

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



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: <u>siemens.com/sce/tp</u>

Training courses For regional Siemens SCE training courses, contact your regional SCE representative: <u>siemens.com/sce/contact</u>

Additional information regarding SCE

siemens.com/sce

Information regarding use

The SCE Learn-/Training Document for the integrated automation solution Totally Integrated Automation (TIA) was prepared for the program "Siemens Automation Cooperates with Education (SCE)" specifically for training purposes for public educational 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 <u>scesupportfinder.i-ia@siemens.com</u>.

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

Use for industrial customer courses is 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 | G | oal 6 |
|---|------|---|
| 2 | R | equirement6 |
| 3 | R | equired hardware and software7 |
| 4 | Т | heory8 |
| | 4.1 | Notes on S7-GRAPH programming language8 |
| | 4.2 | GRAFCET according to EN 608488 |
| 5 | Т | ask9 |
| | 5.1 | Description of control task9 |
| | 5.2 | Technology diagram9 |
| | 5.3 | Switching on 10 |
| | 5.4 | Operating mode selection |
| | 5.5 | EMERGENCY STOP |
| | 5.6 | Manual mode10 |
| | 5.7 | Automatic mode11 |
| | 5.8 | Indicator lights |
| | 5.9 | Reference list14 |
| 6 | Ρ | lanning16 |
| | 6.1 | Sequence diagram of the sorting station17 |
| 7 | S | tructured step-by-step instructions 21 |
| | 7.1 | Retrieving an existing project21 |
| | 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" |
| | 7.5 | Specifying the interface of FB50 "AUTOMATIC_MODE"27 |
| | 7.6 | Structure of the sequencer |
| | 7.7 | Programming of FB50 "AUTOMATIC_MODE" 31 |
| | 7.8 | Programming the organization block OB148 |
| | 7.9 | Result in the LAD programming language53 |
| | 7.10 | Saving and compiling the program54 |

| | 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 | Exe | ercise | 105 |
| | 8.1 | Task – Exercise | 105 |
| | 8.2 | Planning | 105 |
| | 8.3 | Checklist – Exercise | 105 |
| 9 | Ado | litional 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



3 SIMATIC S7-1500 controller

Panel

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.



 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.

8

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

| -P4 aktiviert/active | -53 Tippbetrieb -M1 vorwärts/ Manual -M1 forwards -54 Tippbetrieb -M1 rückwärts/ Manual -M1 forwards -54 Tippbetrieb -M1 rückwärts/ Manual -M1 backwards -56 Zylinder -M4 ausfahren/ cylinder -M4 einfahren/ | -P7 ausgefahren/extended -P6 eingefahren/retracted |
|----------------------|---|---|
|----------------------|---|---|

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" or -Q1 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 | Туре | Identifier | Function | NC/NO |
|-------|------|------------|--|---------------------|
| I 0.0 | BOOL | -A1 | return signal emergency stop ok | NC |
| I 0.1 | BOOL | -K0 | main switch "ON" | NO |
| 10.2 | BOOL | -S0 | mode selector manual(0) / automatic(1) | man = 0 auto = 1 |
| 10.3 | 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 |
| 10.7 | BOOL | -B3 | sensor motor -M1 active (pulse signal for positioning) | NO |
| I 1.0 | BOOL | -B4 | sensor part at slide | NO |
| 11.1 | 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 |
| l 1.5 | BOOL | -S4 | pushbutton manual mode conveyor –M1 | NO |
| l 1.6 | BOOL | -S5 | pushbutton manual mode cylinder -M4 retract | NO |
| l 1.7 | BOOL | -S6 | pushbutton manual mode cylinder -M4 extend | NO |

Learn-/Training Document | TIA Portal Module 052-100, Edition 10/2019 | Digital Industries, FA

| DO | Туре | 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 Analoger Output
- 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".

 $(\rightarrow \text{Project} \rightarrow \text{Retrieve} \rightarrow \text{Select} \text{ a *.zap15_1 archive } \dots \rightarrow \text{Open})$

| Project | Edit | View | Insert | Online | Option |
|------------------|------------------|--------|--------|--------|--------------------|
| New | · | 100.00 | | | Ctrl+0 |
| Close | te proj | ect | | | Ctrl+W |
| Save Save | | | | Ctrl | Ctrl+S +Shift+S |
| Delet Archiv | e proje ve | ct | | | Ctrl+E |
| Retrie | ve | | | | |
| Retrie Multiu | a na serie da na | | | | |

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

| 📑 🔚 Save project 📑 🐰 🗉 😭 🗙 | | _101_CPU1516F\012_101_CPU15 Window Help | | | | Totally | Integrated A | utomation PORT |
|-------------------------------|---|--|-----------|----------|-----------|-------------|--------------|-------------------|
| roject tree | | 012_101_CPU1516F > CPU_1 | | | | fault tag t | table [54] | _ # # |
| Devices | | | | 🕣 Tags 🛛 | 🗉 User co | onstants | System o | constants |
| ý | | * * > E * # | | | | | | E |
| | | Default tag table | | | | | | |
| 012_101_CPU1516F | ~ | Name Import | Data type | Address | Retain | Acces V | Writa Visibl | Supervis |
| Add new device | | 1 <add new=""></add> | | | | | | |
| Bevices & networks | | | | | | | | |
| CPU_1516F [CPU 1516F-3 PN/DP] | | | | | | | | |
| Device configuration | = | | | | | | | |
| 😼 Online & diagnostics | | | | | | | | |
| Program blocks | | | | | | | | |
| 🕨 🙀 Technology objects | | | | | | | | |
| Energy objects | | | | | | | | |
| External source files | | | | | | | | |
| 🕶 🔁 PLC tags | | | | | | | | |
| a Show all tags | | | | | | | | |
| Add new tag table | | | | | | | | |
| Default tag table [54] | | | | | | | | |
| PLC data types | | | | | | | | |

 \rightarrow Use the \blacksquare button to choose the location of the import file.

| l | mport × |
|---|---|
| | Path of import file: |
| | C:\Users\mde\Desktop\sce-052-100-graph-s7-1500-r1904-en\sce-052-100-projects-[] |
| | Elements to be imported: 🗹 Tags |
| | OK Cancel |

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

 $(\rightarrow sce-020-100-tag-table-sorting-station... \rightarrow Open \rightarrow OK \rightarrow OK)$

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



→ You have now imported the tag table of the sorting station. Save your project under the name

052-100_ GRAPH_S7-1500. (\rightarrow Project \rightarrow Save as ... \rightarrow 052-100_GRAPH_S7-1500 \rightarrow Save)

| Project Edit View Insert Online | | Tool | | idow | State and | 💋 Go on | line 🔊 Go off | fline | | × - | • | | Totally Integrated Auto | PORTA |
|--------------------------------------|-----------------|------|------|--------|------------|------------|---------------|----------|---------|---------|---------|--------|--|---------|
| Open | Ctrl+O | | € 01 | 2 101 | CPU1516 | 5F → CPU | 1516F [CPL | J 1516F- | 3 PN/DP | I → PLC | tags ▶ | Tag ta | able sorting station [28] | _ # # X |
| Migrate project Close | Ctrl+W | | | | | | | | | | | | | |
| | | | | | | | | | | | | | Tags 🔲 User co | |
| Save | Ctrl+S | | | 1 | | î (ii) | | | | | | | | |
| Save as Ctrl- | -Shift+S | | | Tag ta | able_sorti | ng station | | | | | | | | |
| Delete project | Ctrl+E | | ^ | | Name | Data type | Address | Retain | Acces | Writa | Visibl | Sup | Comment | |
| Archive | | | 1 | - | -A1 | Bool | %IO.0 | - | | | | 1 | return signal emergency stop ok (nc) | |
| Retrieve | | | 2 | - | -K0 | Bool | %10.1 | | | | | | main switch "ON" (no) | |
| Multiuser | • | | 3 | - | -S0 | Bool | %10.2 | | | | | | mode selector manual(0) / automatic(1) | |
| Card Reader/USB memory | • | | 4 | - | -S1 | Bool | %10.3 | | | | | | pushbutton automatic start (no) | |
| Memory card file | | | = 5 | | -52 | Bool | %10.4 | | | | | | pushbutton automatic stop (nc) | |
| | 10 ₂ | | 6 | - | -B1 | Bool | %10.5 | | | | | | sensor cylinder -M4 retracted (no) | |
| Start basic integrity check | | | 7 | - | -B2 | Bool | %10.6 | | | | | | sensor cylinder -M4 extended (nc) | |
| Print | Ctrl+P | | 8 | - | -B3 | Bool | %10.7 | | | | | | sensor motor -M1 actice (pulse signal for posi | tioning |
| Print preview | | | 9 | - | -B4 | Bool | %11.0 | | | | | | sensor part at slide (no) | |
| C:\Users\mde\Docume\012_101_CPU1516F | | | 10 | - | -B5 | Bool | %11.1 | | | | | | sensor metal part (no) | |
| C:\Users\\012_101_CPU1516F_V14_V1 | 15.1 | | 11 | - | -B6 | Bool | %11.2 | | | | | | sensor part in front of cylinder -M4 (no) | |
| C:\Users\mde\Do\052-100_GRAPH_S7- | 1500 | | 12 | - | -B7 | Bool | %11.3 | | | | | | sensor part at end of conveyor (no) | |
| C:\Users\mde\Do\020-112_OPC UA S7 | 1500 | | 13 | - | -53 | Bool | %11.4 | | | | | | pushbutton manual mode conveyor –M1 forw | ards (|
| C:\Users\mde\Do\092 300_OPC UA S7- | | | 14 | - | -54 | Bool | %11.5 | | | | | | pushbutton manual mode conveyor –M1 bac | wards |
| C:\Users\md\032-600_Global_Data_Bl | ocks | | 15 | - | -\$5 | Bool | %11.6 | | | | | | pushbutton manual mode cylinder -M4 retrac | t (no) |
| Exit | Alt+F4 | | 16 | - | -56 | Bool | %11.7 | | | | | | pushbutton manual mode cylinder -M4 extend | l (no) |
| Unine backups | | 1 | 17 | - | -Q1 | Bool | %Q0.0 | | | | | | conveyor motor -M1 forwards fixed speed | |
| 🕨 🔛 Traces | | | 18 | - | -Q2 | Bool | %Q0.1 | | | | | | conveyor motor -M1 backwards fixed speed | |
| 🕨 📴 Device proxy data | | | 19 | - | -Q3 | Bool | %Q0.2 | | | | | | conveyor motor -M1 variable speed | |
| Program info | | | 20 | - | -M2 | Bool | %Q0.3 | | | | | | cylinder -M4 retract | |
| PLC supervisions & alarms | | | 21 | - | -M3 | Bool | %Q0.4 | | | | | | cylinder -M4 extend | |
| PLC alarm text lists | | | 22 | - | -P1 | Bool | %Q0.5 | | | | | | display "main switch on" | 1 |
| 🕨 🛅 Local modules | | | ~ | < | | 1 | | | - | 1000 | | | | > |
| > Details view | | | | | | | | | | | | Prope | erties 1 Info 👔 🗓 Diagnostics | 124 |

7.3 Creating function block FB50 "AUTOMATIC_MODE"

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



For unrestricted use in educational and R&D institutions. © Siemens 2019. All rights reserved. sce-052-100-graph-s7-1500-r1903-en.docx

→ 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 → Madd new and open → OK)

| Add new block | | | | | X |
|-----------------------|---|---|--|---|---|
| Name: | | | | | |
| AUTOMATIC_MODE | | | | | |
| | Language: | GRAPH | • | | |
| OB | Number: | 50 | • | | |
| Organization block | | O Manual ○ Automatic | | | |
| Function block | Description: Function blocks so that they rer | are code blocks that nain available after th | store their value: e block has been | s permanently in instance data blocks, n executed. | |
| Function | | | | | |
| | | | | | |
| Data block | | | | | |
| | more | | | | |
| > Additional infor | mation | | | | _ |
| Add new and oper | 1 | | | 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" (\rightarrow Properties \rightarrow General \rightarrow Attributes). Select the attributes as shown here.

| 052-100_GRAPH_\$7-1500 > CPL | J_1516F [CPU 1516F-3 PN/DP] 	→ Program blocks 	→ AUTOMATIC_MODE [FB50] | _ • • × |
|--|--|---------|
| Navigation 🔍 🤉 🖶 | 4 ഈ 환 •, 탇 펌 플 ☞ 웹 ± 월 ± 월 안 •, 당 장 용 약 Block interface 낮 수 ↓ ↓ ↓ Ţ 주 ← | 3 |
| Permanent pre-instructio Sequences (1) Permanent post-instructi Alarms | 1: <new sequence=""> Comment</new> | |
| AUTOMATIC_MODE [FB50] | 🖳 Properties 🚺 Info 🚺 🗓 Diagnostics | |
| General Texts FB supe General Information Information Time stamps Compilation Protection Attributes Attributes Information | Attributes | |

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.

| | 1 1 | MATIC_MO | | Data type | Default valu | e Retain | Accessible f | Writa | Visible in | Setpoint | Supe |
|----|-------|--------------|--------------|-------------------|---------------|------------|--------------|-------|------------|----------|------|
| | | Input | | - | - | | | | | | |
| 2 | | OFF_SQ | | Bool | false | Non-retain | | | | | |
| 3 | | INIT_SQ | | Bool | false | Non-retain | | | | | |
| ŧ. | - | ACK_EF | | Bool | false | Non-retain | | | | | 1 |
| 5 | | S_PREV | | Bool | false | Non-retain | | | | | |
| 5 | | S_NEXT | | Bool | false | Non-retain | 8 | | | | |
| 7 | | SW_AUTO |).) | Bool | false | Non-retain | | | | | |
| В | - | SW_TAP | | Bool | false | Non-ret 💌 | | | | | |
| 9 | | SW_TOP | 🚔 Insert rov | N | Ctrl+Enter | Non-retain | | | | | |
| 10 | | SW_MAN | 🚔 Add row | | Alt+Ins | Non-retain | | | | | |
| 11 | | S_SEL | 🗶 Cut | | Ctrl+X | Non-retain | | | | | |
| 12 | | S_ON | Copy | | Ctrl+C | Non-retain | | | | | |
| 13 | | S_OFF | 💼 Paste | | Ctrl+V | Non-retain | | | | | |
| | - | T_PUSH | X Delete | | Del | Non-retain | | | | | |
| 15 | • | Output | Rename | | F2 | | | | | | - |
| | < | | 🗘 Add new | supervision | | | a de de | | | | > |
| Na | | on @ | Update in | nterface | 3 | | | | | | |
| > | Perma | anent pre-in | Go to ner | d point of use | Ctrl+Shift+G | | | | | | |
| > | 🕄 Se | quences (1) | Go to def | inition | Ctrl+Shift+D | | | | | | |
| > | Perma | anent post-i | Cross-ref | | F11 | | | | | | |
| | Alarn | | Cross-ref | erence informatio | n Shift+F11 | | | | | | |
| | | | • | | T1 Trans 1 | | | | | | |

 \rightarrow The static tags must not be deleted.

| | | Na | ime | 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=""></add> | | | | | | | | |
| 7 | | • | InOut | | | - | | | | | |
| 8 | | | <add new=""></add> | | | | | | - | | |
| 9 | | • | Static | | | | | | | | |
| 10 | | | RT_DATA | G7_RTDataPlus | | Non-ret | | | ~ | | Internal data area |
| 11 | | | Trans1 | G7_TransitionP | | Non-ret | | | \checkmark | | Transition structure |
| 12 | | | Step1 | G7_StepPlus_V6 | | Non-ret | | | | | Step structure |
| 13 | | • | Temp | | | | | | | | |
| 14 | | | <add new=""></add> | | | | | | | | |
| 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" (\rightarrow Input \rightarrow ACK_EF \rightarrow Add row)

| | AU | TOMATIC_MOD | DE | | | |
|----|-----|-----------------|-----------|---------------|---------|----|
| | | Name | Data type | Default value | Retain | Ac |
| 1 | | ▼ Input | | | | |
| 2 | | OFF_SQ | Bool | false | Non-ret | |
| 3 | | INIT_SQ | Bool | false | Non-ret | |
| 4 | 5 | Insert row | Ctrl | Enter | No 💌 | |
| 5 | 4 | Add row | | Alt+Ins | | |
| 6 | | | 3 | | | |
| 7 | | Cut | | Ctrl+X | | |
| 8 | | Сору | | Ctrl+C | | |
| 9 | - | Paste | | Ctrl+V | | |
| 10 | < > | < Delete | | Del | Non-ret | |
| 11 | - | Rename | | F2 | Non-ret | |
| 12 | 1 | 🖸 Add new super | vision | | Non-ret | |

- → 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.
- \rightarrow Alternatively, you can also copy and paste these from the tag table.

Learn-/Training Document | TIA Portal Module 052-100, Edition 10/2019 | Digital Industries, FA

| | | Na | me | Data type | Defau | Retain | Ac | Visi | Set | Sup | Comment |
|----|-----|----|---------|-----------|-------|---------|----|----------|-----|-----|---|
| 1 | -00 | - | Input | 21 | | | | | | | |
| 2 | -00 | | OFF_SQ | Bool | false | Non-ret | | | | | |
| 3 | - | | INIT_SQ | Bool | false | Non-ret | | | | | |
| 4 | - | | ACK_EF | Bool | false | Non-ret | | | | | |
| 5 | -00 | | Start | Bool | false | Non-ret | | | | | starting command |
| 6 | - | | -B1 | Bool | false | Non-ret | | | | | sensor cylinder -M4 retracted |
| 7 | | | -B2 | Bool | false | Non-ret | | | | | sensor cylinder -M4 extended |
| 8 | - | | -B4 | Bool | false | Non-ret | | | | | sensor part at slide |
| 9 | - | | -B5 | Bool | false | Non-ret | | | | | sensor metal part |
| 10 | - | | -B6 | Bool | false | Non-ret | | | | | sensor part in front of cylinder -M4 |
| 11 | | | -B7 | Bool | false | Non-ret | | | | | sensor part at end of conveyor |
| 12 | - | • | Output | | | | | | | | |
| 13 | - | | -Q1 | Bool | false | Non-ret | | | | | conveyor motor -M1 forwards fixed speed |
| 14 | | | -M2 | Bool | false | Non-ret | | | | | cylinder -M4 retract |
| 15 | - | | -M3 | Bool | false | Non-ret | | | | | cylinder -M4 extend |

7.6 Structure of the sequencer

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

| | A | UTON | ATIC_MODE | | | | | | | | | |
|----|----|--|--|-----------|-------|-----------|-------|------|--------|-------|---------|--|
| | | Nam | e | Data type | Defau | Retain | Ac | | Visi | Set | Sup | Comment |
| 1 | - | i 🔻 I | nput | | | | | | | | | · |
| 2 | - | | OFF_SQ | Bool | false | Non-ret | | | | | | |
| 3 | | | INIT_SQ | Bool | false | Non-ret | | | | | | × |
| >~ | Pe | Seque 1: <r< th=""><th>ent pre-instr Jences (1) new sequences</th><th>vectio</th><th>Comme</th><th>wsequence</th><th>•</th><th>s</th><th>Ŧ</th><th>₽</th><th></th><th>field</th></r<> | ent pre-instr Jences (1) new sequences | vectio | Comme | wsequence | • | s | Ŧ | ₽ | | field |
| | Т | Frans | sition or | | ľ | * | The | firs | st ste | ep of | the | sequencer is automatically inserted in |
| | S | step- | enabling c | condition | | | the | bloo | ck. T | This | step | will be labeled as initial step and is |
| | | | | | _ | | activ | /e a | t the | e sta | rt of t | the sequencer. |
| > | Pe | erman | ent post-inst | tructi | | | | | | | | |
| > | A | larms | | | | | | | | | | 100% |

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

| 🛋 🛃 🔮 🔮 🐛 🔚 🚍 | 🖂 🗐 + 👷 + 📴 + 🔝 | 10 (D 0 00 |
|-----------------|-----------------|-----------------------|
| юны 🖦 📰 🚽 💳 🗔 🗺 | |] < #0 ~ [0. > |

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.

| ~ | Favorites | |
|-----|---|-------------------------------------|
| 2 | ╞ ┿ ∔ ╁ ╁ _₅ | 부 후 나 |
| | r i T • +s | • • |
| | | |
| | | |
| _ | | |
| * | Basic instructions | |
| Nar | ne | Description |
| - | GRAPH sequence | |
| | E Step and transition | Step and transition [Shift+F5] |
| | - Step | Step |
| | Transition | Transition |
| | Sequence end | Sequence end [Shift+F7] |
| | ₽ s Jump to step | Jump [Shift+F12] |
| | T Alternative branch | Open alternative branch [Shift+F8] |
| | Simultaneous branch | Open simultaneous branch [Shift+F8] |
| | CONTRACTOR OF A REPORT OF A | |

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.
- \rightarrow Open the action table using the " \square " button in the step field.
- \rightarrow Open the input window using the " \mathbb{D} " button at the transition.

| 05 | 2-100_GRAPH_\$7-150 | DO → CPL | _1516 | F [CPU 15 | 516F-3 | PN/DP] | ▶ Prog | gram | blocks | ► AUT | OMAT | IC_MO | DE [F | B50] | | | | | Ě | - - - × |
|-----|-------------------------|-----------|----------------------------------|-----------|--------|--------------|--------|------|--------------------|----------|-----------|---|-------|----------|----------|------|--------|---------|------|----------------|
| | | | | | | | | | | | | | | | | | | | | |
| | AUTOMATIC_MODE | ha ƙa ¥ | (¹ 1) ¹ 1 |) I., E | : 🚍 (| - 💬 : | 🗄 ± 🔏 | ± 법 | ± 😥 | ¢© 60 | \$ | o, œœ́r | | | | | | | | |
| | | Data type | Defau | Retain | Ac | Visi | Set | Sup | Comn | nent | | | | | 9 | | | | | ų. |
| - | Input Input Input Input | Bool | false | Non-ret | H | | | | | | | | | | | | | | | ^ |
| 2 | | | laise | Non-ret | | | | | | • | | | | | | | | | | ~ |
| Nav | rigation 🔍 🔍 | 🖻 🛓 | 中 | 1 + | +, | Ŧī | ₽ + | 8 | > = 1 | - 17 | -01 | -[w] | | CMP>T CM | P>U CONV | NEG | NOT SW | AP CMP> | CMP> | |
| > | Permanent pre-instru | ict | | ncer AUTO | | | • | | | | | | | | | | | 1.000 | | |
| ~ | 🕄 Sequences (1) | 1000 | : seque | | WANC_P | NODE | | | | | | | | | | | | | | |
| - | 🕄 1: sequencer AUTON | IATIC_ | | | | | | | | | | | | | | | | | | |
| | \$30 | | | | | | | | | | | | | | | | | | | |
| | Sitep30 | | | | | | | ¥ | 530 - 5 | tep30: | initial s | tep | | | | | | | × | - 1 |
| - | Transl | | | | | S | tep30 | | Inte | erlock E | vent | Qualif | | Actio | n | 1.00 | | | | |
| | - | | | | | | | | | | | <add n<="" td=""><td>ew></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></add> | ew> | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | 1 | | | | | | | | | | | | |
| | | | T1 - | Trans1: | х | | | is1 | | | | | | | | | | | | |
| | | ٠ | | | | | Į. | | | | | | | | | | | | | |
| | | - | | | | | | | | | | | | | | | | | | |
| | | 2 | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | |
| | | | _ | | | | | | | | | | | | | | | | | |
| | | | | | | | * | | | | | | | | | | | | | |

- \rightarrow 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.



For unrestricted use in educational and R&D institutions. © Siemens 2019. All rights reserved. sce-052-100-graph-s7-1500-r1903-en.docx

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

| Nan | ne | Data type | Defau | Retain | Ac | | Visi | Set | Sup | Comr | ment | | | | | | | - | | | | | |
|--------------|-----------------|-----------|--------------|-----------|--|---------|--------------|-----|-----|-------------|----------|---------|----------|--------------|-------|-------------|-----------|--|------|-----|------|---------------|----------------|
| | ACK_EF | Bool | false | Non-ret. | | | | | | | | | | | | | | | | | | | |
| - | Start | Bool | false | Non | - 🗹 | | | | | starti | ing con | nman | d | | | | | | | | | | |
| | -81 | Bool | false | Non-ret | 🗹 | | | | | sens | or cylin | ider -N | 14 retra | acted | | | | | | | | | |
| | -B2 | Bool | false | Non-ret. | 🗹 | | | | | sens | or cylin | der -N | 14 exte | nded | | | | | | | | | |
| - 🕞 | -B4 | Bool | false | Non-ret | | | | | | sens | or part | atslic | le | | | | | | | | | | |
| 10 7018 | 100 | | | | | | <u> </u> | | | • •• | • 1 | | | | | | | | | | | | 1 |
| vigatio | 100 - 10 | | 中 | + + | · ‡ | Ţ | . ģ | i ← | 8 | > = 1 | ?? | н | -01 | -[w] | | CMP>T | CMP>U | CONV | NEG | NOT | SWAP | CMP> T_MAX | CMP> T_WARN |
| Perma | nent pre-instru | ict | | | | | | | | | | | | | | | | | | | | 0 | 0 |
| 🖸 Sec | uences (1) | | omment | ncer AUT | JMAIIC | _MOD | - | | | | | | | | | | | | | | | | |
| 1: 5 | equencer AUTOR | NATIC | omment | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | |
| S30 Step3 | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | 53 | 0 | | 530 - | Step | 30: ir | nitial st | ep | | | | | |
| 0 | Transl | | | | | | | | | | 1000000 | 12121 | - P | | | | | | 12.1 | | 141 | | |
| | | | | | | | | | | | Step | 30 | 1.16 | Int | erloc | k Ev | ent | Ouali | fier | Ac | tion | | |
| * | | | <hr/> | | | | | | | | Step | 030 | | Int | erloc | k Ev | ent | <add i<="" td=""><td></td><td>Ac</td><td>tion</td><td></td><td></td></add> | | Ac | tion | | |
| * | | | \backslash | | | | | | | | Step | 030 | | Int | erloc | k Ev | ent | | | Ac | tion | | |
| ¥ | | | | | | | | | | | Step | 030 | | Int | erloc | k Ev | vent | | | Ac | tion | | |
| * | | | | | | | | | | | Step | 030 | | Int | erloc | k Ev | vent | | | Ac | tion | | |
| ** | | | | | | | | | | | | 11 | | Int | erloc | k Ev | rent | | | Ac | tion | | |
| ** | | | С Т | 1 - Trans | 1: sta | irt cor | dition | s | ×. | | | | 151 | Int | erloc | k Ev | rent | | | Ac | tion | | |
| | | | C) T | 1 - Trans | 1: sta | irt cor | | 5 | ×. | | | п | 15 1 | Int | erloc | k Ev | vent | | | Ac | tion | | |
| ** | | - | С Т Т | × | | irt cor | dition: & | s | ×. | (| | п | 151 | Int | erloc | k Ev | vent | | | Ac | tion | | |
| ** | | - | T (S) | À | dt 19 | _ | | 5 | ×. | | | п | ns1 | Int | erloc | <u>k Ev</u> | rent | | | Ac | tion | | |
| ** | | - | СЭ Т С | × | <mark>⊲? ?</mark> - ?.7 - | _ | | 5 | ×. | | | п | ns1 | Int | erloc | <u>k Ev</u> | vent | | | Ac | tion | | |
| ** | | | ЭT | X | <mark>থা 12</mark> বা.চ- বা.চ- | - | | 5 | ×. | | | п | ns 1 | Int | erloc | <u>k Ev</u> | rent | | | Ac | tion | | |
| ** | | - | ©T | X | 41.72 <11.75 <11.75 <11.75 | | | 5 | ×. | | | п | 151 | Int | erloc | <u>k Ev</u> | rent | | | Ac | tion | | |
| ** | | | ©T | À | ৰায়া ব্যায়চ ব্যায়চ ব্যায়চ | | | 5 | ×. | | | п | 151 | Int | erloc | <u>k Ev</u> | rent | | | Ac | tion | | |
| ** | | | ЭТ ЭТ | À | 41.72 <11.75 <11.75 <11.75 | | | 5 | ×. | | | п | 151 | Int | erloc | <u>k Ev</u> | rent | | | Ac | tion | | |

 \rightarrow Interconnect the AND logic operation according to the specification in the GRAFCET.



 \rightarrow 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.



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



 \rightarrow 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.
- \rightarrow 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 gualifier.
- \rightarrow Drag the output tag "-Q1" into the action field.



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

| Nam. Single step view | Data type | | Default | value | Retain | | Acces | sible f | Writ | ta | Visible i | n | Setpoir | t | Superv | is Co | mme | ent | | |
|--|------------|------------|------------|---------------------|--------|------|-------|---------|-------|-----|-----------|-----|---------|-------|--------|-------|--------|--------|-------|----------------|
| 📶 = Start | Bool | | false | | Non-re | tain | | ~ | 6 | 2 | | | | | | sti | arting | g comr | mand | |
| 5 🕣 = -B1 | Bool | | false | | Non-re | t 💌 | | - | | 2 | | | | | | se | nsor | cylind | er-M4 | retracted |
| -B2 | Bool | | false | | Non-re | tain | | ✓ | | | | | | | | se | nsor | cylind | er-M4 | extended |
| Vavigation 🔍 🔍 🔶 | + + | | ⊦ ‡, ₪ | Ŧ. | ب 4 | 8 | >=1 | 177 | 4 | -01 | 1 | ⊣"⊢ | CMP>T | CMP>U | CONV | NEG | NOT | SWAP | | CMP> T_WARN |
| Sequences (1) | | | | | | | | | | | | | | | | | | | | |
| ▼ 1: sequencer AUTOMATIC_M(∧ | | tep30 | | | 100 | | | | | | | | | | | | _ | | | |
| ■ T6 ■ T8 | ₩ | 531 | 531 | - Step3 | | | | | | | | | | | | × | - | | | |
| - T6 | ₩ | T1 Trar | 531 | - Step3 Interloc | | | | | Actio | 'n | | | | | | × - | • | | | |

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 ty | /pe | | De | fault v | alue | Re | etain | | Acces | sible f | Writ | ta | Visible | in | Setpoint | Supervis | Comment |
|--|------------|---------------------------|--------|-------------|---------|--------------|--------|----------|--------|--------|---------|-------------|------|---------------|---------------|----------|----------|------------------------------|
| 🧉 📹 🔹 Start | Bool | | | fal | se | | N | on-reta | ain | | ~ | | 2 | |) | | | starting command |
| -B1 | Bool | | 1 | al fal | se | | N | on-ret. | | | < | 6 | 2 | | | | | sensor cylinder -M4 retracte |
| 📶 🕷 -B2 | Bool | | 100 | fal | se | | N | on-reta | ain | | 2 | | 2 | |) | | | sensor cylinder -M4 extende |
| avigation 🔍 🔍 🖶 Permanent pre-instructio | | >=1 [| ?? | - | -01 | -[w] | -"F | CMP>T | CMP>U | CONV | NEG | NOT | 12 | CMP> T_MAX | CMP: T_WAP | > N | | |
| Sequences (1) 1: sequencer AUTOMATIC_M(^ | 0.777.0000 | Step31 ment Iterloc | k -(c) | ŀ- : | | | | | | | | | | | | | | |
| + T8 530 Trins1 Trins1 T2 T7an2 T7an2 T7an2 | ► SI | upervi | | -61" - | _ | Interlo C | ck | | | | | | | | | | | |
| *** 532 *** 536 Step32 Step36 + T3 Trans3 + T8 533 530 | ar ar | ctions | | | | | | | | | | | | | | | | |
| Step3 3 | -(c)- | In | terlo | ck E | vent | Qu | alifie | r | | | | Act | tion | | | | | |
| + Trans4 ** 534 512693 + | | | C)- | | | N | | et as lo | ing as | step i | active | # "- | Q1" | | | | | |
→ Drag tag "-B5" for Transition 2 and tag "-B7" for Transition 7as the step-enabling condition for the step below.

| 5 | | N KA YA | 🦸 🔮 🔩 🖿 🚍 🚍 | 💬 📲 ± 📲 ± 🔛 | 🕐 🚱 🥵 🐨 | | |
|--------|--|--|--|--------------|--------------------------------------|----------------------|--------------|
| | AUTOMATIC_MO | | | | | | |
| | Name | Data type | Defau Retain Ac | Visi Set Sup | Comment | | |
| 9 | -B5 | Bool | false Non-ret 💽 | | sensor metal part | | |
| 10 | -B6 | Bool | false Non-ret 💽 | | sensor part in front of cylinder -M4 | | |
| 1 | -B7 | Bool | 📑 false 🛛 Non 💌 星 | | sensor part at end of conveyor | | |
| 17 | Cutout | | | | | | |
| Na | | • • • | & >=1 [??] -I -ol | | CONV NEG NOT SWAP CMP> CMP> | | |
| > | Permanent pre-in | structions | | Enterine . | Const Const | | |
| ~ | Sequences (1) |) | | | | | |
| < | 326931 Trans2 526932 Trans3 526932 Trans3 526933 Trans4 526934 Trans5 526935 Trans5 526935 Trans5 526936 Trans5 526935 Trans5 526935 Trans6 Will | T7 Trans7 S36 Step36 T6 Trans8 W | ▼ T2: Trans2 Comment #**B5* ▼ T7: Trans7 Comment #**B7* | | | Trans2 Step32 S32 | 17 Trens7 |
| < > | Permanent post-i | | | | | | Step36 \$36 |
| 6 C | 1 | | | | | | |

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



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

| Step31 | Interlock Event | Qualifier | Action | |
|--------|-----------------|-----------|--------|--|
| | -(C)- | N | #"-Q1" | |
| | Define tag | Ctrl+S | hift+1 | |
| | Rename tag | Ctrl+S | hift+T | |
| | Rewire tag | Ctrl+S | hift+P | |
| 2 | X Cut | c | Ctrl+X | |
| | П Сору | | Strl+C | |

\rightarrow Paste the copied row in Step 32 and Step 36.



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

| \$36 🔠 | S36 - Step36 | : convey | or forwards | | Χ. |
|--------|--------------|----------|--------------------|--------|----|
| Step36 | Interlock | Event | Qualifier | Action | |
| | -(C)- | | N | #"-Q1" | |
| | | | <add new=""></add> | | |
| | | | | | |
| | | | | | |

→ 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.



- \rightarrow Drag tag "-B6" onto Transition 3 as the step-enabling condition.
- \rightarrow 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 ¹ 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.

| 533 | | | | | |
|-----------------------|---------------|-----------|--------------------|----------------|----|
| 14 Trans4 534 Ⅲ | \$34 - Step34 | : cylinde | er -M4 extend | | ×. |
| Step34 | Interlock | Event | Qualifier | Action | |
| | -(C)- | | N | # "−M3" | |
| | | | <add new=""></add> | | |
| | | | | | |
| | | | | | |

- \rightarrow 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.

| B | 347 24 | 8 | | X 🗐 | 2) B., | EE | | 3 8 ±. | 🛾 ± 🖞 | i ± 🗊 | 60 | 6 0 - | \$ | 0 | ¢ ∧ | | | | |
|-----------|--|---------------|--------------|-------|---|--------------------------------|-----------|--|-----------|----------|---------|--------------|-----------|-------|--------|------|--------|----|--|
| | and the second second | ATIC_MODE | - | | | | | 1 | | 11 | | | | | | | | 10 | |
| 1 | Name | | Data type | | Defau | | Ac | | . Set | Sup | | | | | | | | | |
| 6 | | -B1 | Bool | | false | Non-ret | 1111 Care | | | | sens | or cyl | inder - | -M4 r | etra | cted | P. | | |
| 7 | | -B2 | Bool | | false | Non | - 🗹 | | | | sens | or cyl | inder | -M4 e | exter | nded | ł | | |
| 0 | - | D.A. | Roal | | falco | Non rot | | | | | | - | ** * * * | lida | | | | l. | |
| | vigation | • | Θ 🖶 | 8 | >=1 | ?? ` – | -01 | -(w) _== | CMP>T | CMP>U | CONV | NEG | NOT | SWA | | | CMP> | | |
| > | Perman | ent pre-instr | uctions | | 222 0 | | 10-2-8 | 1-1 H | | | | | | | · 1 | MAX | T_WARN | | |
| | | | lictionism | \$34: | Step34 | | | | | | | | | | | | | | |
| - 10 - 10 | the second s | ences (1) | | Com | nment | | | | | | | | | | | | | | |
| - | 1: se | quencer AUTO | MATIC_MOD | - | | | | | | | | | | | | | | | |
| | | _ | | • h | nterloc | k -(c)-: | | | | | | | | | | | | | |
| | S30 Step3.0 | | | | | | | | | | | | | | | | | | |
| | -I | 1 rans1 | | | | | 1 | nterlock | | | | | | | | | | | |
| | 25 | | | | | | | С | | | | | | | | | | | |
| | Step31 | | | | | #"-B2 | -0 | | | | | | | | | | | | |
| | - | | i | | | | | | | | | | | | | | | | |
| | T | an s2 - | T7 Trans7 | | | | | | | | | | | | | | | | |
| - | | | 36 | | | | | | | | | | | | | | | | |
| | Step3 2 | Stej | p36 | | | | | | | | | | | | | | | | |
| | | 3 rans3 - | T8 Trans8 | • • s | upervi | sion -(v) | er um | | | | | | | | | | | | |
| | \$33 Step3 3 | | * | - A | Actions | : cylinde | r-M4 ex | tend | | | | | | | | | | | |
| | 1 14 | 4 | 1 | | retroms |) | | | | | | | | | | | | | |
| | T | rans+ | | | | | | | | | | | | | | | | | |
| | 534 Step34 | | | -(c)- | 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | Contract and Contract Contract | Event | Qualif | | | | | | tion | | | | | |
| | 11 | 5 rans5 | | | -(| C)- | | N -: | Set as lo | ong as s | step is | active | e #"- | -M3" | | | | | |
| | \$35 | | | | | | | <add n<="" td=""><td>:w></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></add> | :w> | | | | | | | | | | |
| | Step3 5 | | | | | | | | | | | | | | | | | | |
| | +7 | 6 rans6 | | | | | | | | | | | | | | | | | |
| | * | | | | | | | | | | | | | | | | | | |

- → 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.
- \rightarrow Select and copy Step 34 and the associated Transition 5.



 \rightarrow Select Transition 5 and insert the copied step with transition.

| | _ | Insert element | • |
|-----|--------|----------------|--------------|
| H5 | т5 | Define tag | Ctrl+Shift+I |
| 1.2 | Trans5 | Rename tag | Ctrl+Shift+T |
| | 537 🏢 | Rewire tag | Ctrl+Shift+P |
| | Step37 | X Cut | Ctrl+X |
| H5 | 19 | Сору | Ctrl+C |
| 1.2 | Trans9 | Paste | Ctrl+V |

- \rightarrow Change the step number and the tag designations of the inserted step.
- \rightarrow Change the transition number and the tag designations of the inserted transition.

| Step34 | A THE OWNER AND A CONTRACTOR | : cylinder -M4 extend Event Qualifier | Action | |
|----------|------------------------------|--|--------|--|
| | -(C)- | N | #"-M3" | |
| | | <add new=""></add> | | |
| | ns 5 | | | |
| | NUMBER OF STREET | 4_1: cylinder -M4 exte | | |
| Step34_1 | | Event Qualifier | Action | |
| - n | -(C)- | N | #"-M3" | |
| | | <add new=""></add> | | |
| тъ | ns5_1 | | | |
| L | II) | | | |
| Step35 | | | | |
| Stepss | | | | |
| | | | | |
| | Dr.G | | | |

- \rightarrow 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 ¹ at the second input and enter T#500MS for the time.

 \rightarrow Designate the transitions as shown here.

| ÷ | 1 | ŧ | ‡ ₅ | Ŧ | ₽ | ┙ | 8 | >=1 | ·?? | - | -01 | -[w] | ⊣∓⊢ | CMP>T | CMP>U | CONV | NEG | NOT | SWAP | | CMP> T_WARN |
|------|-------|--------------|-----------------|---------|-------------|--------|---------|--------|--------|---|-----|--------------|-----|-------|-------|------|--------|-----------------------|-------------------------------|--------|-------------------------------|
| T5 - | Tran | s5: c #"- | | r exter | nded | | | | | | | | | ► C | | than | step a | | <mark>on tim</mark> Irans5 | e | |
| | | | | | | | | | | | | | | | | - | St | 5341 tep34_ | .1 | ·· \$3 | 41 - Ste Interloo -(C)- |
| T51 | - Tra | ns5_1 | 1: cyl | | extend > | ed and | l waiti | ng tim | e 0.5s | | | | | | × | : | | | 151 Trans 5 _. | _1 | |
| | | | _1.T — IMS — | IN1 | ime | - | | | #"-B2 | | 8 | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | \$35 | 113 | | |

- → 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.
- \rightarrow 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.

Learn-/Training Document | TIA Portal Module 052-100, Edition 10/2019 | Digital Industries, FA

| R | 8 | e o e | EX 🖏 K | X 学 | * | L E | | 98 | ± \$ | 1± 12 | i ± 🗱 | • ¢© | ¢ 6⊖ : | ₽. | 0. | 00 | | | | |
|---|------------------|-----------------------------|--------------|-------|----------|--------------------|--------------|---|---------------|----------------|--------|---------|-----------|-------|------|------|------|--------------------|--|------|
| | AUTOM | ATIC_MODE | | | | | | | | | | | | | | | | | | |
| | Name | 8 | Data type | | Defau | Retain | Ac | | Visi | Set | Sup | Com | nment | | | | | | | |
| 6 | | -B1 | Bool | | false | Non | - | | | | | sen | is or cyl | inder | -M4 | retr | acte | ed | | |
| 7 | - | -B2 | Bool | | false | Non-re | et 🗹 | | | | | sen | is or cyl | inder | -M4 | exte | ende | ed | | |
| 2 | | P.4 | Roal | | falsa | Mane | + 🗖 | | | 0 | | | corp2 | - | lido | - | | | | |
| | vigation | | 0, 🖶 | 8 | >=1 | ·[?] - | -01 | -[w] | | CMP>T | CMP>U | CONV | NEG | NOT | sv | VAP | | > CMP> X T_WARN | | |
| | Perman | ent pre-instru | uctions | COF. | C+++-7 | - | | | | | | | | | | | | 1- | | |
| | 🕄 Sequ | ences (1) | | \$35: | oment | | | | | | | | | | | | | | | |
| - | 1: se | quencer AUTO | | Con | nment | | | | | | | | | | | | | | | |
| | | | | • | nterlo | ck -(c)-: | | | | | | | | | | | | | | |
| • | Step3 1 | 2 rans2 ** Sa Step | T8 Trans8 | • 5 | Super | #"-E vision -(' | n " 0 | Interlo C | ck | | | | | | | | | | | |
| | 533 Step33 | 4 | | - / | Action | s: cylind | der -M4 re | tract | | | | | | | | | | | | |
| 4 | 337 | ans4 | - | -(0)- | | nterlock | Event | Qu | alifie | er | | | | Ac | tio | n | | | | |
| 4 | 5341 Step3+_1 | 5 an:5 | | | | -(C)- | | N <ac< th=""><th>-Se Id nev</th><th>et as lo w></th><th>ing as</th><th>step is</th><th>s active</th><th>e #"-</th><th>-M2</th><th></th><th></th><th></th><th></th><th></th></ac<> | -Se Id nev | et as lo w> | ing as | step is | s active | e #"- | -M2 | | | | | |
| | Step3 5 | 6 rans6 | | | | | | | | | | | | | | | | | | |

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



- \rightarrow At the end of the sequence, we set a jump to Step 30.
- \rightarrow 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.)

| | 536 | | 536 - Step36 | : convey | yor forwards | | 5 |
|---|--------|------|--------------|----------|-----------------------------|-------------|---|
| S | Step36 | | Interlock | Event | Qualifier | Action | |
| | | | -(C)- | | N <add new=""></add> | #"-Q1" | |
| | | | | | | | |
| | 5361 | ans8 | | | nveyor forward | | 9 |
| | Tra | ans8 | | | nveyor forward Qualifier | s Action | 5 |
| | 5361 | ans8 | | | | | - |

- \rightarrow 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 ¹ at the second input and enter **T#2S** for the time.
- \rightarrow Designate the transitions.

| 中 | 4 | ŧ | \$s | £ | ₽ | ┙ | 8 | >=1 | ·?? | Ä | -01 | - [w] | -"⊢ | CMP>T | CMP>U | CONV | NEG | NOT | SWAP | CMP> T_MAX | CMP > T_WARN | | |
|------|-------|----------------|-----|---------|--------|---------|-------|-----|-----|---|-----|--------------|-----|-------|-------|------|------------|------|-----------|---------------|-----------------------------------|------------------------------|-------------|
| T8 - | Trans | 8: se #*-B7 | | t end o | ofconv | veyor p | assed | l | | | | | | | | | T8 Tra | ins8 | | | | | |
| | | | | | | | | | | | | | | | 402 | | 61 36_1 | | azazatero | | and the state of the state of the | onveyor forward Qualifier | s Action |
| | | | | | | | | | | | | | | | | | | | | (C)- | | N <add new=""></add> | #"-Q1" |
| | | | | | | | | | | | | | | | | | | | | | | | |

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



 \rightarrow The S7-GRAPH sequencer for sequencer AUTOMATIC_MODE is now finished.



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".

 $(\rightarrow \text{Network } 1:... \rightarrow \text{Call AUTOMATIC}_MODE \text{ for test})$

| | | Na | me | Data type | Default value | Comment |
|-----|-----|-----|------------------------|---------------|----------------|---------------------------------------|
| Ř | -00 | - | Input | | | |
| 2 | -00 | | Initial_Call | Bool | | Initial call of this OB |
| 3 | - | | Remanence | Bool | | =True, if remanent data are available |
| 100 | -00 | - | Temp | | | |
| 5 | | | <add new=""></add> | | | |
| 5 | -00 | - | Constant | | | |
| | | | <add new=""></add> | | | |
| | - | 1 | | | harmed harward | |
| • | Blo | ock | title: "Main Program S | weep (Cycle)" | | |

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

| K Siemens - C:\Users\mde\Documents\Automation\05 | 2-100_GF | RAPH_\$7-1500\052-100_GRA | PH_\$7-1500 | | |
|--|----------|---|----------------------|-----------------|-------------------------------------|
| Project Edit View Insert Online Options Tools | | | 🕈 Go offline 🛛 🛔 👔 | | Totally Integ |
| Project tree 🛛 🔲 🖣 | 052-10 | 0_GRAPH_\$7-1500 → CPU | _1516F [CPU 1510 | 5F-3 PN/DP] 🕨 P | rogram blocks 🕨 Main [OB1] |
| Devices | | | | | |
| | 14 18 | # # L = A = C |] 2 : 2 : 2 : | = 😥 🍋 🖕 d | ≣ %≣ 4≱ ⊊ I ₌ %_ 61 ¢1 |
| | Mair | | Ja- a- a- 6 | | |
| ▼ 1 052-100 GRAPH S7-1500 | 10000000 | Jame | Data type | Default value | Comment |
| Add new device | 1 📶 🕇 | Input | 31 | | |
| Devices & networks | 2 📶 🗉 | and the second se | Bool | | Initial call of this OB |
| T CPU_1516F [CPU 1516F-3 PN/DP] | 3 📶 🖷 | Remanence | Bool | | =True, if remanent data are availal |
| Device configuration | 4 📶 🗖 | Temp | | | |
| 🖳 Online & diagnostics | 5 . | Add new> | | | |
| 🔻 🛃 Program blocks | 6 🕣 🗖 | Constant | | | |
| Add new block | 7 | <add new=""></add> | | | |
| - Main [OB1] | | | | | |
| - AUTOMATIC_MODE [FB50] | - Block | k title: "Main Program Sweep | (Cycle)" | | |
| 🕨 📴 Technology objects | Comm | ent | | | |
| Energy objects | | | | | |
| External source files | ▼ Ne | etwork 1: Call AUTOMATIC_M | DDE for test | | |
| 🕨 📜 PLC tags | Co | mment | | | |
| LC data types | _ | | | | |
| Watch and force tables | | 1.23 | UTOLUTIC MODE CO | 501 | |
| Online backups | | -10° | AUTOMATIC_MODE [FB | 00] | |

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

| Call options | × |
|--------------|---|
| Single | Data block Name AUTOMATIC_MODE_DB Number Manual Automatic If you call the function block as a single instance, the function block saves its data in its own instance data block. |
| | more |
| | OK Cancel |

→ 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)

| | Sav | | | | | | | | | | | | |
|--------------------|---|--|---|------|---|--------|-------|---|---|-----------------------------|--|-----------------|----------|
| Proj | ject tre | e | | | • • | 52-100 | _GRAI | PH_S/ | -1500 | | CPU_1516F [C | PU 1516 | F-3 PN/L |
| D | evices | | | | | | | | | | | | |
| 1 | | | | | | 1 .X | st st | | = | | 🗩 🗶 🖉 | - <u>19</u> + [| |
| | | | | | | | | | | _ | | | Blocki |
| | • 🔚 | External sour | rce files | | ~ | - | | | 1 | - | | | 1 + 1 |
| | | PLC tags | | | CONTROL OF THE OWNER | & >= | 1 ?? | 4 | -01 | \rightarrow | -[=] | | |
| | | how all t | aas | | | | | | | | | | |
| | | 😽 Add new t | 11 T 10 T | | = | Ma | tuork | 1. 0 | IL ALLTC | MATIC | MODE for test | | |
| | | 📕 Default ta | - | | | 8.77 | | r: u | III AUIC | INAIIC | _MODE IOI lest | | |
| | | | _sorting station | [28] | | Con | nment | | | | | | |
| | • C a | PLC data type | 25 | | ~ | | | | | | | | |
| | and the second se | | | | 100205 | | | | | | | | |
| VI | Details | view | | | | | | | | | %DB1 | | |
| ▼ | Details | view | | | | | | | | | %DB1 "AUTOMATIC_ MODE_DB" | | |
| | Details | view | | | | | | | | | *AUTOMATIC_ MODE_DB* %FB50 | | |
| | | | Details | | | | | | | "Al | *AUTOMATIC_ MODE_DB* | E" | |
| | Details Name -P3 | Data type Bool | | | ~ | | | | — EN | | *AUTOMATIC_ MODE_DB* %FB50 | E" | |
| | lame | Data type | Details | | | | | false | - OF | F_SQ | *AUTOMATIC_ MODE_DB* %FB50 | E. | |
| N | Name -P3 | Data type Bool | Details %Q0.7 | | | | | false false | | F_SQ T_SQ | *AUTOMATIC_ MODE_DB* %FB50 | E. | |
| | lame -P3 -P4 | Data type Bool Bool | Details %Q0.7 %Q1.0 | | | | | false false | - OF | F_SQ T_SQ | *AUTOMATIC_ MODE_DB* %FB50 | E" | |
| | lame -P3 -P4 -P5 | Data type Bool Bool Bool | Details %Q0.7 %Q1.0 %Q1.1 | | | | | false false false %10.2 | | F_SQ T_SQ K_EF | *AUTOMATIC_ MODE_DB* %FB50 | E" | |
| | Name -P3 -P4 -P5 -P6 | Data type Bool Bool Bool Bool Bool | Details %Q0.7 %Q1.0 %Q1.1 %Q1.2 | | | | | false false false %10.2 "-S0" | | F_SQ T_SQ K_EF | *AUTOMATIC_ MODE_DB* %FB50 | E" | |
| | Name -P3 -P4 -P5 -P6 -P7 | Data type Bool Bool Bool Bool Bool Bool | Details %Q0.7 %Q1.0 %Q1.1 %Q1.2 %Q1.3 | | | | | false false false %40.2 "-SO" false | — OFI — INIT — ACH — Sta — -B1 | F_SQ T_SQ K_EF | *AUTOMATIC_ MODE_DB* %FB50 | E. | |
| | lame -P3 -P4 -P5 -P6 -P7 -Q1 | Data type Bool Bool Bool Bool Bool Bool Bool | Details %Q0.7 %Q1.0 %Q1.1 %Q1.2 %Q1.3 %Q0.0 | | | | | false false false %10.2 *S0 false false | — OFI — INIT — ACH — Sta — -B1 — -B2 | F_SQ T_SQ (_EF | *AUTOMATIC_ MODE_DB* %FB50 | | 54 |
| | lame -P3 -P4 -P5 -P6 -P7 -Q1 -Q2 | Data type Bool Bool Bool Bool Bool Bool Bool Boo | Details %Q0.7 %Q1.0 %Q1.1 %Q1.2 %Q1.3 %Q0.0 %Q0.1 | | | | | false false false %40.2 "-SO" false false | - OFI - INIT - ACI - Sta B1 B2 B4 | F_SQ T_SQ (_EF .rt | *AUTOMATIC_ MODE_DB* %FB50 | Q1 | false |
| | Name -P3 -P4 -P5 -P6 -P7 -Q1 -Q2 -Q3 | Data type Bool Bool Bool Bool Bool Bool Bool Boo | Details %Q0.7 %Q1.0 %Q1.1 %Q1.2 %Q1.3 %Q0.0 %Q0.1 %Q0.2 | | | | _ | false false false %40.2 *-SO* false false false | - OFI - INIT - ACI - Sta B1 B2 B4 B5 | F_SQ T_SQ <_EF | *AUTOMATIC_ MODE_DB* %FB50 | -Q1 -M2 | — false |
| | Vame -P3 -P4 -P5 -P6 -P7 -Q1 -Q2 -Q3 -Q3 -S0 | Data type Bool Bool Bool Bool Bool Bool Bool Boo | Details %Q0.7 %Q1.0 %Q1.1 %Q1.2 %Q1.3 %Q0.0 %Q0.1 %Q0.2 | | | | | false false false %40.2 *-SO false false false false | - OFI - INIT - ACI - Sta B1 B2 B4 | F_SQ T_SQ (_EF | *AUTOMATIC_ MODE_DB* %FB50 | -Q1 -M2 | |

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

| "-S0" — | | | |
|--------------------------------------|------|-------|--|
| -B | 81 | | |
| -m *-B1* | Bool | %10.5 | |
| - B2" | Bool | %10.6 | |
| 📶 [*] -B3* | Bool | %10.7 | |
| - 1 "-B4" | Bool | %11.0 | |
| ************************************ | Bool | %11.1 | |
| 🕣 "-B6" | Bool | %11.2 | |
| -m *-B7* | Bool | %11.3 | |

- \rightarrow 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.

| | %DB1 "AUTOMATIC_ MODE_DB" | | |
|---------|---------------------------------|----|---------|
| | %FB50 | | 1 |
| | "AUTOMATIC_MODE" | | |
| | EN | | |
| false — | OFF_SQ | | |
| false — | | | |
| false — | | | |
| %40.2 | | | |
| "-so" — | Start | | |
| %40.5 | | | |
| "-B1" — | -81 | | |
| %10.6 | | | |
| "-B2" | -82 | | |
| %1.0 | | | |
| "-B4" — | -84 | | %Q0.0 |
| %1.1 | - | Q1 | |
| "-B5" — | -85 | | %Q0.3 |
| %11.2 | | M2 | |
| "-B6" — | -86 | | %Q0.4 |
| %1.3 | - | ΜЗ | - "-M3" |
| *-B7* — | -87 EN | NO | 27 |
| | | | |

7.9 Result in the LAD programming language



7.10 Saving and compiling the program

→ To save your project, select the " Gave project and button in the menu. To compile all blocks, click the

"Program blocks" folder and select the button " \square " for compiling in the menu. (\rightarrow \square Save project \rightarrow Program blocks \rightarrow \square)

| HA | Siemens - C:\Users\mde\Documents\Automation\ | 052-100 |
|------------|--|---------|
| | oject Edit View Insert Online Options Tool 🛉 🎦 🗔 Save project 昌 🐰 🏥 间 🗙 🏷 ± 🧭 | |
| | Project tree | |
| | Devices | |
| | | |
| nime | ▼ 052-100_GRAPH_S7-1500 | ^ |
| C programm | Add new device | |
| L Pr | CPU_1516F [CPU 1516F-3 PN/DP] | |
| F | Device configuration | |
| | ♥ Online & diagnostics ▼ | |
| | Add new block | |
| | - Main [OB1] | = |
| | AUTOMATIC_MODE [FB50] | |
| | AUTOMATIC_MODE_DB [DB1] | |

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

| | <u>s</u> | roperties | i Info | o 追 🗄 | Diagno | ostics 📑 🗉 |
|--|---|-----------|--------|-------|----------|------------|
| General 👔 Cross-references | Compile Energy Suite Synt | ax | | | | |
| 3 👍 🚯 Show all messages | | | | | | |
| Compiling finished (errors: 0; warnings: | 2) | | | | | |
| Path | Description | Go to | ? Err | ors V | Varnings | Time |
| CPU_1516F | | ~ | 0 | 2 | 2 | 2:59:08 PM |
| Program blocks | | 7 | 0 | 2 | 2 | 2:59:08 PM |
| ✓ AUTOMATIC_MODE (FB | | ~ | 0 | 2 | 2 | 2:59:08 PM |
| Sequence 1 | Step Step30 does not contain actions. | ~ | ? | | | 2:59:08 PM |
| Sequence 1 | Step Step33 does not contain actions. | 7 | ? | | | 2:59:08 PM |
| | Block was successfully compiled. | | | | | 2:59:13 PM |
| AUTOMATIC_MODE_DB | Block was successfully compiled. | ~ | | | | 2:59:13 PM |
| Main (OB1) | Block was successfully compiled. | ~ | | | | 2:59:13 PM |
| | Compiling finished (errors: 0; warnings: 2) | | | | | 2:59:14 PM |

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 (\rightarrow .

| VIDA | Siemens - C:\Users\mde\Documents\Automation\052-100 | | _ ¤ X |
|-------|--|---|--|
| 1000 | iject Edit View Insert Online Options Tools Wind C 🔁 🔒 Save project 블 💥 🗐 📬 🗙 🏷 🛨 🦓 🛨 🚮 | w Help 👖 🖉 Go online 🖉 Go offline 🏭 🖪 🖪 🗶 🚽 🛄 🏼 Search in projects 🖓 | Fotally Integrated Automation PORTAL |
| | Project tree 🔲 🖡 | 052-100 GRAPH_S7-1500 	CPU_1516F [CPU 1516F-3 PN/DP] 	Program blocks 	Main [(| DB1] 🗕 🖬 🖬 🗙 📢 |
| | Devices | Download to device | - 1 |
| | 11 III III III III III III III III III | ,ã, ã ở ở 🐁 🖹 🗖 🚍 🗩 📲 🖢 📲 થ ± 🗐 й 🕫 🚱 🧐 🧐 🖓 💶 🗄 🦌 🤇 | |
| ning | 052-100_GRAPH_\$7-1500 | Block interface | <u>ā</u> , |
| i i i | Add new device | & >=1 ??? → -ol → -[=] | ns |
| gra | A Devices & networks | | |
| h | CPU_1516F [CPU 1516F-3 PN/DP] | Network 1: Call AUTOMATIC_MODE for test | |
| FC | Device configuration | Comment | Testing |
| | 😡 Online & diagnostics | Comment | ti |
| | ▼ 🙀 Program blocks | | Di la construcción de la constru |
| | Add new block | %DB1 "AUTOMATIC | |
| | - Main [OB1] | MODE_DB* | Tasks |
| | AUTOMATIC_MODE [FB50] | %FB50 | as |
| | System blocks | "AUTOMATIC_MODE" | N I |
| | Technology objects | En En | |
| | Energy objects | false — OFF_SQ | ∃ Libraries ≡ |
| | External source files | false — INIT_SQ | = bra |
| | PLC tags | false — ACK_EF | rie |
| | PLC data types | %40.2 "-50" — Start | 0 |
| | Watch and force tables | | |
| | Online backups | %0.5 "-81" | |
| | 🕨 🚰 Traces | 940.6 | |
| | Device proxy data | "-82" -0 -82 | |
| | Program info | 541.0 | |
| | ✓ Details view | "-B4" | |
| | | 9km.n Q1 Q1 Q1 | |
| | | "-85" — -85 %Q0.3 | |
| | Name Address | %11.2 -M2 *-M2* | |
| | | "-86" — -86 %Q0.4 | |
| | | %1.3 -M3 "-M3" | |
| | | "-87" — -87 ENO — | |
| | | | ~ |
| | | 100% | |
| | | Properties 11 Info | i) Diagnostics |
| | 🖣 Portal view 🔚 Overview 📲 Default tag | | nming language of the sele |

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).





 \rightarrow The initial step (Step 30) is already active.

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

| 052-100_GRAPH_S7-1500 → (8 8 8 8 8 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 | | | | | | | | 9± | 2ª± | 않 ± 🖻 | 60 | (a 4 | ž 🔗 | 00- | | | | | | | |
|--|--------|--------|--------------|-----|-----|------------|---------------------------|--------|-----------------|-------------------------|---------------|--------------|------------------|------|-----|-------|-------|------|-----|-----|------|
| | ICA | - | | | | | 0 | | | lock int | | | | | | | | | | | |
| 9 | 1 | | | | | | | | 1 | | | | | | | | | | | | |
| Navigation 🔍 🍳 🖶 | Instar | nce DB | : AUT | OMA | | IODE | E_DB [| [DB1] | | | | | | | | | | | | | |
| > Permanent pre-instructio | 卓 | 中 | | H | L . | ŧ, | Ŧ | 쿠 | 4 | 8 | = 1 77 | P - | 01 | -[w] | | CMP>T | CMP>U | CONV | NEG | NOT | SWAP |
| Sequences (1) | T | * | 100 | 1 | | • s | * | • | | | 18 | <u>.</u> | | - | 3.5 | | | | | | |
| ▼ 1: sequencer AUTOMATIC_M | | | | | | | | | | | | | | | | | | | | | |
| 530 Step30 | | | | | | | | | | | | | | | | | | | | | |
| Step30 Trans1 see Step31 Trans2 Trans2 |] | | 5.81 | | S | tep | activa | tion t | ime (T) | | | | 5_282M 5_282M | | | | | | | | ~ |
| Step30 | - | | 531 tep31 | Ĩ | S | tep 531 | activa - Ste | tion t | ime (T) conv | eyor for | ards | T#569 | 5_282M | | | | | | | | × _ |
| Step30 Trans1 Trans2 Trans2 | HCEPH | | 531 tep31 | | S | tep 531 | activa - Ste Interl | tion t | ime (T) conv | | ards | T# 563 | | | | #"-0 | 1" | | | TRU | |
| Step30 Trans1 Trans2 Trans2 Trans3 Trans3 Trans3 Trans3 Trans3 Trans3 Trans3 Trans3 Trans3 | Here | | | | S | tep 531 | activa - Ste | tion t | ime (T) conv | eyor forv t Qua N | ards | T# 563 | tion | | | #"-0 | 1" | | | TRU | |
| Step30 Trans1 Here Trans1 Here Trans2 Trans2 Trans2 Step32 Step32 Trans3 Trans3 Trans3 Step33 Step35.1 | | | | | S | tep 531 | activa - Ste Interl | tion t | ime (T) conv | eyor forv t Qua N | ards ifier | T# 563 | tion | | | #"-0 | 1" | | | TRU | |
| Step30 •••• •••• •••• •••• •••• •••• •••• •••• •••• •••• •••• •••• ••••• ••••• •••••• ••••••••••••••••••••••••• | - | | | | S | tep 531 | activa - Ste Interl | tion t | ime (T) conv | eyor forv t Qua N | ards ifier | T# 563 | tion | | | #"-0 | 1" | | | TRU | |
| Step30 Trans1 Trans2 Trans2 Trans2 Trans2 Trans3 Trans3 Trans3 Trans3 Step36 Step33 Step34 Step34 | | | | | S | tep 531 | activa - Ste Interl | tion t | ime (T) conv | eyor forv t Qua N | ards ifier | T# 563 | tion | | | #"-0 | 1" | | | TRU | |
| Step30 Trans1 HT2 Trans2 Trans2 Trans2 Trans2 Step32 Step32 Step33 Step33 Step33 Step33 Step34 Step34 Step34 Step34 Step34 Trans5 | | 5 | tep31 | | S | tep 531 | activa - Ste Interl | tion t | ime (T) conv | eyor forv t Qua N | ards ifier | T# 563 | tion | | | #"-0 | 1" | | | TRU | |
| Step30 Trans1 Trans2 Trans2 Trans2 Trans3 Trans3 Trans3 Trans3 Trans3 Trans3 Trans4 Step34 Step34 Step33 Step34 Step34 Step34 Step34 Step34 Step34 Step34 Step34 Step34 | | | tep31 | | s | tep 531 | activa - Ste Interl | tion t | ime (T) conv | eyor forv t Qua N | ards ifier | T# 563 | tion | | | #"-0 | 1" | | | TRU | |
| Step30 Image: Trians1 Image: Trians1 Image: Trians2 Image: Trians2 Image: Trians3 Image: Trians4 Image: Trians5 Image: Trians5 Image: Trians5 | | S | tep31 | 12 | \$ | tep 531 | activa - Ste Interl | tion t | ime (T) conv | eyor forv t Qua N | ards ifier | T# 563 | tion | | | #"-0 | 1" | | | TRU | |

 \rightarrow 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.



 \rightarrow The cylinder is retracted again in Step 35.

| 1 - Trans5_1: cylinder extende X _ * Time * Step34_1.T - IN1 T# 500ms - IN2 | | Uninterrupted st Step activation t | | | T#465_994MS T#465_994MS | | |
|--|--------|---------------------------------------|-------|--------------------|----------------------------|--------|------|
| | | \$35 - Step35 | | | - | | × |
| | Step35 | Interlock | Event | Oualifier | Action | | |
| | 100 | -(C)- | | Ň | #"-M2" | #"-M2" | TRUE |
| | | | | <add new=""></add> | | | |
| FALSE | Trans6 | | | | | | |

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



 \rightarrow 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.
- \rightarrow 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.
- \rightarrow **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.
- \rightarrow 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
- \rightarrow To use this function, switch the sequencer to manual mode and enable the synchronization.
- → Select the "Preceding transition satisfied" option.



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

| | B: AUTO | | | | | | | | | | | | | | | | | | | | | 10 | > | Call environment |
|-------|------------|---------|------------|---------|--------|----|-------|-------|-------|---------|------|--------|-------|---------|---------|---------|--------|-------|------|------|----------------|------|---|--------------------------------|
| 中 | 4 | + | + | Ŧ | 쿽 | 4 | 8 | > = 1 | ?? | н | -01 | -[w] | | CMP>T | CMP>U | CONV | NEG | NOT | SWAP | CMP> | CMP> T_WARN | | > | Call hierarchy |
| 1. 35 | 17 | | ▼ S | • | | | | | 1 | | 1 | | | | | | | | | - | | ~ | ~ | Sequence control |
| TEA | Trans | - | P 1 | 25.7.2 | 1.1. | | | | | 151 | | | | | | | | | | | | ^ | N | Mode: |
| 151 - | Trans | 5_1: | cylinae | erexte | nded a | ·• | | | T | frans 5 | _1 | | | | | | | | | | | | 1 | Automatic |
| | | | | > | | | | | | | | | | | | | | | | | | | | Automatic |
| | | | | Time | | | | | | | | | | | | | | | | | | | | 🔘 Semiautomatic mode |
| T# 31 | M_235_ | | | | | | | | | | | | | | | | | | | | | | | Ignore transition |
| | 2012 0 000 | 34_1.T | 2263 | | | | | | | | | | | | | | | | | | | | | Manual mode |
| | 1# | 500ms | | 2 | | | | | | | Unir | terrur | ted s | ten act | ivation | time (| ш)- т# | 105 8 | 2MRP | | | | | |
| | | | | | | | | | | | Step | activ | ation | time (T |): | unite (| | 105_8 | | | | | | Step with transition |
| | | | | | | | -423- | | \$35 | | \$3 | 5 - St | ep35 | : cylin | nder -M | 4 retra | ct | | | | | | | Next |
| | | | | | | | | 5 | tep35 | | | | lock | Ever | | ualifie | | Actio | | | | | | Select step manually |
| | | | | | | | | | | | | -(C)- | | | N | dd nev | | #"-M2 | | | 1 | • | | |
| | | | | | | | | | | | | | | | 4 | ad nev | v> | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | Enable Disable |
| | | | | | | | | | 24 | 16 | here | | | | | | - | | | | | -11 | | System synchronization |
| 6 - 1 | rans6 | : cylin | der re | tracted | 1 | 2 | :; | | | frans 6 | | | | | | | | | | | | | | Enable synchronization |
| | | | - Ť | | | | - 1 | | | | | | | | | | | | | | | - | | Preceding transition satisfied |
| | - 1 | FALSE | | | | | - 1 | | | | | | | | | | | | | | | | | Interlock condition satisfied |
| | | #"-B1" | | | | | - 1 | | | | | | | | | | | | | | | - | | |
| | | | | | | | | | | | | | | | | | | | | | | 1976 | | Step to activate: |
| | | | | | | | | | - | | | | | | | | | | | | | | | Enable |

 \rightarrow Switch the sequencer back to automatic mode.



 \rightarrow The sequencer can now continue running in automatic mode.

7.15 Creating function block FB30 "SIGNAL_LAMPS"

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



- → 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: $30 \rightarrow \blacksquare$ 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.

| | | -11 | Name | Data type | Default value | Retain | Acce | Writa | Visible | Setpoi. | Sup | Comment |
|----|----|-----|----------------------------|-------------------|---------------|------------|------|-------|---------|----------|-----|-------------------------------|
| 1 | - | 0.0 | Input | JF | | | | | | | | |
| 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 f |
| 5 | - | | Output | | | | | | | | | |
| 6 | | | Add new> | | | | | | | | | |
| 7 | | 1 | InOut | | | | | 1 | | | | |
| 8 | | | Add new> | | | | | | | | | |
| 9 | | 1 | ▼ Static | | | | | | | | | |
| 10 | | | RT_DATA | G7_RTDataPlus_V6 | | Non-retain | | | | V | | Internal data area |
| 11 | | | Trans1 | G7_TransitionPlus | | Non-retain | | | | V | | Transition structure |
| 12 | - | | Step1 | G7_StepPlus_V6 | | Non-retain | | | | V | | Step structure |
| 13 | -0 | 1 | Temp | | | | | | | | | |

- \rightarrow 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" (\rightarrow Input: ACK_EF \rightarrow Add row).

| | | Name | Data type | | Default value | Retain | A |
|----|---|---------------------|------------|--------|---------------|------------|---|
| 1 | | 🔽 🔻 Input | | | | | |
| 2 | | OFF_SQ | Bool | | false | Non-retain | |
| 3 | | INIT_SQ | Bool | | false | Non-retain | |
| 4 | 5 | Insert row | Ctrl+Enter | | false | Non-ret | - |
| 5 | | Add row | Alt+Ins | | | | |
| 6 | 4 | ¥ Cut | Ctrl+X | - | | | |
| 8 | | 📃 Сору | Ctrl+C | | | | |
| 9 | - | Paste | Ctrl+V | | | | |
| 10 | * | 🗙 Delete | Del | lus_V6 | | Non-retain | |
| 11 | * | Rename | F2 | onPlus | | Non-retain | |
| 12 | 4 | Add new supervision | | s_V6 | | Non-retain | |

- → 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.

| | | Na | ime | Data type | Default value | Retain | Acce | Writa | Visible | Setpoi | Sup | Comment |
|----|-----|----|--------------------|-----------|---------------|------------|------|-------|---------|--------|-----|---------------------------------------|
| 1 | -00 | • | Input | | | | | | | | | |
| 2 | - | | OFF_SQ | Bool | false | Non-retain | | | | | | Turn sequence off |
| 3 | - | | INIT_SQ | Bool | false | Non-retain | | | | | | Set sequence to initial state |
| 4. | -00 | | ACK_EF | Bool | false | Non-retain | | | | | | Acknowledge all errors and faults |
| 5 | -00 | | -A1 | Bool | false | Non-retain | | | | | | notification EmergOFF |
| 6 | - | | -KO | Bool | false | Non-retain | | | | | | plant [*] on [*] |
| 78 | - | | -50 | Bool | false | Non-retain | | | | | | mode selector manual(0) / automatic(1 |
| в | - | | -B1 | Bool | false | Non-retain | | | | | | sensor cylinder -M4 retracted |
| 9 | -00 | | -B2 | Bool | false | Non-retain | | | | | | sensor cylinder -M4 extended |
| 10 | - | | <add new=""></add> | | | | | | | | | |
| 11 | - | • | Output | | | | | | | | | |
| 12 | -00 | | -P2 | Bool | false | Non-retain | | | | | | display "manual mode" |
| 13 | - | | -P3 | Bool | false | Non-retain | | | | | | display "automatic mode" |
| 14 | - | | -P4 | Bool | false | Non-retain | | | | | | display "emergency stop activated" |
| 5 | - | | -P6 | Bool | false | Non-retain | | | | | | display cylinder -M4 "retracted" |
| 6 | - | | -P7 | Bool | false | Non-retain | | | | | | display cylinder -M4 "extended" |

 \rightarrow Alternatively, you can also copy and paste these from the tag table.

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.
- \rightarrow Open the action table using the " \square " button in the step field.
- \rightarrow Open the input window using the " \mathbb{B} " button at the transition.

| 052 | -10 | 0_GRAPH_ | S7-1500 | ► CP | U_1516 | 5F [CPU | 1516 | F-3 PN/DP] | ► Prog | ram blo | icks ▶ | SIGNAL_ | LAMPS | [FB30] | | | | | | _ # = × |
|-----|-----|--|---------|--------|--------|---------|-------|------------|----------------|------------------------|----------|-----------------------------|-----------|---|-----------------|---------|--------|-----------|-----|----------|
| 8 | ×. | 980 | S S | ы Řя | X ≓' : | e 🔒 | | | 8 ± 2 : | - 121 ± | 10 | € ₀ 🍄 | A 00 | ņ | | | | | | |
| | | AL_LAMPS | | | | | | | | | | | | | | | | | | |
| | N | lame | | Data t | ype | Default | value | Retain | Acce | Writa | Visible | Setpoi | Sup | . Comment | | | | | | |
| 1 | • 🗊 | Input | | | | | | | | | | | | | | | | | | <u>^</u> |
| 2 . | • | OFF_S | Q | Bool | | false | | Non-retain | | | | | | Turn sequenc | e off | | | | | |
| 3 . | • | INIT_SC | 2 | Bool | | false | | Non-retain | | | | | | Set sequence | to initial stat | te | | | | |
| 4 | • | ACK_EI | F | Bool | | false | | Non-retain | | | | | | Acknowledge | | faults | | | | |
| | • | | | Bool | | false | | Non-retain | | | | | | notification E | mergOFF | | | | | |
| 6 . | • | -K0 | | Bool | | false | | Non-retain | | | | | | plant "on" | | | | | | ~ |
| > | 3 S | ananent pre- equences (l: sequencer | (1) | 6 | Comm | quencer | | | | 20 [[] p20 Trans | 520 1 | ⊡ ⊣ - Step2(nterlock |): initia | -[w] → → αw> Istep t Qualifier ≪Add new> | Action | NEG NOT | r swap | CMF> CMF> | × _ | |

- \rightarrow 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.

| SIGNAL_I | amps | ਾਲੇ ਲੱਭ | ≝" ¤ _ | E | | 월 ± - 22 | ± 留 ± | 12 00 | 6 ₀ 🍄 | 0, oo | | | | | | | E |
|---|--|---------------|---------------------------------|-------------|----------------|------------|------------|----------------------------|--------------|----------|-----------------|--------------------------------|-----------|---------|---|---------------------------|-----|
| Name | | Data type | Default | value | Retain | Acce. | Writa. | . Visible . | Setpoi | Sup | Comment | | | | | | |
| | OFF_SQ | Bool | false | | Non-retain | n 🗌 | | | | | Turn seque | nce off | | | | | |
| - | INIT_SQ | Bool | false | | Non-retain | ı 🗌 | | | | | Set seque | nce to initia | l state | | | | |
| | ACK_EF | Bool | false | | Non-retain | | | | | | Acknowled | lge all error | rs and fa | ults | | | |
| | -A1 | Bool | false | | Non-retain | | | | | | notification | EmergOF | F | | | | |
| A CONTRACTOR OF | -КО | | false | | Non-ret | | | | | | plant "on" | | | | | | |
| | -50 | Bool | false | | Non-retain | n 🔽 | | | | | mode sele | ctor manua | 1(0) / au | A | A | | |
| Permaner Seque 1: seque | nt pre-instructi | 🖸 1: | 中 ↓ sequencer ment | + SIGNAL | ţ, Ŧ ∟lamps | | - | >=1 [7] | } - | -01 -[| w] <u>-</u> F ° | | | | | CMP> CMP> T_MAX T_WARN | |
| Seque 1: seque S20 Step20 | nt pre-instructi nces (1) | D ₩ 1: Com | sequencer | | | | - | 1 | | | | MP>T CMP>U | | | | CMP> CMP> T_MAX T_WARN | × |
| Permaner Seque 1: seque | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | | | | - | >=1 | | 0 - Step | w] <u>- F</u> o | ирэт смрэц step Qualifia | conv i | | | CMP> CMP> TWAX TWARN | × - |
| Seque Seque 1: seque | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | | | | - | >=1 [7] | | 0 - Step | w] → F α | ирэт смрэц step | conv i | IEG NOT | | CMP> CMP> T_MAX T_WARN | × _ |
| Permaner Seque 1: seque | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | | | | - | >=1 [7] | | 0 - Step | w] → F α | ирэт смрэц step Qualifia | conv i | IEG NOT | | CMP> CMP> T_MAX T_WARN | × - |
| Seque Seque 1: seque Step20 | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | | | | - | >=1 [7] | | 0 - Step | w] → F α | ирэт смрэц step Qualifia | conv i | IEG NOT | | CMP> CMP> T_MAX T_WARN | × - |
| Seque Seque 1: seque Step20 | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | SIGNAL | LAMPS | ₽ + | <u>ء</u> د | >=1 177 | | 0 - Step | w] → F α | ирэт смрэц step Qualifia | conv i | IEG NOT | | CMP> CMP> T_MAX T_WARN | × - |
| Seque Seque 1: seque Step20 | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | SIGNAL | | ₽ + | <u>ء</u> د | >=1 177 | 52 | 0 - Step | w] → F α | ирэт смрэц step Qualifia | conv i | IEG NOT | | CMP> CMP> TMAX TWAEN | × - |
| Seque Seque 1: seque Step20 | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | SIGNAI | nain switch | ₽ + | <u>ء</u> د | >=1 127 520 Step20 | 52 | 0 - Step | w] → F α | ирэт смрэц step Qualifia | conv i | IEG NOT | | CMP> CMP> TMAX TMARN | × - |
| Seque Seque 1: seque Step20 | nt pre-instructi nces (1) Jencer SIGNAL_L/ | D ₩ 1: Com | sequencer | SIGNAI | LAMPS | ₽ + | <u>ء</u> د | >=1 127 520 Step20 | 52 | 0 - Step | w] → F α | ирэт смрэц step Qualifia | conv i | IEG NOT | | CMP> CMP> TMAX TMARN | × - |

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



- \rightarrow Drag "Open simultaneous branch" onto the green square \blacksquare above Step 22.

 \rightarrow Drag "Open simultaneous branch" onto the green square \blacksquare above Steps 23 and 24.



 \rightarrow Drag "Close branch" onto the green square \blacksquare below Steps 22 to 25.



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

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



- \rightarrow Insert "-K0" with a negation as the step-enabling condition in the window of Transition 2.
- → Designate T2 Trans2: as "main switch off".



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



- \rightarrow Change to the single step view.
- \rightarrow Interconnect Step 21 / Step 21a and designate the action.

| Na | vigation 🔍 🍳 🖶 | 8 | >=1 | [??] | н | -01 | -(w) _= | CMP>T | CMP>U | CONV | NEG | NOT | SWAP | CMP> | CMP> T.WARN |
|----|---|------|--------|---------|-----------------------|-----------------------|---------------|---------|--------|---------|--------|-----|------|-----------|----------------|
| > | Permanent pre-instructio | | | | | | 1000 | | | | | | | - Charles | - Carola |
| v | Sequences (1) | S21: | | | | | | | | | | | | | |
| | 1: sequencer SIGNAL_LAMPS | Cor | nment | 5 | | | | | | | | | | | |
| | - T2 S30 Step20 - Trans1 - S312 Step21b - S312 Step21b | - | Interl | ock -(i | c)-: #"-A1" | | nterlock C | | | | | | | | |
| | T2 Trans2 | • | Super | visio | n -(v)- | ÷ | | | | | | | | | |
| | ¥ S20 | - 1 | Actio | ns: d | isplay | "emerg | ency stop | activat | ed" | | | | | | |
| | - | -(c) | - | Interl | ock | Event | Qualif | ier | | | | Act | tion | | |
| | • | • | | -(C)- | | the the second second | N - ⊲Add n | | ong as | step is | active | ¢"- | P4" | | |

 \rightarrow Interconnect Step 212 / Step 21b and designate the action.

| Navigation 🔍 🔍 🖶 | 8 | >=1 ?? | - | -01 | -[w] | | CMP>T C | MP>U CO | | NOT | SWAP | CMP> TWARN |
|--|--|-------------|---------|--------------|---|--------|-----------|----------|----------|--------|------|---------------|
| > Permanent pre-instructio | \$212 | : Step21b | | | | | | | | | | |
| Sequences (1) | Contraction of the local division of the loc | mment | | | | | | | | | | |
| ▼ 1: sequencer SIGNAL_LAMPS | COI | linnent | | | | | | | | | | |
| | • | Interlock - | (c)-: | | | | | | | | | |
| 520 Ti Ti Tens1 Star | | Supervisio | #"-S0" | - - 0 | Interlo C | ick | | | | | | |
| ¥ 520 | | | | | | | | | | | | |
| de la | • | Actions: o | display | "manu | al mo | de" | | | | | | |
| | | | | | | | | | | | | |
| | -(c) | - Inter | lock | Event | | | | | | | tion | |
| | | -(C)- | | | | | t as long | g as ste | p is act | ive ∳" | -P2" | |
| | | | | | <ac< td=""><td>dd new</td><td>Þ</td><td></td><td></td><td></td><td></td><td></td></ac<> | dd new | Þ | | | | | |
| | | | | | | | | | | | | |

 \rightarrow Interconnect Step 213 / Step 21c and designate the action.

| Navigation 🔍 🔍 🛃 | & >=1 | [??] | ol | -fw] | CMP> | T CMP>U | CONV NEG | NOT | SWAP | CMP> | CMP> T WARN |
|---|----------------------------|----------|----------------|----------------|-------|-----------|-------------|-------|------|-------|----------------|
| > Permanent pre-instructio | | | | 2.400 | | | | | | 1_MAX | LWARN |
| ✓ Sequences (1) | \$213: Ste | | | | | | | | | | |
| LAMPS | Commer | 12 | | | | | | | | | |
| 5712 40 5213 40 52 5212 349 216 552 349 216 552 | ▼ Inter | |)-: "-so" — | Interlock C | | | | | | | |
| | | | -(v)-: | | | | | | | | |
| | Action | ons: dis | play auto | matic mo | de" | | | | | | |
| | -(c)- | Interlo | ck Even | t Qual | ifier | | | Ac | tion | | |
| | | -(C)- | | N | | long as s | tep is acti | /e #" | -P3" | | |

 \rightarrow Interconnect Step 214 / Step 21d and designate the action.

| Navigation 🔍 🍳 🖶 | 8 | > = 1 | [??] ⊣ | -01 | -[w] | o | MP>T CMP>U | J CONV | NEG | NOT | SWAP | | CMP> TWARN |
|--|------|-------------------------|---------------|--------------|-------------|-----------------|------------|---------|--------|-------|------|--------|---------------|
| > Permanent pre-instructio | - | | - | | 107907 | 76 | | | | | | 1_0100 | 1.0000 |
| ✓ Sequences (1) | | l: Step mment | | | | | | | | | | | |
| | • | Interio | ock -(c)-: | | | | | | | | | | |
| 5113 ^{ere} 514 ^{ere} 53 Rep21c Step21d Step | | | #"-B | 1* | Interl C | 110.000.000 | | | | | | | |
| | | Super | vision -(v | <i>(</i>)-: | | | | | | | | | |
| | • | Action | ns: displa | y cylind | ler "ex | tended" | | | | | | | |
| | -(c) | - | nterlock | Ever | nt Qu | Jalifier | | | | Ac | tion | | |
| | | | -(C)- | | | -Set dd new: | as long as | step is | active | e ∳". | -P6" | | |

 \rightarrow Interconnect Step 215 / Step 21e and designate the action.

| Navigation 🔍 🍳 🖶 | .8 | > = 1 ?? | - - o | ı -{w] _= | | CONV NEG | NOT : | | CMP> TWARN |
|----------------------------------|--------|--------------------------|--------------|----------------|-----------------------|---------------|--------|----|---------------|
| > Permanent pre-instructio | \$215. | Step21e | | | | | | | |
| ✓ Sequences (1) | | ment | | | | | | | |
| | ▼ In | terlock -(| c)-: | | | | | | |
| d ^{ers} 5315 Stop2te | | | ≠"-B2" — | Interlock C | | | | | |
| | | upervisior ctions: di | | ider "retracte | d" | | | | |
| | -(c)- | Interl | ock Eve | ent Qualif | ier | | Acti | on | |
| | | -(C)- | | N - ⊲Add n | Set as long as ew> | step is activ | e #"-P | 7" | |

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

- \rightarrow Open OB1 and delete the block call in Network 1.
- → Call function block "SIGNAL_LAMPS[FB30]" in Network 1.
- \rightarrow Confirm the name of the data block.
- \rightarrow 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 $\boxed{1}$ save project button in the menu. To compile all blocks, click the "Program blocks" folder and select the button $\boxed{1}$ for compiling in the menu. (→ $\boxed{1}$ save project → Program blocks → $\boxed{1}$)
- → After successful compilation, the complete controller with the created program, as previously described in the modules for hardware configuration, can be downloaded (\rightarrow \blacksquare).
- \rightarrow Test the SIGNAL_LAMPS block.
7.18 Creating function block FB20 "CLOCK_PULSE"

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

| Project tree | |
|-------------------------------|---|
| Devices | |
| Ŭ | |
| ▼ 3052-100_GRAPH_S7-1500 | ^ |
| 🏙 Add new device | |
| Devices & networks | |
| CPU_1516F [CPU 1516F-3 PN/DP] | = |
| Device configuration | |
| 🛂 Online & diagnostics | |
| 🔻 🔜 Program blocks | |
| 📑 Add new block | |
| 🖀 Main [OB1] | |
| AUTOMATIC_MODE [FB50] | |
| SIGNAL_LAMPS [FB30] | |
| AUTOMATIC_MODE_DB [DB1] | |
| SIGNAL_LAMPS_DB [DB2] | |

→ 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".

| Name: CIOCK_PULSE | | | | |
|-----------------------|------------------|-------------------------------|-----------------------------------|----------------------|
| | Language: | GRAPH | | |
| OB | Number: | 20 | • | |
| Organization block | | 💽 Manual | | |
| DIOCK | | Automatic | | |
| | Description: | | | |
| FB | Function blocks | s are code blocks that : | ore their values permanently in i | nstance data blocks, |
| Function block | so that they rer | main available after the | block has been executed. | |
| | | | | |
| | | | | |
| | | | | |
| FC | | | | |
| Function | | | | |
| | | | | |
| СВ | more | | | |
| В | | | | |

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.

| | | Na | me | Data type | Default value | Retain | Access | Writa | Visibl | Setpo | Sup | Comment |
|----|---|----|--------------------|----------------|---------------|----------|--------|-------|--------|----------|-----|-----------------------------------|
| 1 | - | • | Input | | | | | | | | | |
| 2 | - | | OFF_SQ | Bool | false | Non-reta | | | | | | Turn sequence off |
| 3 | - | | INIT_SQ | Bool | false | Non-reta | | | | | | Set sequence to initial state |
| 4 | - | | ACK_EF | Bool | false | Non-reta | | | | | | Acknowledge all errors and faults |
| 5 | - | • | Output | | | | | | | | | |
| 6 | | | <add new=""></add> | | | | | | | | | |
| 7 | | • | InOut | | - | | | 100 | - | | | |
| В | | | <add new=""></add> | | | | | | | | | |
| 9 | - | • | Static | | | | | | | | | |
| 10 | - | | RT_DATA | G7_RTDataPlus | | Non-reta | | | | | | Internal data area |
| 11 | - | | Trans1 | G7_Transition | | Non-reta | | | | | | Transition structure |
| 12 | - | | Step1 | G7_StepPlus_V6 | | Non-reta | | | | V | | Step structure |
| 13 | - | - | Temp | | | | | | | | | |

- \rightarrow 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.

| | CL | oc | K_PULSE | | | | | | | | | |
|----|----|----|--------------------|----------------|---------------|----------|--------|-------|--------|----------|-----|-----------------------------------|
| | | Na | ime | Data type | Default value | Retain | Access | Writa | Visibl | Setpo | Sup | Comment |
| 1 | - | - | Input | | | | | | | | | |
| 2 | - | | OFF_SQ | Bool | false | Non-reta | | | | | | Turn sequence off |
| З | - | - | INIT_SQ | Bool | false | Non-reta | | | | | | Set sequence to initial state |
| 4 | - | | ACK_EF | Bool | false | Non-reta | | | | | | Acknowledge all errors and faults |
| 5 | - | • | Output | | | | | | | | | |
| 6 | - | | Clock_Pulse | Bool | false | Non-reta | | | | | | clock pulse 1Hz |
| 7 | | | <add new=""></add> | | | | | | | | | |
| 8 | - | - | InOut | | | | | | | | | |
| 9 | | | <add new=""></add> | | | | | | | | | |
| 10 | - | - | Static | | | | | | | | | |
| 11 | - | | RT_DATA | G7_RTDataPlus | | Non-reta | | | | | | Internal data area |
| 12 | - | | Trans1 | G7_Transition | | Non-reta | | | | V | | Transition structure |
| 13 | - | | Step1 | G7_StepPlus_V6 | | Non-reta | | | | V | | Step structure |
| 14 | - | - | Temp | | | | | | | | | |

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.
- \rightarrow Open the action table using the " \mathbb{I} " button in the step field.
- \rightarrow Open the input window using the " \mathbb{B} " button at the transition.
- → Now drag the comparator "Greater than step activation time" onto the green square as the stepenabling 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.



- \rightarrow Open the action table using the " \square " button in the step field.
- \rightarrow Open the input window using the " \mathbb{D} " button at the transition.
- → Now drag the comparator "Greater than step activation time" onto the green square as the stepenabling 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".
- \rightarrow Drag a jump to the double arrow and select Step 100 as the jump destination.

| 052-100_GRAPH_\$7-1500 | CPU_1516F [CPU 1516F-3 PN/DP] Program blocks CLOCK_PULSE [FB20] | _ 🖬 🖬 🗙 |
|---------------------------|---|---------|
| | | |
| 물 🗟 🖻 🖥 🖂 🗃 활 왕 | ká ká 🖻 🖻 🐛 🖿 🚍 💬 🕲 ± 😫 ± 😫 ± 😭 🥙 📞 🍄 🖉 | |
| | Block interface | |
| Navigation 🔍 🔍 🖶 | | |
| > Permanent pre-instr | | |
| ✓ Sequences (1) | 1: sequencer ClOCK_PULSE Comment | |
| ▼1: sequencer ClOCK_PULSI | comment | |
| 172 | | |
| \$1.00 Step100 | T 2 | |
| Ti | S100 | |
| 51 01 Step 101 | Step100 | |
| T2 Trans2 | T1 - Trans1: pulse off time X = TT Trans1 | |
| ¥ 51.00 | | |
| | Time | |
| | #Step100.T IN1 T#500MS IN2 | |
| | | |
| | 5101 III S101 - Step101: clock pulse 1Hz | |
| | Step101 Interlock Event Qualifier Action | |
| | <add new=""></add> | |
| | | |
| | 12 | |
| | T2 - Trans2: pulse on time X - Trans2 | |
| | > | |
| | #Step101.T IN1 | |
| | T#500MS - IN2 | |
| | | |
| | \$100 | |

 \rightarrow 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. \rightarrow
- Call function block of the clock pulse in Network 1. \rightarrow
- Confirm the name of the data block. \rightarrow
- \rightarrow Interconnect the pulse tag of the block with the global tag "-P1" of the sorting station.

| Project tree | | | -100_GRA | | | | 80112200-0 | | | NAMES OF TAXABLE | | | | locks | ▶ Mai | n [OB |
|-------------------------------|---|------|--------------------|------|----------|---------------------|----------------------|---------|------------|------------------|----------|-----|------|-------|-------|------------|
| Devices | | | | | | | | | | | | | | | | |
| E | | 1.38 | <mark>∦ ∌</mark> € | | = 🖻 | | © 2 | ± .21 ± | <u>명</u> ± | - 12 | 60 6 | . 📾 | (# D | G= 1 | - ×- | G (|
| | | | | 1 | | _ | | _ | - (| | nterface | | | | - | |
| ▼ 7 052-100 GRAPH \$7-1500 | ~ | | R Ex | - | | | 17 - 11 | | | 1 + 1 | 1 1 | | | | | |
| Add new device | | 8 | >=1 ?? | t H | -01 | \rightarrow | -[=] | | | | | | | | | |
| Devices & networks | | | | | | | 10 | 1.58 | | | | | | | | |
| CPU_1516F [CPU 1516F-3 PN/DP] | | 1.20 | lock title: | Mai | n Progra | m Swe | eep (Cyd | le) | | | | | | | | |
| Device configuration | | Col | mment | | | | | | | | | | | | | |
| V. Online & diagnostics | | - | Network | 1: 0 | all CLOC | K PUL | SE for | test | | | | | | | | |
| ▼ 🙀 Program blocks | | | Comment | | | | | | | | | | | | | |
| Add new block | | | comment | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| AUTOMATIC_MODE [FB50] | | | | | | | 1200 Contract (1997) | PULSE | | | | | | | | |
| CLOCK_PULSE [FB20] | | | | | | | | B" | | | | | | | | |
| SIGNAL_LAMPS [FB30] | | | | | | | %FI | 320 | | £ | | | | | | |
| AUTOMATIC_MODE_DB [DB1] | | | | | | | CLOCK | _PULSE" | | | | | | | | |
| CLOCK_PULSE_DB [DB3] | | | | | — EN | | | | | | | | | | | |
| SIGNAL_LAMPS_DB [DB2] | | | | | e — OFI | | | | | %00 | 15 | | | | | |
| 🕨 🔙 System blocks | | | | | | | | Clock | k Pulse | - *-P1 | | | | | | |
| Technology objects | | | | | e — ACH | | | | ENC | | | | | | | |
| Energy objects | | | | | | 0 11 000 | | | | | | | | | | |

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

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 |
|-----------|--|
| Ν | Set to 1 as long as step is active: |
| | 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: |
| | CALL " <fcname>" (parameter list)</fcname> |
| | Function blocks (FB) with syntax: |
| | CALL " <fbname>", "<dbname>" (parameter list)</dbname></fbname> |
| S | Set to 1 |
| | As soon as the step is active, the operand is set to "1" and then remains at "1". |
| R | Set to 0: |
| | As soon as the step is active, the operand is set to "0" and then remains at "0". |

The following actions cannot be linked with events:

| Qualifier | Description |
|-----------|--|
| D | On delay: |
| | 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. |
| L | Set for limited time: |
| | 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. |
| TF | Turn off timer: |
| | 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. |

| 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") |
| LO | 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. |

The following events are defined for GRAPH:

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"

- \rightarrow The GRAPH function block for release of the controller is to be programmed and tested.
- \rightarrow 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".

| Name: RELEASE | | | | | |
|-----------------------|--|---|---------------------------------------|-------------------------------------|-----------------|
| Organization block | Language: Number: | GRAPH 10 Manual Automatic | • | | |
| Function block | Description: Function block so that they ren | s are code blocks that main available after th | store their value e block has beer | s permanently in insta executed. | nce data blocks |
| Function | | | | | |
| Data block | more | | | | |
| Additional info | rmation | | | | |

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.

| | | Na | ime | 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=""></add> | | | | | | | | | |
| 7 | | • | InOut | | | | | | | | | |
| 8 | | | <add new=""></add> | | | | | | | | | |
| 9 | - | • | Static | | | | | | | | | |
| 10 | - | | RT_DATA | G7_RTDataPlus_V6 | | Non-ret | | | | \checkmark | | Internal data area |
| 11 | - | | Trans1 | G7_TransitionPlus | | Non-ret | | | | Image: A start of the start of | | Transition structure |
| 12 | - | | Step1 | G7_StepPlus_V6 | | Non-ret | | | | V | | Step structure |
| 13 | - | • | Temp | | | | | | | | | |

- \rightarrow 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" (\rightarrow Input: ACK_EF \rightarrow Add row).

| | | Nan | ne | Data type | Default value | Retain | Acce | Wri | Visibl | Setpo | Sup | Comment |
|-----|---------------------------|------|-----------------|------------------|---------------|--------------------|------|-----|--------|---|-----|--|
| R. | | - | Input | | | | | | | | | |
| | | | OFF_SQ | Bool | false | Non-ret | | | | | | Turn sequence off |
| 3 | | | INIT_SQ | Bool | false | Non-ret | | | | | | Set sequence to initial state |
| F. | | | nsert row | Ctrl+Enter | false | Non 💌 | | | | | 1- | Acknowledge all errors and faults |
| 100 | • | -# / | Add row | Alt+Ins | | | - 8 | | | | | |
| | Ð | | | Ctrl+X Ctrl+C | | | | | | | | |
|) | | | Paste Delete | Ctrl+V Del | - | | | | ġ | | | |
| 10 | \$ \$ | | Rename | F2 | | Non-ret Non-ret | | | | × | | Internal data area Transition structure |
| 2 | 2 🔨 💭 Add new supervision | | | | | Non-ret | | | | Image: A start of the start of | | Step structure |

- \rightarrow 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.

| | RE | LE | ASE | | | | | | | | | |
|----|----|-----|--------------------|-----------|---------------|---------|------|-----|---------|-------|-----|--|
| | - | N | lame | Data type | Default value | Retain | Acce | Wri | Visibl | Setpo | Sup | Comment |
| 1 | - | | lnput | | | | | | | | | |
| 2 | - | | OFF_SQ | Bool | false | Non-ret | | | | | | Turn sequence off |
| 3 | | | INIT_SQ | Bool | false | Non-ret | | | | | | Set sequence to initial state |
| ŧ. | - | 1 = | ACK_EF | Bool | false | Non-ret | | | | | | Acknowledge all errors and faults |
| 5 | - | | -A1 | Bool | false | Non-ret | | | | | | notification EmergOFF ok |
| 5 | - | 1 = | -KO | Bool | false | Non-ret | | | | | | plant "on" |
| 7. | - | | -50 | Bool | false | Non-ret | | | | | | mode selector manual(0) / automatic(1) |
| 3 | - | = | -51 | Bool | false | Non-ret | | | | | | pushbutton automatic start |
| 2 | - | 1 = | -52 | Bool | false | Non-ret | | | | | | pushbutton automatic stop |
| 0 | - | | -53 | Bool | false | Non-ret | | | | | | pushbutton inching operation conveyor -M1 forwards |
| 11 | | = | -54 | Bool | false | Non-ret | | | | | | pushbutton inching operation conveyor -M1 backward |
| 12 | - | = | -55 | Bool | false | Non-ret | | | | | | pushbutton inching operation cylinder -M4 retract |
| 13 | | 1 = | -56 | Bool | false | Non-ret | | | | | | pushbutton inching operation cylinder -M4 extend |
| 14 | | | <add new=""></add> | | | | | | | | | |
| 15 | - | • | Output | | | | | | | | | |
| 16 | - | = | -P1 | Bool | false | Non-ret | | | | | | display "main switch on" |
| 17 | - | | Man/Auto-OFF | Bool | false | Non-ret | | | | | | switch off seqzencer OPERATING_MODES |
| 18 | - | | Automatic | Bool | false | Non-ret | | | | | | switch off seqzencer AUTOMATIC_MODE |
| 19 | - | = | Signal_Lamp | Bool | false | Non-ret | | | | | | initialize seqzencer SIGNAL_LAMPS |
| 20 | - | 1 = | Clock_Pulse | Bool | false | Non-ret | | | | | | initialize seqzencer ClOCK_PULSE |
| 21 | - | | Man/Auto_INIT | Bool | false | Non-ret | | | | | | initialize seqzencer OPERATING_MODES |
| 22 | - | = | Release | Bool | false | Non-ret | | | | | | operational release |

 \rightarrow Alternatively, you can also copy and paste these from the tag table.

7.24 Programming of FB10 "RELEASE"

- → First, we assign the sequencer the name "sequencer RELEASE" by overwriting the text <new sequence>.
- \rightarrow Open the action table using the " \mathbb{I} " button in the step field.
- \rightarrow Open the input window using the " \mathbb{B} " button at the transition.
- \rightarrow 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.
- \rightarrow For this, program indicator light "-P1" using event S1 and qualifier R.
- \rightarrow Program the remaining actions in Step 1.



 \rightarrow Programmable events:

| Step | 51 | Interlock | Even | t Qualifier | Action | | | |
|------|---------------|-----------|--|--|---------------|--|--|--|
| | | | 51 | | ✓ R | #"-P1" | | |
| | | | A1 -/ L0 -I L1 -0 R1 -I S0 -0 | No event Alarm acknowle ncoming interlo Dutgoing interlo ncoming regist Dutgoing step | ock condition | :o-OFF" .c_OFF .amps_INII 11se_INIT | | |
| | T1 Trans 1 | | 51 - Incoming step V0 - Outgoing monitoring error V1 - Incoming monitoring error | | | | | |

- → 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.
- \rightarrow For this, program indicator light "-P1" using event S1 and qualifier S.
- \rightarrow 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 ¹ below Step 2. The alternative branch with Transition 3 is inserted.



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



- \rightarrow Open the input window using the " \mathbb{D} " button at Transition 3.
- \rightarrow Insert an AND logic operation with eight inputs in the window of Transition 3.
- \rightarrow 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.
- \rightarrow Open the action table for Step 3 using the " \square " button in the step field.
- \rightarrow 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 "^{bb}" button at Transition 4 and designate T4 Trans4: as "emergency stop operated or main switch off".
- \rightarrow Insert an OR logic operation with two inputs in the window of Transition 4.
- \rightarrow Interconnect the OR logic operation according to the specification in the GRAFCET.
- \rightarrow Drag a jump to the double arrow below Transition 4 and select Step 1 as the jump destination.



- \rightarrow The GRAPH function block is now finished and can be called for testing in OB1.
- \rightarrow Open OB1 and delete the block call in Network 1.
- \rightarrow Call function block of the release in Network 1.
- \rightarrow Confirm the name of the data block.
- \rightarrow 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.
- \rightarrow Interconnect the tags of the block with the local Temp tags of OB1.

| | Ma | in | | | | |
|---|-----|----|------------------|-----------|---------------|--------------------------------------|
| | | Na | me | Data type | Default value | Comment |
| 1 | - | • | Input | | | |
| 2 | | • | Temp | | | |
| 3 | | | Man/Auto-OFF | Bool | | switch off seqzencer OPERATING_MODES |
| 4 | - | | Automatic_OFF | Bool | | switch off seqzencer AUTOMATIC_MODE |
| 5 | | | Sinal_Lamps_INIT | Bool | | initialize seqzencer SIGNAL_LAMPS |
| 6 | - | | Clock_Pulse_INIT | Bool | | initialize seqzencer ClOCK_PULSE |
| 7 | - | | Man/Auto_INIT | Bool | | initialize seqzencer OPERATING_MODES |
| 8 | -00 | | Release | Bool | | operational release |



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

7.25 Creating function block FB40 "OPERATING_MODES"

- \rightarrow Finally, the GRAPH function block "OPERATING_MODES" is to be programmed and tested.
- \rightarrow 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".

| Add new block | | | | | × |
|---------------------|----------------------|--|---|---|---|
| Name: | | | | | |
| OPERATING_MODES | | | | | |
| Organization | Language: Number: | GRAPH 40 Manual | • | | |
| block | | Automatic | | | |
| Function block | | : are code blocks that nain available after the | | s permanently in instance data blocks, n executed. | |
| Data block | more | | | | |
| > Additional infor | mation | | | | _ |
| Add new and open | | | | OK Cancel | |

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.

| | | Na | me | 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=""></add> | | | | | | | | | |
| 7 | - | • | InOut | | | | | | | | | |
| 8 | | | <add new=""></add> | | | | | | | | | |
| 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 | | | | Image: A start of the start of | | Step structure |
| 13 | | • | Temp | | | | | | | | | |

- \rightarrow 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" (\rightarrow Input: ACK_EF \rightarrow Add row).

| | | Na | me | 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 | - | in. | ACK EF | Rool | false | Non 🔻 | | | | | 1 | Acknowledge all errors and faults |
| 5 | | - | Insert row | Ctrl+Er | nter | | | | | | | |
| 6 | | * | Add row | Alt+ | -Ins | | ā | ā | | | | |
| 7 | - | 0.0 | Cut | Ctr | rl+X | | | | | | | |
| 8 | | 1 | Сору | Ctr | rl+C | | | | | | | |
| 9 | - | ì | Paste | Cti | rl+V | | | | | | | |
| 10 | - | × | Delete | | Del | Non-ret | | | | | | Internal data area |
| 11 | - | 20 | Rename | | F2 | Non-ret | | | | | | Transition structure |
| 12 | - | (Th | Add new supervis | ion i | | 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.
- \rightarrow Alternatively, you can also copy and paste these from the tag table.

| | Ur | | ATING_MODES | Terretoria | W. | | | | | | | |
|----|-----|-----|----------------------|------------|---------------|---------|------|-----|---------|-------|-----|--|
| | | Nar | 117 | Data type | Default value | Retain | Acce | Wri | Visible | Setpo | Sup | Comment |
| 1 | | • | Input | | | | | | | | | |
| 2 | -00 | | OFF_SQ | Bool | false | Non-ret | | | | | | Turn sequence off |
| 3 | | | INIT_SQ | Bool | false | Non-ret | | | | | | Set sequence to initial state |
| 4 | -01 | | ACK_EF | Bool | false | Non-ret | | | | | | Acknowledge all errors and faults |
| 5 | | 1 = | Release | Bool | false | Non-ret | | | | | | |
| 6 | - | | Clock_Pulse | Bool | false | Non-ret | | | | | | |
| 7 | -00 | | -50 | Bool | false | Non-ret | | | | | | mode selector manual(0) / automatic(1) |
| 8 | -00 | | -\$1 | Bool | false | Non-ret | | | | | | pushbutton automatic start |
| 9 | -00 | | -52 | Bool | false | Non-ret | | | | | | pushbutton automatic stop |
| 10 | -00 | | -53 | Bool | false | Non-ret | | | | | | pushbutton inching operation conveyor -M1 forwards |
| 11 | | | -54 | Bool | false | Non-ret | | | | | | pushbutton inching operation conveyor -M1 backward |
| 12 | - | | -55 | Bool | false | Non-ret | | | | | | pushbutton inching operation cylinder -M4 retract |
| 13 | -01 | | -56 | Bool | false | Non-ret | | | | | | pushbutton inching operation cylinder -M4 extend |
| 14 | - | | -B1 | Bool | false | Non-ret | | | | | | sensor cylinder -M4 retracted |
| 15 | - | | -B2 | Bool | false | Non-ret | | | | | | sensor cylinder -M4 extended |
| 16 | | | <add new=""></add> | | | | | | | | | |
| 17 | -00 | • | Output | | | | | | - | | | |
| 18 | -00 | - | -P5 | Bool | false | Non-ret | | | | | | display "automatic mode" started |
| 19 | | | -Q1 | Bool | false | Non-ret | | | | | | conveyor motor -M1 forwards fixed speed |
| 20 | - | = | -Q2 | Bool | false | Non-ret | | | | | | conveyor motor -M1 backwards fixed speed |
| 21 | -00 | | -M2 | Bool | false | Non-ret | | | | | | cylinder -M4 retract |
| 22 | - | | -MB | Bool | false | Non-ret | | | | | | cylinder -M4 extend |
| 23 | -00 | | Automatic_OFF | Bool | false | Non-ret | | | | | | switch off seqzencer AUTOMATIC_MODE |
| 24 | -00 | | Automatic_INIT | Bool | false | Non-ret | | | | | | initialize seqzencer AUTOMATIC_MODE |
| 25 | -00 | | Automatic_Mode_Start | Bool | false | Non-ret | | | | | | 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.
- \rightarrow Remember to designate the action tables and the transition windows.
- \rightarrow The finished block is shown on the following pages.



\rightarrow Sequencer of the operating mode selection

→ Steps S10 and S11, Transitions T1 and T2



For unrestricted use in educational and R&D institutions. © Siemens 2019. All rights reserved. sce-052-100-graph-s7-1500-r1903-en.docx

 \rightarrow Steps S12 and S13, Transitions T3 and T4



 \rightarrow Step S12 single step view

| S12: Ste | ep12 | | | | |
|----------|---------------|-----------|-----------------------------------|--------|--|
| Comm | | | | | |
| ▼ Inte | erlock -(c)-: | [| | | |
| | #Clock_Puls | | nterlock C | | |
| | 25 | | | | |
| ► Sup | ervision -(v |)-: | | | |
| ▼ Act | ions: displa | y "automa | atic mode" flash | | |
| -(c)- | Interlock | Event | Qualifier | Action | |
| _ | -(C)- | Lvent | N - Set as long as step is active | | |
| | | | <add new=""></add> | | |



 \rightarrow Transitions T7 and T8

→ Step S14: Transitions T5 and T6



\rightarrow Step S14 single step view

| S14: Ste | ep14 | | | | |
|----------|-------------------------|-----------|--|---------|--|
| Comm | ent | | | | |
| ▼ Inte | erlock -(c)-: | | | | |
| | #"-53 #"-54 #"-B1 | | & Interlock C | | |
| ► Sup | pervision -(v) |)-: | | | |
| ▼ Act | tions: inching | g operati | on conveyor -M1 forwards | | |
| -(c)- | Interlock | Event | Qualifier | Action | |
| | -(C)- | | N -Set as long as step is active <add new=""></add> | \$"-Q1" | |
| | | | | 1 | |

 \rightarrow Step S142 single step view

| 5142: 1 | Step14b | | | | |
|---------|-------------------------|-----------|--|--------|--|
| Comm | ent | | | | |
| ▼ Int | erlock -(c)-: | | | | |
| | #*-54 #*-53 #*-B1 | | & Interlock | | |
| ► Su | pervision -(v) | -: | | | |
| ▼ Ac | tions: inching | g operati | on conveyor -M1 backwards | | |
| -(c)- | Interlock | Event | Qualifier | Action | |
| | -(C)- | | N -Set as long as step is active <add new=""></add> | | |

 \rightarrow Step S143 single step view

| S143: 3 | Step14c | | | | |
|---------|-------------------------|------------|---|----------------|--|
| Comm | ent | | | | |
| ▼ Int | erlock -(c)-: | | | | |
| | #"-55 #"-56 #"-81 | | & Interlock | | |
| ► Su | pervision -(v) | -: | | | |
| ▼ Ac | tions: inching | g operatio | on cylinder -M4 retract | | |
| -(c)- | Interlock | Event | Qualifier | Action | |
| | -(C)- | | N - Set as long as step is active <add new=""></add> | # "-M2" | |

 \rightarrow Step S144 single step view

| S144: Step14d | | | |
|--|-------------------------------|-----------------|--|
| Comment | | | |
| ▼ Interlock -(c)-: | | | |
| * #*-55* 0 #*-82* 0 * | Interlock C | | |
| Supervision -(v)-: | | | |
| ← Actions: inching operation cylind | er -M4 extend, sequencer AUTC | DMATIC_MODE off | |

| -(c)- | 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=""></add> | | |

- → 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 seqzencer OPERATING_MODES |
| 6 | | | Automatic_OFF | Bool | switch off seqzencer AUTOMATIC_MODE |
| 7 | | | Sinal_Lamps_INIT | Bool | initialize seqzencer SIGNAL_LAMPS |
| 8 | - | | Clock_Pulse_INIT | Bool | initialize seqzencer ClOCK_PULSE |
| 9 | - | | Man/Auto_INIT | Bool | initialize seqzencer OPERATING_MODES |
| 10 | - | | Release | Bool | operational release |



- \rightarrow The block of the release is called in Network 1.
- \rightarrow Call function block of the clock pulse in Network 2.
- \rightarrow 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.
- \rightarrow Interconnect the existing Temp tag #Clock_Pulse_INIT.

| • | Network 2: call block CLOCK_PULSE | | | | | |
|---|-----------------------------------|----------------------------|--|--|--|--|
| | Comment | | | | | |
| | %DB3 "CLOCK_PULSE_ DB" | | | | | |
| | "cic | %FB20 DCK_PULSE" | | | | |
| | <mark>EN</mark> false — OFF_SQ | | | | | |
| | #Clock_Pulse_INIT - INIT_SQ | Clock_Pulse — #Clock_pulse | | | | |
| | false — ACK_EF | ENO — | | | | |

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

Network 3: call block SIGNAL_LAMPS Comment %DB2 SIGNAL LAMPS DB" %FB30 "SIGNAL_LAMPS" ... - EN false - OFF_SQ #Sinal_Lamps INIT -- INIT_SQ false - ACK_EF %Q0.6 "-P2" -P2 %i0.0 "-A1" --A1 %Q0.7 - "-P3" -P3 %10.1 "-KO" -- -KO %Q1.0 - "-P4" -P4 %10.2 *-S0* --- -SO %Q1.2 - "-P6" %10.5 -P6 %Q1.3 - "-P7" %10.6 -P7 -"-B2" -0 -B2 ENO -

For unrestricted use in educational and R&D institutions. © Siemens 2019. All rights reserved. sce-052-100-graph-s7-1500-r1903-en.docx

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

| 4 | | • | Temp | | | |
|----|---|---|----------------------|------|---|--|
| 5 | - | | Man/Auto-OFF | Bool | switch off seqzencer OPERATING_MODES | |
| 5 | | | Automatic_OFF | Bool | switch off seqzencer AUTOMATIC_MODE | |
| 1 | - | | Sinal_Lamps_INIT | Bool | initialize seqzencer SIGNAL_LAMPS | |
| 3 | | | Clock_Pulse_INIT | Bool | initialize seqzencer ClOCK_PULSE | |
| | - | | Man/Auto_INIT | Bool | initialize seqzencer OPERATING_MODES | |
| 0 | - | | Release | Bool | operational release | |
| 1 | | | Clock_pulse | Bool | clock pulse 1Hz | |
| 2 | | | -Q1_Manual | Bool | conveyor forwards in manual mode | |
| 3 | - | | -Q2_Manual | Bool | conveyor motor backwards in manual mode | |
| 4 | | | -M2_Manual | Bool | cylinder retract in automatic mode | |
| 5 | - | | -M3_Manual | Bool | cylinder extend in automatic mode | |
| 6 | | | Automatic_OFF_Manual | Bool | switch off seqzencer AUTOMATIC_MODE | |
| 7 | - | | Automatic_INIT | Bool | initialize seqzencer AUTOMATIC_MODE | |
| 18 | | | Automatic_Mode_Start | Bool | start automatic mode | |







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

| 4 | | • | Temp | | |
|----|----|---|----------------------|------|--|
| 5 | - | | Man/Auto-OFF | Bool | switch off seqzencer OPERATING_MODES |
| 6 | | | Automatic_OFF | Bool | switch off seqzencer AUTOMATIC_MODE |
| 7 | - | | Sinal_Lamps_INIT | Bool | initialize seqzencer SIGNAL_LAMPS |
| 8 | | | Clock_Pulse_INIT | Bool | initialize seqzencer ClOCK_PULSE |
| 9 | - | | Man/Auto_INIT | Bool | initialize seqzencer OPERATING_MODES |
| 10 | -0 | | 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 seqzencer AUTOMATIC_MODE |
| 17 | - | | Automatic_INIT | Bool | initialize seqzencer 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. \rightarrow Create networks 6 to 9 to assign the outputs correctly.



- → Then, click the "Program blocks" folder and select the button 1 for compiling all blocks in the menu. (→ $\fbox{3}$ save project → Program blocks → 1)
- → After successful compilation, the complete controller with the created program, as previously described in the modules for hardware configuration, can be downloaded (\rightarrow \blacksquare).
- \rightarrow 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-bystep 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

SCE Learn-/Training Document siemens.com/sce/module

SCE Trainer Packages siemens.com/sce/tp

SCE Contact Partners siemens.com/sce/contact

Digital Enterprise siemens.com/digital-enterprise

Industrie 4.0 siemens.com/future-of-manufacturing

Totally Integrated Automation (TIA) siemens.com/tia

TIA Portal siemens.com/tia-portal

SIMATIC Controller siemens.com/controller

SIMATIC Technical Documentation siemens.com/simatic-docu

Industry Online Support support.industry.siemens.com

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. $\ensuremath{\textcircled{}}$ Siemens 2019

siemens.com/sce