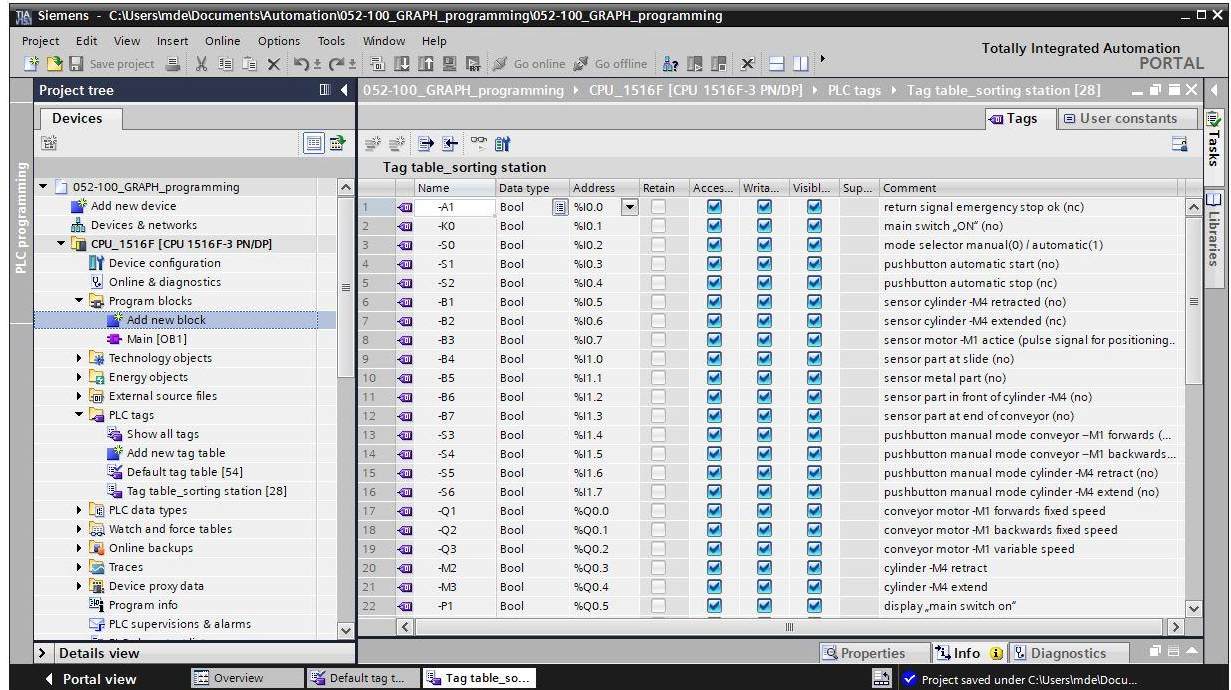
→ You have now imported the tag table of the sorting station. Save your project under the name 052-100_GRAPH_S7-1500. (→ Project → Save as ... → 052-100_GRAPH_S7-1500 → Save)



7.3 Creating function block FB50 "AUTOMATIC_MODE"

→ In the project tree under Program blocks, click on "Add new block" to create a new function block.

(→ Program blocks → Add new block → )



- Name your new function block "AUTOMATIC_MODE", set the language to GRAPH and manually choose the number 50 for the FB number. Select the "Add new and open" check box. You are then taken automatically to your created function block in the project view. Click "OK". (→ Name: AUTOMATIC_MODE → Language: GRAPH → Manual → Number: 50 → ☒ Add new and open → OK)

Add new block

Name: AUTOMATIC_MODE

Organization block

Function block

Function

Data block

Language: GRAPH

Number: 50

☒ Manual

☐ Automatic

Description:

Function blocks are code blocks that store their values permanently in instance data blocks, so that they remain available after the block has been executed.

[more...](#)

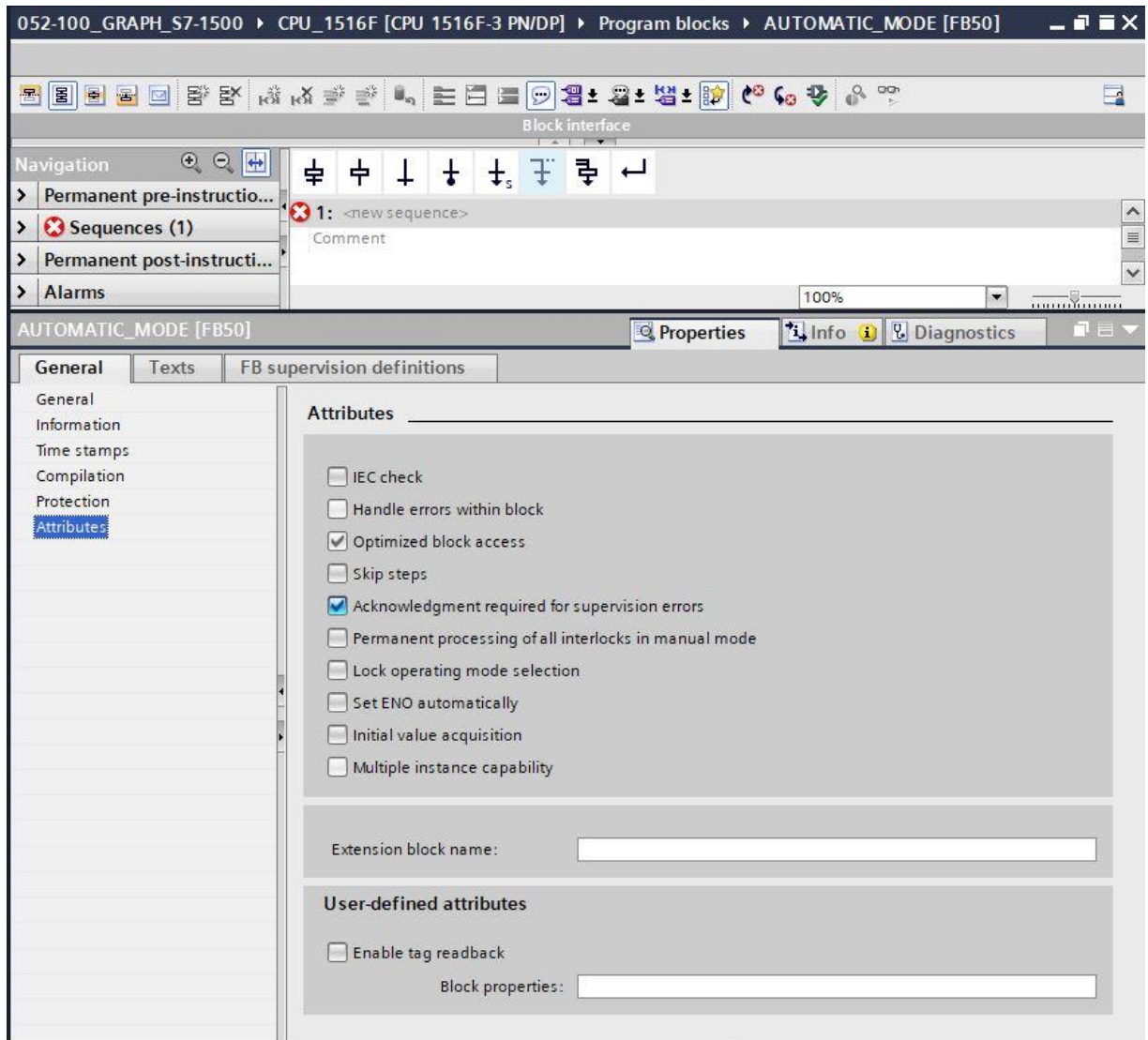
> Additional information

☒ Add new and open

OK Cancel

7.4 Block properties of FB50 "AUTOMATIC_MODE"

- If you selected "Add new and open", the project view opens with a GRAPH editor for programming the block you just created.
- To check the specific properties of the GRAPH block, select "Properties" and select "Attributes" under "General" (→ Properties → General → Attributes). Select the attributes as shown here.



Note:

- You can find detailed information on the attributes in the manuals or the online help.

7.5 Specifying the interface of FB50 "AUTOMATIC_MODE"

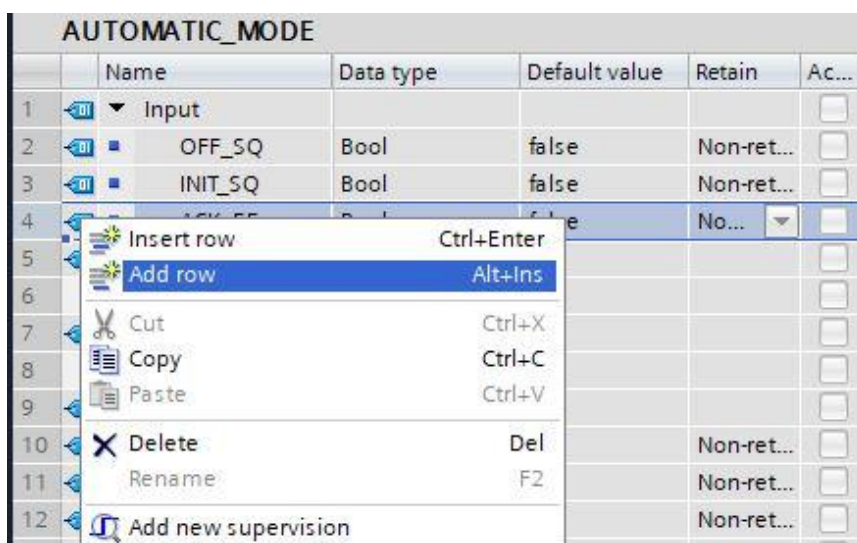
- You can find the interface description of your function block in the upper section of your programming view. Based on the default settings in the TIA Portal, the local tags of the standard interface parameters have already been created. If necessary, these default settings can be changed in the settings of the TIA Portal.
- We need only the first three input tags. The rest of the input tags and all the output tags can be deleted.

The screenshot shows the 'AUTOMATIC_MODE' function block interface in the TIA Portal. A context menu is open over the 'Delete' option, indicating the deletion of input tags. The interface table lists 15 tags: 1-14 are inputs (OFF_SQ, INIT_SQ, ACK_EF, S_PREV, S_NEXT, SW_AUTO, SW_TAP, SW_TOP, SW_MAN, S_SEL, S_ON, S_OFF, T_PUSH) and 15 is an output. The context menu options include 'Delete', 'Rename', 'Add new supervision', 'Update interface', 'Go to next point of use', 'Go to definition', 'Cross-references', and 'Cross-reference information'.

- The static tags must not be deleted.

AUTOMATIC_MODE										
	Name	Data type	Default value	Retain	Ac...	...	Vi...	Se...	Supervis...	Comment
1	Input									
2	OFF_SQ	Bool	false	Non-ret...						Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...						Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...						Acknowledge all errors and faults
5	Output									
6	<Add new>									
7	InOut									
8	<Add new>									
9	Static									
10	RT_DATA	G7_RTDataPlus...		Non-ret...						Internal data area
11	Trans1	G7_TransitionP...		Non-ret...						Transition structure
12	Step1	G7_StepPlus_V6		Non-ret...						Step structure
13	Temp									
14	<Add new>									
15	Constant									

- The specified GRAFCET for the automatic mode is a function-related, process-oriented description of our control task, independent of the sensor wiring. This means that instead of examining signal states (normally open contact/normally closed contact), the station state (cylinder retracted) is looked at. This GRAFCET describes the step-by-step sequence with equipment identifiers, as in the mechatronics engineer exam.
- According to EN 81346-2 the minus sign is placed before the letter (-B1) of the product aspect, thus in the case of -B1 the component that signals the retracted cylinder, independent of the wiring. For output assignments, e.g. -Q1, the component that is being controlled.
- The GRAPH function blocks created by us are to be created in just as function-related a manner with the same designations.
- The same tag designations can be used for global tags as well as local tags in the TIA Portal, which is why we can take the needed tags from the GRAFCET for the automatic mode and from the Tag table sorting station.
- Select the last row of the input tags with the right mouse buttons and select menu item "Add row" (→ Input → ACK_EF → Add row)

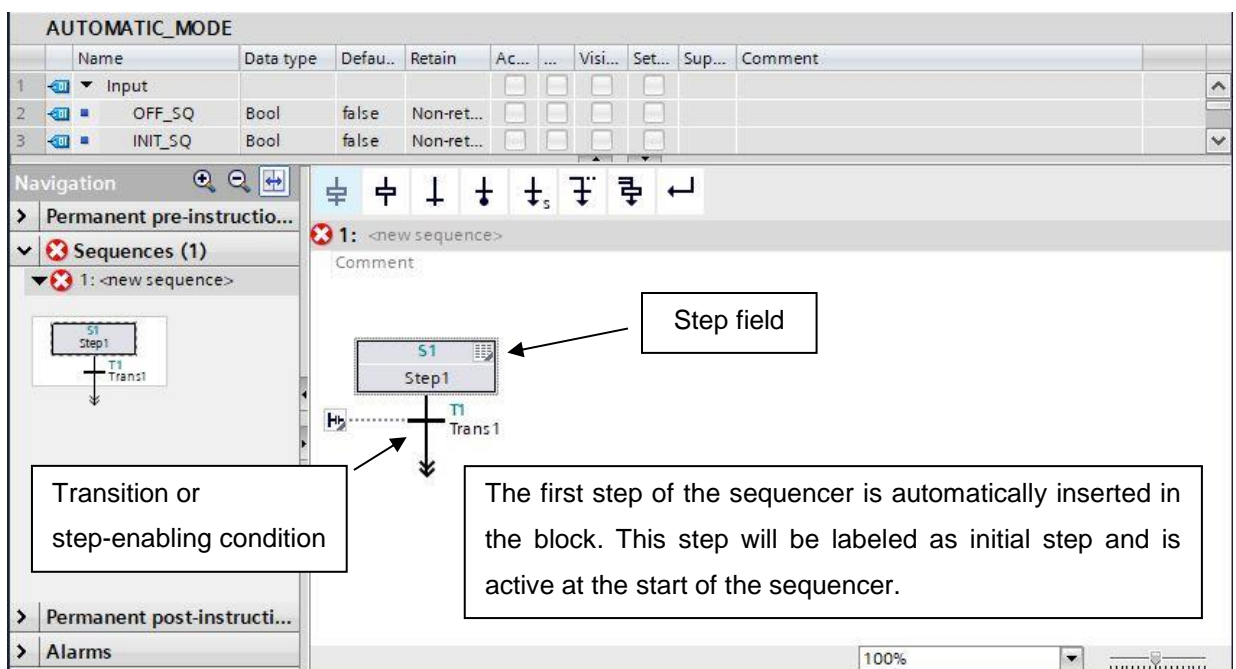


- Add parameter #Start as the input interface under Input in the added line and confirm the input with the Enter key. Data type "Bool" is assigned automatically. This will be retained. Enter the associated comment "Start command".
- Add additional binary input parameters #-B1, #-B2, #-B4 to #-B7 under Input and check their data types. Add descriptive comments for these.
- Add the binary output parameters #-Q1, #-M2 and #-M3 under Output and check their data types. Add descriptive comments for these.
- Alternatively, you can also copy and paste these from the tag table.

AUTOMATIC_MODE										
	Name	Data type	Defau...	Retain	Ac...	...	Visi...	Set...	Sup...	Comment
1	▼ Input									
2	OFF_SQ	Bool	false	Non-ret...						
3	INIT_SQ	Bool	false	Non-ret...						
4	ACK_EF	Bool	false	Non-ret...						
5	Start	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			starting command
6	-B1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor cylinder -M4 retracted
7	-B2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor cylinder -M4 extended
8	-B4	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor part at slide
9	-B5	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor metal part
10	-B6	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor part in front of cylinder -M4
11	-B7	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor part at end of conveyor
12	▼ Output									
13	-Q1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			conveyor motor -M1 forwards fixed speed
14	-M2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			cylinder -M4 retract
15	-M3	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			cylinder -M4 extend

7.6 Structure of the sequencer

After the local tags have been declared, we can start creating the sequencer.



A sequencer consists of a series of steps that are activated in a fixed sequence depending on the conditions for the transition to the next step.

The execution of a sequencer always begins with an initial step.

A step is exited when any pending faults are eliminated or confirmed and the transition condition following the step is satisfied.

The next step, which follows the transition whose condition is met, becomes active.

Simultaneous branches allow multiple steps following the transition to be enabled simultaneously.

A jump to any step of this or another sequencer of the same function block is available at the end of the sequencer. This allows a cyclic operation of the sequencer. A sequence end can also be placed at the end of the sequencer. The sequence ends here when the sequence end is reached.

Active step

An active step is a step whose actions are currently being executed.

The step becomes active when the conditions of the previous transition are satisfied. In addition, it becomes active as soon as it is defined as the initial step and the sequencer has been initialized or when it is called by an event-dependent action.

Objects of S7-GRAPH

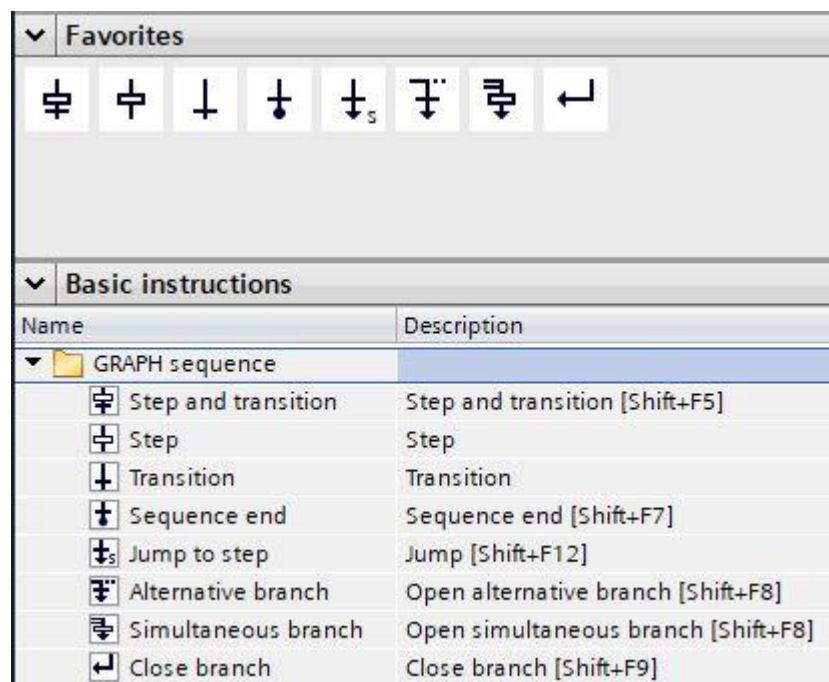


Different views of the sequencer can be selected with the first five buttons of the toolbar.

The sixth button is used to create a new sequencer in the seventh button to delete a sequencer.

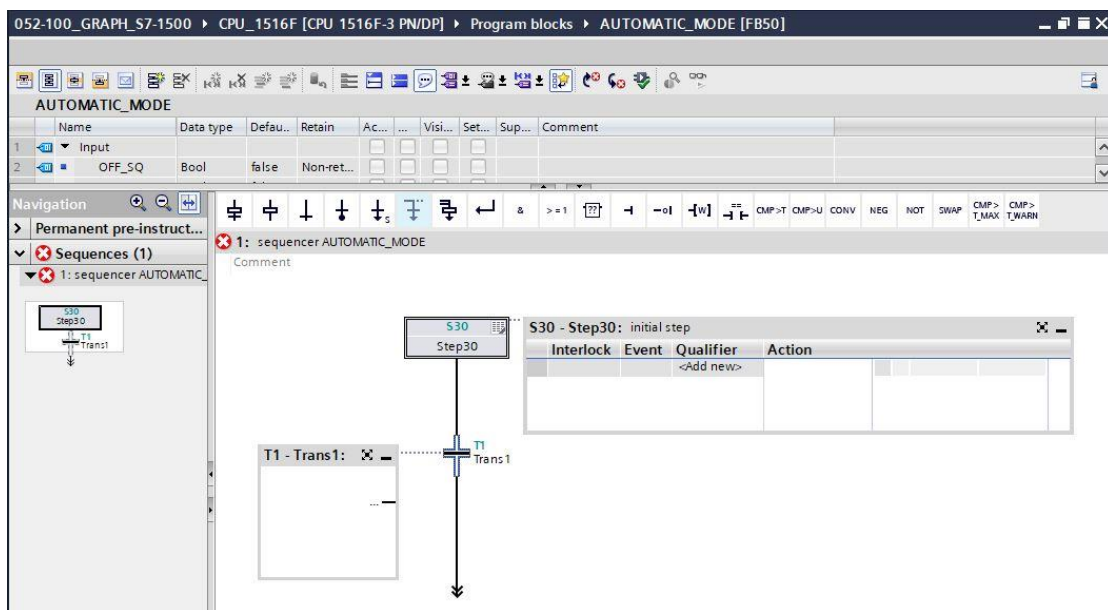
Elements of a sequencer

The following elements of a sequencer can be directly selected under Favorites.

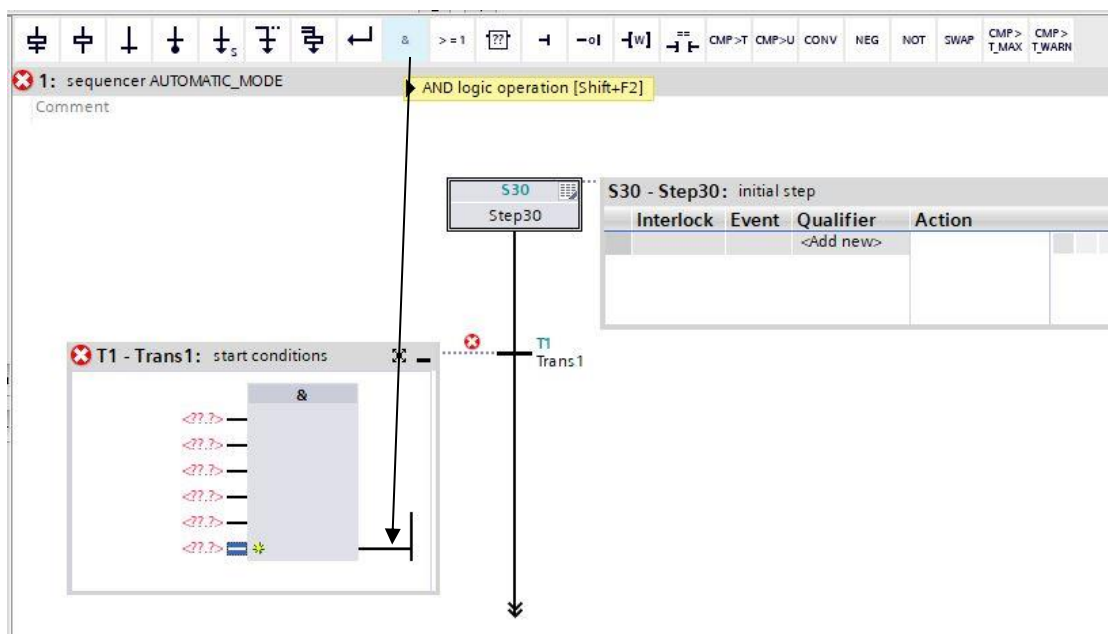


7.7 Programming of FB50 "AUTOMATIC_MODE"

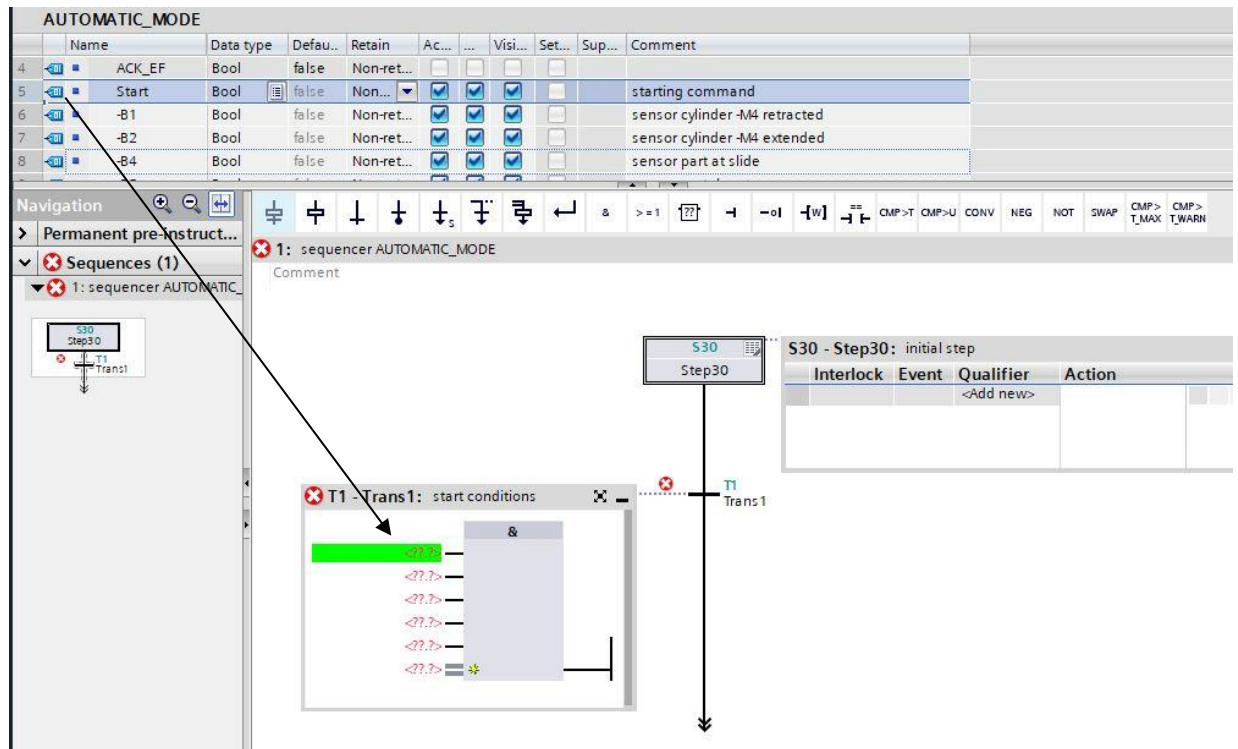
- First, we assign the sequencer the name "sequencer AUTOMATIC_MODE" by overwriting the text <new sequence>.
- The step number and the designation of the step tag must now be changed. To do so, click inside the step field and enter a new number and designation.
- Open the action table using the "📄" button in the step field.
- Open the input window using the "H" button at the transition.



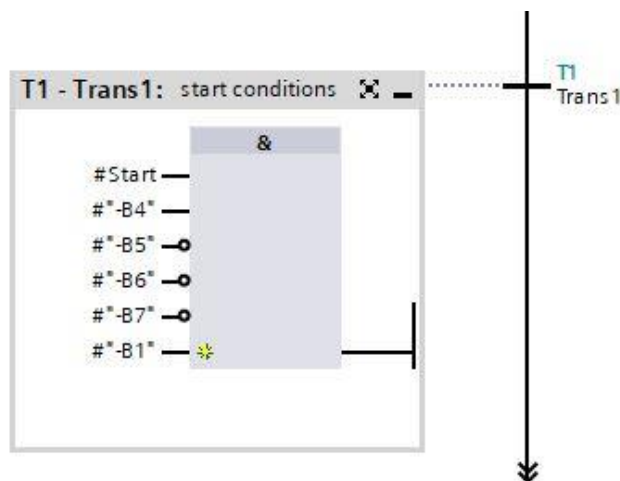
- Insert an AND logic operation with six inputs in the window of Transition 1.
- Designate **T1 – Trans1:** as "start conditions", and designate the **S30 – Step 30:** as "initial step" in the action table.



→ To create the interconnection, drag the "Start" tag from the interface to the first input of the AND logic operation.



→ Interconnect the AND logic operation according to the specification in the GRAFCET.

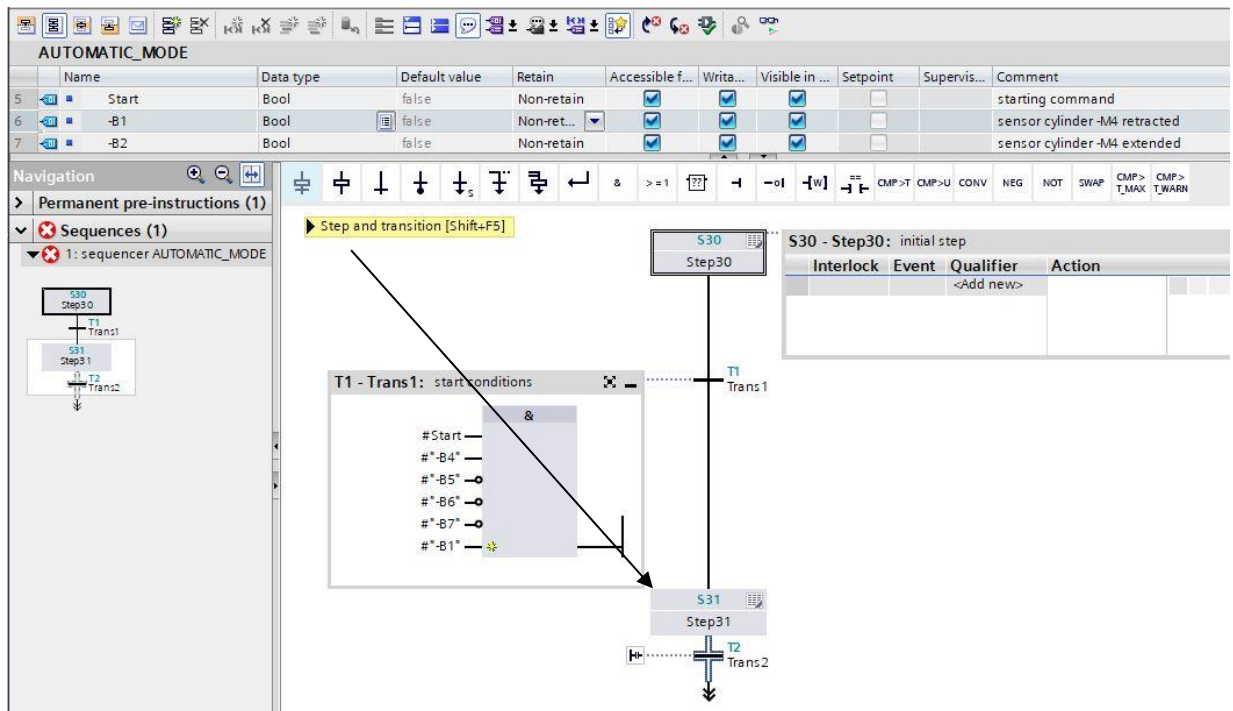


→ Here, the initial step has no actions and is thus finished.

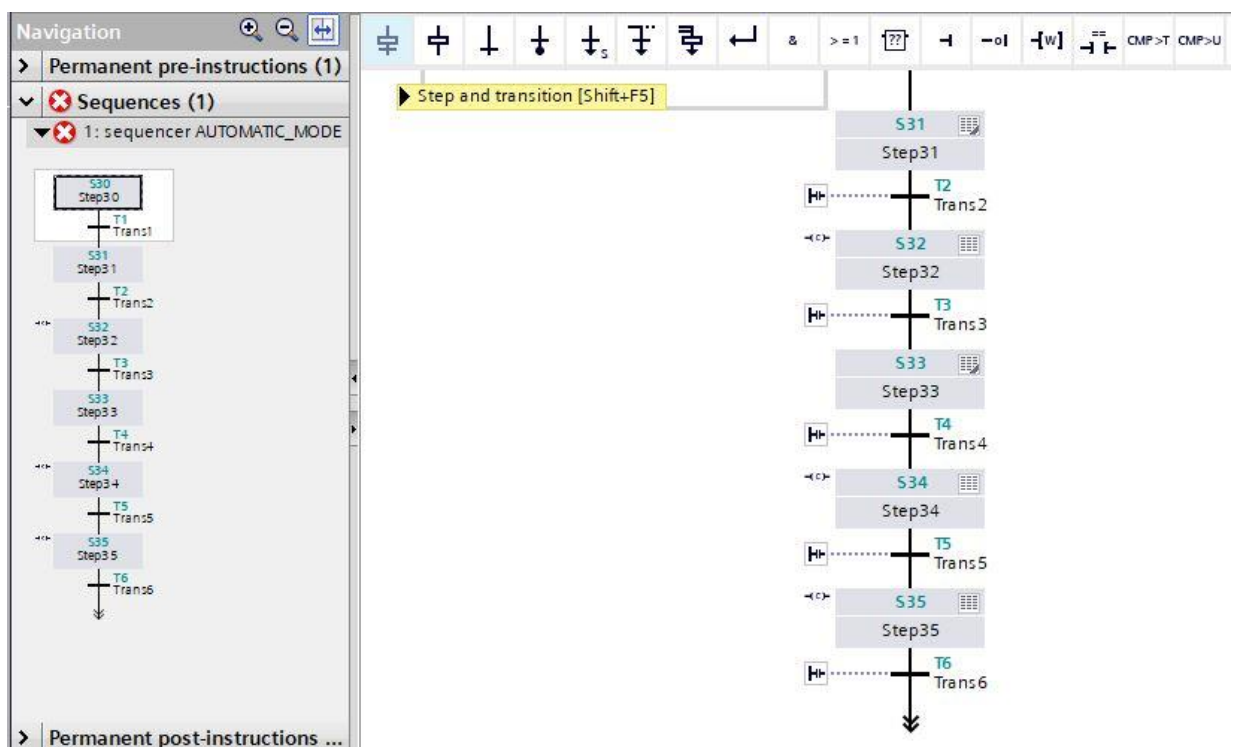
Note:

- To avoid confusion with the global tags from the "Tag table sorting station", the local tags should be moved from the interface description using Drag & Drop. Local tags always start with a number sign #.

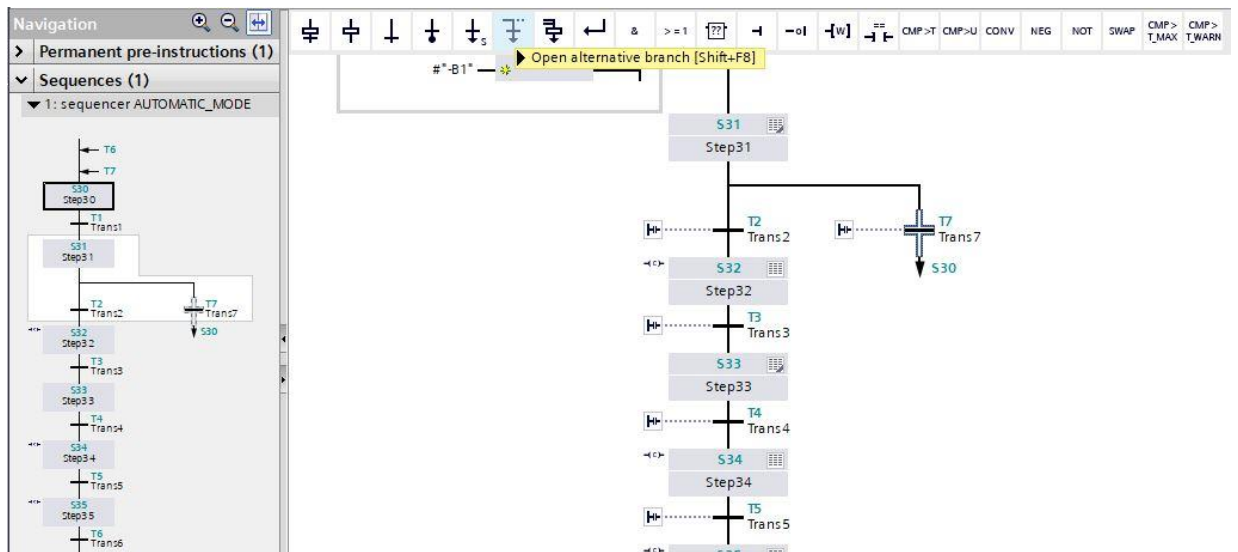
- Drag "Step and transition" onto the double arrow below Transition 1 to insert the next step with transition. The numbering continues automatically.



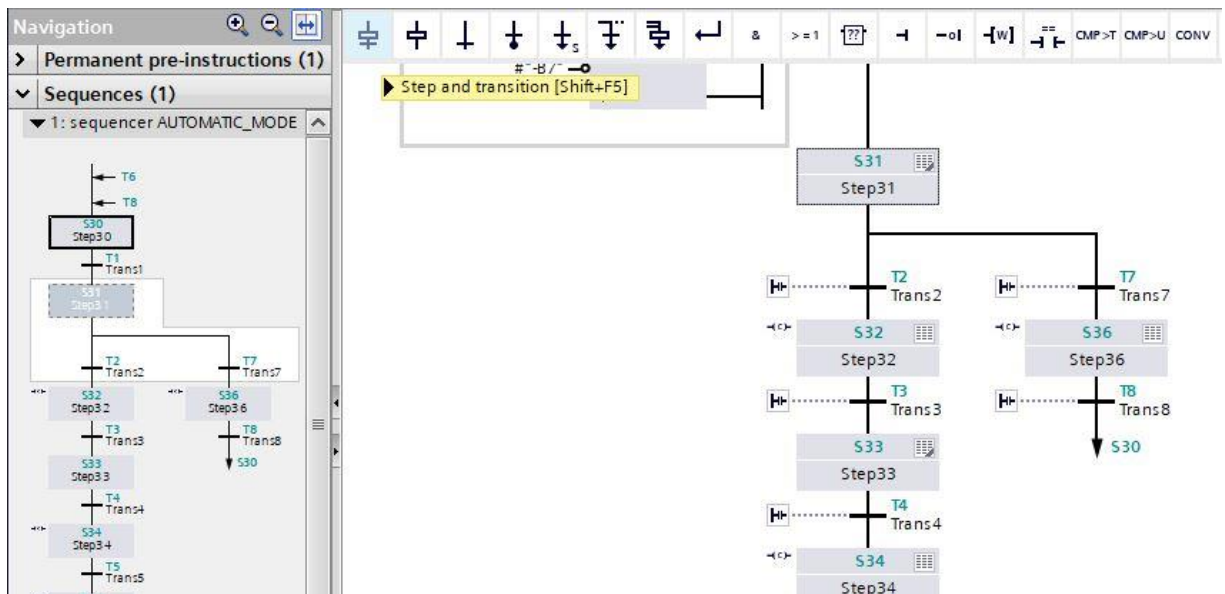
- Use Drag & Drop to move Steps S32 to S35 with their respective transitions into the work window.



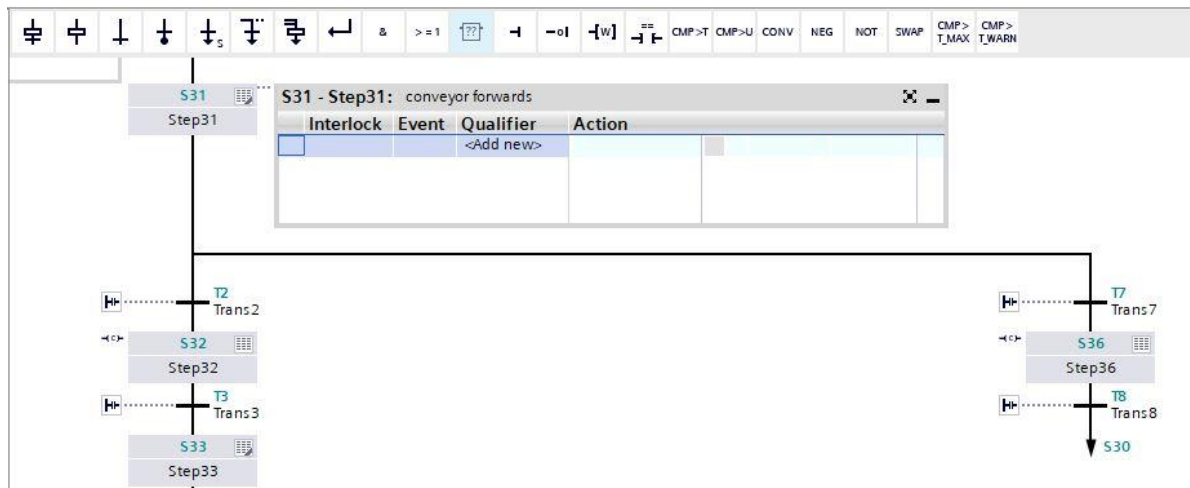
- After Step 31, the sequencer is divided using an alternative branch. Use Drag & Drop to move "Open alternative branch" onto the green square below Step 31. The alternative branch with Transition 7 is inserted.



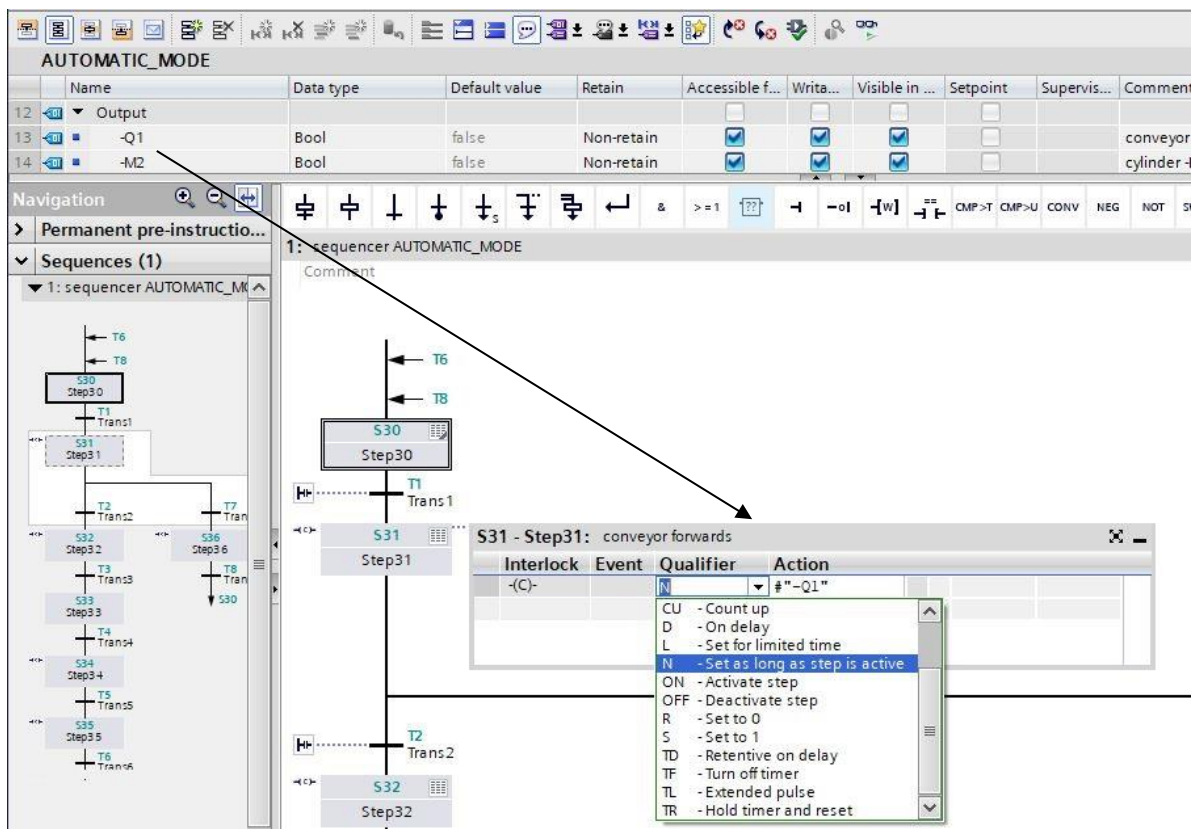
- Use Drag & Drop to move "Step and transition" onto the double arrow below Transition 7 to insert Step 36 with Transition 8.




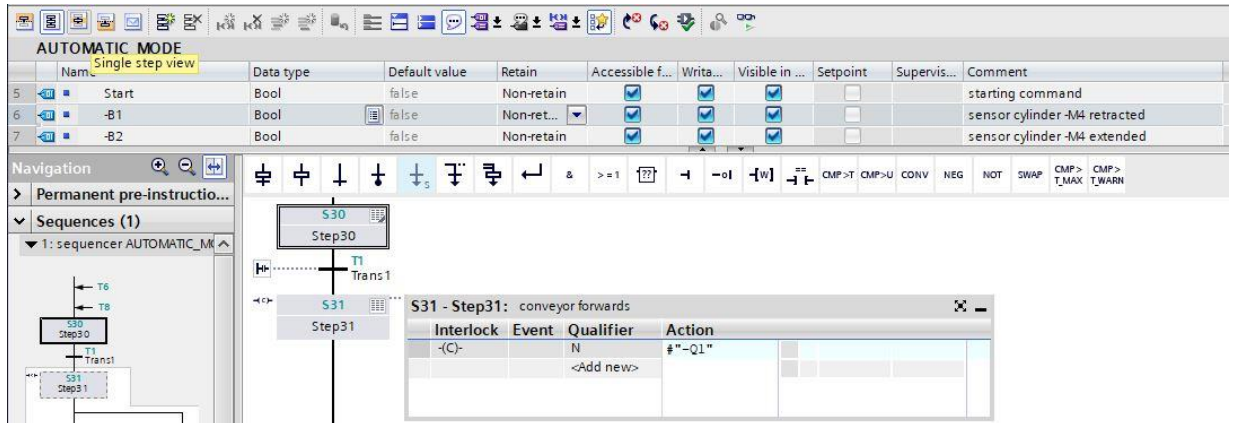
→ Now, open the action table in Step 31.



- In Step 31 the conveyor motor -M1 it is to be driven forwards at a fixed speed. To do so, set output -Q1 as the action, but only as long as Step 31 is active and sensor -B1 signals that the cylinder is in retracted position.
- Identify the action field in Step 31 with "conveyor forwards".
- Set an interlock condition "-(C)-" under Interlock and choose "Set as long as step is active" as qualifier.
- Drag the output tag "-Q1" into the action field.



→ Double-click Step 31 or press the single step view button  in order to input the interlock condition in the single step view.

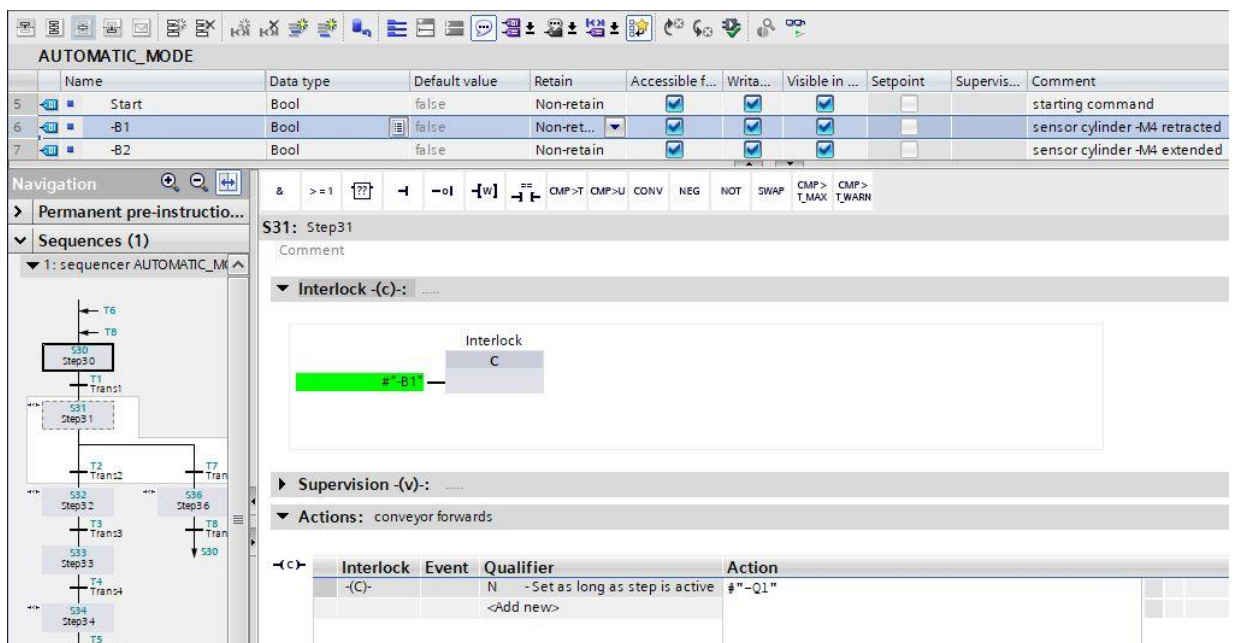


Name	Data type	Default value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Comment
5 Start	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		starting command
6 -B1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		sensor cylinder -M4 retracted
7 -B2	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		sensor cylinder -M4 extended

Interlock	Event	Qualifier	Action
-(C)-	N	<Add new>	#"-Q1"

Note:

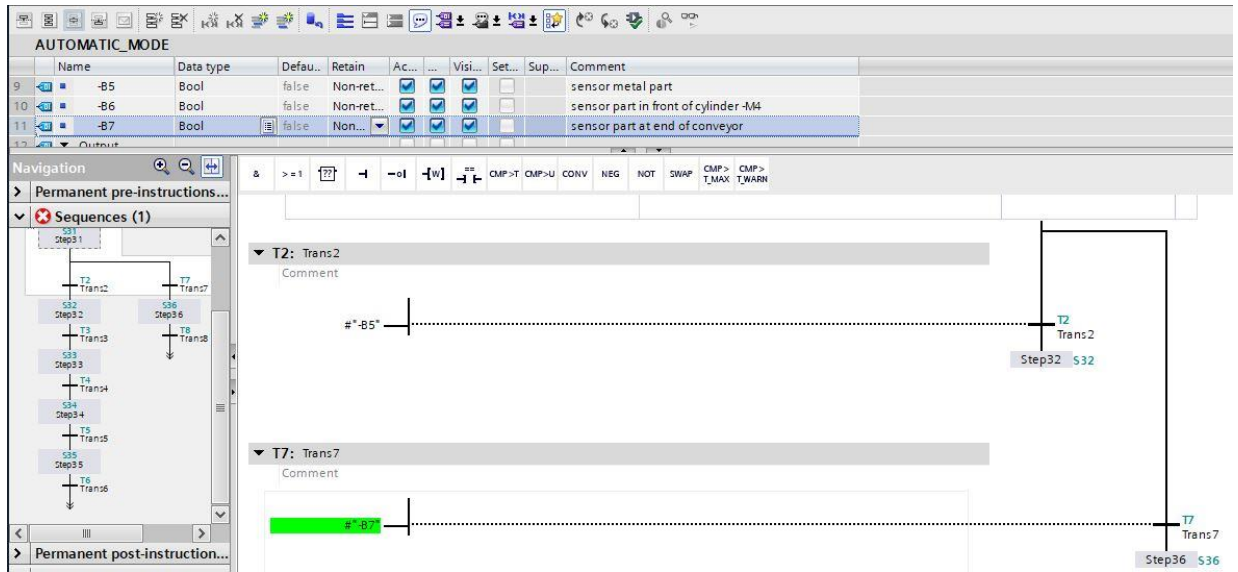
- You can find detailed information on interlock conditions/interlock in the manuals or the online help.
- Tag "-B1" can now be set as the condition for the interlock in the single step view. Drag tag "-B1" onto the input of Interlock C.




Name	Data type	Default value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Comment
5 Start	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		starting command
6 -B1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		sensor cylinder -M4 retracted
7 -B2	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		sensor cylinder -M4 extended

Interlock	Event	Qualifier	Action
-(C)-	N	- Set as long as step is active	#"-Q1"

- Drag tag "-B5" for Transition 2 and tag "-B7" for Transition 7 as the step-enabling condition for the step below.



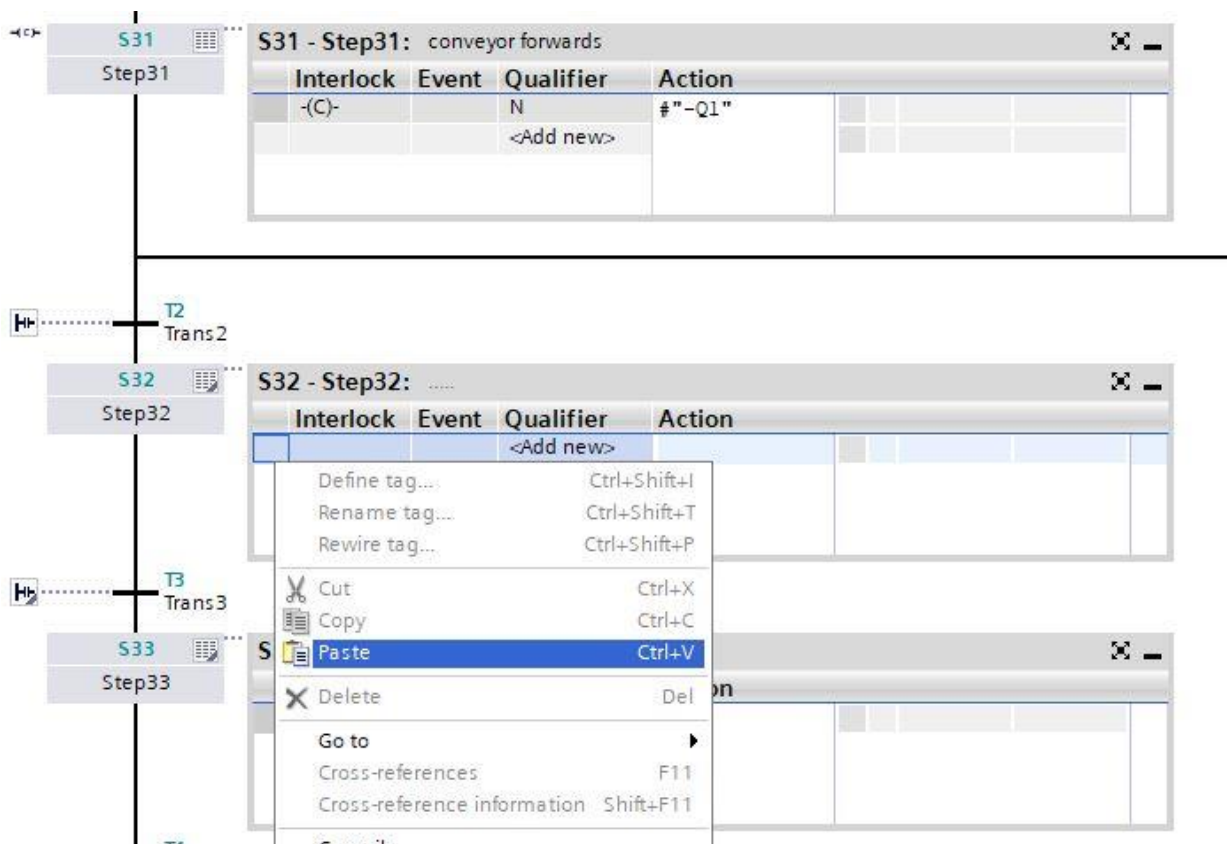
- Switch to the sequencer view  and enter the designation "sensor metal part" for Transition 2 and "sensor part at end of conveyor" for Transition 7.



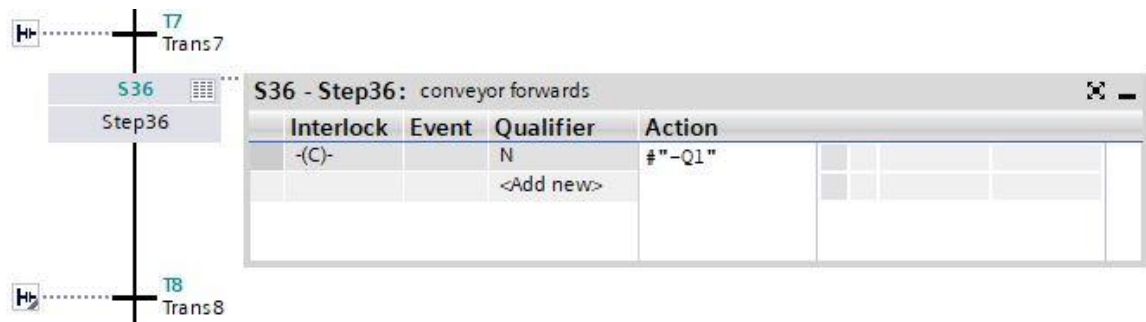
→ Select and copy the first row in the action window from Step 31.




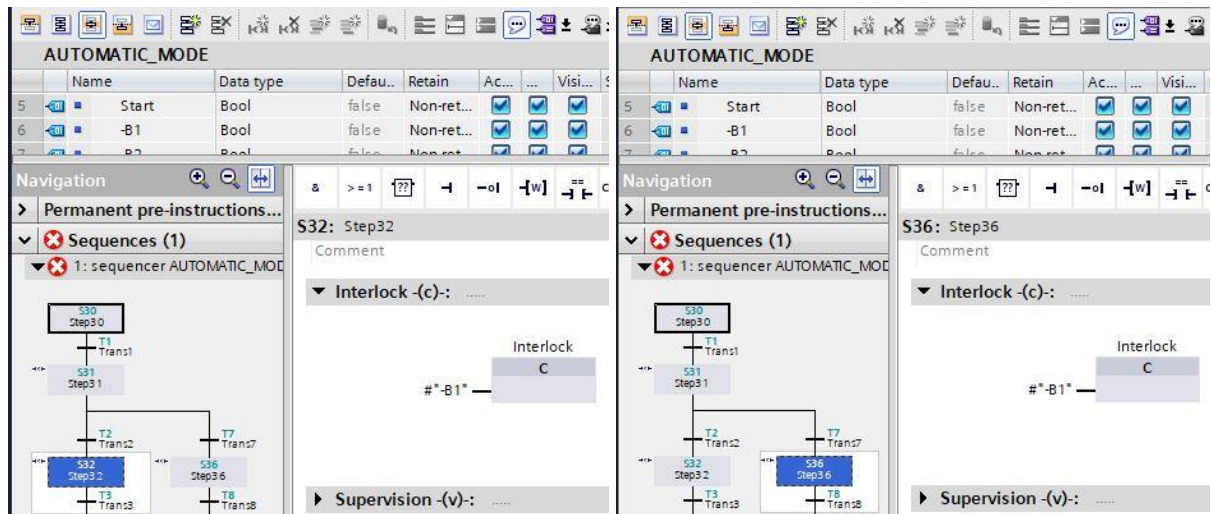
→ Paste the copied row in Step 32 and Step 36.



→ Identify the action fields in Steps 32 and 36 with "conveyor forwards".

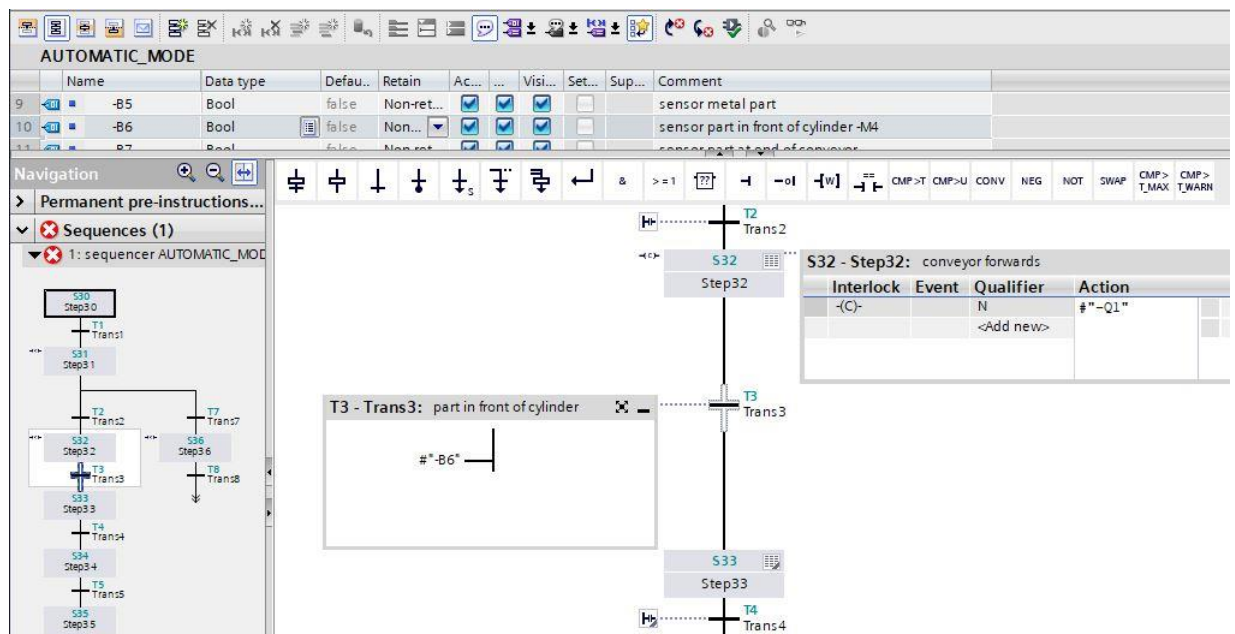



→ As shown previously for Step 31, the interlock condition "-B1" must now be set as Interlock C in Step 32 and Step 36 in the single step view  by dragging from the interface. The -(C)- to the left of the step field indicates that an interlock is programmed in this step.

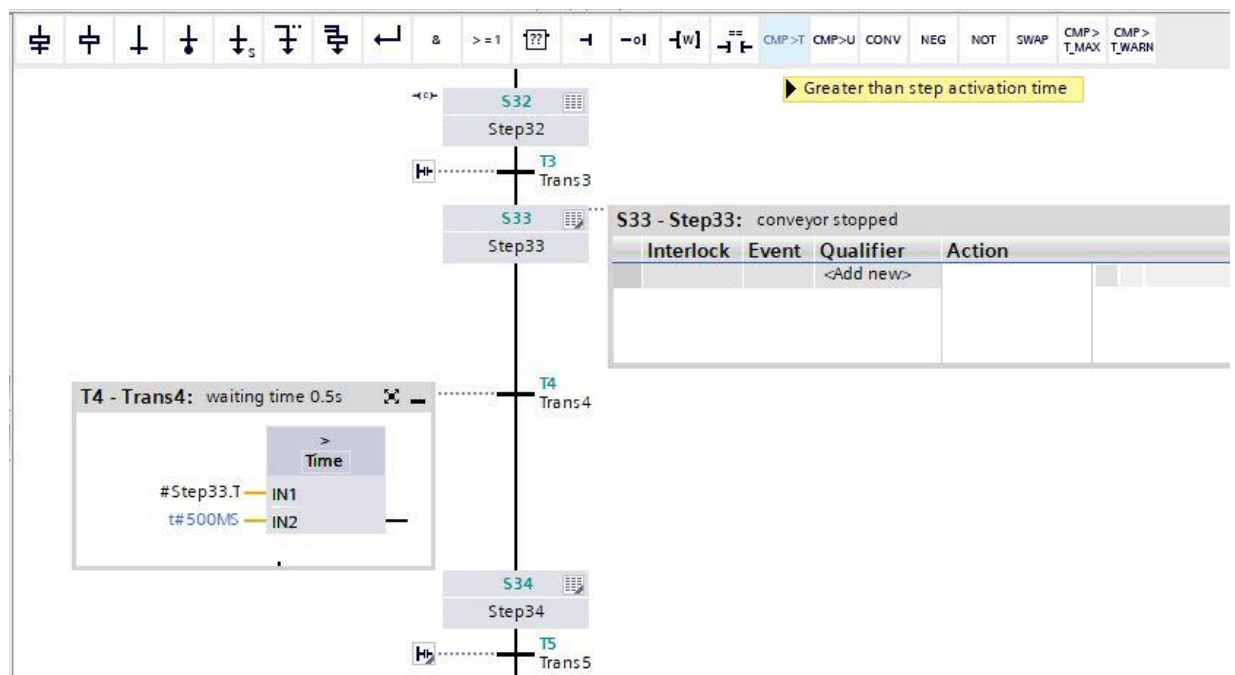


→ Drag tag "-B6" onto Transition 3 as the step-enabling condition.

→ Identify Transition 3 with "part in front of cylinder".




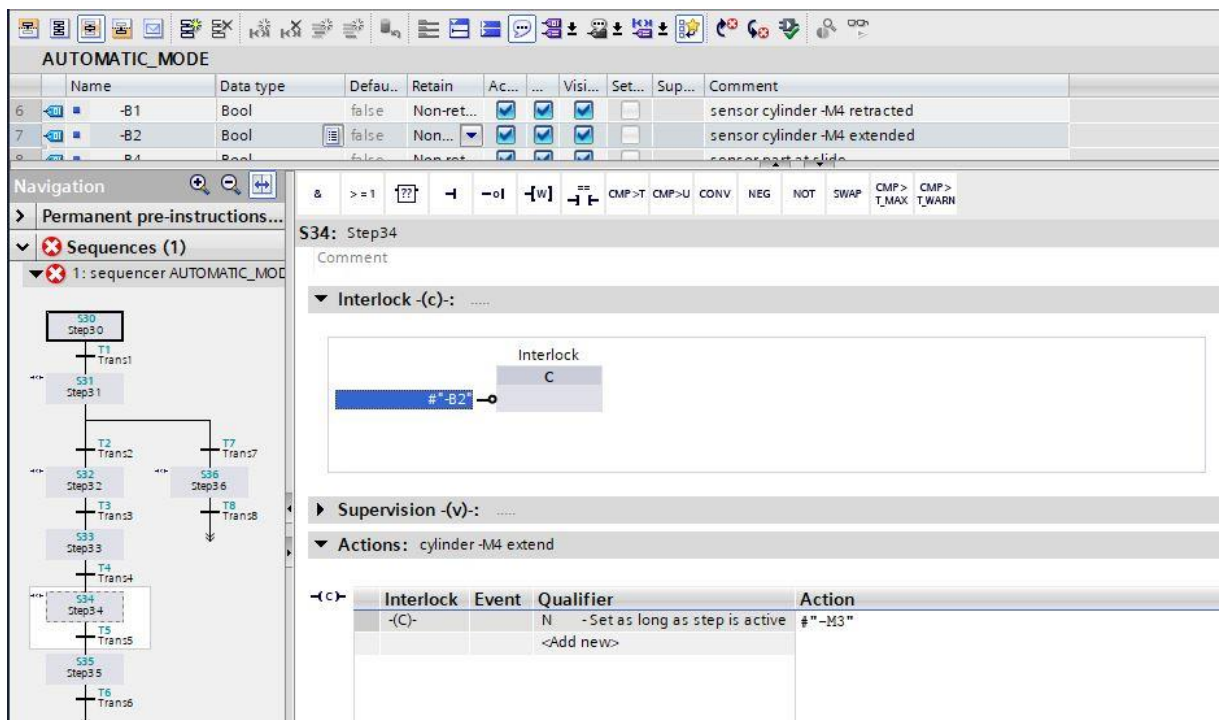
- The conveyor will be stopped in Step 33. An action does not have to be programmed here, because the advancement to Step 33 causes conveyor operation "-Q1" in Step 32 to stop due the "Set as long as step is active" qualifier. After a wait time of 0.5 seconds, the next step is to be activated.
- Drag the comparator "Greater than step activation time" onto the green square  in the window of Transition 4 and enter **T#500MS** for the time.
- Identify the action window with "conveyor stopped" and the transition window with "waiting time 0.5s"



- In Step 34, cylinder -M4 for pushing the metal part is to be extended by the control of "-M3" shown here, but only as long as it has not reached its end position.



- Double-click Step 34 or press the single step view button  in order to input the interlock condition in the single step view.
- Tag "-B2" can now be set as the condition for the interlock in the single step view. Drag tag "-B2" onto the input of Interlock C. Negate tag "-B2" because the cylinder is only driven as long as it has not reached its end position.



AUTOMATIC_MODE

Name	Data type	Default	Retain	Ac...	...	Visi...	Set...	Sup...	Comment
6	-B1	Bool	false	Non-ret...		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor cylinder -M4 retracted
7	-B2	Bool	false	Non...		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor cylinder -M4 extended

Navigation

- Permanent pre-instructions...
- Sequences (1)
 - 1: sequencer AUTOMATIC_MOD

S34: Step34

Comment

Interlock -(c)-:

Interlock

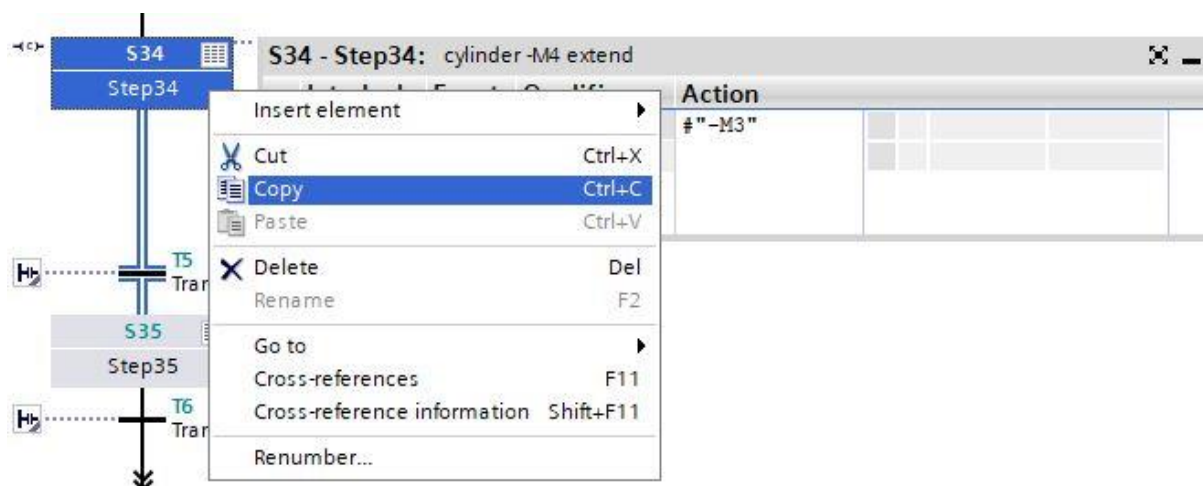
#'-B2'

Supervision -(v)-:

Actions: cylinder -M4 extend

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-M3"
		<Add new>	

- The transition to Step 34 of the GRAFCET means: When the cylinder has reached its front-end position "-B2", a wait time of 0.5 seconds begins. An intermediate step must be inserted for this in S7-GRAPH.
- Select and copy Step 34 and the associated Transition 5.



S34 - Step34: cylinder -M4 extend

Insert element

- Cut Ctrl+X
- Copy Ctrl+C**
- Paste Ctrl+V
- Delete Del
- Rename F2
- Go to
- Cross-references F11
- Cross-reference information Shift+F11
- Renumber...

Action

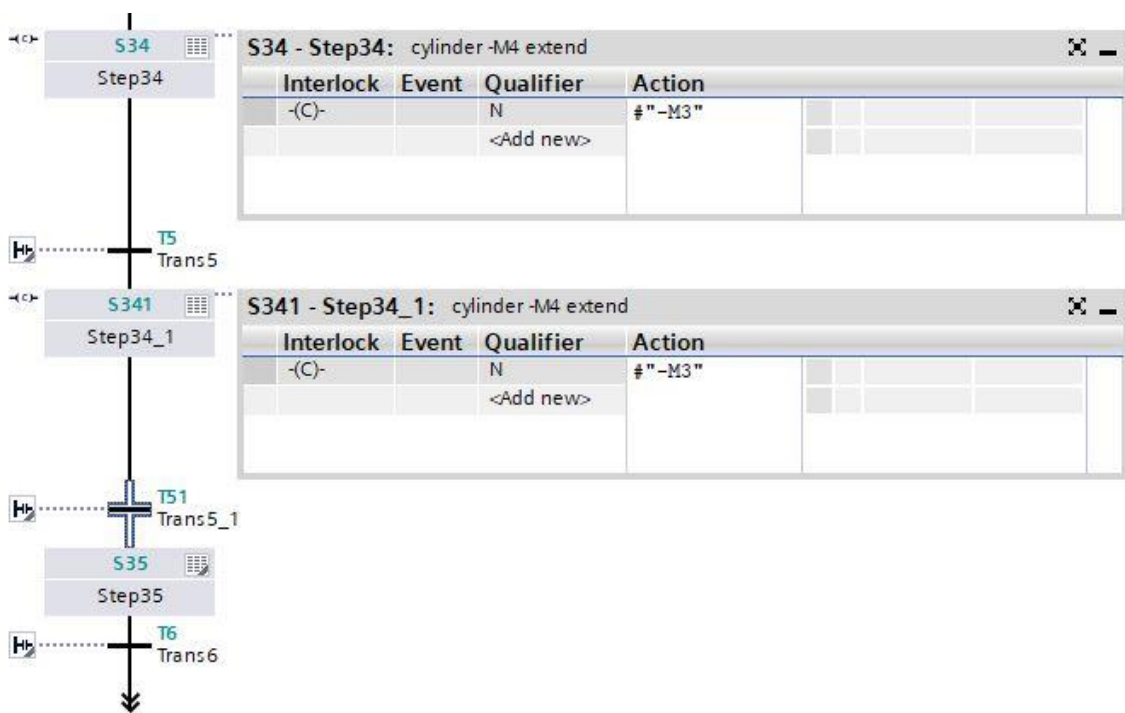
#"-M3"

→ Select Transition 5 and insert the copied step with transition.




→ Change the step number and the tag designations of the inserted step.

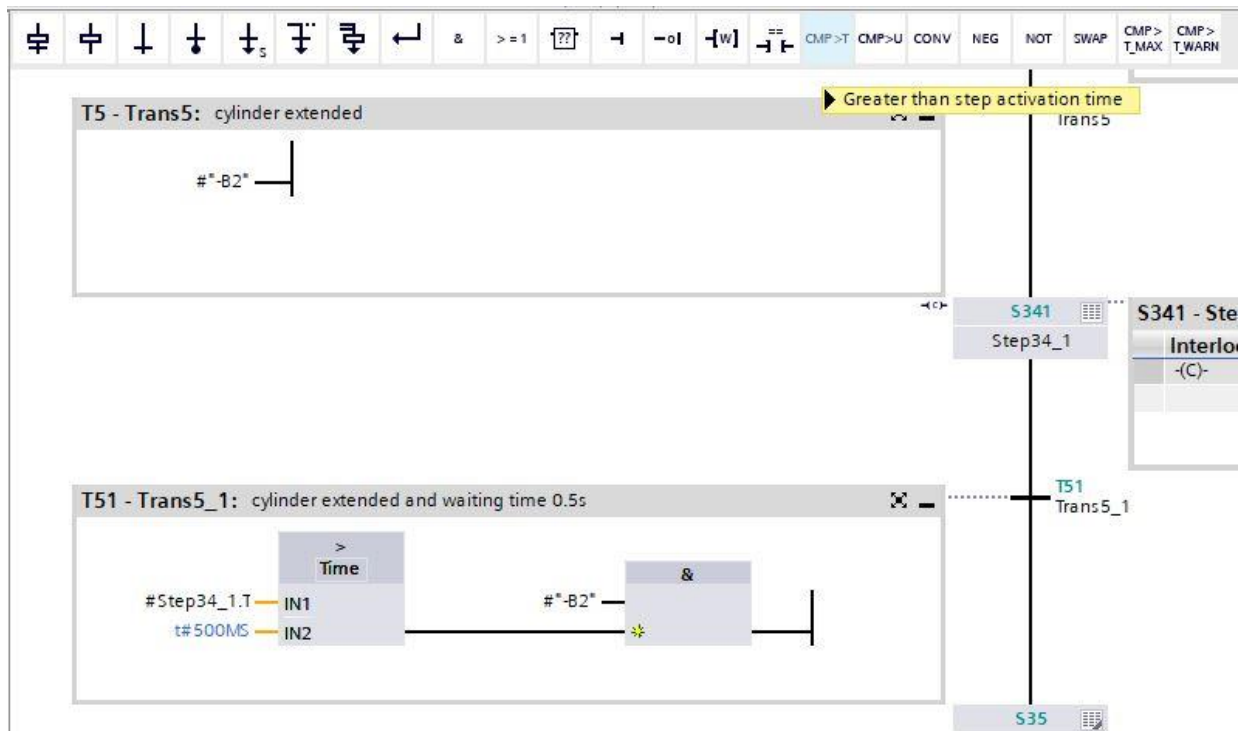
→ Change the transition number and the tag designations of the inserted transition.




→ Drag tag "-B2" onto Transition 5 as the step-enabling condition.

→ First, drag an AND logic operation onto Transition 51 as the step-enabling condition, and then drag tag "-B2" onto the first input of the AND logic operation. Now drag the comparator "Greater than step activation time" onto the green square  at the second input and enter **T#500MS** for the time.

→ Designate the transitions as shown here.



- In Step 35, cylinder -M4 for pushing the metal part is to be retracted by the control of "-M2" shown here, but only as long as it has not reached its end position.
- Double-click Step 35 or press the single step view button  in order to input the interlock condition in the single step view.
- Tag "-B1" can now be set as the condition for the interlock in the single step view. Drag tag "-B1" onto the input of Interlock C. Negate tag "-B1" because the cylinder is only driven as long as it has not reached its end position.

AUTOMATIC_MODE

Name	Data type	Defau...	Retain	Ac...	...	Visi...	Set...	Sup...	Comment
6	-B1	Bool	false	Non...					sensor cylinder -M4 retracted
7	-B2	Bool	false	Non-ret...					sensor cylinder -M4 extended
8	-B3	Bool	false	Non-ret...					sensor part at slide

Navigation

> Permanent pre-instructions...

Sequences (1)

1: sequencer AUTOMATIC_N

S35: Step35

Comment

Interlock -(c)-:

Interlock C

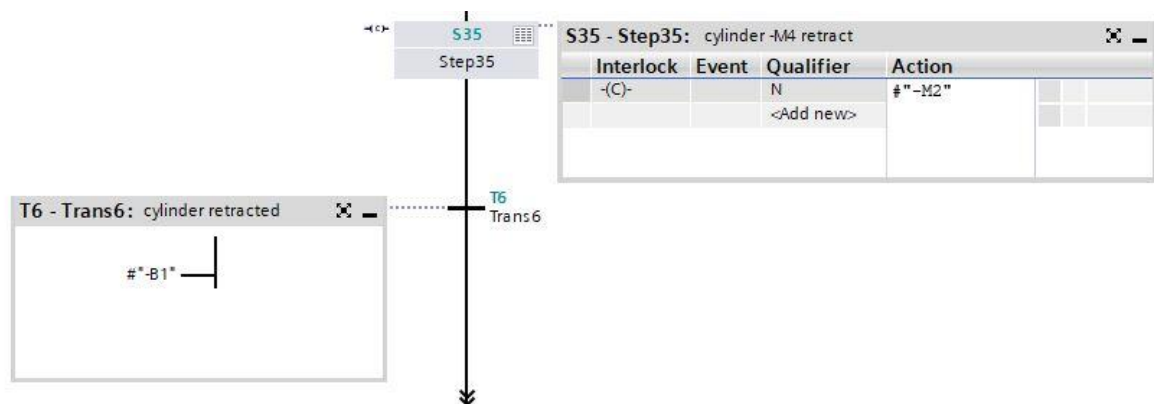
#"-B1"

Supervision -(v)-:

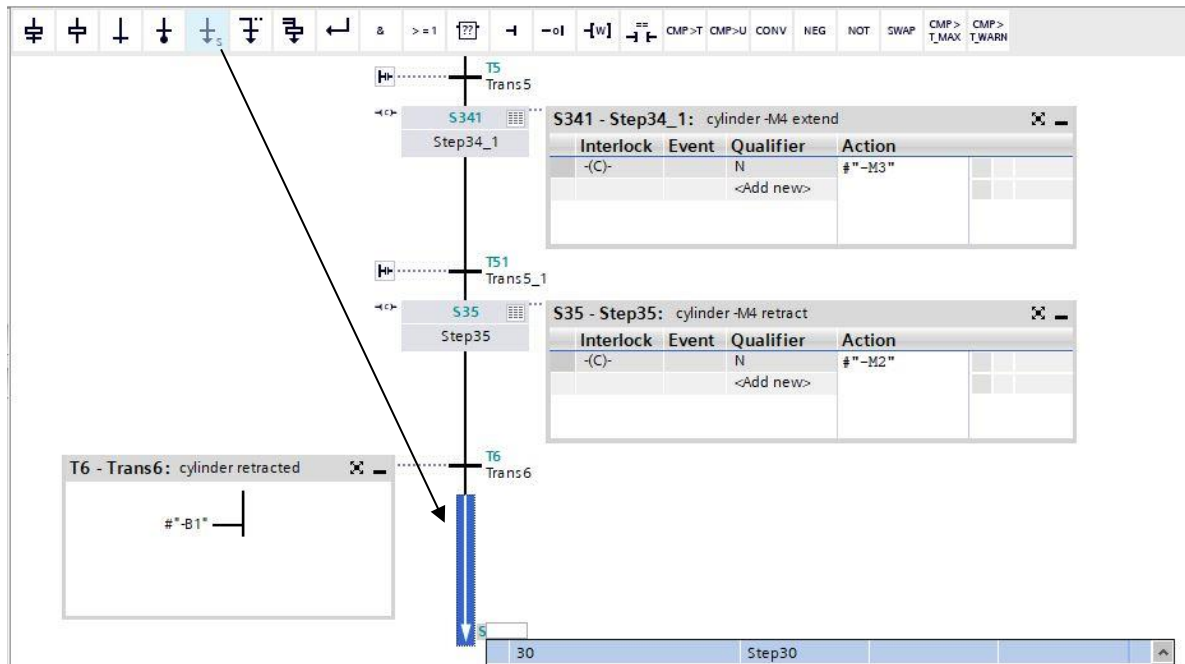
Actions: cylinder -M4 retract

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-M2"
		<Add new>	

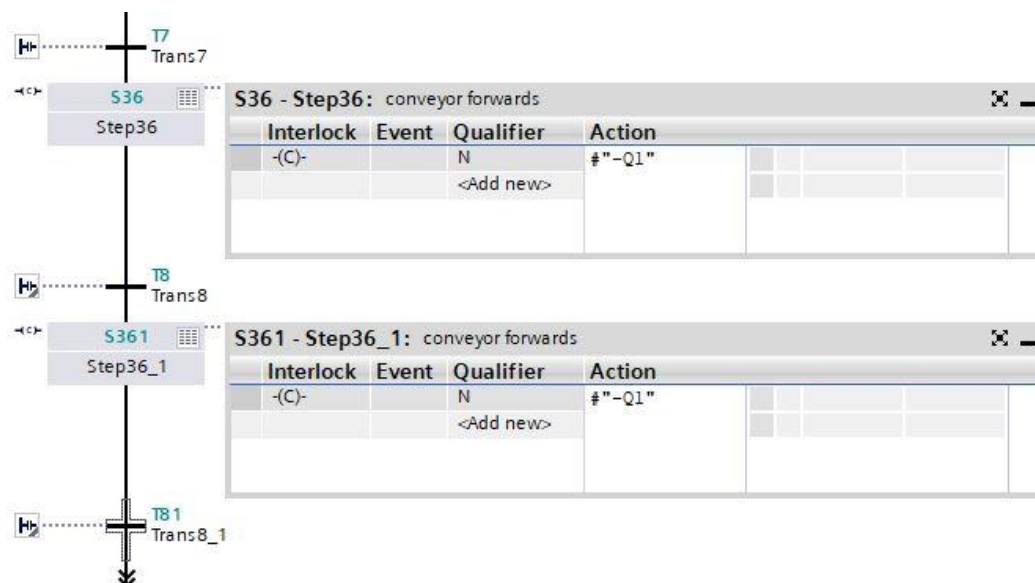
- Switch to the sequence view and enter the designation "cylinder retracted" for Transition 6.
- Drag tag "-B1" onto Transition 6 as the step-enabling condition.




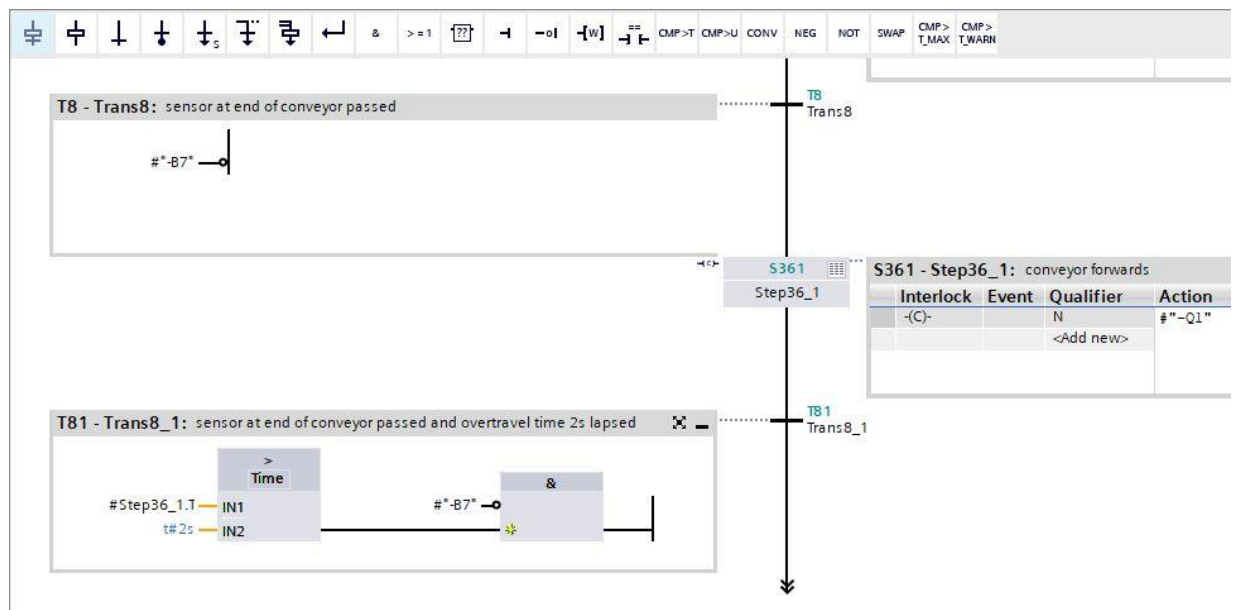
- At the end of the sequence, we set a jump to Step 30.
- Drag a jump to the double arrow and select Step 30 as the jump destination.



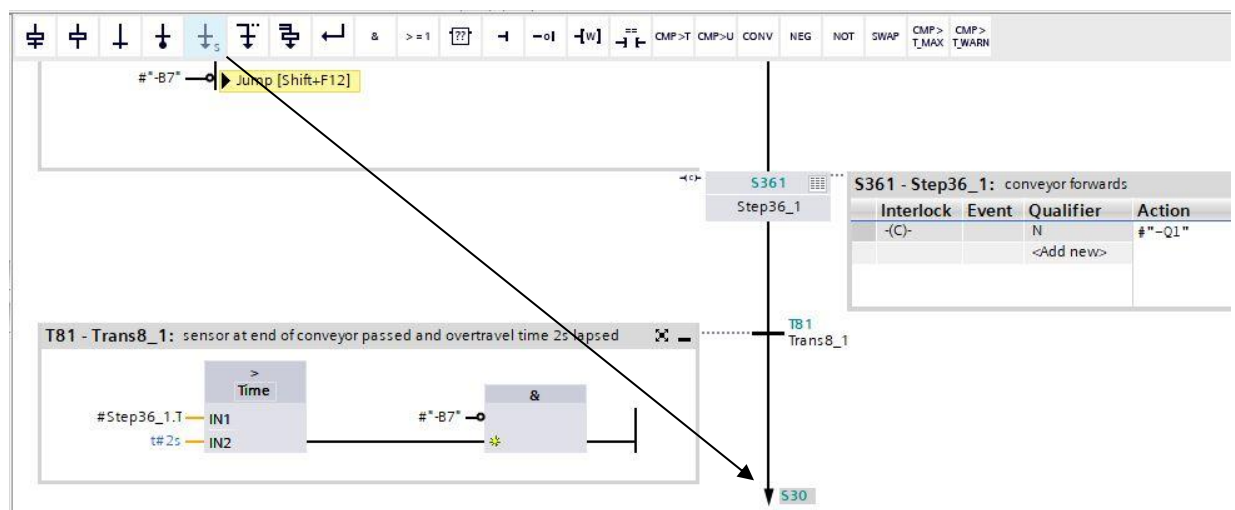
- The transition after Step 36 of the GRAFCET means that a wait time of 2 seconds begins when the plastic part has passed light sensor "-B7". In other words, when "-B7" is no longer activated, the wait time starts. An intermediate step must be inserted for this in S7-GRAPH like for Step 34.
- Repeat the procedure used for Step 34. (→ Select and copy Step 36 and Transition 8. → Select Transition 8 and insert the copied step with transition. → Change step number to S361 and step tag to Step36_1. → Change transition number to T81 and tag designation Trans8_1.)



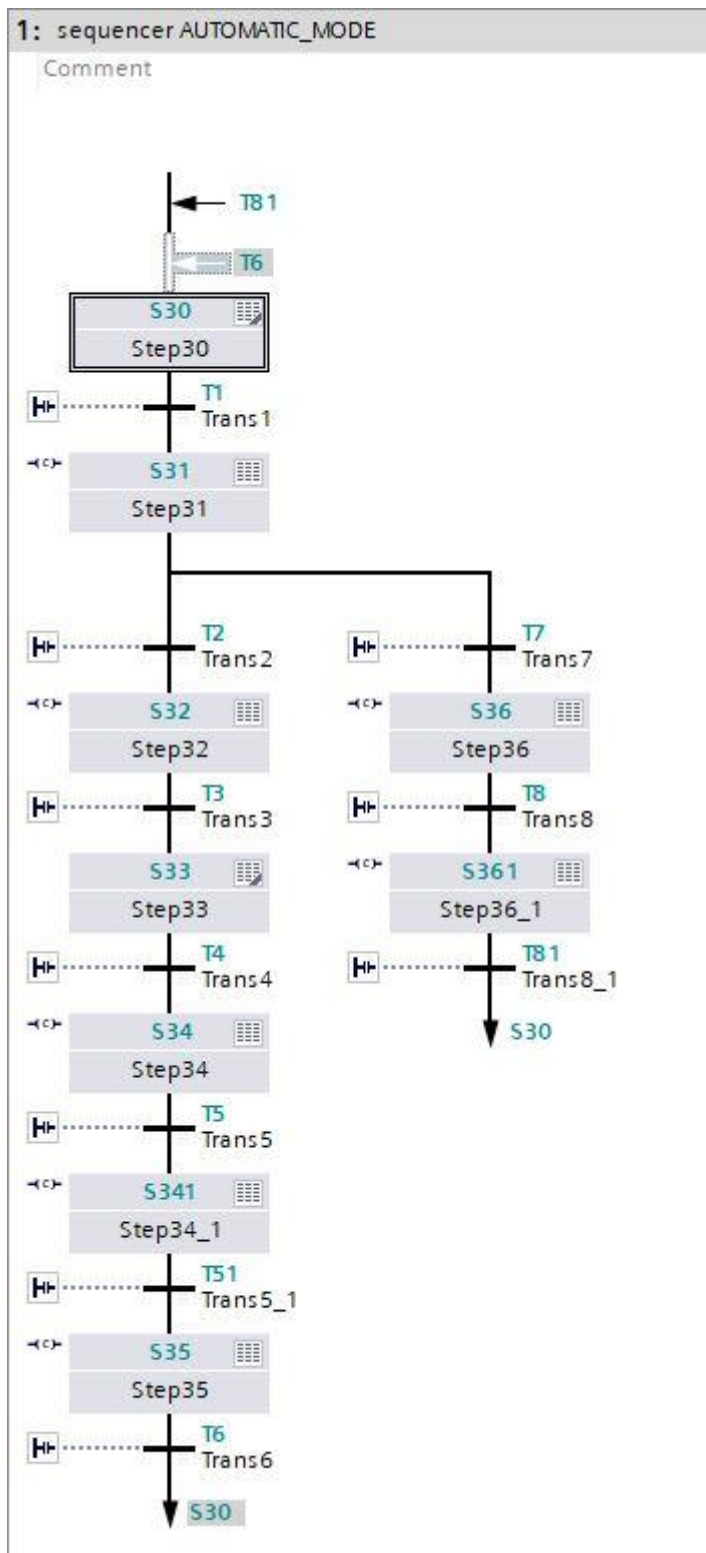
- Drag tag "-B7" with Negation onto Transition 8 as the step-enabling condition.
- First, drag an AND logic operation onto Transition 81 as the step-enabling condition, and then drag tag "-B7" with a negation onto the first input of the AND logic operation. Now drag the comparator "Greater than step activation time" onto the green square  at the second input and enter **T#2S** for the time.
- Designate the transitions.



- At the end of the sequence, we set a jump to Step 30.
- Drag a jump to the double arrow and select Step 30 as the jump destination.



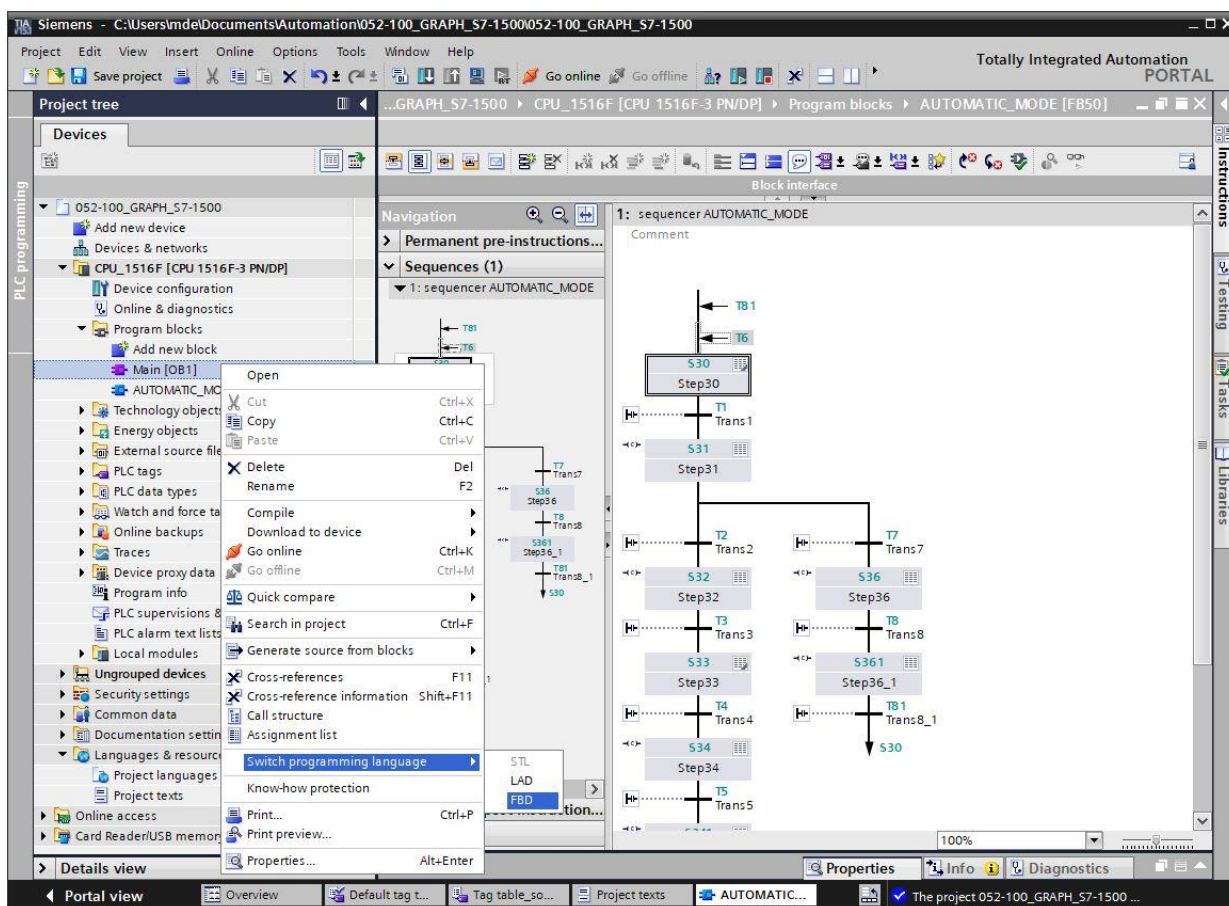
→ The S7-GRAPH sequencer for sequencer AUTOMATIC_MODE is now finished.



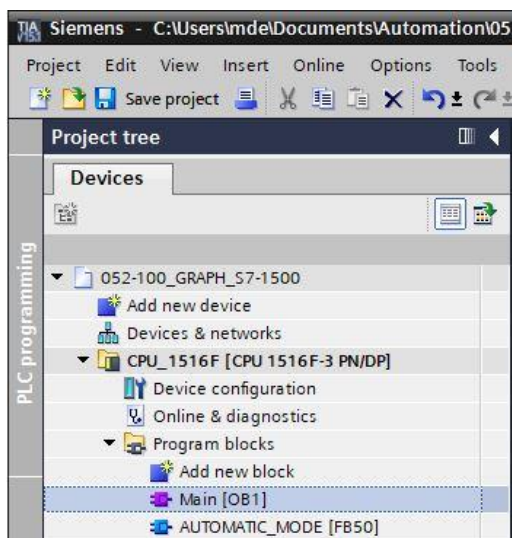
→ To save your project, select the “ Save project” button in the menu.

7.8 Programming the organization block OB1

→ Before programming organization block "Main [OB1]", we switch the programming language to FBD (Function Block Diagram). To do this, first click on "Main [OB1]" in the "Program blocks" folder. (→ CPU_1516F [CPU 1516F-3 PN/DP] → Program blocks → Main [OB1] → Switch programming language → FBD)

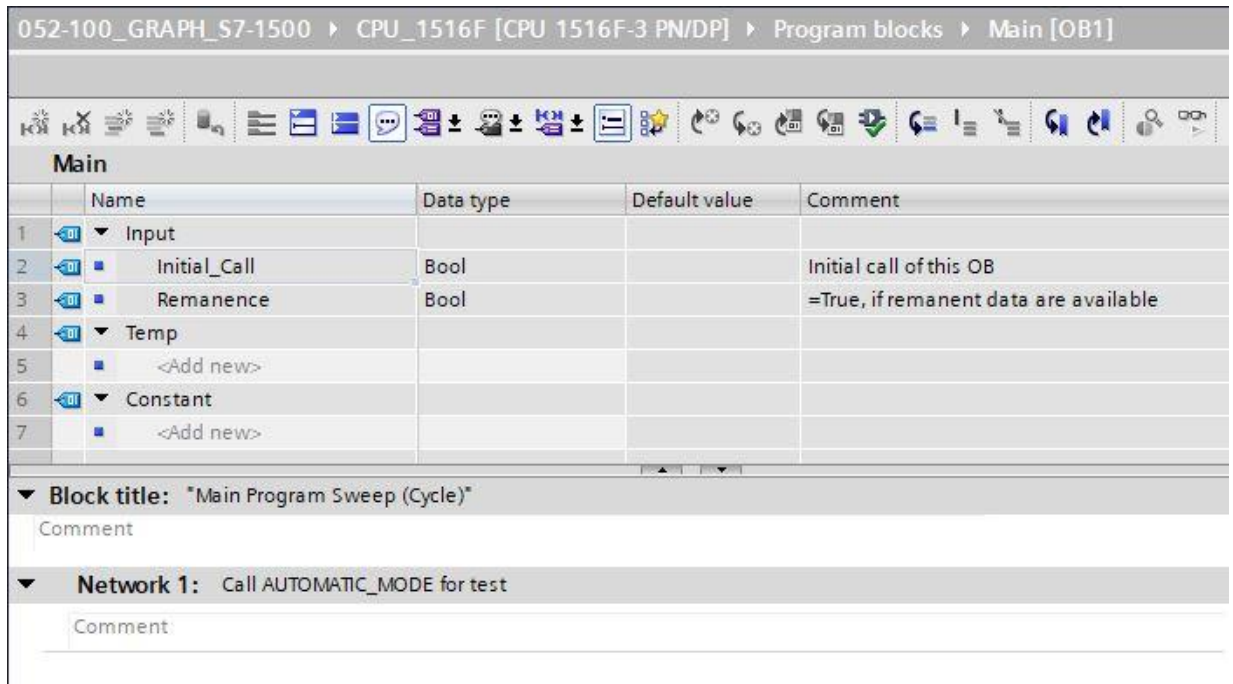


→ Open the "Main [OB1]" organization block with a double-click.

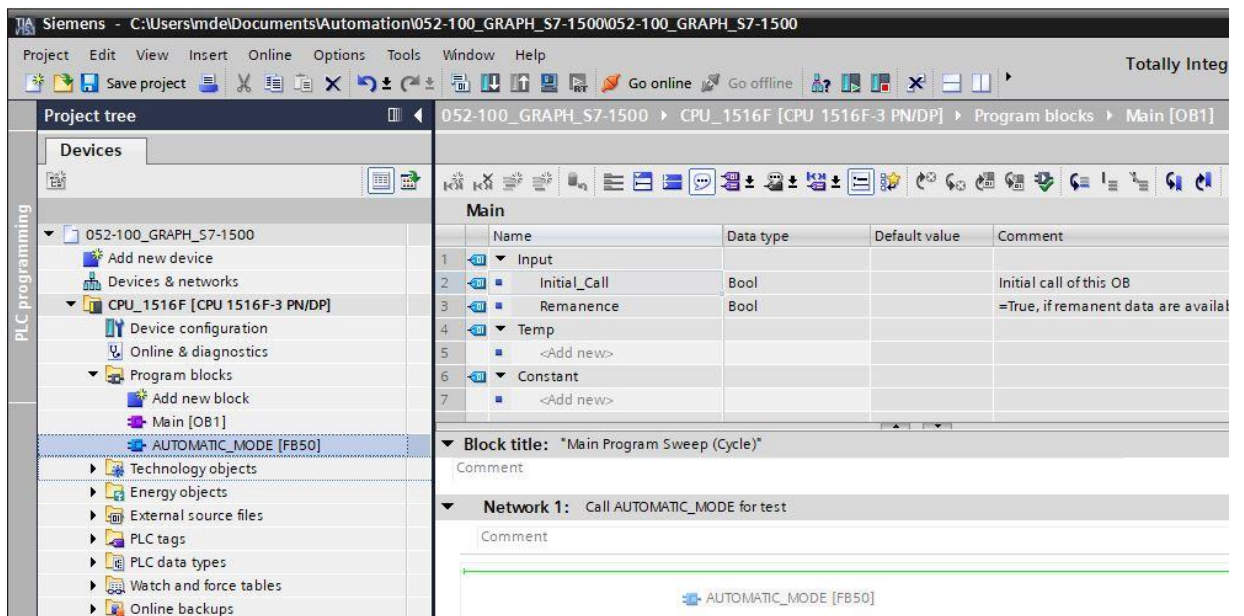


→ Assign Network 1 the name "Call AUTOMATIC_MODE for test".

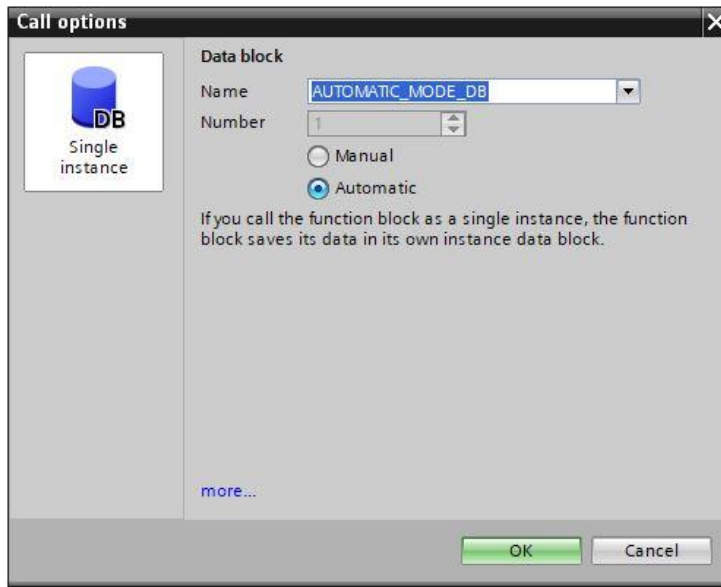
(→ Network 1:... → Call AUTOMATIC_MODE for test)



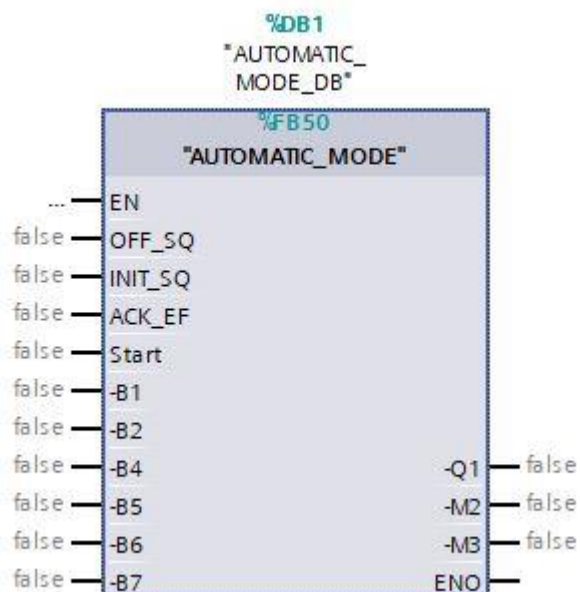
→ Use Drag & Drop to move your "AUTOMATIC_MODE [FB50]" function block onto the green line in Network 1.



- The instance data block for this call of FB50 is created automatically. Accept the default name and confirm with "OK".



- A block with the interface you defined, the instance data block and I/Os EN and ENO are inserted in Network 1.



- Select the "Tag table sorting station" in the project tree and move the desired global tags from the Details view to the interface of the block using Drag & Drop (→ Tag table sorting station → Details view → -S0 → Start)

The screenshot shows the Siemens TIA Portal interface. The project tree on the left displays the 'Tag table sorting station' under 'PLC tags'. The details view on the right shows the 'Network 1: Call AUTOMATIC_MODE for test' block. The block interface shows the following inputs and outputs:

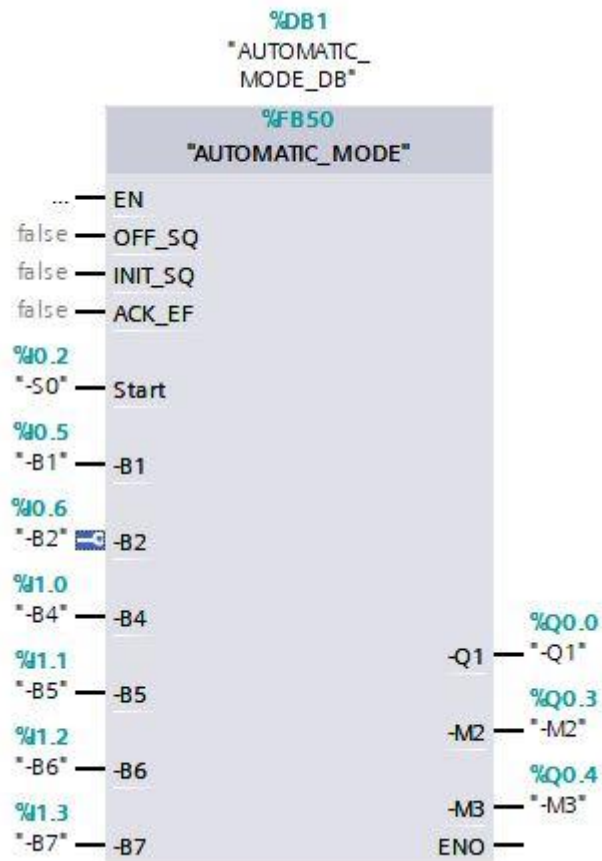
- EN: ...
- OFF_SQ: false
- INIT_SQ: false
- ACK_EF: false
- %I0.2: %I0.2
- S0: Start
- B1: false
- B2: false
- B4: false
- B5: false
- B6: false
- B7: false
- Q1: false
- M2: false
- M3: false
- ENO: ...

- Or enter the starting letters (e.g. "-B") of the desired global tag and select the global input tag "-B1" from the displayed list.

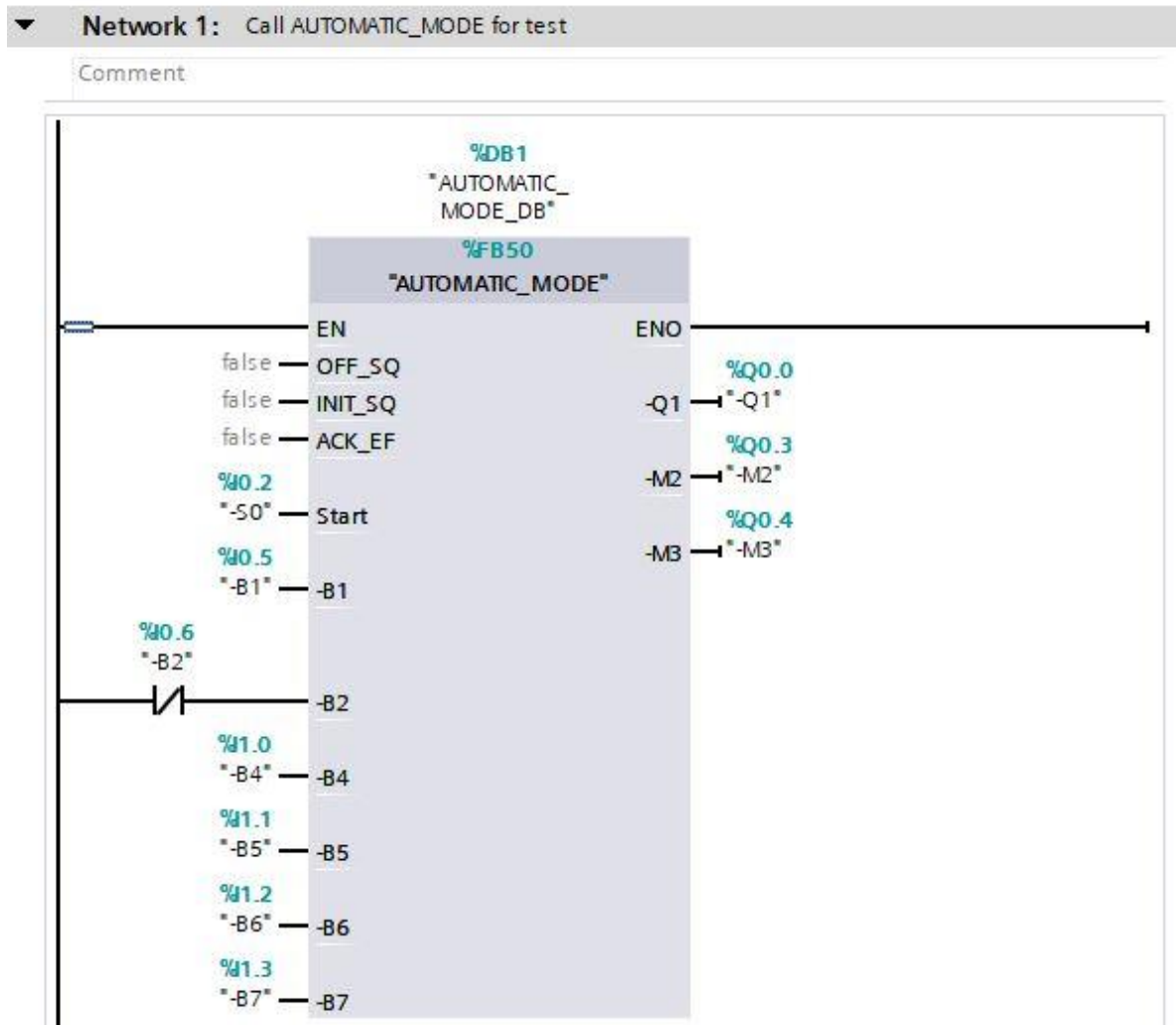
The screenshot shows the Siemens TIA Portal interface. The project tree on the left displays the 'Tag table sorting station' under 'PLC tags'. The details view on the right shows the 'Network 1: Call AUTOMATIC_MODE for test' block. The block interface shows the following inputs and outputs:

- EN: ...
- OFF_SQ: false
- INIT_SQ: false
- ACK_EF: false
- %I0.2: %I0.2
- S0: Start
- B1: false
- B2: false
- B4: false
- B5: false
- B6: false
- B7: false
- Q1: false
- M2: false
- M3: false
- ENO: ...





- Insert the other input and output tags.
- Negate input "-B2", because this sensor is wired as a normally closed contact. The controller recognizes that the cylinder is in extended position ("-B2" is activated) when no voltage is present at the terminal of input I0.6, i.e. a 0 signal is present.

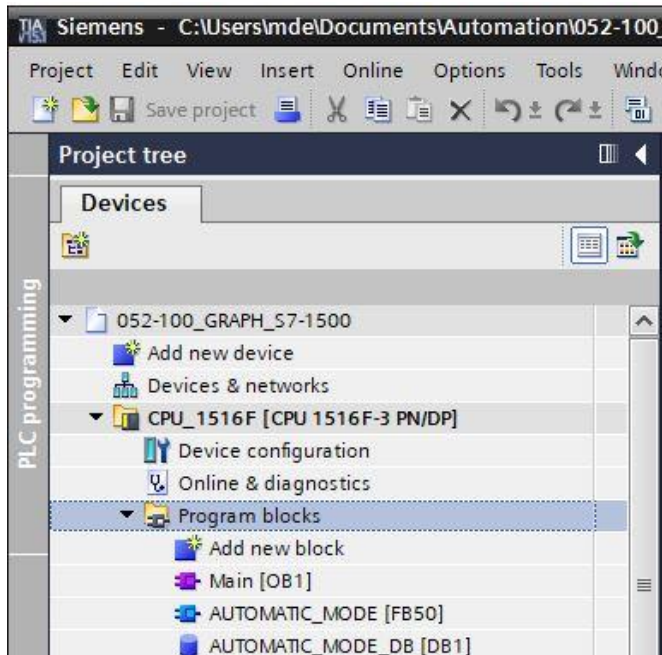


7.9 Result in the LAD programming language

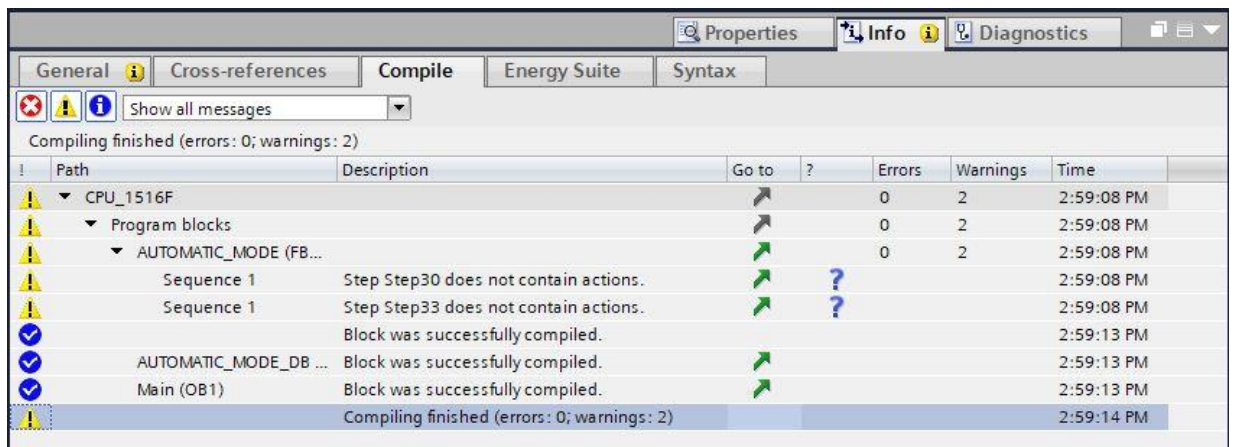


7.10 Saving and compiling the program


- To save your project, select the “ Save project” button in the menu. To compile all blocks, click the “Program blocks” folder and select the button “” for compiling in the menu. (→  Save project → Program blocks → )

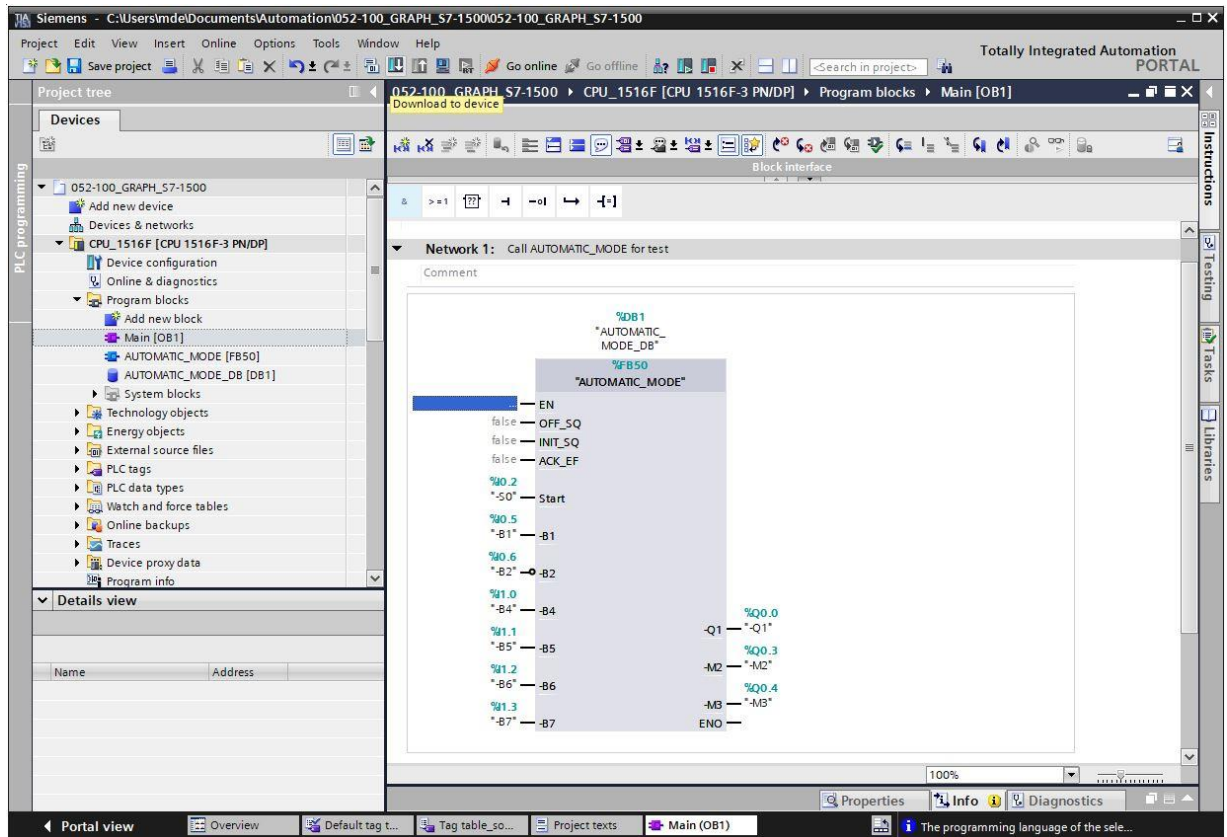


- The "Info", "Compile" area shows which blocks were successfully compiled.



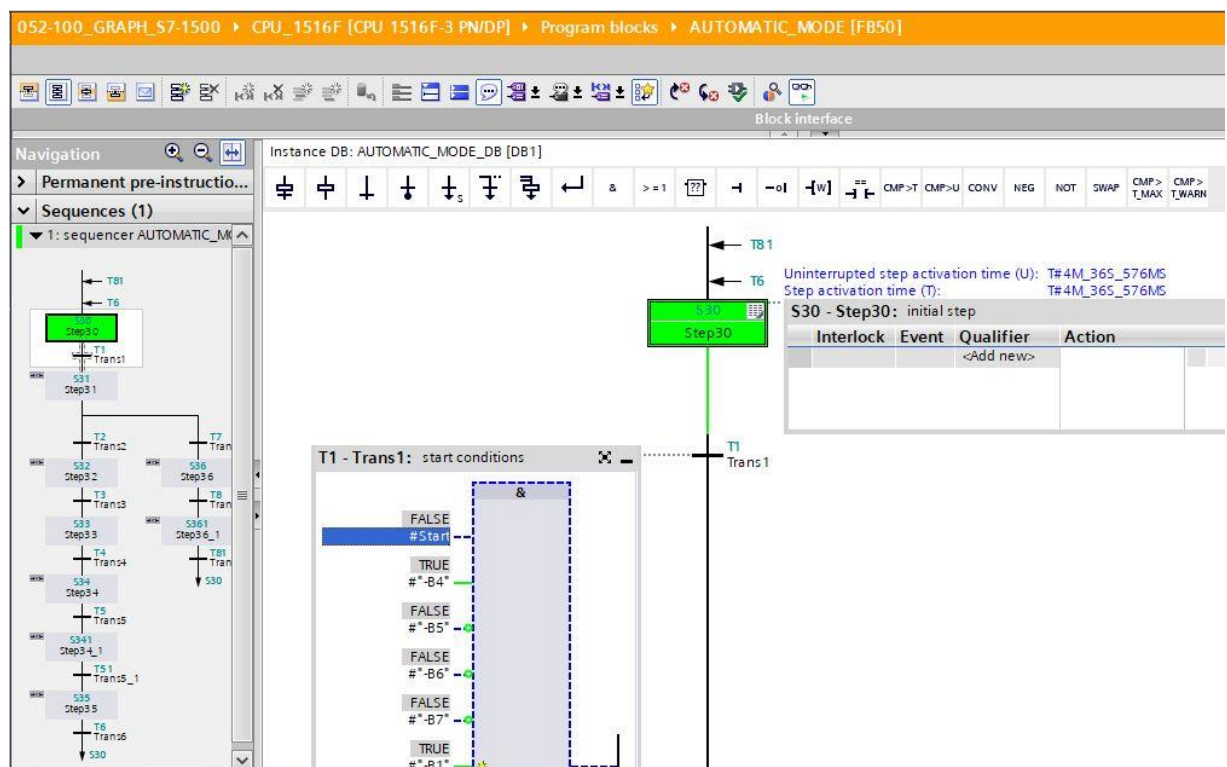
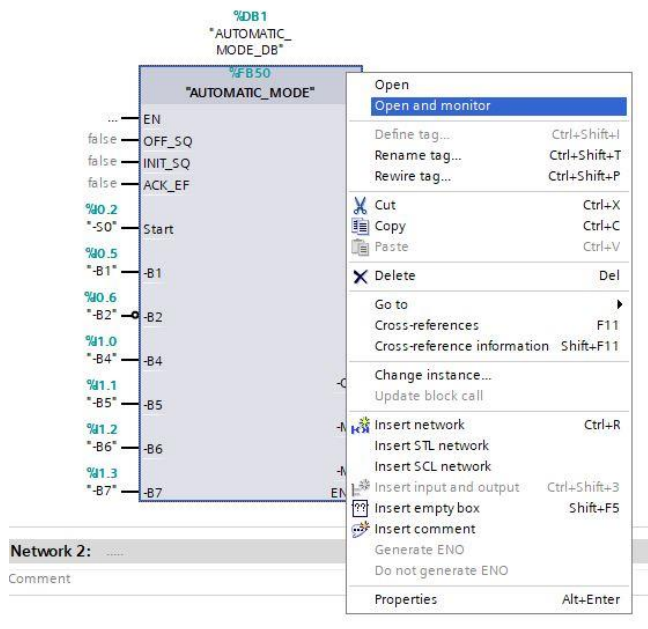
7.11 Downloading the program

→ After successful compilation, the complete controller with the created program, as previously described in the modules for hardware configuration, can be downloaded (→ ).



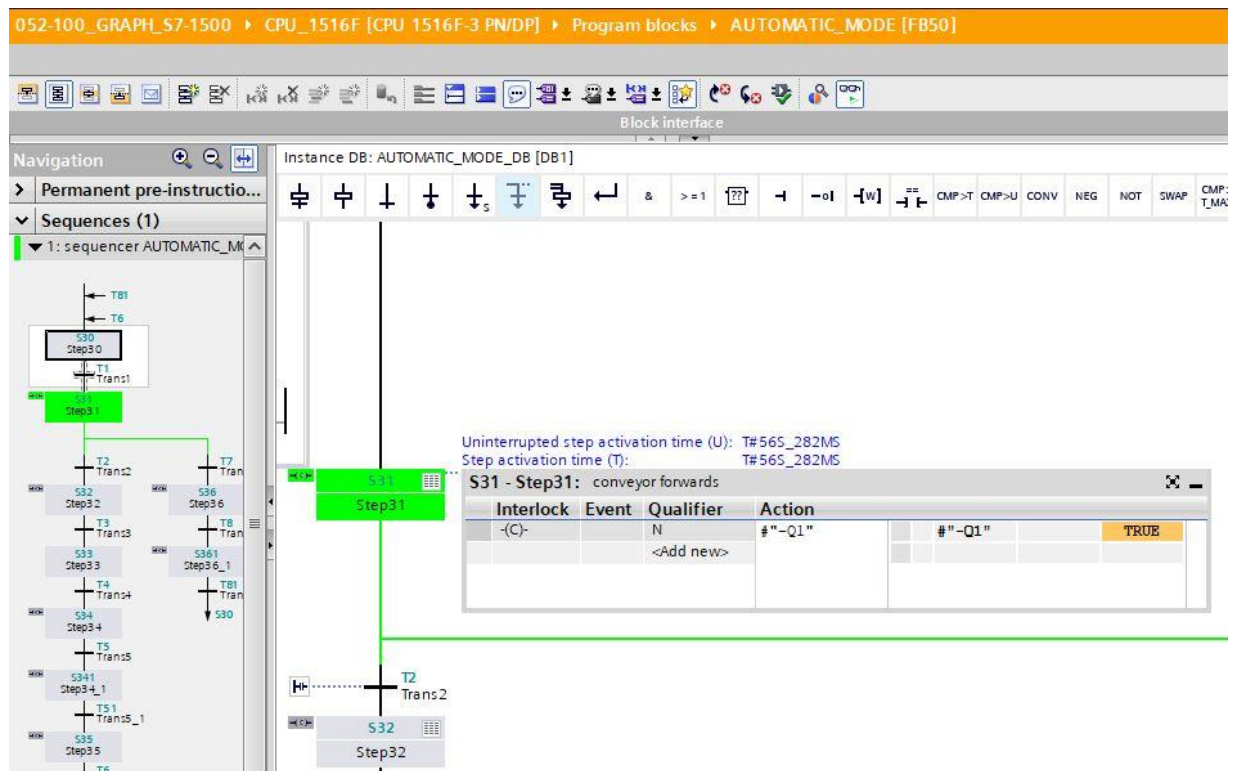
7.12 Monitoring program blocks

→ The "AUTOMATIC_MODE" [FB50] function block called in the "Main [OB1]" organization block can be selected directly for "Open and monitor" after right-clicking (→ "AUTOMATIC_MODE" [FB50] → Open and monitor).

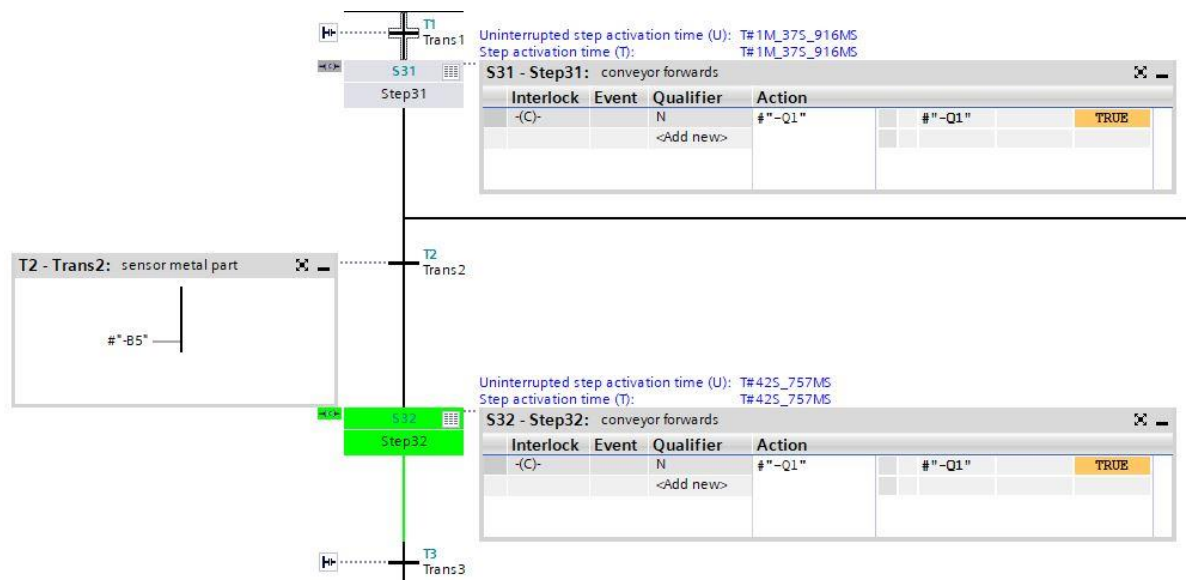


→ The initial step (Step 30) is already active.

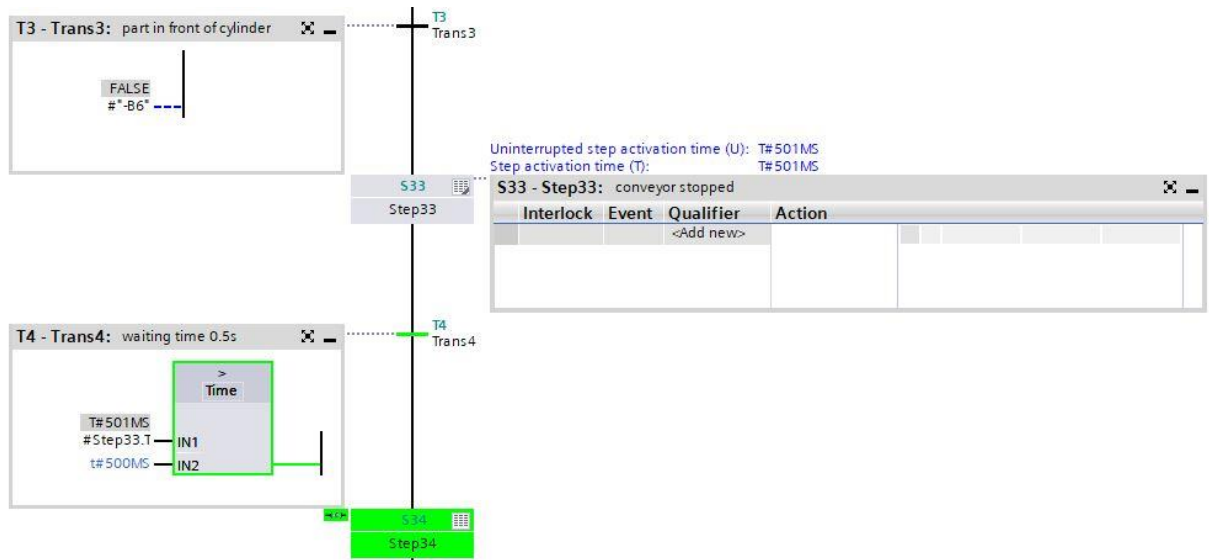
→ If the starting conditions are satisfied, the sequencer jumps to Step 31 and sets "-Q1" to TRUE.



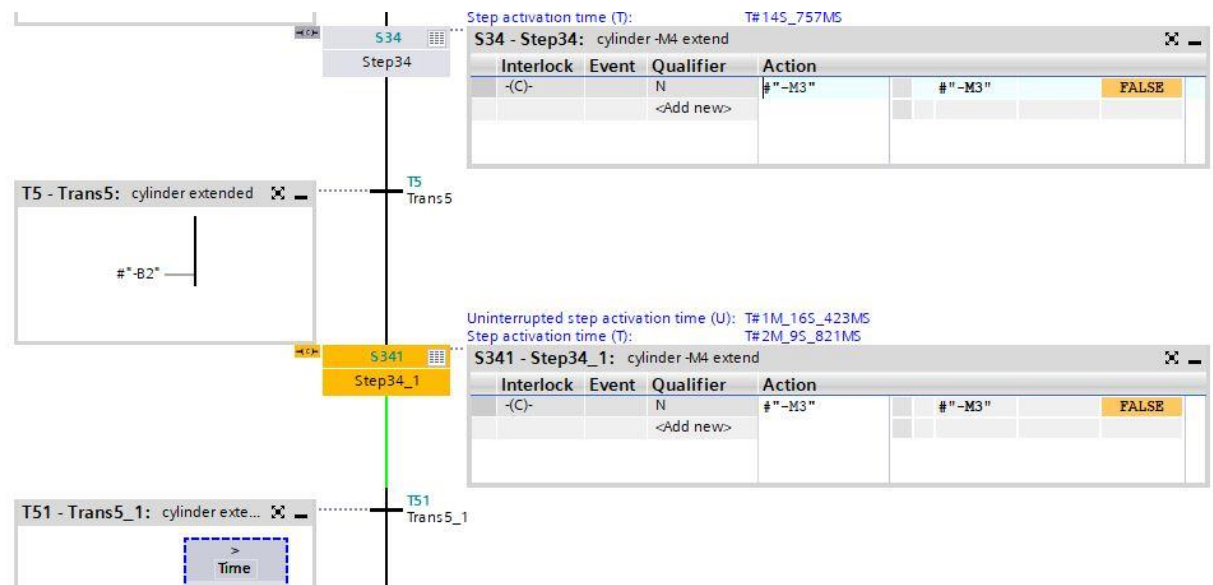
→ If a metal part is now detected by sensor "-B5", the sequencer jumps to Step 32 and "-Q1" remains set to TRUE.



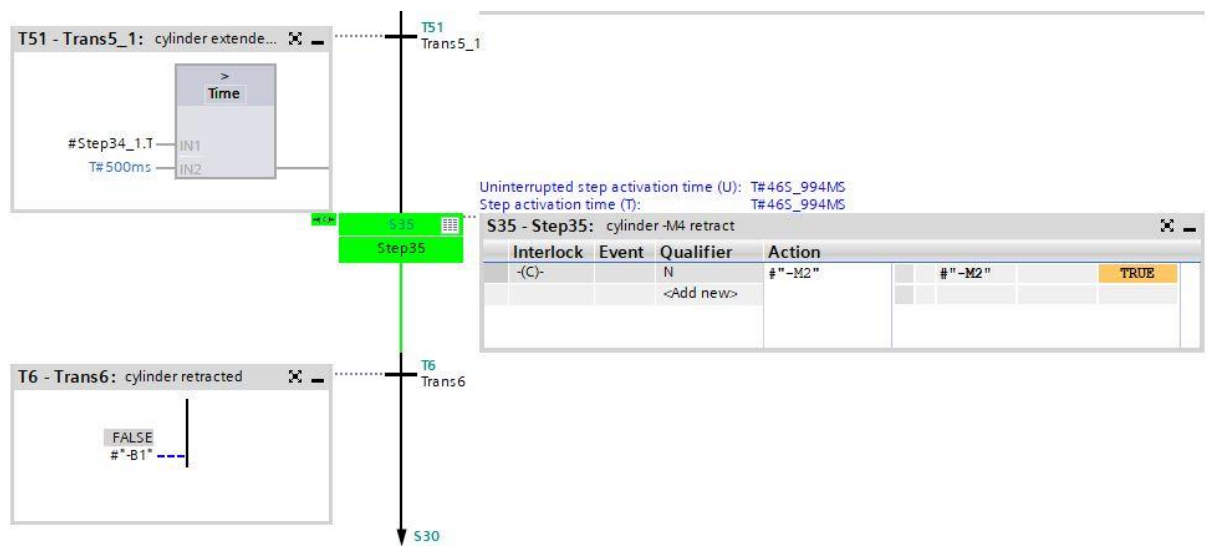
→ If sensor "-B6" now detects the metal part in front of the cylinder, the conveyor stops and the wait time of 0.5 seconds starts. After the time elapses, the sequencer jumps ahead to Step 34.



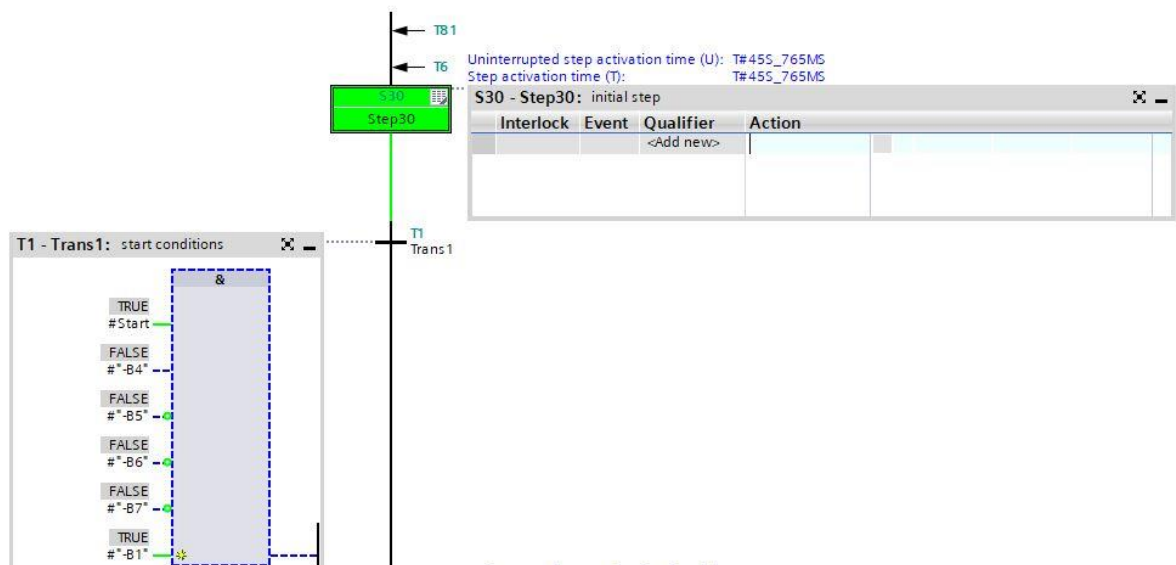
→ The cylinder extends and after it reaches the end position, the sequencer jumps to the next step (Step 341). The time of 0.5 seconds starts. As long as an interlock is active, the step is displayed in orange. After the time elapses, the sequencer jumps ahead to Step 35.



→ The cylinder is retracted again in Step 35.



→ When the cylinder has retracted, the sequencer jumps to Step 30 and waits for the next part. A new cycle can begin.



→ Test the next sequence with a plastic part.

7.13 Sequencer in test mode

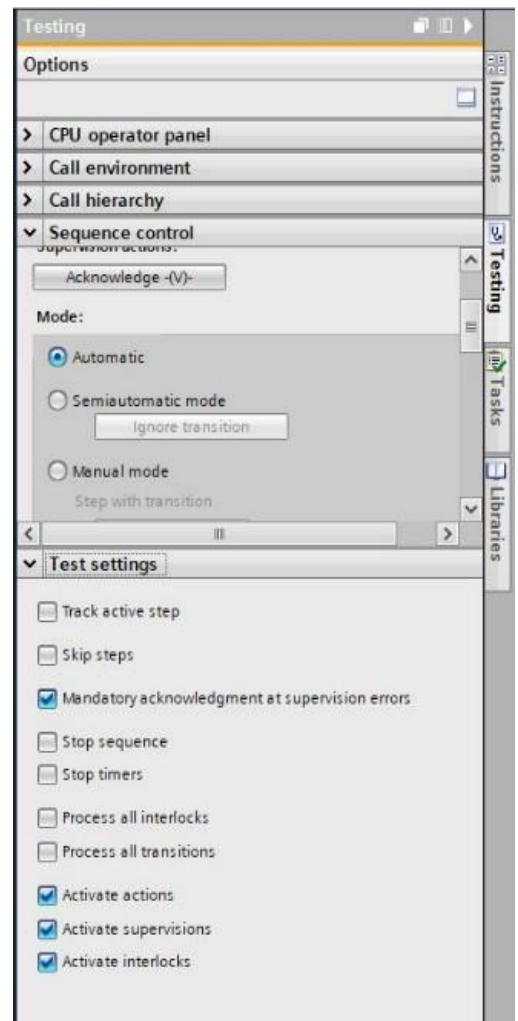
- The "**Testing**" window contains the "**Sequence control**" test function that can be used to test the GRAPH sequence control in all operating modes.
- All inputs and settings have the same effect as the corresponding FB parameters.
- The inputs in the "Sequence control" dialog window can differ from the settings that the programmer has made for compiling the sequencer. The settings made here take precedence over the other settings that were made when compiling.

Setting of operating mode:

- Select one of the options to put the sequencer into the selected operating mode. For example, you can have your sequencer run in manual mode instead of automatic mode. In manual mode, you can activate or deactivate any step you like.

Proceed as follows:

- 1. Enter the step number of the step that you want to execute in the "Step number" field or click the desired step.
- 2. Select the action that is to be performed with the step:
 - **Enable:** The selected step is activated even if the condition of the previous transition is not satisfied.
 - **Disable:** The selected step is deactivated.
 - Note that you when you enable a step the currently active step will be deactivated because only one step at a time may be active in a linear sequence. You can thus enable only one step at a time: Exception for simultaneous branches: Here, one step in each branch can be activated.
- Once you have controlled your sequencer in manual mode, you can return to automatic mode.
- In semi-automatic mode, you have the option of using the "Ignore transition" button to advance the sequencer even when transition conditions are not satisfied.
- Additional parameters can be set under "Test settings".



7.14 Synchronization of the sequencer

GRAPH helps you locate possible synchronization points between the process and sequencer. A process is no longer synchronous when it is brought into a different state manually. This can happen, for example, when you switch to manual mode in which you can enable any step you like – even when the condition of its preceding transition is not satisfied.

In order for the process, which may have changed in the meantime, to run automatically again and to find the possible synchronization points, the synchronization function is available with two options:

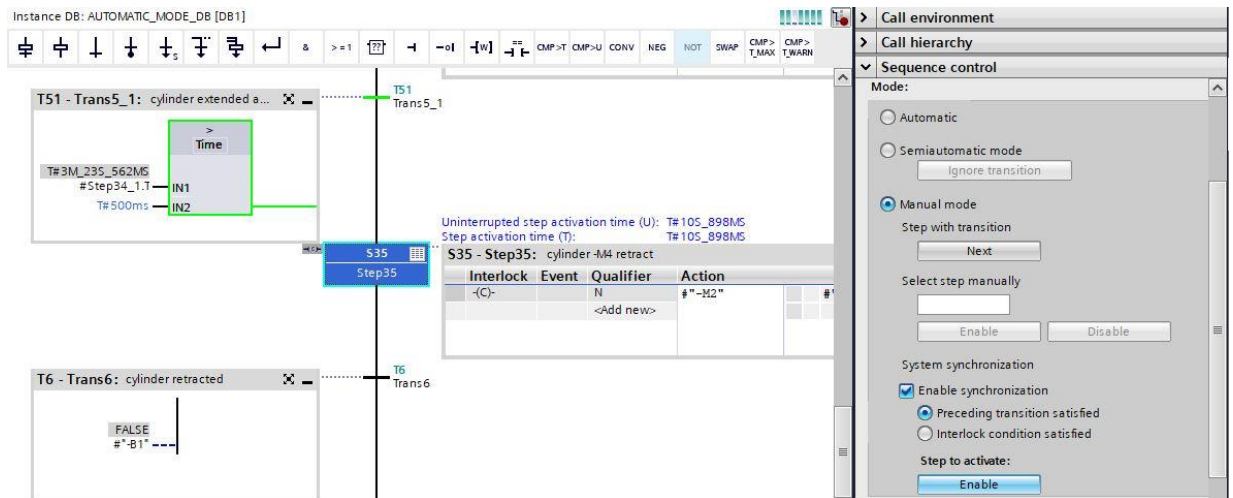
- Preceding transition satisfied
 - Interlock condition satisfied
- To use this function, switch the sequencer to manual mode and enable the synchronization.
- Select the "Preceding transition satisfied" option.

The screenshot displays the TIA Portal software interface. The main window shows the GRAPH editor with a ladder logic diagram. The diagram includes a transition labeled 'T51 - Trans5_1: cylinder extended a...' and a step labeled 'S35 - Step35: cylinder -M4 retract'. The step S35 has a table with the following data:

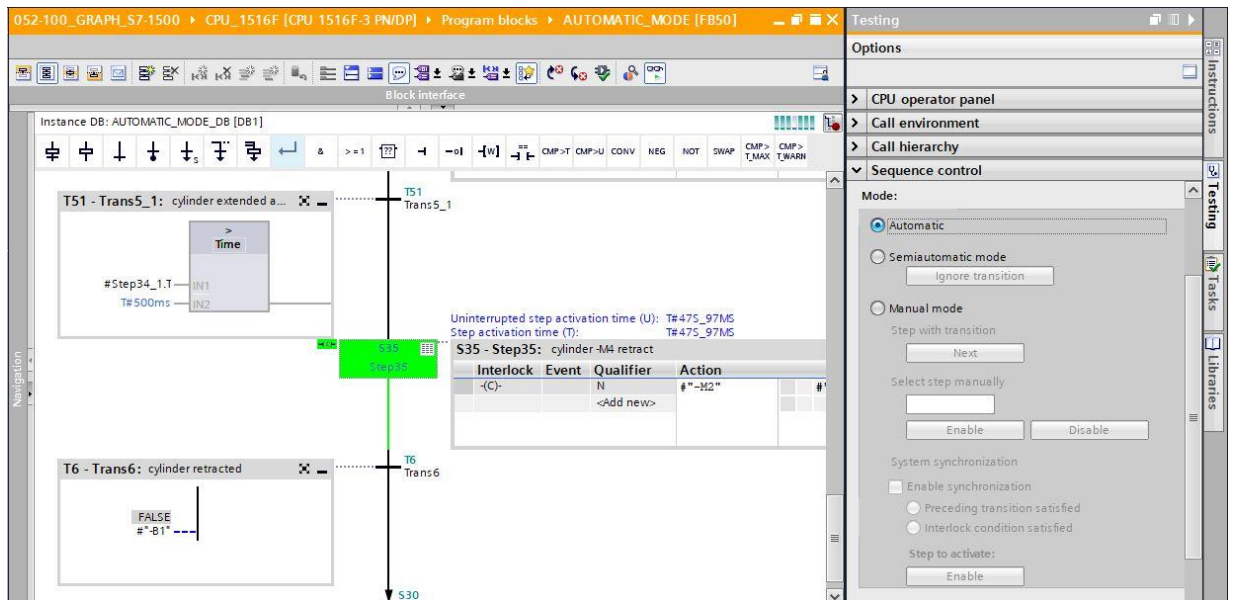
Interlock	Event	Qualifier	Action
-(C)	N	#"-M2"	

Below the step S35, there is a transition labeled 'T6 - Trans6: cylinder retracted' and a step labeled 'S30'. The right-hand side of the interface shows the 'Testing' options panel. The 'Mode' section has three radio buttons: 'Automatic', 'Semiautomatic mode', and 'Manual mode'. The 'Manual mode' radio button is selected. Below the 'Mode' section, there are buttons for 'Step with transition' (Next) and 'Select step manually' (Enable, Disable). The 'System synchronization' section has a checked box for 'Enable synchronization'. Below this, there are two radio buttons: 'Preceding transition satisfied' (selected) and 'Interlock condition satisfied'. At the bottom, there is a 'Step to activate:' section with an 'Enable' button.

→ Right-click the suggested step to select it. In this example, it is assumed that the cylinder is extended.




→ Switch the sequencer back to automatic mode.

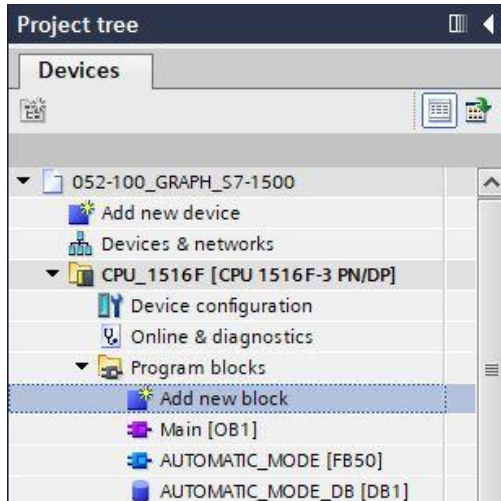



→ The sequencer can now continue running in automatic mode.

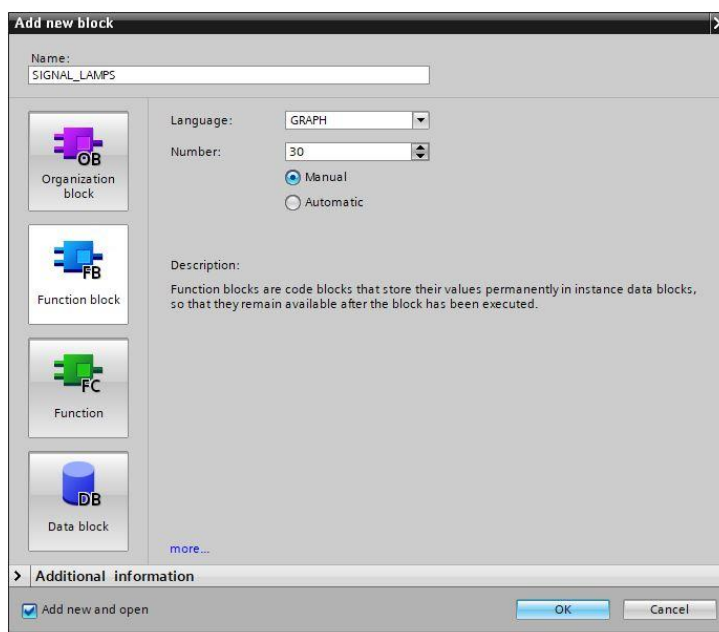
7.15 Creating function block FB30 "SIGNAL_LAMPS"

- Next, the GRAPH function block for controlling the signal lamps is to be programmed and tested.
- In the project tree under Program blocks, click on "Add new block" to create a new function block.

(→ Program blocks → Add new block →  FB)



- Name your function block "SIGNAL_LAMPS", set the language to GRAPH and manually choose the number 30 for the FB number. Select the "Add new and open" check box. You are then taken automatically to your created function block in the project view. Click "OK".
- (→ Name: SIGNAL_LAMPS → Language: GRAPH → Manual → Number: 30 →  Add new and open → OK)



7.16 Specifying the interface of FB30 "SIGNAL_LAMPS"

- If you selected "Add new and open", the project view opens with a GRAPH editor for programming the block you just created.
- You can find the interface description of your function block in the upper section of your programming view. Based on the default settings in the TIA Portal, the local tags of the standard interface parameters have already been created. If necessary, these default settings can be changed in the settings of the TIA Portal.
- We need only the first three input tags. The rest of the input tags and all the output tags can be deleted.

SIGNAL_LAMPS										
	Name	Data type	Default value	Retain	Acce...	Writa...	Visible ..	Setpoi...	Sup...	Comment
1	Input									
2	OFF_SQ	Bool	false	Non-retain						Turn sequence off
3	INIT_SQ	Bool	false	Non-retain						Set sequence to initial state
4	ACK_EF	Bool	false	Non-retain						Acknowledge all errors and fa
5	Output									
6	<Add new>									
7	InOut									
8	<Add new>									
9	Static									
10	RT_DATA	G7_RTDataPlus_V6		Non-retain						Internal data area
11	Trans1	G7_TransitionPlus_...		Non-retain						Transition structure
12	Step1	G7_StepPlus_V6		Non-retain						Step structure
13	Temp									

- The static tags must not be deleted.
- The same tag designations can be used for global tags as well as local tags in the TIA Portal, which is why we can take the needed tags from the GRAFCET for controlling the signal lamps from previously created blocks (e.g.: FB50) or the Tag table sorting station.
- Select the last row of the input tags with the right mouse button and select menu item "Add row" (→ Input: ACK_EF → Add row).

	Name	Data type	Default value	Retain	Acce...
1	Input				
2	OFF_SQ	Bool	false	Non-retain	
3	INIT_SQ	Bool	false	Non-retain	
4	ACK_EF	Bool	false	Non-ret...	
5					
6					
7					
8					
9					
10					
11					
12					

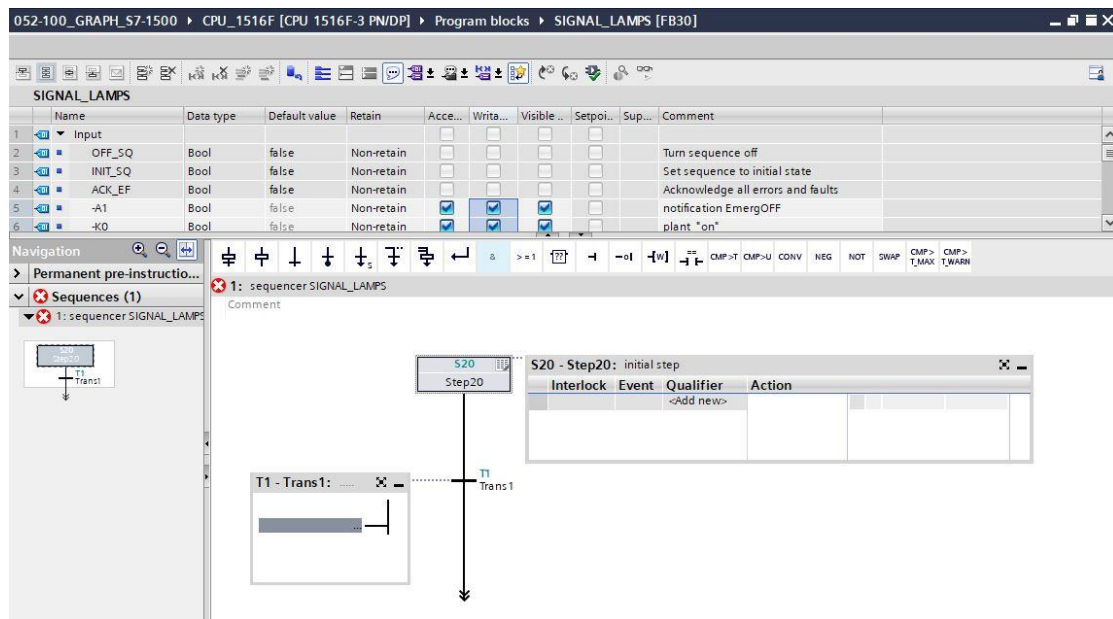
Insert row Ctrl+Enter
 Add row Alt+Ins
 Cut Ctrl+X
 Copy Ctrl+C
 Paste Ctrl+V
 Delete Del
 Rename F2
 Add new supervision

- Add parameter #-A1 as the input interface under Input in the added line and confirm the input with the Enter key. Data type "Bool" is assigned automatically. This will be retained. Finally, enter the associated comment "notification EmergOFF".
- Add additional binary input parameters #-K0, #-S0, #-B1 and #-B2 under Input and check their data types. Add descriptive comments for these.
- Add the binary output parameters #-P2, #-P3, #-P4, #-P6 and #-P7 under Output and check their data types. Add descriptive comments for these.
- Alternatively, you can also copy and paste these from the tag table.

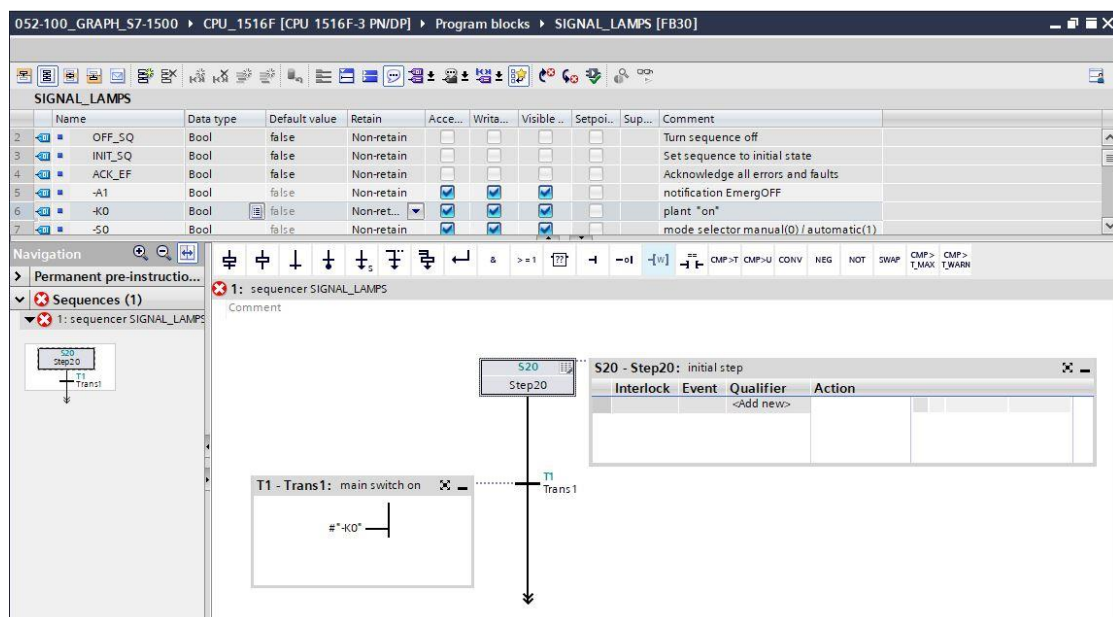
SIGNAL_LAMPS										
	Name	Data type	Default value	Retain	Acce...	Writa...	Visible ..	Setpoi...	Sup...	Comment
1	▼ Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
2	OFF_SQ	Bool	false	Non-retain	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Turn sequence off
3	INIT_SQ	Bool	false	Non-retain	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Set sequence to initial state
4	ACK_EF	Bool	false	Non-retain	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Acknowledge all errors and faults
5	-A1	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		notification EmergOFF
6	-K0	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		plant "on"
7	-S0	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		mode selector manual(0) / automatic(1)
8	-B1	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		sensor cylinder -M4 retracted
9	-B2	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		sensor cylinder -M4 extended
10	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
11	▼ Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
12	-P2	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display „manual mode“
13	-P3	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display „automatic mode“
14	-P4	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display „emergency stop activated“
15	-P6	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display cylinder -M4 „retracted“
16	-P7	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display cylinder -M4 „extended“

7.17 Programming of FB30 "SIGNAL_LAMPS"

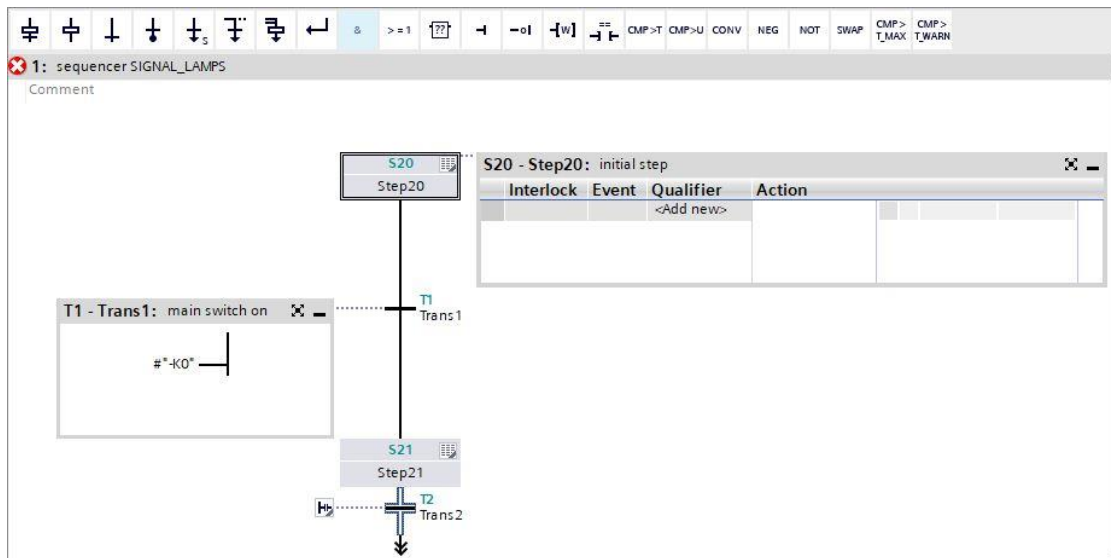
- First, we assign the sequencer the name "sequencer SIGNAL_LAMPS" by overwriting the text <new sequence>.
- The step number and the designation of the step tag must now be changed. To do so, click inside the step field and enter a new number and designation.
- Open the action table using the "📄" button in the step field.
- Open the input window using the "H" button at the transition.



- Insert "-K0" as the step-enabling condition in the window of Transition 1.
- Designate **T1 – Trans1:** as "main switch on", and designate **S20 – Step 20:** as "initial step" in the action table.

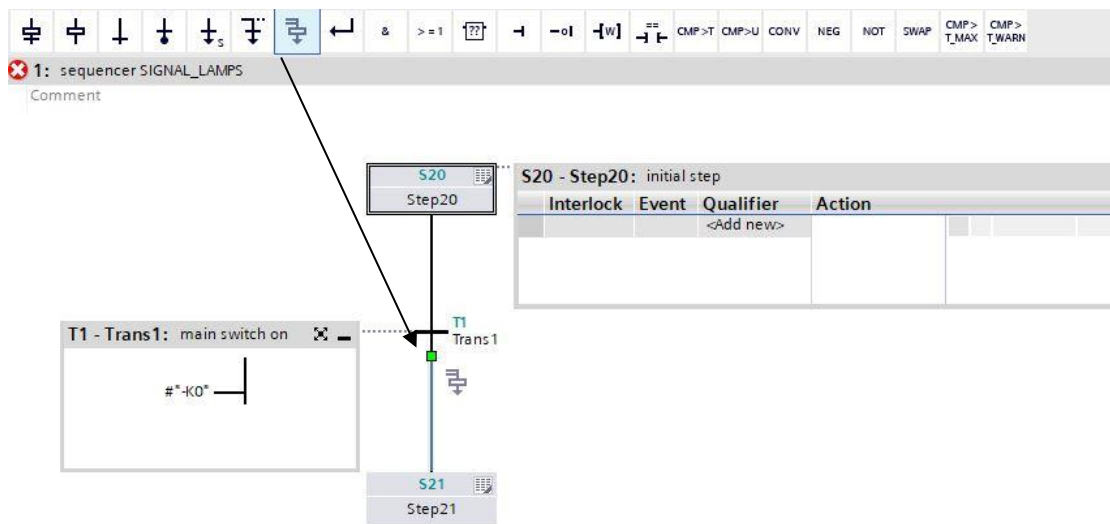



- Drag "Step and transition" onto the double arrow below Transition 1 to insert the next step with transition. The numbering continues automatically.

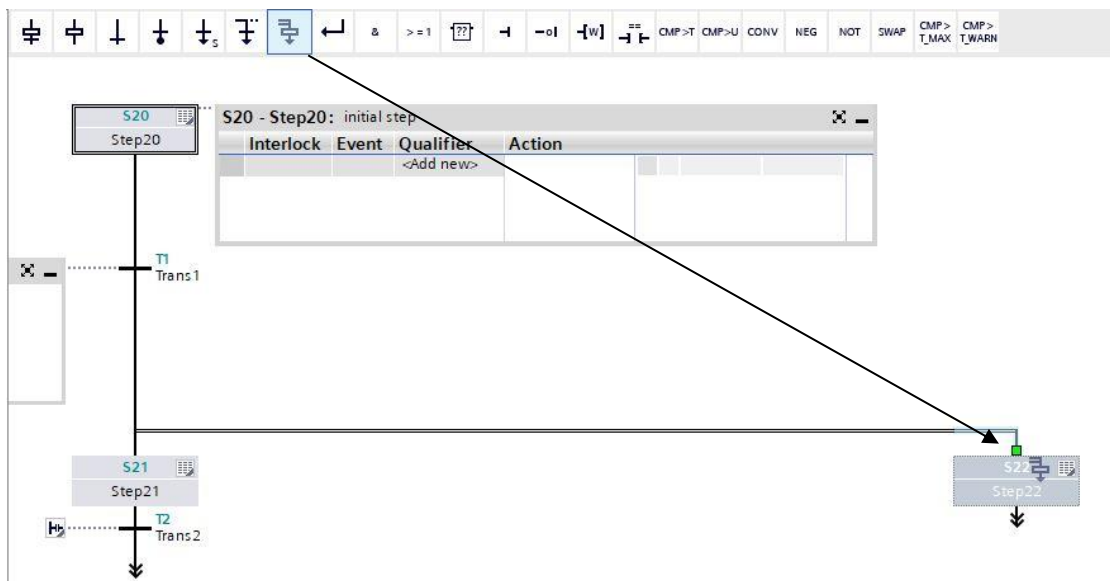



- In Step 21 of the GRAFCET for controlling the indicator lights, five actions with five different interlocks are specified. Because only one interlock per step can be programmed in S7-GRAPH, additional parallel steps must be inserted using simultaneous branches. The five actions with interlock are then distributed among five parallel steps.

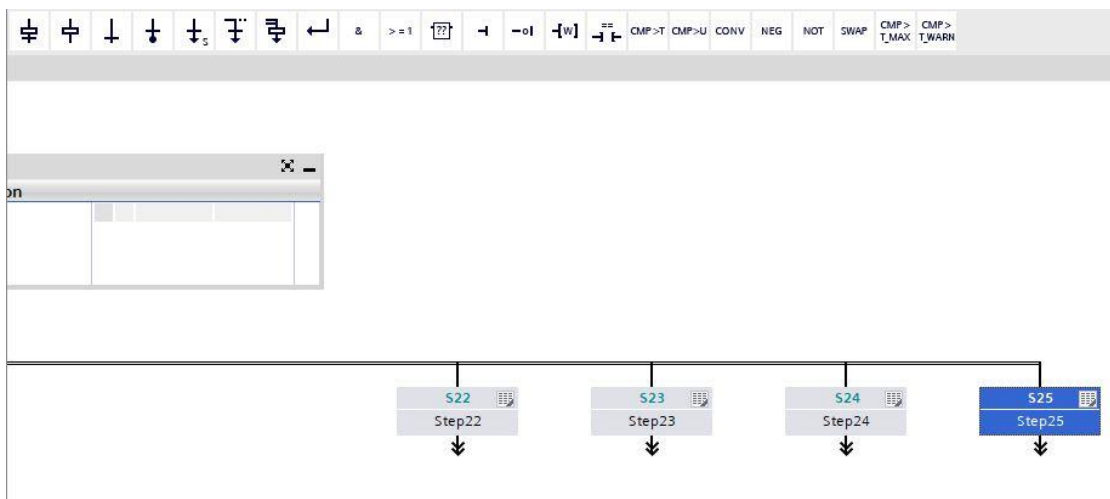
- Drag "Open simultaneous branch" onto the green square in the window of Transition 1.




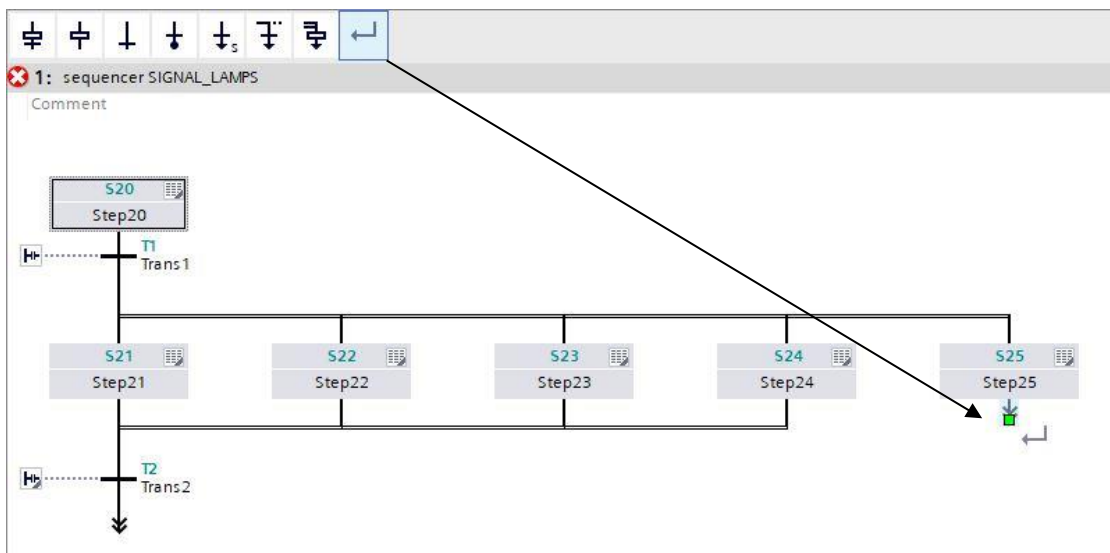
→ Drag "Open simultaneous branch" onto the green square  above Step 22.



→ Drag "Open simultaneous branch" onto the green square  above Steps 23 and 24.

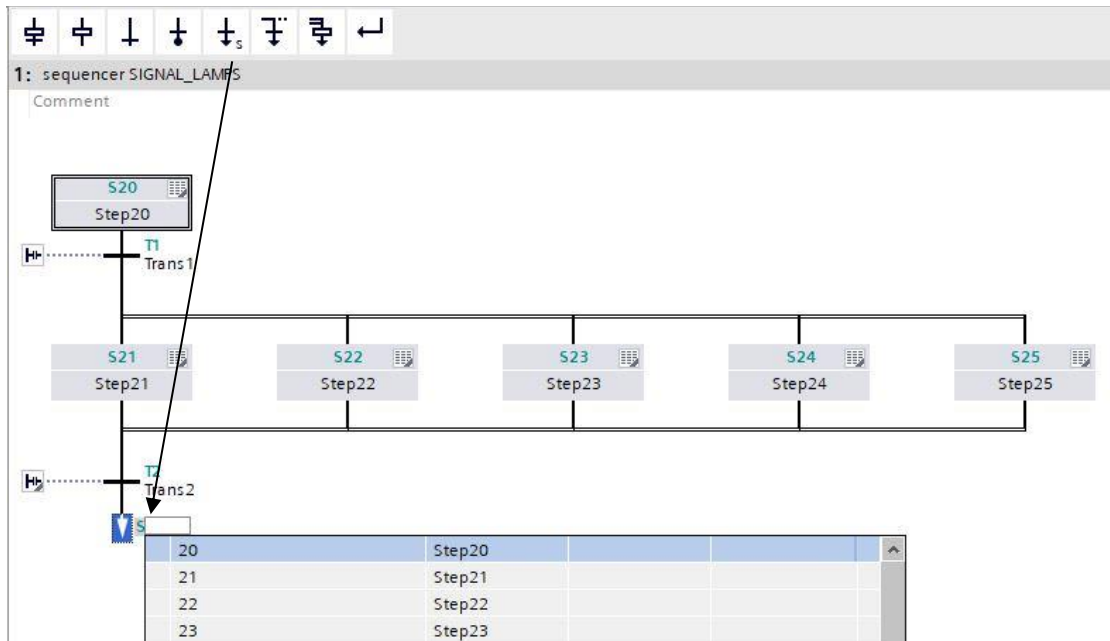


→ Drag "Close branch" onto the green square  below Steps 22 to 25.



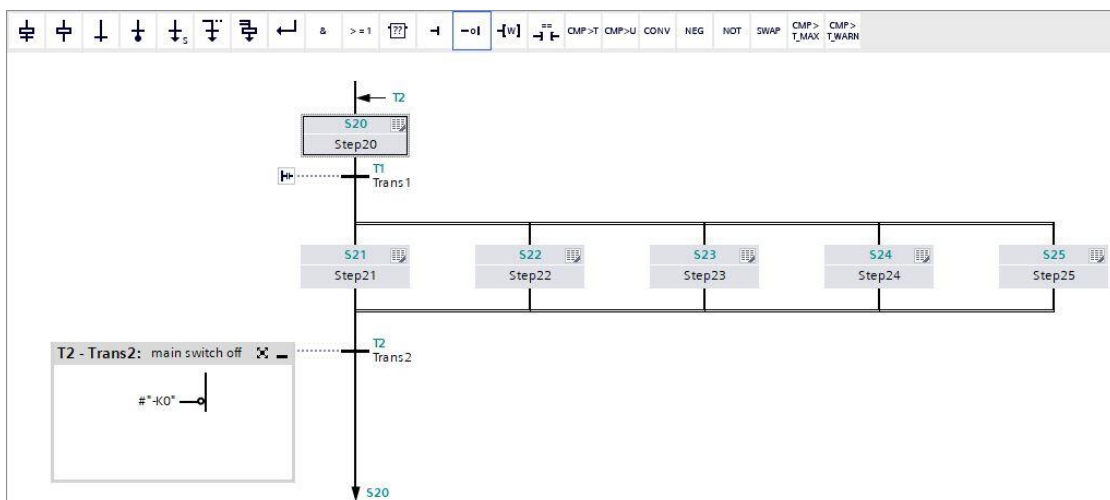
At the end of the sequence, we set a jump to Step 20.

→ Drag a jump to the double arrow and select Step 20 as the jump destination.

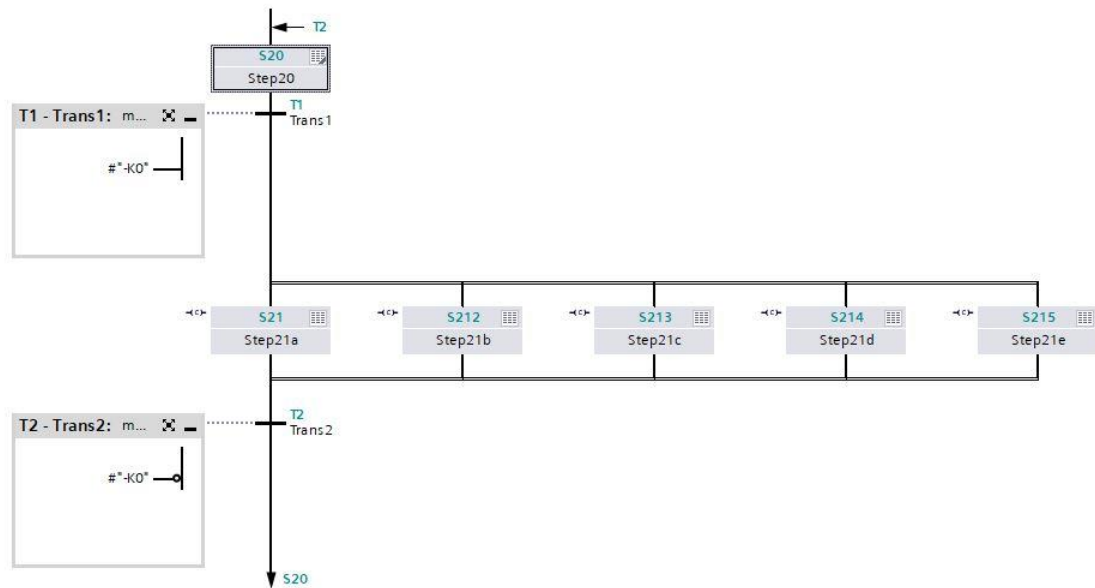


→ Insert "-K0" with a negation as the step-enabling condition in the window of Transition 2.

→ Designate **T2 – Trans2:** as "main switch off".



→ To keep a uniform structure, the inserted steps should be renamed as shown here.



→ Change to the single step view.

→ Interconnect Step 21 / Step 21a and designate the action.

Navigation: Permanent pre-instruction... Sequences (1) 1: sequencer SIGNAL_LAMPS

S21: Step21a

Comment

Interlock -(c)-: ...

Interlock C

#"-A1"

Supervision -(v)-: ...

Actions: display „emergencystop activated“

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-P4"
		<Add new>	

→ Interconnect Step 212 / Step 21b and designate the action.

Navigation: Permanent pre-instruction... Sequences (1) 1: sequencer SIGNAL_LAMPS

S212: Step21b

Comment

Interlock -(c)-: ...

Interlock C

#"-S0"

Supervision -(v)-: ...

Actions: display „manual mode“

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-E2"
		<Add new>	

→ Interconnect Step 213 / Step 21c and designate the action.

The screenshot shows the LAD editor for Step 21c. The left sidebar displays the 'Sequences (1)' tree with 'LAMP5' and 'S213 Step 21c'. The main workspace shows the 'Interlock -(c)-' section with a logic diagram: a normally open contact labeled '#"-50"' is connected to a coil labeled 'Interlock C'. Below this, the 'Supervision -(v)-' section is empty. The 'Actions: display „automatic mode“' section contains a table with the following data:

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-P3"
		<Add new>	

→ Interconnect Step 214 / Step 21d and designate the action.

The screenshot shows the LAD editor for Step 21d. The left sidebar displays the 'Sequences (1)' tree with 'S213 Step 21c' and 'S214 Step 21d'. The main workspace shows the 'Interlock -(c)-' section with a logic diagram: a normally open contact labeled '#"-81"' is connected to a coil labeled 'Interlock C'. Below this, the 'Supervision -(v)-' section is empty. The 'Actions: display cylinder „extended“' section contains a table with the following data:

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-P6"
		<Add new>	

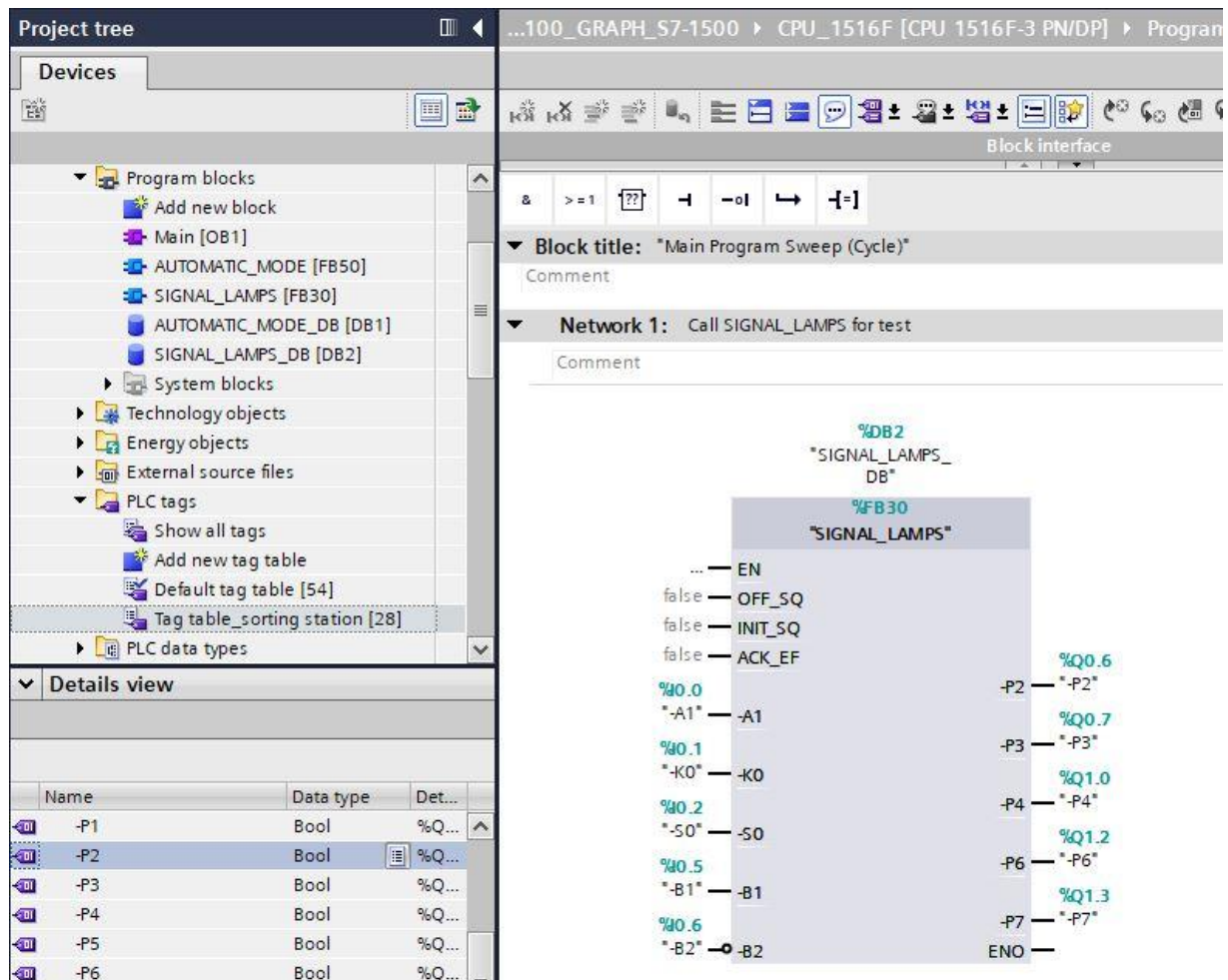
→ Interconnect Step 215 / Step 21e and designate the action.

The screenshot shows the LAD editor for Step 21e. The left sidebar displays the 'Sequences (1)' tree with 'S213 Step 21c' and 'S215 Step 21e'. The main workspace shows the 'Interlock -(c)-' section with a logic diagram: a normally open contact labeled '#"-82"' is connected to a coil labeled 'Interlock C'. Below this, the 'Supervision -(v)-' section is empty. The 'Actions: display cylinder „retracted“' section contains a table with the following data:

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-P7"
		<Add new>	

→ The GRAPH function block is now finished and can be called for testing in OB1.

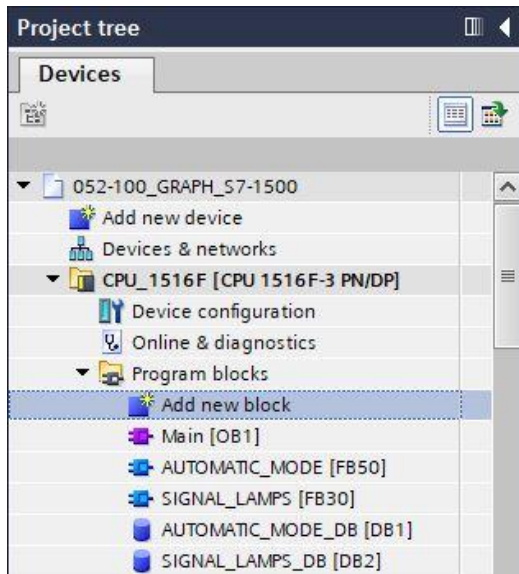
- Open OB1 and delete the block call in Network 1.
- Call function block "SIGNAL_LAMPS[FB30]" in Network 1.
- Confirm the name of the data block.
- Interconnect the tags of the block with the global tags of the sorting station.
- Negate input "-B2", because this sensor is wired as a normally closed contact. The controller recognizes that the cylinder is in extended position ("-B2" is activated) when no voltage is present at the terminal of input I0.6, i.e. a 0 signal is present.



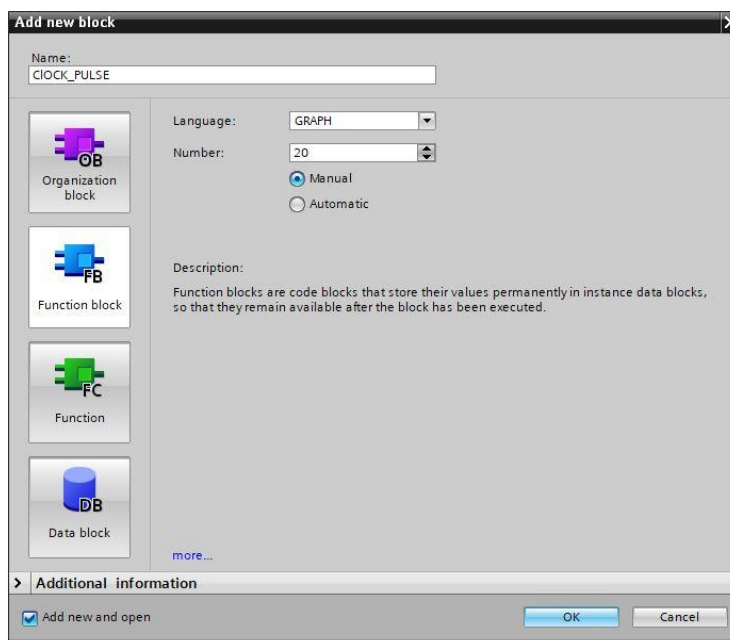
- To save your project, select the Save project button in the menu. To compile all blocks, click the "Program blocks" folder and select the button for compiling in the menu. (→ Save project → Program blocks →)
- After successful compilation, the complete controller with the created program, as previously described in the modules for hardware configuration, can be downloaded (→)
- Test the SIGNAL_LAMPS block.

7.18 Creating function block FB20 "CLOCK_PULSE"

- Next, the GRAPH function block for generating a clock pulse is to be programmed and tested.
- In the project tree under Program blocks, click on "Add new block" to create a new function block.



- Name your new function block "CLOCK_PULSE", set the language to GRAPH and manually choose the number 20 for the FB number. Select the "Add new and open" check box. You are then taken automatically to your created function block in the project view. Click "OK".



7.19 Specifying the interface of FB20 "CLOCK_PULSE"

- If you selected "Add new and open", the project view opens with a GRAPH editor for programming the block you just created.
- You can find the interface description of your function block in the upper section of your programming view. Based on the default settings in the TIA Portal, the local tags of the standard interface parameters have already been created. If necessary, these default settings can be changed in the settings of the TIA Portal.
- We need only the first three input tags. The rest of the input tags and all the output tags can be deleted.

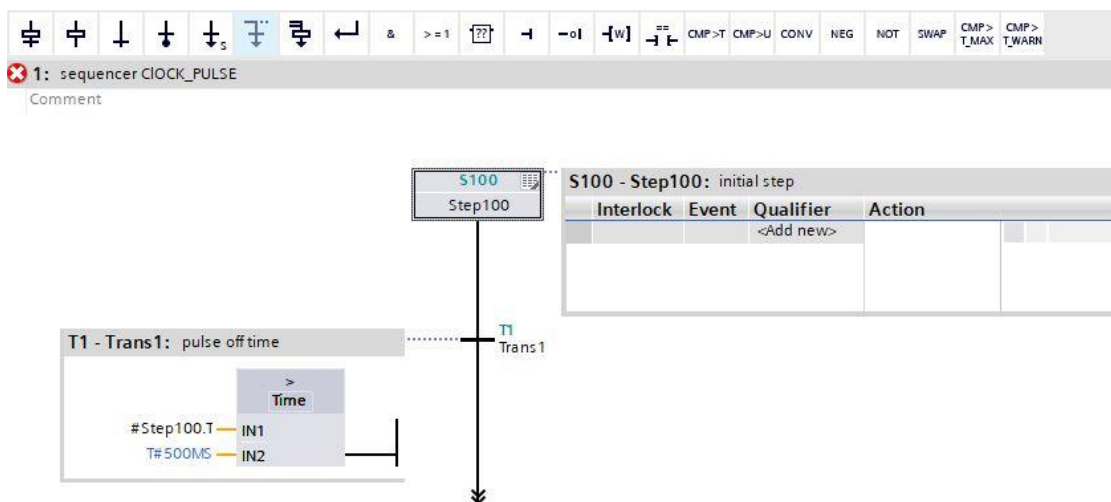
	Name	Data type	Default value	Retain	Access...	Writa...	Visibl...	Setpo...	Sup...	Comment
1	▼ Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
2	OFF_SQ	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Acknowledge all errors and faults
5	▼ Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
6	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
7	▼ InOut				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
8	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
9	▼ Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
10	▶ RT_DATA	G7_RTDataPlus...		Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Internal data area
11	▶ Trans1	G7_Transition...		Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Transition structure
12	▶ Step1	G7_StepPlus_V6		Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Step structure
13	▼ Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

- The static tags must not be deleted.
- Add the binary output parameter #Clock_Pulse under Output and check the data type. Enter "clock pulse 1Hz" as comment.

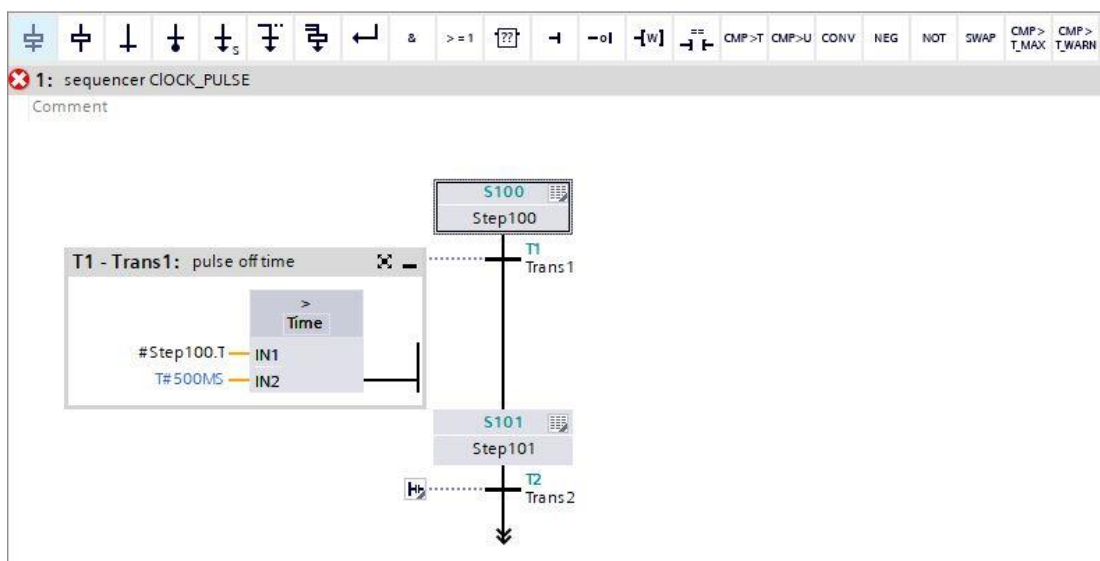
CLOCK_PULSE										
	Name	Data type	Default value	Retain	Access...	Writa...	Visibl...	Setpo...	Sup...	Comment
1	▼ Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
2	OFF_SQ	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Acknowledge all errors and faults
5	▼ Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
6	Clock_Pulse	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		clock pulse 1Hz
7	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
8	▼ InOut				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
9	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
10	▼ Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
11	▶ RT_DATA	G7_RTDataPlus...		Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Internal data area
12	▶ Trans1	G7_Transition...		Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Transition structure
13	▶ Step1	G7_StepPlus_V6		Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Step structure
14	▼ Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

7.20 Programming of FB20 "CLOCK_PULSE"

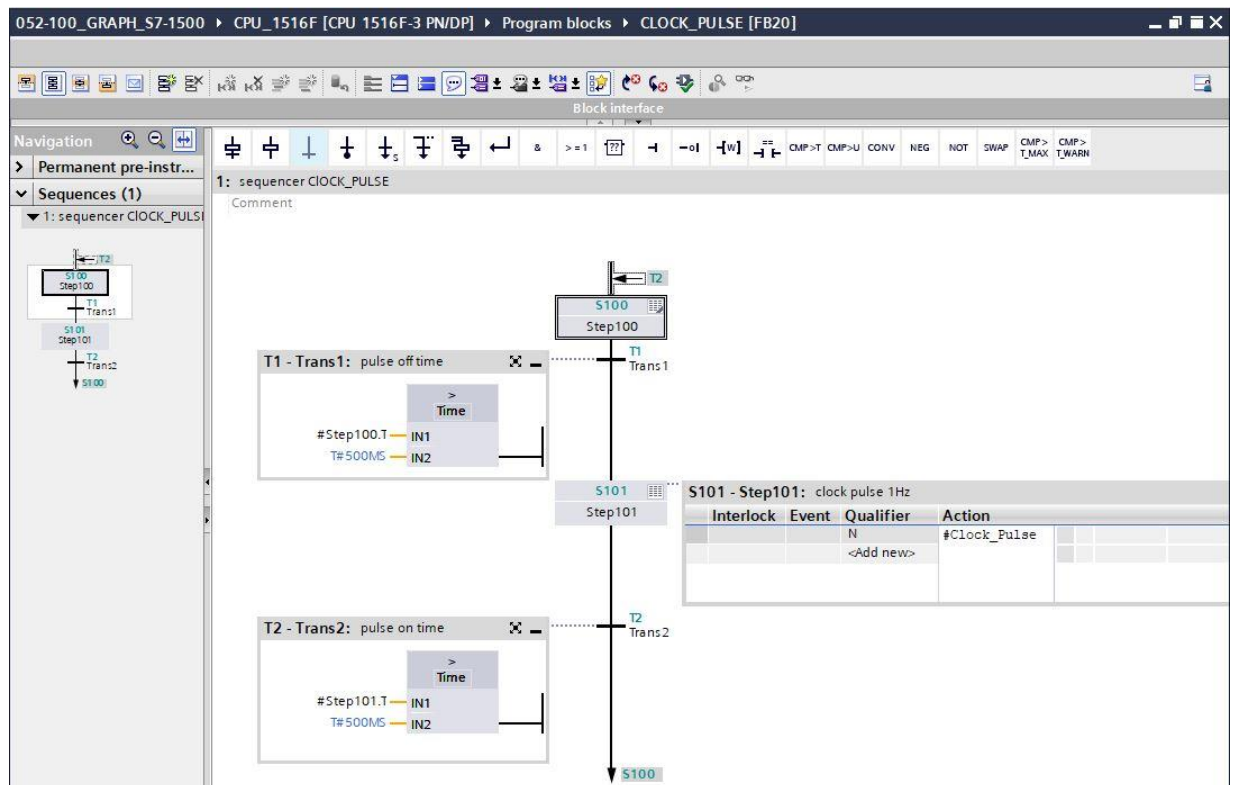
- First, we assign the sequencer the name "sequencer CLOCK_PULSE" by overwriting the text <new sequence>.
- The step number and the designation of the step tag must now be changed. To do so, click inside the step field and enter a new number and designation.
- Open the action table using the "📄" button in the step field.
- Open the input window using the "H" button at the transition.
- Now drag the comparator "Greater than step activation time" onto the green square as the step-enabling condition and enter **T#500MS** for the time.
- Designate **T1 – Trans1:** as "pulse off time", and designate **S100 – Step 100:** as "initial step" in the action table.



- Drag "Step and transition" onto the double arrow below Transition 1 to insert the next step with transition. The numbering continues automatically.

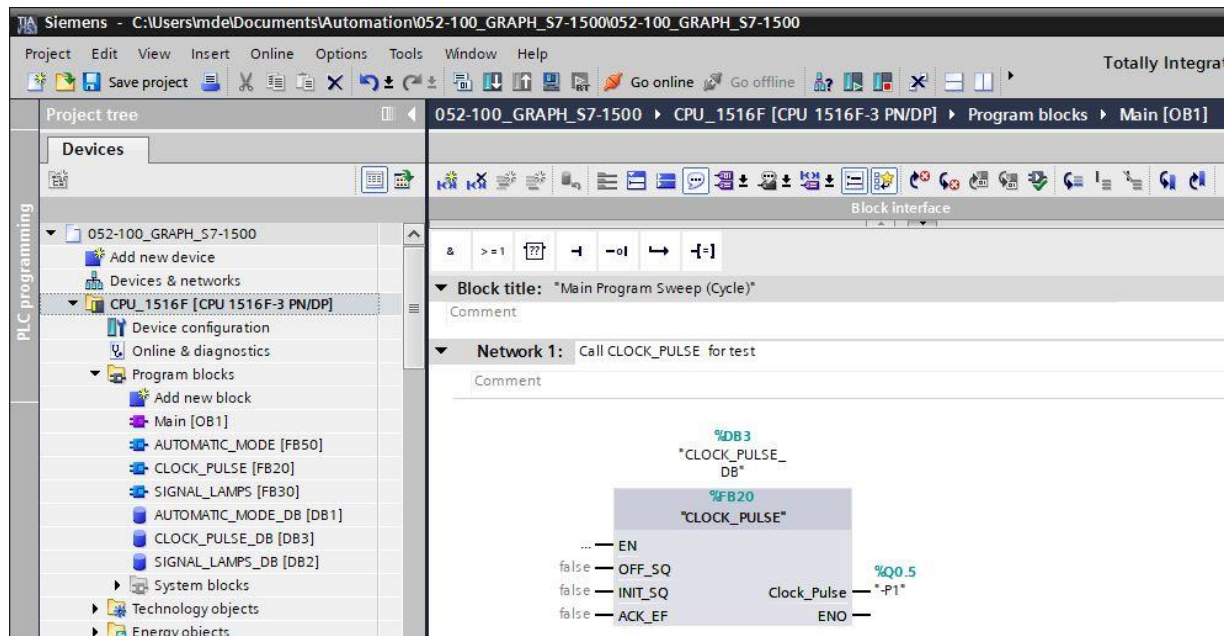


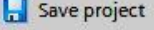

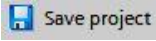


- Open the action table using the "📄" button in the step field.
- Open the input window using the "Hb" button at the transition.
- Now drag the comparator "Greater than step activation time" onto the green square as the step-enabling condition and enter T#500MS for the time.
- Designate T2 – Trans2: as "pulse on time", and designate S101 – Step 101: as "clock pulse 1 Hz".
- Drag a jump to the double arrow and select Step 100 as the jump destination.



- The GRAPH function block is now finished and can be called for testing in OB1.

- Open OB1 and delete the block call in Network 1.
- Call function block of the clock pulse in Network 1.
- Confirm the name of the data block.
- Interconnect the pulse tag of the block with the global tag "-P1" of the sorting station.



- To save your project, select the  button in the menu.
- Then, click the "Program blocks" folder and select the button  for compiling all blocks in the menu. (→  → Program blocks → )
- After successful compilation, the complete controller with the created program, as previously described in the modules for hardware configuration, can be downloaded (→ )
- Test the CLOCK_PULSE block.

7.21 General notes on use of events

In this chapter, event-controlled actions are also used for the GRAPH function block "RELEASE".

An event defines when an action is to be executed. You must specify an event for some action qualifiers.

Below are the actions that always need an event, together with the associated events and a description:

Qualifier	Events	Description
CS	S1, S0, L1, L0, V1, V0, A1, R1	Set counter value: As soon as the defined event occurs, the counter is set to the specified counter value. You can specify the counter value as a tag or constant of data type WORD (C#0 to C#999).
CU	S1, S0, L1, L0, V1, V0, A1, R1	Count up: As soon as the defined event occurs, the counter is incremented by "1". The counter value can be incremented until the limit of "999" is reached. When the limit is reached, the counter value is no longer incremented on a positive edge.
CD	S1, S0, L1, L0, V1, V0, A1, R1	Count down: As soon as the defined event occurs, the counter is decremented by "1". The counter value can be decremented until the limit of "0" is reached. When the limit is reached, the counter value is no longer decremented on a positive edge.
CR	S1, S0, L1, L0, V1, V0, A1, R1	Reset counter: As soon as the defined event occurs, the counter is reset to "0".
TL	S1, S0, L1, L0, V1, V0, A1, R1	Extended pulse: As soon as the defined event occurs, the timer starts. During the specified duration, the timer status has signal state "1". After the time expires, the timer status has signal state "0".
TD	S1, S0, L1, L0, V1, V0, A1, R1	Retentive on delay: As soon as the defined event occurs, the timer starts. During the specified duration, the timer status has signal state "0". After the time expires, the timer status has signal state "1".
TR	S1, S0, L1, L0, V1, V0, A1, R1	Hold timer and reset: As soon as the defined event occurs, the timer is stopped. Timer status and time value are reset to "0".
ON	S1, S0, L1, L0, V1, V0, A1, R1	Activate step
OFF	S1, S0, L1, L0, V1, V0, A1, R1	Deactivate step

The use of events is optional for the following actions.

Qualifier	Description
N	<p>Set to 1 as long as step is active:</p> <p>The signal state of the operand is "1" as long as the step is active. The specified block is called as long as the step is active. The step is also regarded as active in the cycle in which the S1 event occurs. This qualifier is also used for block calls. Functions (FC) with syntax:</p> <p>CALL "<FCName>" (parameter list)</p> <p>Function blocks (FB) with syntax:</p> <p>CALL "<FBName>", "<DBName>" (parameter list)</p>
S	<p>Set to 1</p> <p>As soon as the step is active, the operand is set to "1" and then remains at "1".</p>
R	<p>Set to 0:</p> <p>As soon as the step is active, the operand is set to "0" and then remains at "0".</p>

The following actions cannot be linked with events:

Qualifier	Description
D	<p>On delay:</p> <p>n seconds after the step activation, the operand is set to "1" and remains at "1" for the duration of the step activation. This does not apply if the duration of the step activation is shorter than n seconds. You can specify the time as a constant or as a PLC tag of data type TIME/DWORD.</p>
L	<p>Set for limited time:</p> <p>When the step is active, the operand is set to "1" for n seconds. The operand is then reset. The operand is also reset if the step activation time is less than the time duration. You can specify the time as a constant or as a PLC tag of data type TIME/DWORD.</p>
TF	<p>Turn off timer:</p> <p>The timer status is set to "1" as soon as the step is activated. The timer runs when the step is deactivated, and timer status is set to "0" only after the time expires.</p>

The following events are defined for GRAPH:

Event	Signal evaluation	Description
S1	Positive edge	Step is activated (signal state = "1")
S0	Negative edge	Step is deactivated (signal state = "0")
V1	Positive edge	Supervision condition is satisfied, i.e. fault occurs (signal state = "1")
V0	Negative edge	Supervision condition is no longer satisfied, i.e. fault is eliminated (signal state = "0")
L0	Positive edge	Interlock condition is satisfied, i.e. fault is eliminated (signal state = "1")
L1	Negative edge	Interlock condition is not satisfied, i.e. fault occurs (signal state = "0")
A1	Positive edge	An alarm is being acknowledged.
R1	Positive edge	Incoming registration.

You can additionally combine actions for which you use the "S1", "V1", "A1", or "R1" event with an interlock. The actions are then only executed if the conditions of the interlock are satisfied.

Interlock

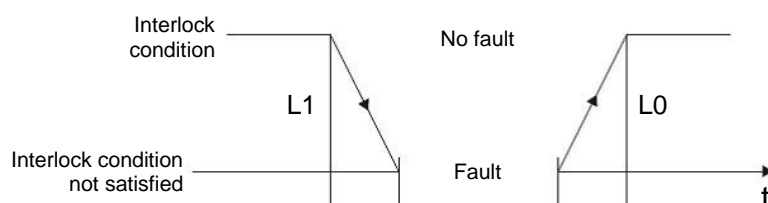
An interlock is a programmable condition that influences the execution of individual actions. If the combination of conditions is satisfied, actions combined with interlock are executed. Otherwise the execution is prevented.

If the combination of conditions is not satisfied, there is a fault:

- Actions combined with interlock are not executed.
- An interlock error is signaled (event L1).
- The alarms specified for the interlock are displayed.
- The fault does not affect the advancement to the next step, however.

A programmed interlock is indicated in each view type with the letter C to the left of the step.

The following figure shows the signal evaluation for an interlock:



L1: Interlock condition no longer satisfied (incoming fault)

L0: Interlock condition satisfied (outgoing fault)

Supervision

A supervision is a programmable condition that can prevent the advancement from one step to the next. If the combination of conditions is satisfied, there is a fault and event V1 is signaled.

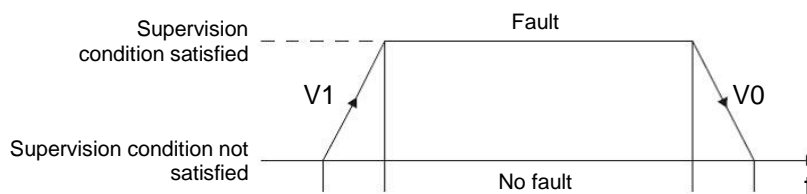
- The sequencer does not advance to the next step.
- The current step stays active, however.
- The step activation time (step activation time U) is paused as soon as the condition is satisfied.
- The alarms specified for the interlock are displayed.

If the combination of conditions is not satisfied, there is a fault: If the condition of the following transition is satisfied, the sequencer advances to the next step.

A programmed supervision is indicated in each view type with the letter V to the left of the step.

You can use supervision conditions, for example, to monitor the step activation time; in other words, the time that has elapsed since activation of the step.

The following figure shows the signal evaluation for a supervision:



V1: Supervision error occurs

V0: Supervision error is eliminated

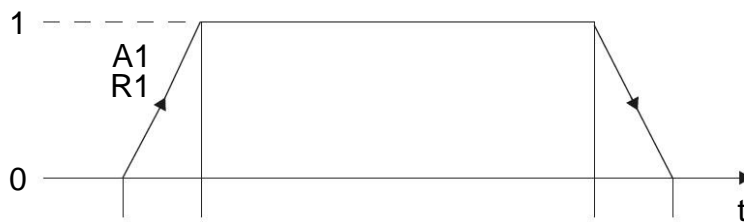
Alarm and registration

A registration is an event that is triggered outside the block. Registrations are queried via a positive signal edge at either the "REG_S" or "REG_EF" input parameter.

If the registration occurs via the "REG_S" input parameter, the event is only forwarded to the active step that is indicated at the "S_NO" output parameter.

If the registration occurs via the "REG_EF" input parameter, the event is forwarded to all currently active steps.

The following figure shows the signal evaluation for an alarm and the registration:



A1: An alarm is being acknowledged.

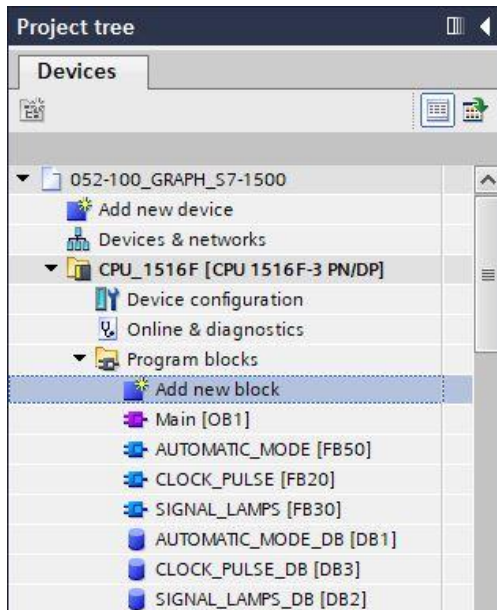
R1: Incoming registration (positive edge at input REG_EF/REG_S)

Note:

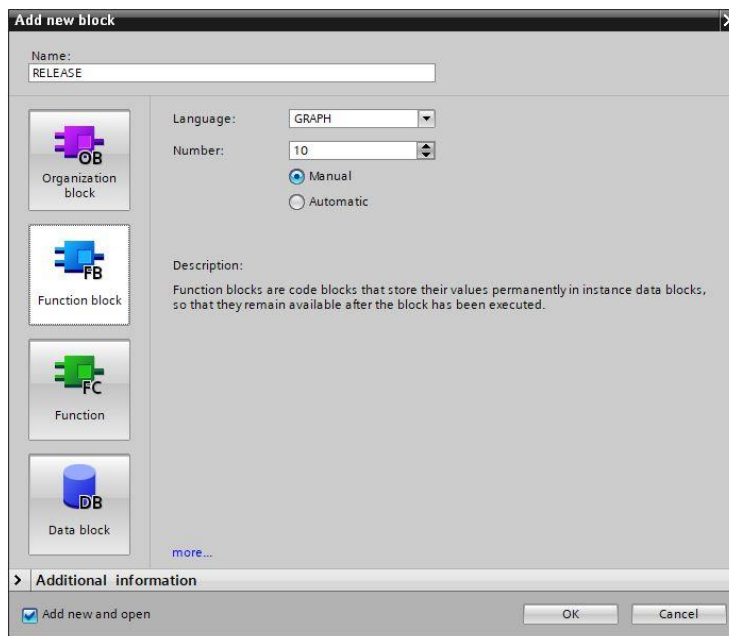
- You can find detailed information on all actions, events and event-controlled actions in the manuals or the online help.

7.22 Creating function block FB10 "RELEASE"

- The GRAPH function block for release of the controller is to be programmed and tested.
- In the project tree under Program blocks, click on "Add new block" to create a new function block.



- Name your new function block "RELEASE", set the language to GRAPH and manually choose the number 10 for the FB number. Select the "Add new and open" check box. You are then taken automatically to your created function block in the project view. Click "OK".



7.23 Specifying the interface of FB10 "RELEASE"

- If you selected "Add new and open", the project view opens with a GRAPH editor for programming the block you just created.
- You can find the interface description of your function block in the upper section of your programming view. Based on the default settings in the TIA Portal, the local tags of the standard interface parameters have already been created. If necessary, these default settings can be changed in the settings of the TIA Portal.
- Here again, we need only the first three input tags. The rest of the input tags and all the output tags can be deleted.

	Name	Data type	Default value	Retain	Acce...	Wri...	Visibl...	Setpo...	Sup...	Comment
1	▼ Input									
2	OFF_SQ	Bool	false	Non-ret...						Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...						Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...						Acknowledge all errors and faults
5	▼ Output									
6	<Add new>									
7	▼ InOut									
8	<Add new>									
9	▼ Static									
10	▶ RT_DATA	G7_RTDataPlus_V6		Non-ret...				✓		Internal data area
11	▶ Trans1	G7_TransitionPlus...		Non-ret...				✓		Transition structure
12	▶ Step1	G7_StepPlus_V6		Non-ret...				✓		Step structure
13	▼ Temp									



- The static tags must not be deleted.
- The same tag designations can be used for global tags as well as local tags in the TIA Portal, which is why we can take the needed tags from the GRAFCET for controlling the signal lamps, from previously created blocks (e.g.: FB50, FB30, FB20) or from the Tag table sorting station.
- Select the last row of the input tags with the right mouse button and select menu item "Add row" (→ Input: ACK_EF → Add row).

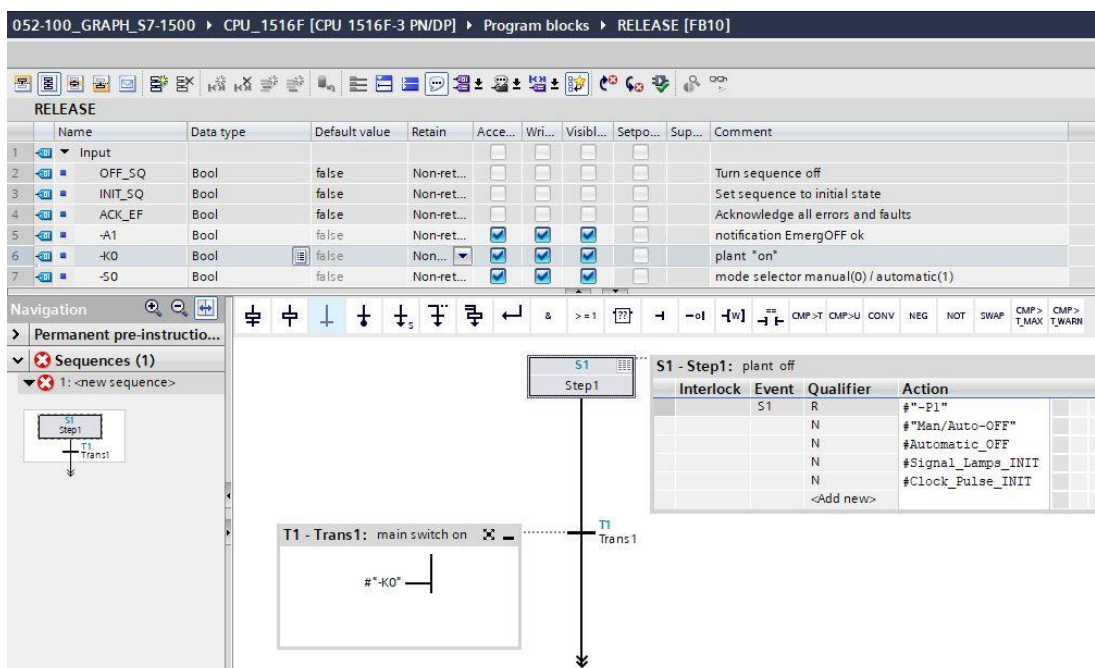
	Name	Data type	Default value	Retain	Acce...	Wri...	Visibl...	Setpo...	Sup...	Comment
1	▼ Input									
2	OFF_SQ	Bool	false	Non-ret...						Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...						Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...						Acknowledge all errors and faults
5	<div> <div>Insert row</div> <div> <div>Ctrl+Enter</div> <div>Alt+Ins</div> </div> </div>									
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										
35										
36										
37										
38										
39										
40										
41										
42										
43										
44										
45										
46										
47										
48										
49										
50										
51										
52										
53										
54										
55										
56										
57										
58										
59										
60										
61										
62										
63										
64										
65										
66										
67										
68										
69										
70										
71										
72										
73										
74										
75										
76										
77										
78										
79										
80										
81										
82										
83										
84										
85										
86										
87										
88										
89										
90										
91										
92										
93										
94										
95										
96										
97										
98										
99										
100										

- You can copy and paste the input tags #-A1, #-K0, #-S0 from the SIGNAL_LAMPS block.
- Add additional binary input parameters #-S0 to #-S6 under Input and check their data types. Add descriptive comments for these.
- Add the binary output parameters #-P1, #Man/Auto-OFF, #Automatic_OFF, #Signal_Lamp_INIT, #Clock_Pulse_INIT, #Man/Auto_INIT and #Release under Output and check their data types. Add descriptive comments for these.
- Alternatively, you can also copy and paste these from the tag table.

RELEASE										
	Name	Data type	Default value	Retain	Acce...	Wri...	Visibl...	Setpo...	Sup...	Comment
1	▼ Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
2	OFF_SQ	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Acknowledge all errors and faults
5	-A1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		notification EmergOFF ok
6	-K0	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		plant "on"
7	-S0	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		mode selector manual(0) / automatic(1)
8	-S1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		pushbutton automatic start
9	-S2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		pushbutton automatic stop
10	-S3	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		pushbutton inching operation conveyor -M1 forwards
11	-S4	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		pushbutton inching operation conveyor -M1 backwards
12	-S5	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		pushbutton inching operation cylinder -M4 retract
13	-S6	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		pushbutton inching operation cylinder -M4 extend
14	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
15	▼ Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
16	-P1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display „main switch on“
17	Man/Auto-OFF	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		switch off sequencer OPERATING_MODES
18	Automatic_...	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		switch off sequencer AUTOMATIC_MODE
19	Signal_Lamp...	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		initialize sequencer SIGNAL_LAMPS
20	Clock_Pulse_...	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		initialize sequencer CLOCK_PULSE
21	Man/Auto_INIT	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		initialize sequencer OPERATING_MODES
22	Release	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		operational release

7.24 Programming of FB10 "RELEASE"

- First, we assign the sequencer the name "sequencer RELEASE" by overwriting the text <new sequence>.
- Open the action table using the  button in the step field.
- Open the input window using the  button at the transition.
- Insert "-K0" as the step-enabling condition in the window of Transition 1.
- Designate **T1 – Trans1:** as "main switch on", and designate **S20 – Step 20:** as "system off" in the action table.
- In the GRAFCET for release of the controller, the first action in the first step is to retentively turn off the indicator light " main switch on" on activation of the step.
- For this, program indicator light "-P1" using event **S1** and qualifier **R**.
- Program the remaining actions in Step 1.



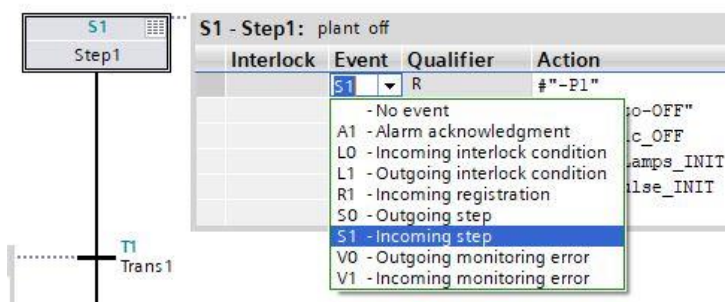
The screenshot displays the Siemens TIA Portal interface for programming the FB10 "RELEASE" block. The top window shows the "RELEASE [FB10]" block with a table of inputs and their actions. The bottom window shows the GRAFCET diagram with Step 1 and Transition 1. The action table for Step 1 is open, showing the configuration for event S1 with qualifier R and action #'-P1'.

Name	Data type	Default value	Retain	Acc...	Wri...	Visibl...	Setpo...	Sup...	Comment
1	Input								
2	OFF_SQ	Bool	false	Non-ret...					Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...					Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...					Acknowledge all errors and faults
5	-A1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		notification EmergOFF ok
6	-K0	Bool	false	Non...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		plant "on"
7	-S0	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		mode selector manual(0) / automatic(1)

The GRAFCET diagram shows Step 1 (S1) and Transition 1 (T1). The action table for Step 1 is open, showing the configuration for event S1 with qualifier R and action #'-P1'.

Interlock	Event	Qualifier	Action
S1	R		#'-P1'
	N		#"Man/Auto-OFF"
	N		#Automatic_OFF
	N		#Signal_Lamps_INIT
	N		#Clock_Pulse_INIT
	<Add new>		


- Programmable events:

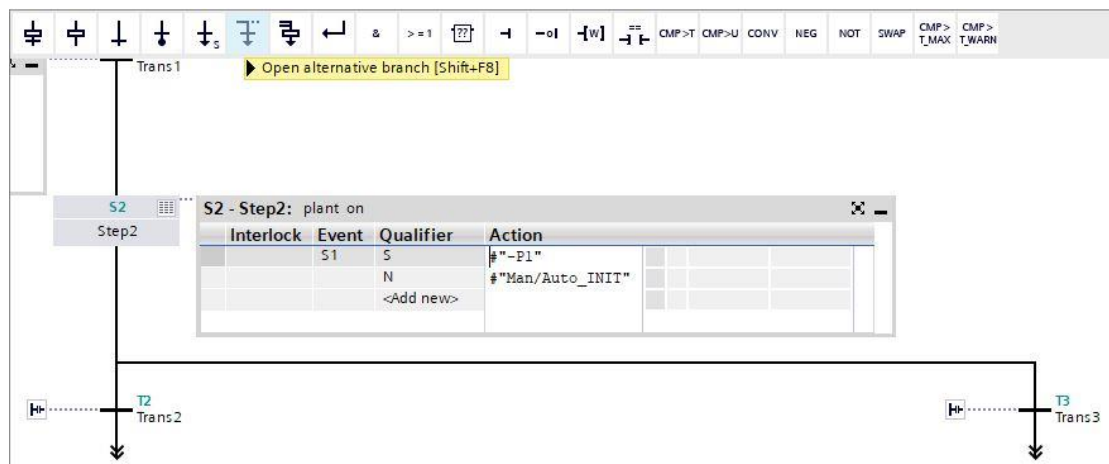


The screenshot displays the Siemens TIA Portal interface for programming the FB10 "RELEASE" block. The top window shows the "RELEASE [FB10]" block with a table of inputs and their actions. The bottom window shows the GRAFCET diagram with Step 1 and Transition 1. The action table for Step 1 is open, showing the configuration for event S1 with qualifier R and action #'-P1'.

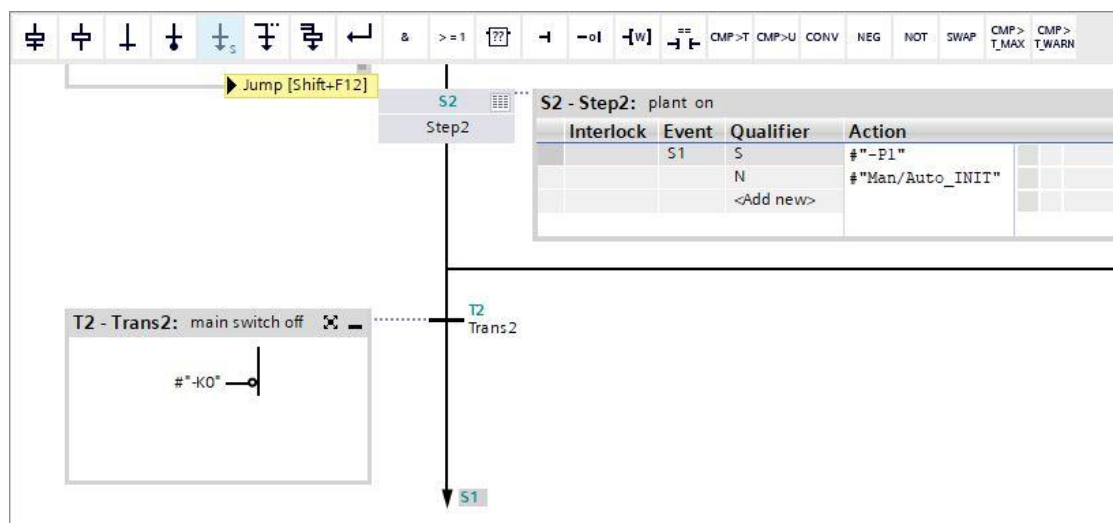
Interlock	Event	Qualifier	Action
S1	R		#'-P1'
	N		#"Man/Auto-OFF"
	N		#Automatic_OFF
	N		#Signal_Lamps_INIT
	N		#Clock_Pulse_INIT
	<Add new>		



The GRAFCET diagram shows Step 1 (S1) and Transition 1 (T1). The action table for Step 1 is open, showing the configuration for event S1 with qualifier R and action #'-P1'.

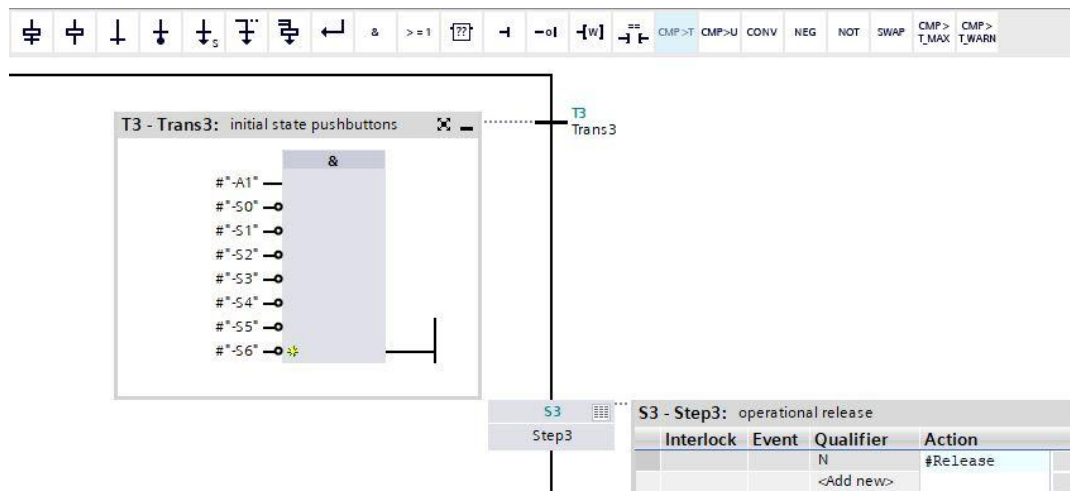
- Drag "Step and transition" onto the double arrow below Transition 1 to insert the next step with transition. The numbering continues automatically.
- Open the action table and designate **S2 – Step 2:** as " main switch on".
- In the GRAFCET for release of the controller, the first action in the second step is to retentively switch on the indicator light " main switch on" on activation of the step.
- For this, program indicator light "-P1" using event **S1** and qualifier **S**.
- Program the remaining actions in Step 2.
- After Step 2, the sequencer is divided using an alternative branch. Use Drag & Drop to move "Open alternative branch" onto the green square  below Step 2. The alternative branch with Transition 3 is inserted.




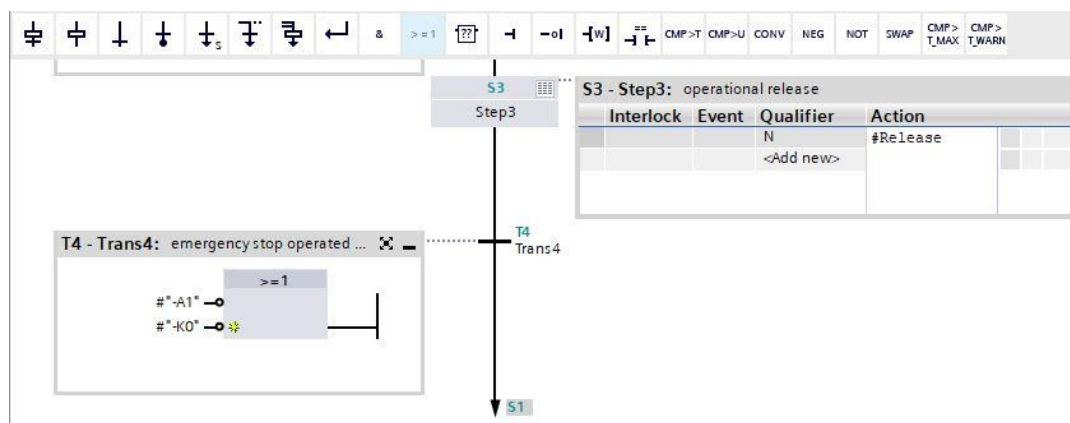
- Insert "-K0" with a negation as the step-enabling condition in the window of Transition 2.
- At the end of the sequence, we set a jump to Step 1.
- Drag a jump to the double arrow and select Step 1 as the jump destination.



- Open the input window using the " " button at Transition 3.
- Insert an AND logic operation with eight inputs in the window of Transition 3.
- Interconnect the AND logic operation according to the specification in the GRAFCET.
- Designate **T3 – Trans3:** as "initial state pushbuttons", and designate **S3 – Step 3:** as "operational release" in the action table.
- Open the action table for Step 3 using the " " button in the step field.
- Program the actions shown here in Step 3.

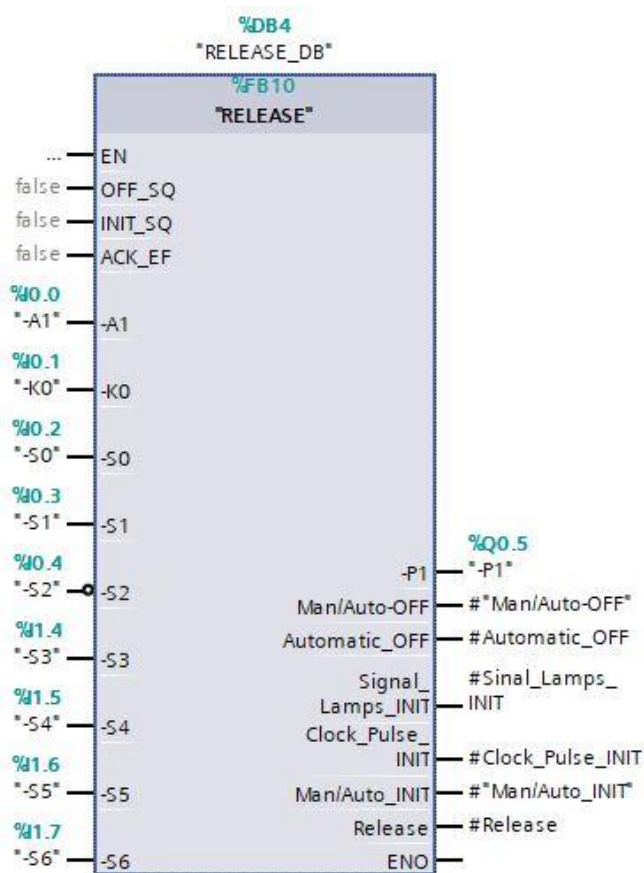




- Drag "Step and transition" onto the double arrow below Transition 3 to insert the next step with transition. The numbering continues automatically.
- Open the input window using the " " button at Transition 4 and designate **T4 – Trans4:** as "emergency stop operated or main switch off".
- Insert an OR logic operation with two inputs in the window of Transition 4.
- Interconnect the OR logic operation according to the specification in the GRAFCET.
- Drag a jump to the double arrow below Transition 4 and select Step 1 as the jump destination.



- The GRAPH function block is now finished and can be called for testing in OB1.
- Open OB1 and delete the block call in Network 1.
- Call function block of the release in Network 1.
- Confirm the name of the data block.
- Interconnect the tags of the block with the global tags of the sorting station.
- Create the local Temp tags shown here in the interface of OB1.
- Interconnect the tags of the block with the local Temp tags of OB1.

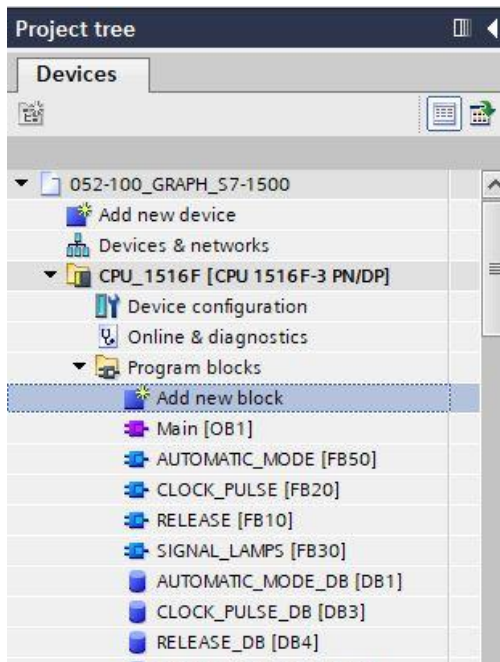
Main				
	Name	Data type	Default value	Comment
1	Input			
2	Temp			
3	Man/Auto-OFF	Bool		switch off sequencer OPERATING_MODES
4	Automatic_OFF	Bool		switch off sequencer AUTOMATIC_MODE
5	Sinal_Lamps_INIT	Bool		initialize sequencer SIGNAL_LAMPS
6	Clock_Pulse_INIT	Bool		initialize sequencer CLOCK_PULSE
7	Man/Auto_INIT	Bool		initialize sequencer OPERATING_MODES
8	Release	Bool		operational release



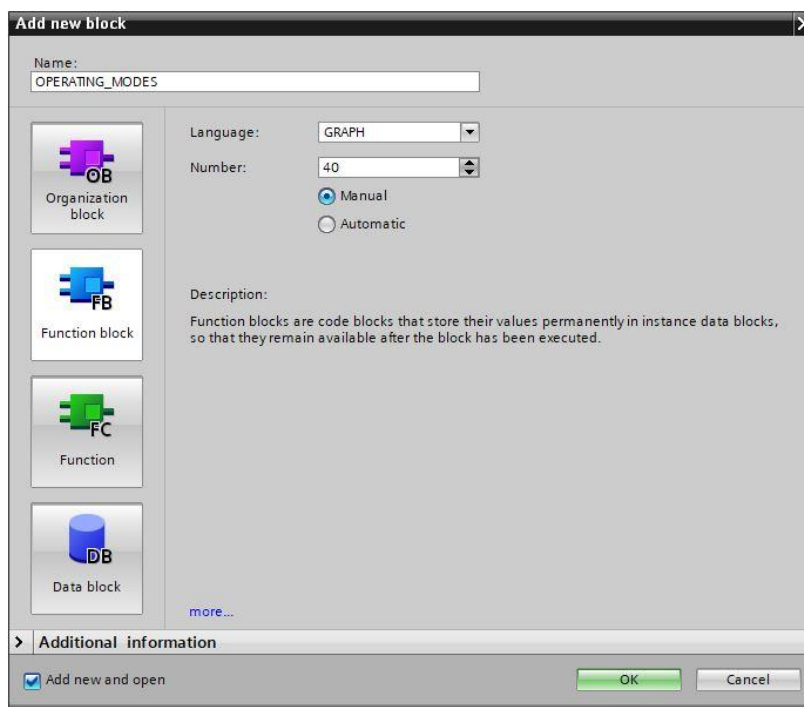
- Save  Save project your project and download it  to your controller
- Test the "RELEASE" [FB10] block.

7.25 Creating function block FB40 "OPERATING_MODES"

- Finally, the GRAPH function block "OPERATING_MODES" is to be programmed and tested.
- In the project tree under Program blocks, click on "Add new block" to create a new function block.



- Name your new function block "OPERATING_MODES". Then set the language to GRAPH and manually choose the number 40 for the function block number. Select the "Add new and open" check box. You are then taken automatically to your created function block in the project view. Click "OK".



7.26 Specifying the interface of FB40 "OPERATING_MODES"

- If you selected "Add new and open", the project view opens with a GRAPH editor for programming the block you just created.
- You can find the interface description of your function block in the upper section of your programming view. Based on the default settings in the TIA Portal, the local tags of the standard interface parameters have already been created. If necessary, these default settings can be changed in the settings of the TIA Portal.
- We need only the first three input tags. The rest of the input tags and all the output tags can be deleted.

	Name	Data type	Default value	Retain	Acce...	Wri...	Visible...	Setpo...	Sup...	Comment
1	▼ Input									
2	OFF_SQ	Bool	false	Non-ret...						Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...						Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...						Acknowledge all errors and faults
5	▼ Output									
6	<Add new>									
7	▼ InOut									
8	<Add new>									
9	▼ Static									
10	▶ RT_DATA	G7_RTDataPlu...		Non-ret...				✓		Internal data area
11	▶ Trans1	G7_Transition...		Non-ret...				✓		Transition structure
12	▶ Step1	G7_StepPlus_...		Non-ret...				✓		Step structure
13	▼ Temp									

- The static tags must not be deleted.
- The same tag designations can be used for global tags as well as local tags in the TIA Portal, which is why we can take the needed tags from the GRAFCET for controlling the signal lamps, from previously created blocks (e.g.: FB50, FB30, FB20, FB10) or from the Tag table sorting station.
- Select the last row of the input tags with the right mouse button and select menu item "Add row" (→ Input: ACK_EF → Add row).

	Name	Data type	Default value	Retain	Acce...	Wri...	Visible...	Setpo...	Sup...	Comment
1	▼ Input									
2	OFF_SQ	Bool	false	Non-ret...						Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...						Set sequence to initial state
4	ACK_EF	Bool	false	Non...						Acknowledge all errors and faults
5	Insert row									
6	Add row									
7	Cut									
8	Copy									
9	Paste									
10	Delete			Non-ret...				✓		Internal data area
11	Rename			Non-ret...				✓		Transition structure
12	Add new supervision			Non-ret...				✓		Step structure

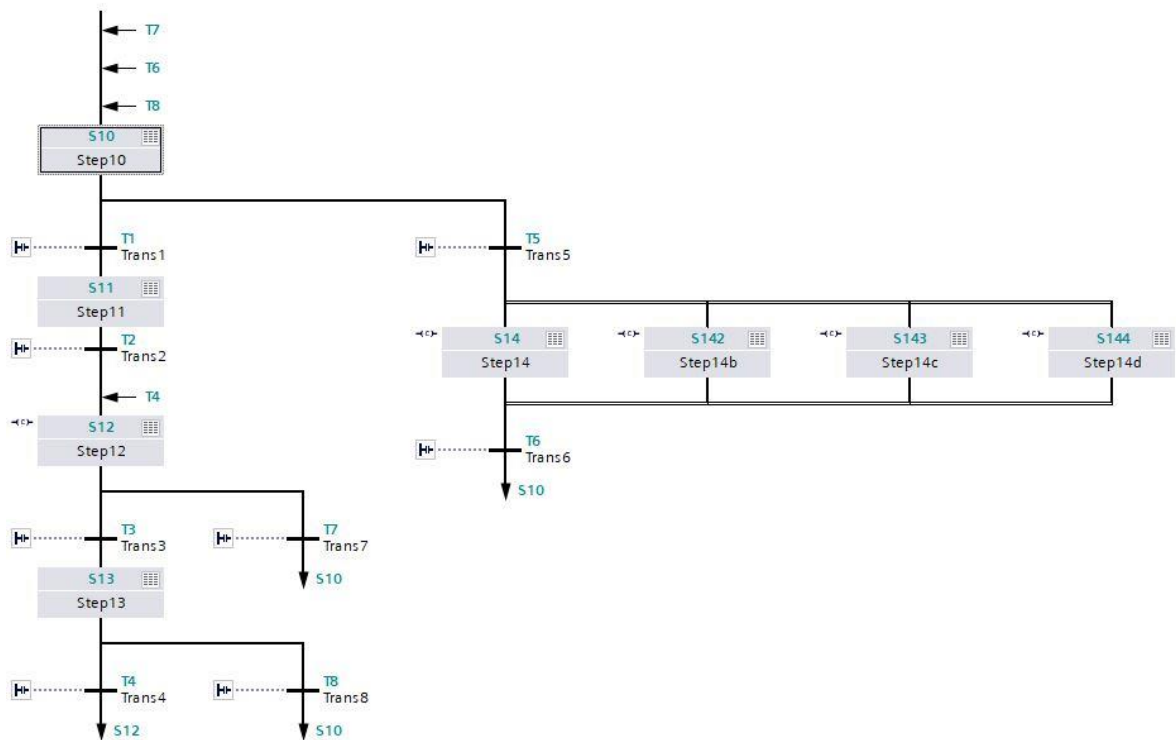
- All tags, except for the output tags #-P5 and #Automatic_Mode_Start, can be copied and pasted from the previously created function blocks.
- Alternatively, you can also copy and paste these from the tag table.

OPERATING_MODES										
	Name	Data type	Default value	Retain	Acce...	Wri...	Visible...	Setpo...	Sup...	Comment
1	▼ Input									
2	OFF_SQ	Bool	false	Non-ret...						Turn sequence off
3	INIT_SQ	Bool	false	Non-ret...						Set sequence to initial state
4	ACK_EF	Bool	false	Non-ret...						Acknowledge all errors and faults
5	Release	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
6	Clock_Pulse	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
7	-S0	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			mode selector manual(0) / automatic(1)
8	-S1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			pushbutton automatic start
9	-S2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			pushbutton automatic stop
10	-S3	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			pushbutton inching operation conveyor -M1 forwards
11	-S4	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			pushbutton inching operation conveyor -M1 backwards
12	-S5	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			pushbutton inching operation cylinder -M4 retract
13	-S6	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			pushbutton inching operation cylinder -M4 extend
14	-B1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor cylinder -M4 retracted
15	-B2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			sensor cylinder -M4 extended
16	<Add new>									
17	▼ Output									
18	-P5	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			display „automatic mode“ started
19	-Q1	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			conveyor motor -M1 forwards fixed speed
20	-Q2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			conveyor motor -M1 backwards fixed speed
21	-M2	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			cylinder -M4 retract
22	-M3	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			cylinder -M4 extend
23	Automatic_OFF	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			switch off sequencer AUTOMATIC_MODE
24	Automatic_INIT	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			initialize sequencer AUTOMATIC_MODE
25	Automatic_Mode_Start	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			start automatic mode

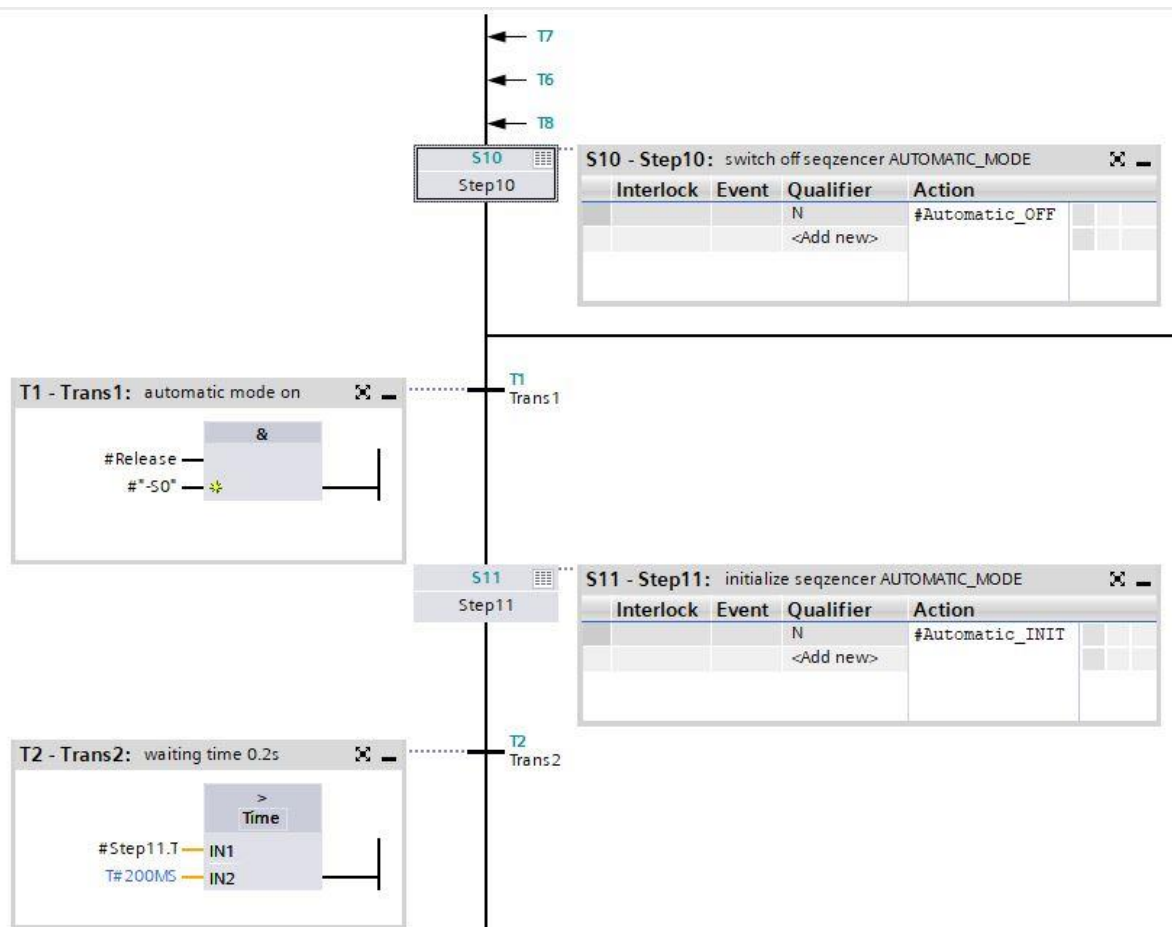
7.27 Programming of FB40 "OPERATING_MODES"

- First, we assign the sequencer the name "sequencer OPERATING_MODES" by overwriting the text <new sequence>.
- The procedure for creating a GRAPH function block is already familiar from the previous GRAPH function block programming.
- Try to create the GRAPH function block FB40 according to the specified GRAFCET of the operating mode selection on your own.
- Remember to designate the action tables and the transition windows.
- The finished block is shown on the following pages.

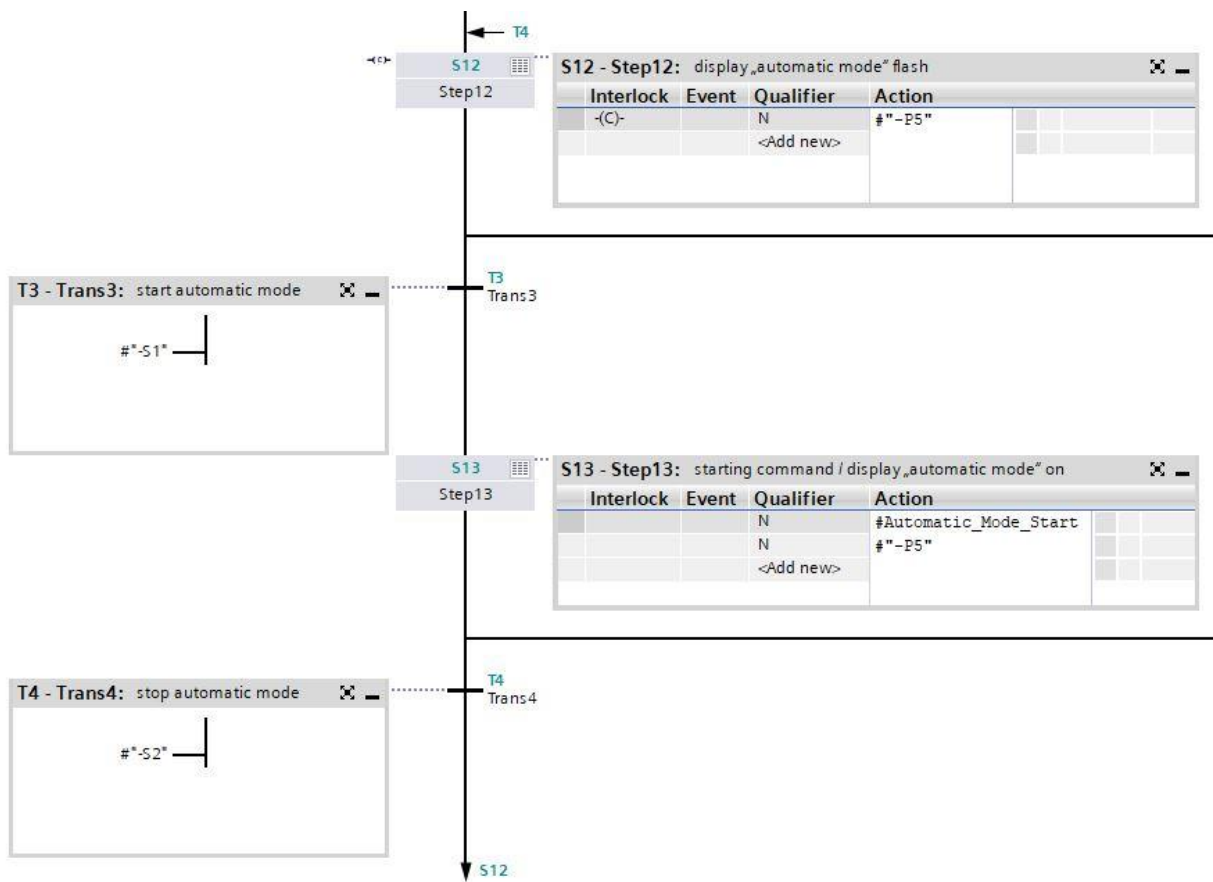
→ Sequencer of the operating mode selection



→ Steps S10 and S11, Transitions T1 and T2



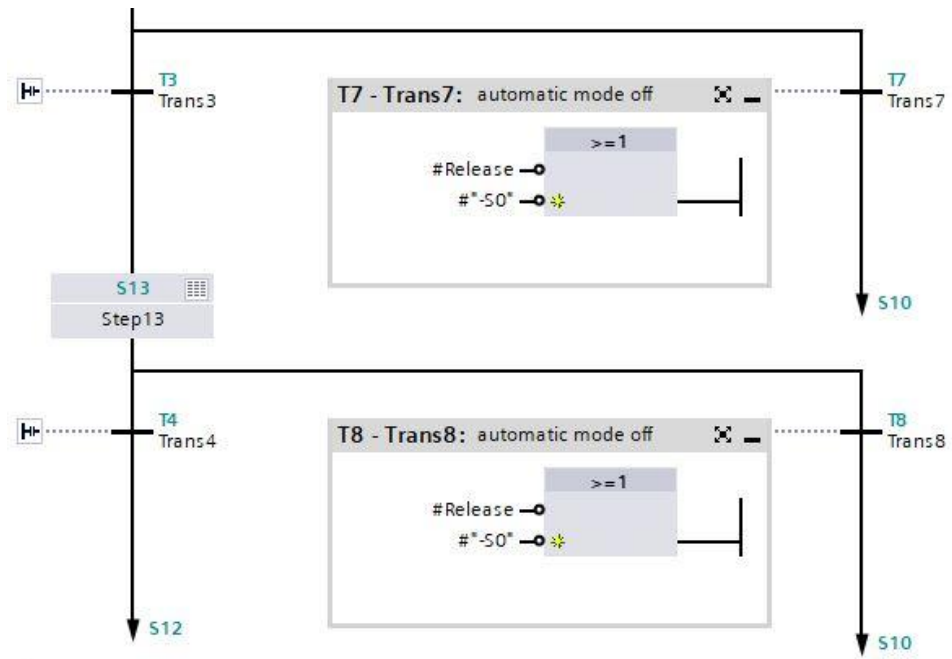
→ Steps S12 and S13, Transitions T3 and T4



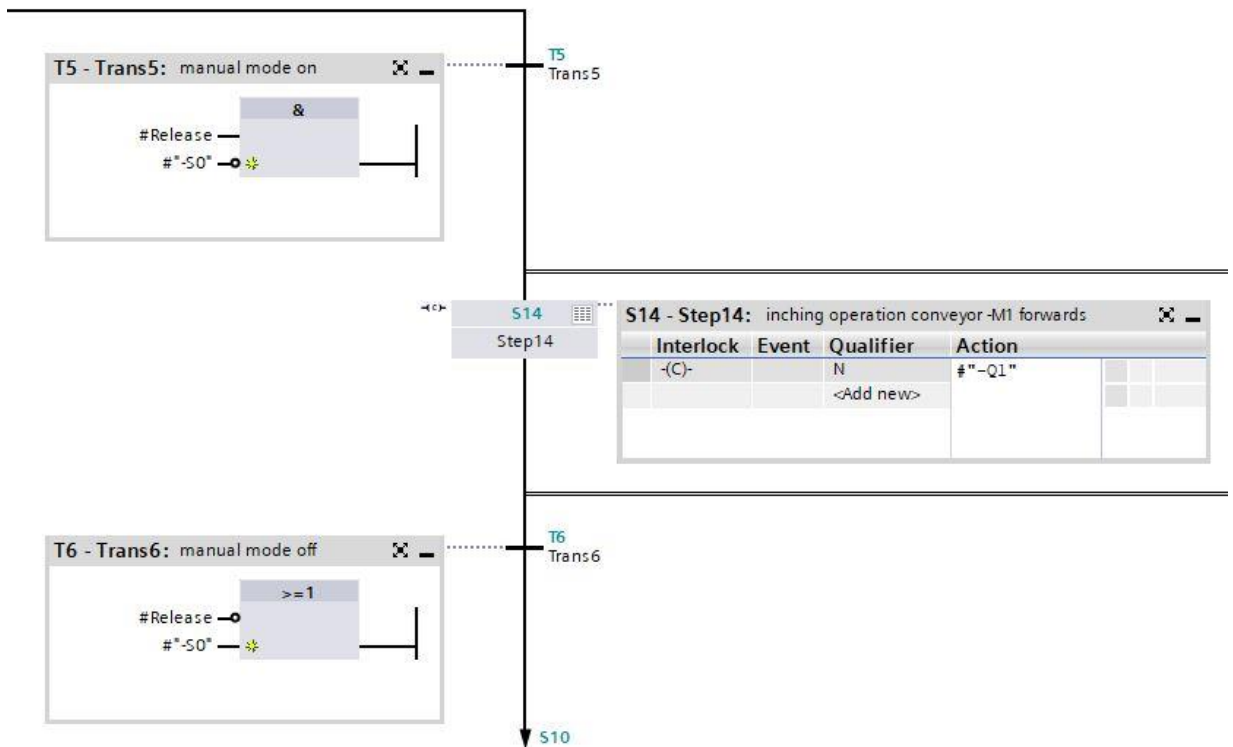
→ Step S12 single step view



→ Transitions T7 and T8



→ Step S14: Transitions T5 and T6




→ Step S14 single step view

S14: Step14

Comment

▼ **Interlock -(c)-:**



► **Supervision -(v)-:**

▼ **Actions:** inching operation conveyor -M1 forwards


Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-Q1"
		<Add new>	

→ Step S142 single step view

S142: Step14b

Comment

▼ **Interlock -(c)-:**



► **Supervision -(v)-:**

▼ **Actions:** inching operation conveyor -M1 backwards


Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-Q2"
		<Add new>	

→ Step S143 single step view

S143: Step14c

Comment

▼ **Interlock -(c)-:**



► **Supervision -(v)-:**

▼ **Actions:** inching operation cylinder -M4 retract


Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-M2"
		<Add new>	

→ Step S144 single step view

S144: Step14d

Comment

▼ **Interlock -(c)-:**



► **Supervision -(v)-:**

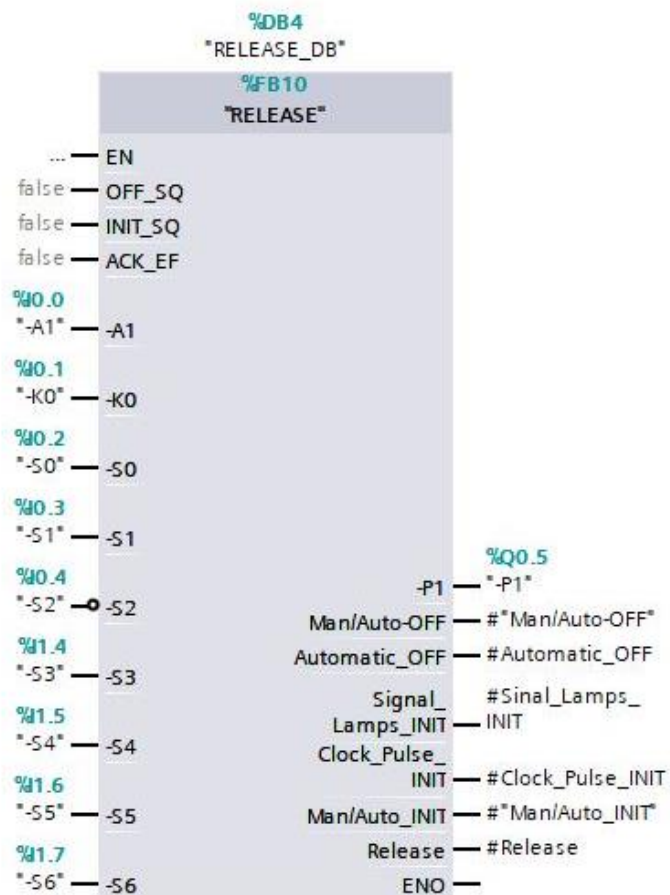
▼ **Actions:** inching operation cylinder -M4 extend, sequencer AUTOMATIC_MODE off

Interlock	Event	Qualifier	Action
-(C)-		N - Set as long as step is active	#"-M3"
		N - Set as long as step is active	#Automatic_OFF
		<Add new>	

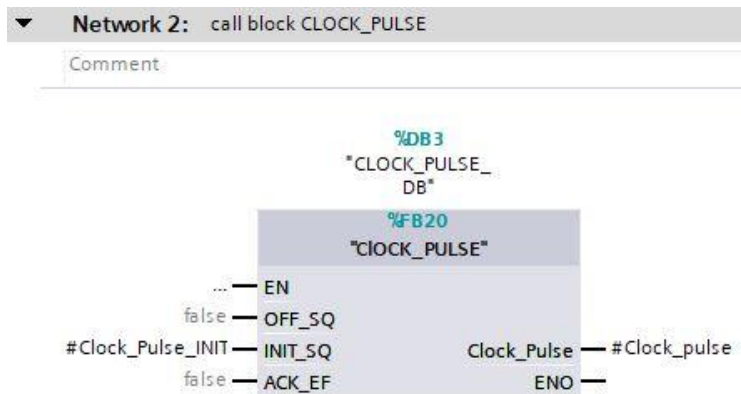
→ The GRAPH function block is now finished and can be called together with the other blocks in OB1.

→ Open OB1.

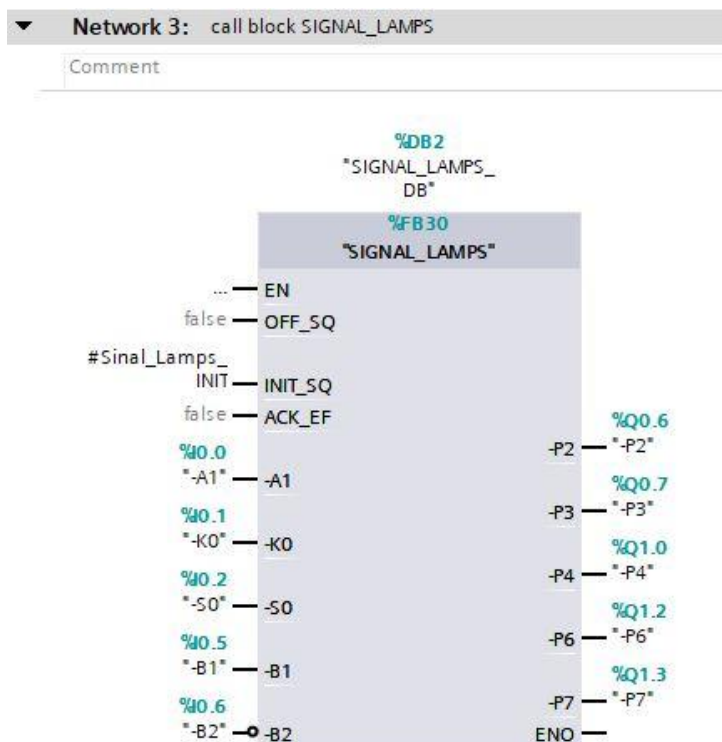
4	Temp			
5	Man/Auto-OFF	Bool		switch off sequencer OPERATING_MODES
6	Automatic_OFF	Bool		switch off sequencer AUTOMATIC_MODE
7	Sinal_Lamps_INIT	Bool		initialize sequencer SIGNAL_LAMPS
8	Clock_Pulse_INIT	Bool		initialize sequencer CLOCK_PULSE
9	Man/Auto_INIT	Bool		initialize sequencer OPERATING_MODES
10	Release	Bool		operational release



- The block of the release is called in Network 1.
- Call function block of the clock pulse in Network 2.
- Choose the existing data block DB3 of the clock pulse.
- Create a local TEMP tag named "#Clock_pulse" in the interface of OB1 and interconnect it with the block of the clock pulse.
- Interconnect the existing Temp tag #Clock_Pulse_INIT.



- Call function block of the signal lamps in Network 3.
- Choose the existing data block DB2 of the signal lamp.
- Interconnect the tags of the block with the global tags of the sorting station.
- Interconnect the existing Temp tag #Signal_Lamps_INIT.

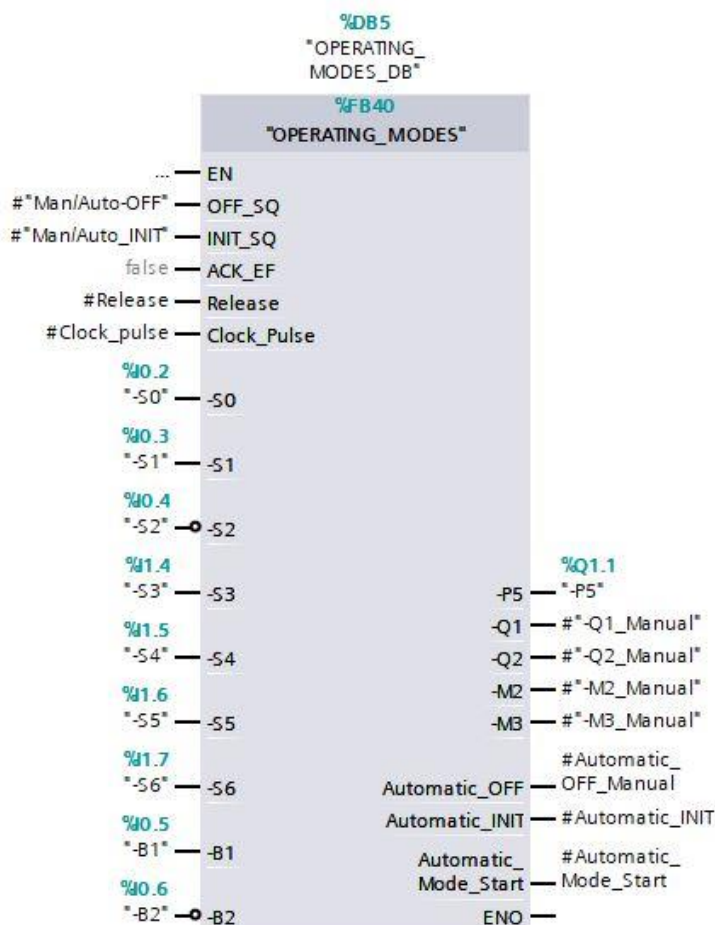


- Call function block of the operating mode selection in Network 4.
- Confirm the data block.
- Interconnect the tags of the block with the global tags of the sorting station.
- Create the local Temp tags in the interface of OB1.
- Interconnect the tags of the block with the local Temp tags of OB1.

4	Temp			
5	Man/Auto-OFF	Bool		switch off seqencer OPERATING_MODES
6	Automatic_OFF	Bool		switch off seqencer AUTOMATIC_MODE
7	Sinal_Lamps_INIT	Bool		initialize seqencer SIGNAL_LAMPS
8	Clock_Pulse_INIT	Bool		initialize seqencer CLOCK_PULSE
9	Man/Auto_INIT	Bool		initialize seqencer OPERATING_MODES
10	Release	Bool		operational release
11	Clock_pulse	Bool		clock pulse 1Hz
12	-Q1_Manual	Bool		conveyor forwards in manual mode
13	-Q2_Manual	Bool		conveyor motor backwards in manual mode
14	-M2_Manual	Bool		cylinder retract in automatic mode
15	-M3_Manual	Bool		cylinder extend in automatic mode
16	Automatic_OFF_Manual	Bool		switch off seqencer AUTOMATIC_MODE
17	Automatic_INIT	Bool		initialize seqencer AUTOMATIC_MODE
18	Automatic_Mode_Start	Bool		start automatic mode

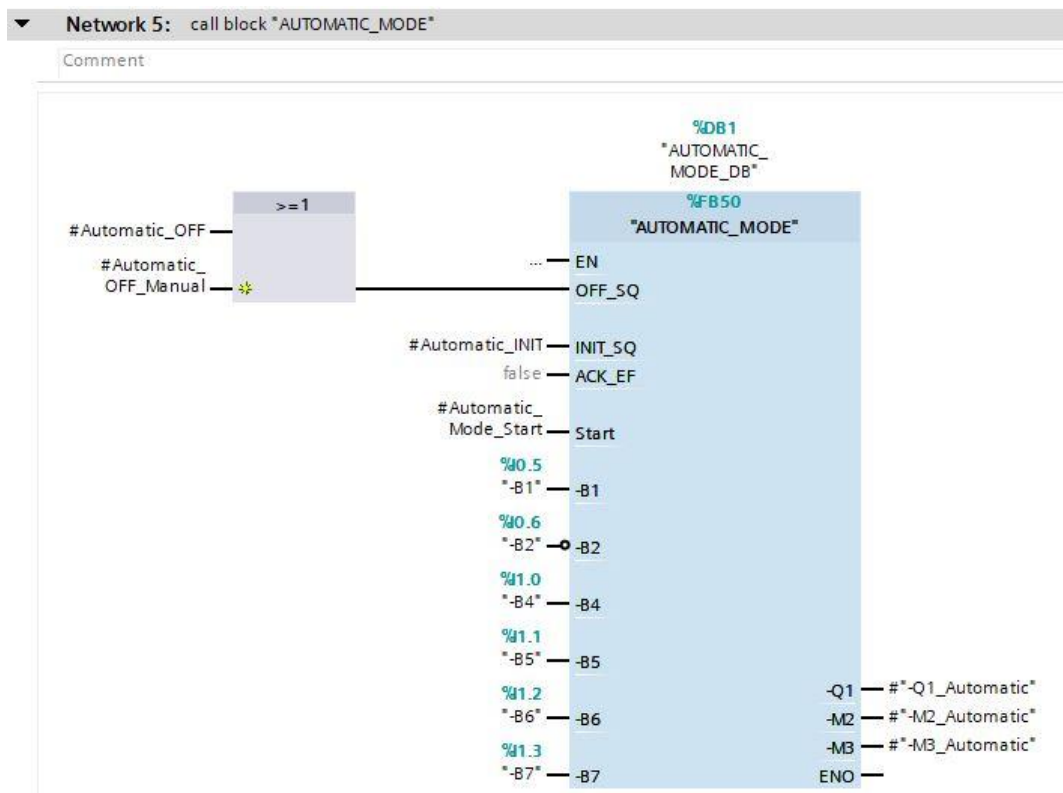
▼ **Network 4:** call block OPERATING_MODES

Comment



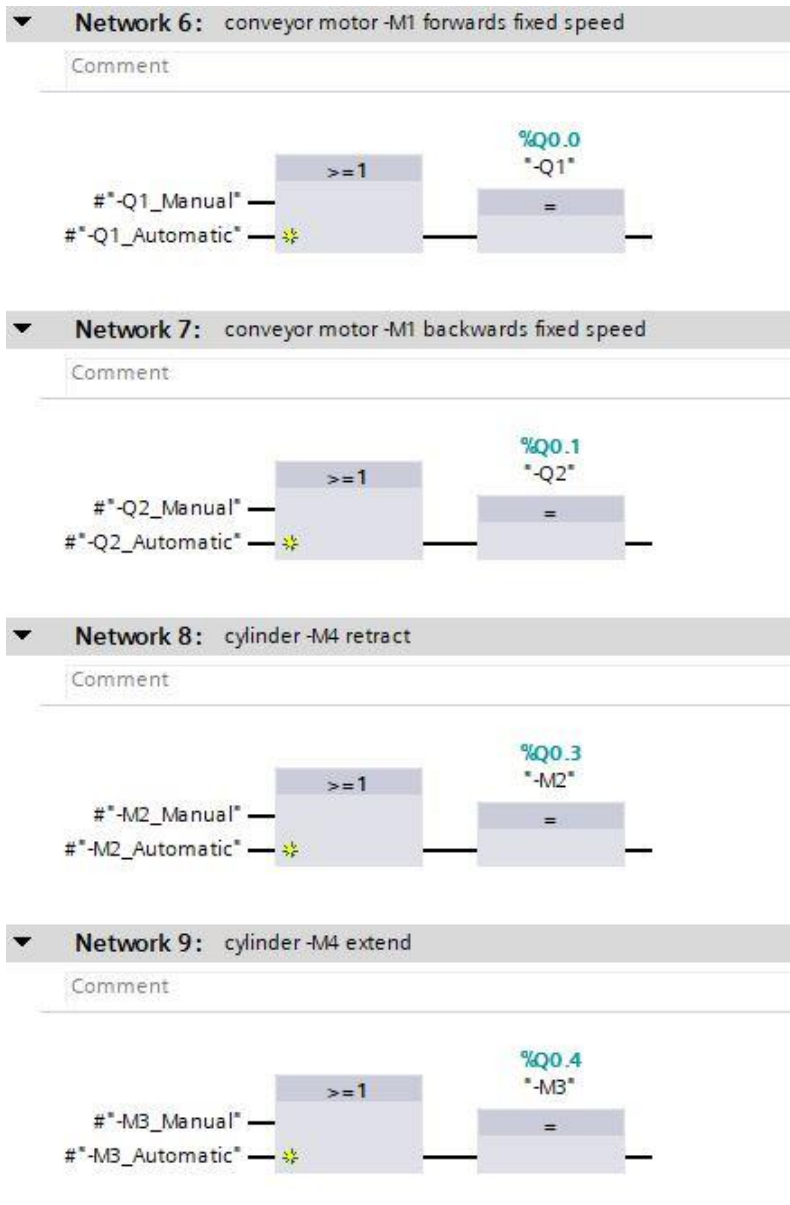
- Call function block of the automatic mode in Network 5.
- Choose the existing data block DB1 of the automatic mode.
- Interconnect the tags of the block with the global tags of the sorting station.
- Create the local Temp tags in the interface of OB1.
- Interconnect the tags of the block with the local Temp tags of OB1.






4	Temp			
5	Man/Auto-OFF	Bool		switch off sequencer OPERATING_MODES
6	Automatic_OFF	Bool		switch off sequencer AUTOMATIC_MODE
7	Sinal_Lamps_INIT	Bool		initialize sequencer SIGNAL_LAMPS
8	Clock_Pulse_INIT	Bool		initialize sequencer CLOCK_PULSE
9	Man/Auto_INIT	Bool		initialize sequencer OPERATING_MODES
10	Release	Bool		operational release
11	Clock_pulse	Bool		clock pulse 1Hz
12	-Q1_Manual	Bool		conveyor forwards in manual mode
13	-Q2_Manual	Bool		conveyor motor backwards in manual mode
14	-M2_Manual	Bool		cylinder retract in automatic mode
15	-M3_Manual	Bool		cylinder extend in automatic mode
16	Automatic_OFF_Manual	Bool		switch off sequencer AUTOMATIC_MODE
17	Automatic_INIT	Bool		initialize sequencer AUTOMATIC_MODE
18	Automatic_Mode_Start	Bool		start automatic mode
19	-Q1_Automatic	Bool		conveyor forwards in automatic mode
20	-Q2_Automatic	Bool		conveyor motor backwards in automatic mode
21	-M2_Automatic	Bool		cylinder retract in automatic mode
22	-M3_Automatic	Bool		cylinder extend in automatic mode



- The OR logic operation is needed at input OFF_SQ because the sequencer can be switched off from either the release or the operating mode selection.

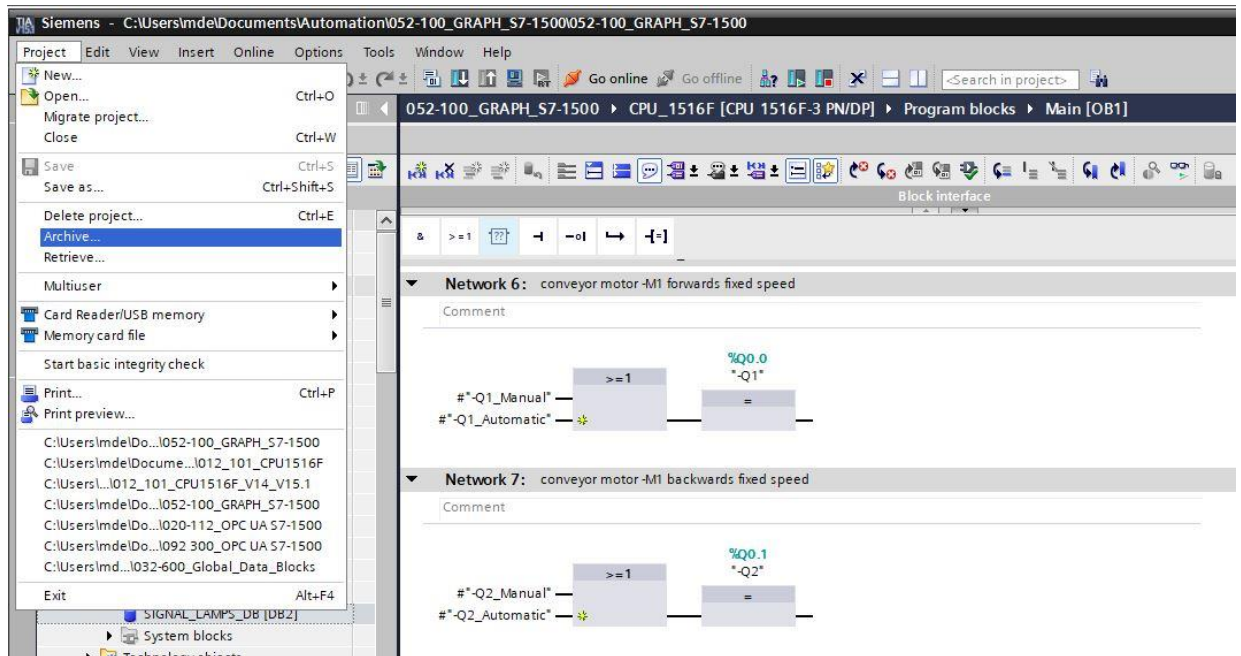
→ Create networks 6 to 9 to assign the outputs correctly.



- To save your project, select the  Save project button in the menu.
- Then, click the "Program blocks" folder and select the button  for compiling all blocks in the menu. (→  Save project → Program blocks → )
- After successful compilation, the complete controller with the created program, as previously described in the modules for hardware configuration, can be downloaded (→ )
- Test the program of the sorting station.

7.28 Archiving the project

→ As the final step, we want to archive the complete project. Select the → "Archive ..." command in the → "Project" menu. Select a folder where you want to archive your project and save it as file type "TIA Portal project archive". (→ Project → Archive → TIA Portal project archive → sce-052-100-graph-s7-1500.... → Save)



7.29 Checklist – step-by-step instructions

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

No.	Description	Checked
1	Function block "AUTOMATIC_MODE" with sequencer created in GRAPH	
2	Function block "AUTOMATIC_MODE" successfully downloaded and tested	
3	Function block "SIGNAL_LAMPS" with sequencer created in GRAPH	
4	Function block "SIGNAL_LAMPS" successfully downloaded and tested	
5	Function block "CLOCK_PULSE" with sequencer created in GRAPH	
6	Function block "CLOCK_PULSE" successfully downloaded and tested	
7	Function block "RELEASE" with sequencer created in GRAPH	
8	Function block "RELEASE" successfully downloaded and tested	
9	Function block "OPERATING_MODES" with sequencer created in GRAPH	
10	Function block "OPERATING_MODES" successfully downloaded and tested	
11	Project successfully archived	

8 Exercise

8.1 Task – Exercise

In this exercise, the function block PRG_SORTING_STATION [FB1] is to be added to the created control program

The existing block calls in OB1 are to be implemented in function block PRG_SORTING_STATION [FB1]. The block is to be library-suitable, meaning that only local tags are to be used in the block.

The block calls of the GRAPH function blocks will therefore be implemented with a parameter instance.

The function block is to be planned, programmed and tested.

Only the function block PRG_SORTING_STATION [FB1] together with the associated data block is permitted to be called in OB1.

The automatic mode is to be changed in such a way that it does not have to be restarted for each part but rather continues operating cyclically.

8.2 Planning

Plan the implementation of the tasks on your own.

8.3 Checklist – Exercise

The following checklist helps trainees/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	Library-suitable function block "PRG_SORTING_STATION" with call of sequencers created	
2	Function block "PRG_SORTING_STATION" called in organization block "Main" [OB1]	
3	Changed blocks successfully downloaded and tested	
4	Project successfully archived	

9 Additional information

You can find additional information as an orientation aid to familiarize yourself or deepen your knowledge, for example: Getting Started, videos, tutorials, apps, manuals, programming guidelines and trial software/firmware, at the following link:

[Advanced Programming](#)

Preview "Additional information"

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

- [TIA Portal Videos](#)
- [TIA Portal Tutorial Center](#)
- [Getting Started](#)
- [Programming Guideline](#)
- [Easy Entry in SIMATIC S7-1200](#)
- [Download Trial Software/Firmware](#)
- [Technical Documentation SIMATIC Controller](#)
- [Industry Online Support App](#)
- [TIA Portal, SIMATIC S7-1200/1500 Overview](#)
- [TIA Portal Website](#)
- [SIMATIC S7-1200 Website](#)
- [SIMATIC S7-1500 Website](#)

Additional information

Siemens Automation Cooperates with Education

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

SCE Learn-/Training Document

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

SCE Trainer Packages

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

SCE Contact Partners

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

Digital Enterprise

[siemens.com/digital-enterprise](https://www.siemens.com/digital-enterprise)

Industrie 4.0

[siemens.com/future-of-manufacturing](https://www.siemens.com/future-of-manufacturing)

Totally Integrated Automation (TIA)

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

TIA Portal

[siemens.com/tia-portal](https://www.siemens.com/tia-portal)

SIMATIC Controller

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

SIMATIC Technical Documentation

[siemens.com/simatic-docu](https://www.siemens.com/simatic-docu)

Industry Online Support

support.industry.siemens.com

Industry Mall catalog and ordering system

mall.industry.siemens.com

Siemens

Digital Industries, FA

P.O. Box 4848

90026 Nuremberg

Germany

Errors excepted and subject to change without prior notice.

© Siemens 2019

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