Programming Manual

English

SCU Series
with SafePLC²

Read and follow the programming manual before initial startup/ integration of the module.

Observe safety instructions!

Keep for future use!

Programming manual for SCU series devices

Status: 05/2022

NOTE

The German version is the original version of the programming manual.

- Contact the manufacturer immediately if the manual is missing!
- Always keep the manual at hand
- Make sure that the manual is complete!
- Obtain this manual only through the original publisher!

Subject to modification.

The content of this documentation has been compiled extremely carefully according to our current level of information.
Nevertheless, we indicate that this document cannot always be updated simultaneously with the technical progress.

Information and specifications can be changed at any time. For the current version, please consult www.bbh-products.de.

Manufacturer information:

BBH Products GmbH
Böttgerstraße 40
D-92637 Weiden
GERMANY

Phone: +49 (0)961-48244-0
E-mail: info@bbh.net
Web: www.bbh-products.de

Responsible for the compilation of the documents:
Gerhard Bauer, Managing Director BBH Products

INFORMATION:
Before programming the unit, the unit must be completely installed and put into operation. For this purpose, all connected components must be installed and put into operation, and the connections must be connected. For installing, putting into operation and connecting, please read and observe the SCU installation manual.

INFORMATION:
The documentation (installation manual, programming manual) are disposable via the download of BBH Products GmbH.

1. Tables

1.1. Table of contents

1.2. List of Figures

2. General information

2.1. Important information for use

IMPORTANT!

Read carefully before use!

Please read this instruction manual carefully. Keep it near the machine to consult it in case of questions.

This manual is aimed at the following target groups:

▪ People projecting safe drive systems:	engineers and technicians
▪ Assembly, electrical installation, maintenance:	industrial electricians and service technicians
▪ Commissioning, operation, configuration:	engineers and technicians

2.2. Scope of delivery

The units of the SCU series are delivered with the plugs for entries or exits and for the voltage supply (in mated condition).

The corresponding product information for SCU series are attached to the units. The product information contain i. a. the download link for the complete documentation.

INFORMATION:
The documentations (installation manual, programming manual) are freely available via the online download.

INFORMATION:
Programming software, dongle (hardlock), programming cable etc. must be ordered separately.

2.3. Terms

The term "**safe**" is used in accordance with or according to the following standards: DIN EN ISO 13849-1, DIN EN 61508-1:2011-02 (cf. the section Relevant standards). **"Safe functions for application to PL e or SIL 3"** indicates functions in the sense of the above standards with corresponding integrity (reliability).

The units SCU, SDU, SIO, SSB manufactured by BBH are units implementing safety relevant functions with safe communication via FSoE, and unsafe communication by means of EtherCAT. Internally, these functions are configured as two-channel systems: system A and system B.

2 General information

The system software **SafePLC²** serves to program and configure the units SCU, SDU, SIO, SSB manufactured by BBH.

2.4. Symbols and signal words

Symbol / signal word	Meaning
⚠	Calls your attention to the use and the effects of safety information.
⚠ **DANGER**	Calls your attention to a dangerous situation **causing** death or severe injury if it is not avoided.
⚠ **WARNING**	Calls your attention to a dangerous situation that **can cause** death or severe injury if it is not avoided.
⚠ CAUTION	Calls your attention to a dangerous situation that can cause slight to moderate injury if it is not avoided.
NOTICE:	Calls your attention to possible material damage and other important information.

2.5. Valid documents

The following documents must be read carefully and must be considered during installation:

- Installation manual SCU
  ➔ *HB-37500-810-10-xxF-EN SCU Installation manual*

- Programming manual SafePLC2
  ➔ *HB-37480-820-01-xxF-EN Programming Manual SafePLC^2*

- Error list SCU series
  ➔ *HB-37500-813-02-xxF-EN Error list SCU*
  ➔ *HB-37500-813-02-xxF-EN Error list SCU-SDU modules*

2.6. General safety information

NOTICE:
Work may only be carried out after the SCU installation manual and the SafePLC² programming manual have both been read thoroughly.

⚠ **DANGER**
Programming or changing the programming can cause malfunctions. Malfunctions can cause unexpected starts of the complete system.

⚠ **WARNING**
Inputs and outputs for standard functions, or the digital and analogue data transmitted by communication units must not be used for safety-related applications. Data errors can cause malfunctions which can cause unexpected starts of the system.

NOTICE:
All work concerning electrical installation must be carried out in according to the installation manual “HB-37500-810-10-xxF-EN SCU installation manual“.

3. Requirements

3.1. SCU unit

The SCU is a Master unit for FSoE communication to read encoder data and switching statuses of external Slave units to implement safety functions.

The SCU master can send, receive and process safe data via FSoE and non-safe data via EtherCAT. In the EtherCAT network, the SCU only functions as Slave.

3.2. Network connection

3.2.1. EtherCAT

The EtherCAT network consists of one Master and a certain number of Slaves.
Data transfer is carried out via Ethernet connections existing between the EtherCAT Master and every participant, and, as a rule, running serially from participant to participant. Via these connections Ethernet-Frames are sent. Every network participant ´must first read the received data, filter the data addressed to it, and insert the exit data into the Frame. Having passed all Slaves, the Frame is sent back to the EtherCAT Master.

As thus every participant can influence the transfer of a message, the network must be exactly defined or specified. This specification is carried out via the ESI files, fixing the participants and their properties.

Data transfer is always initiated by the Master – Data transfer is carried out in the EtherCAT network with a transfer time of at best a few µs.

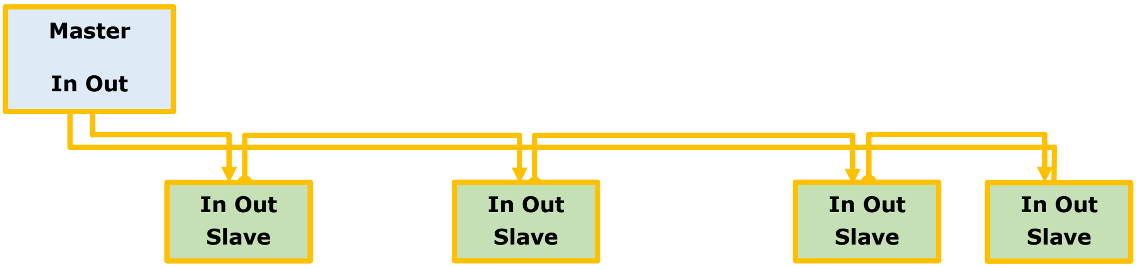


Figure 1: EtherCAT network

3.2.1.1. EtherCAT data transfer

Basically, a difference is made between process data that must be transferred cyclically, and acyclic data, such as configuration data and diagnostic data.

The cyclic process data are assigned to the PDOs (Process Data Objects).
Length and content of the PDOs can be either fixed or variable. The variable content is fixed by the PDO mapping. The PDO's possibilities are fixed by the individual participant's description file (ESI file).

Primarily, acyclic data services are SDOs (Service Data Objects), but they can also be EoE (Ethernet over EtherCAT) or FoE (File over EtherCAT). Also here, the possibilities of the acyclic services are fixed by the individual participant's description file (ESI file).

3.2.2. FSoE

FSoE (Fail Safe over EtherCAT) is the safe data transfer via the EtherCAT network. User data are compiled into data packages, and are supplemented with an additionally transferred checksum. The transfer is carried out every 1ms.
Furthermore, the transfer is monitored via timer (Watchdog) which is checked in every network participant. Thus, any interruption of the data transfer can be safely recognized. The checksum is calculated with a CRC 16 (16-Bit-Cyclic Redundancy Check), and allows to recognize a fault with a residual error probability of < 10^{-9} and thus allows a safe data transfer suitable for the use up to PL e or SIL 3.

Among others, the following units are suitable as FSoE-Slaves:

- SSB = Safe Sensor Box — Slave unit by BBH to read 6 axis
- SDU = Safe Drive Unit — Slave unit by BBH to read one axis
- SIO = Safe IO — Slave unit by BBH to read IOs
- EL 1904 — Slave unit by Beckhoff (to read inputs)
- EL 2904 — Slave unit by Beckhoff (to switch outputs)
- AX 5805 — Slave unit by Beckhoff (to read axis data)
- AX 5806 — Slave unit by Beckhoff (to read axis data)
- SDC — Slave unit by Keba (to read axis data and IOs)

In general, modules from other manufacturers can also be integrated as FSoE slaves, provided that they offer FSoE communication.

3.2.3. FSoE and EtherCAT

The Ethernet frame of the EtherCAT network can contain non-safe as well as safe data. The safe data are called FSoE data, and are assigned to the FSoE Master-and-Slave protocol stack. These data are transferred cyclically, and are thus included in the PDO of the respective participant.

The SCU is designed as FSoE Master, and it starts the safe data transfer via FSoE. Furthermore, as EtherCAT Slave, the SCU participates in the non-safe EtherCAT network.

A separate EtherCAT Master starts the non-safe data transfer via EtherCAT.

The PDOs with the included FSoE files are transmitted cyclically. The cycle time of the transfer is fixed in the configuration of the EtherCAT -Masters. As a rule, it should be adjusted many times shorter than the cycle time of the FSoE master to grant the update of the data within the Watchdog control time of the FSoE Master.

The **SCU** (FSoE Master) starts the safe data transfer via FSoE, receives from the Slave groups (e. g. SIO, SDU, SSB) the data from inputs and evaluates these data. Afterwards, the SCU can implement safety functions, and can switch outputs accordingly – this can be done either by the proper outputs of the units, or by the outputs of the Slave units. Furthermore, as EtherCAT Slave, the SCU is participant in the non-safe EtherCAT network. A separate EtherCAT Master starts the non-safe transfer via EtherCAT.

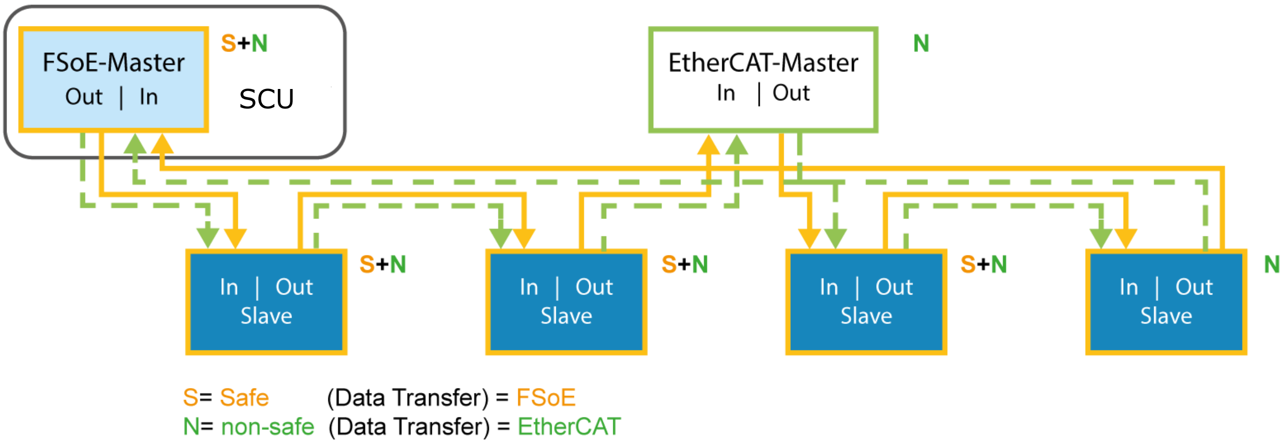


Figure 2 Network FsoE

4. Selecting and parameterizing the SCU via SafePLC^2

The program "SafePLC^2" is a graphics-oriented software to create a PLC-based monitoring program for an SCU system.

This programming software allows the graphic processing of consecutively running programs by means of function blocks, and the adjustment of sensor functions, actuator functions, and other technical functions.

About this manual

This manual explains the bases of SafePLC^2. The manual explains the most important dialogue windows, and the key processes to be followed by using practical examples. These practical examples are constructed in a way that permits you in principle to start with any chapter. Experiences in working with a mouse, with dialogue windows, selection menus etc. are of advantage. Furthermore, you should be acquainted with the basic principles of a programmable logic controller.

4.1. Terms

PLC

The term PLC means **P**rogrammable **L**ogic **C**ontrol. The term PLC is exclusively used in the SCU system.

SafePLC^2

Programming software for the graphical processing of consecutively running programs by means of function blocks, and for the adjustment of sensor functions, actuator functions, and other technical functions.

Function block

Block in an PLC control, influencing the programming sequence of an PLC program either physically or logically. A physical function block (hardware) is p. ex. a button or an output. But a function block is also a logical link of input and output signals within an PLC (e. g. AND or OR).

Function block diagram (Function block language)

Graphically oriented, descriptive programming language based on function blocks according to IEC 1131 to visualize logical connections of the inputs and the outputs of the function blocks of an PLC control. The function block diagram (FBD) shows the function blocks and their logical link in graphical form.

Input / output

Position at a function block where the logical link with other function blocks is possible.

Logical connection

A certain connection between:
a.) an output and an input of a function block.
b.) an input of the PLC and an input of a function block.
c.) an output of a function block, and an output of the PLC

Connector

Connection point between the beginning and the end of a logical connection with an input and an output of a function block.

Attribute

Non-graphical function of a function block. An attribute consists of an identifier and a value.

Routes

Horizontal and vertical connections of logical connections in a function block diagram to avoid intersections with function blocks, and to connect logical connections with identical connectors in an early phase (according to the distance and according to the target function block).

Signal list

Signal lines to and from the PLC in tabular form.

Signal cell

Eligible zone of the signal list that can be communicated.

List of PLC input signals

Signal lines of the PLC in tabular form. In *SafePLC²* the user can fix the PLC entries. The entries have unambiguous numbers, and must be assigned to the inputs of a function block.

List of PLC output signals

Signal lines of the PLC in tabular form. In ***SafePLC²***, the user can fix the PLC outputs. Like the inputs, also the outputs have an unambiguous identification number.

Instruction list (IL)

Programming language similar to Assembler that can be loaded into a central SCU block. ***SafePLC²*** serves to create a statement list on the basis of defined function blocks, their attributes and their connections.

Compilation

Compilation and verification of the function block diagram created by ***SafePLC²***, and of the parameters connected with the function block diagram.

Function block group

Classification of the function blocks according to their positioning capability in the function block diagram (input, output, logic).

Kinds of function blocks

More exact identification of a function block within a group (e. g. "Emergency Stop").

Message window

Multiline output window, embedded in a Windows toolbar. The message window displays errors, warnings and information from the program to the user. The message window can be switched on or off.

Configuration

Configuration is the general term for a monitoring program and the connected parameters for permitted deviations, or rather, minimum values or maximum values. In this connection, it should be noted that a monitoring program is always accompanied by further data, the program can refer to.

4.2. Installation

This chapter describes the installation procedure and its requirements.

4.2.1. System requirements

To install the program, the following system requirements are necessary:

Minimum system requirements:

OS: Windows XP, Windows Vista, Windows 7, Windows 8 or higher (32 Bit / 64 Bit)
Processor: Intel® Pentium® 4 or AMD Athlon™ Dual Core, 3.0 GHz or higher
Memory: 2 GB
HDD: 500 MB free memory

Recommended system requirements:

Processor: Intel® Core™ i3 or AMD Quad Core, 3.0 GHz or higher
Memory: 4 GB or more

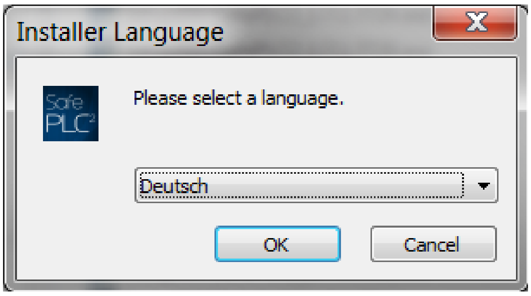
The program uses .Net framework 3.5 and 4.0, but the installation wizard also installs the program if the files are missing. For .Net installation, either local files in the component folder, or files from the internet can be used. If no internet connection is possible, the program is installed, but the user must install .Net 3.5 and 4.0.

The installation wizard installs the VC 2010 distribution data.
The installation wizard also installs the following drivers:

- Matrix-USB driver (driver for dongle)
- FTDI CDM driver (RS485 - USB) –for the connection between PC and PLC to transfer programs from SafePLC² to the PLC hardware

4.2.2. Installation process

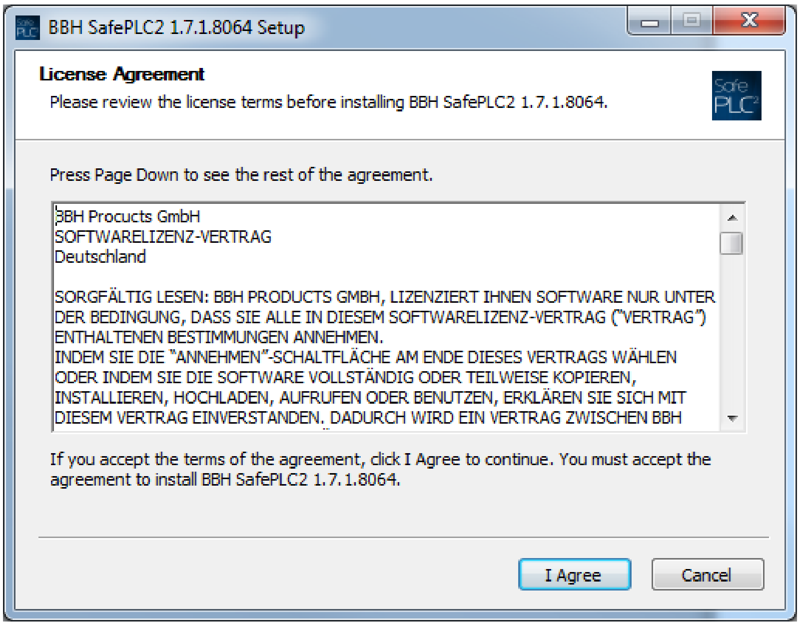
Administrator rights are necessary only for installation. Normal users can use the installed program.



Start the installation of the program by double-clicking on the file "SetupBBHSafePLC2_X.X.X.XXXX.exe" with the left mouse-key. Then, the following window appears:
By opening the menu, you can choose the installation language (English or Deutsch [German]).

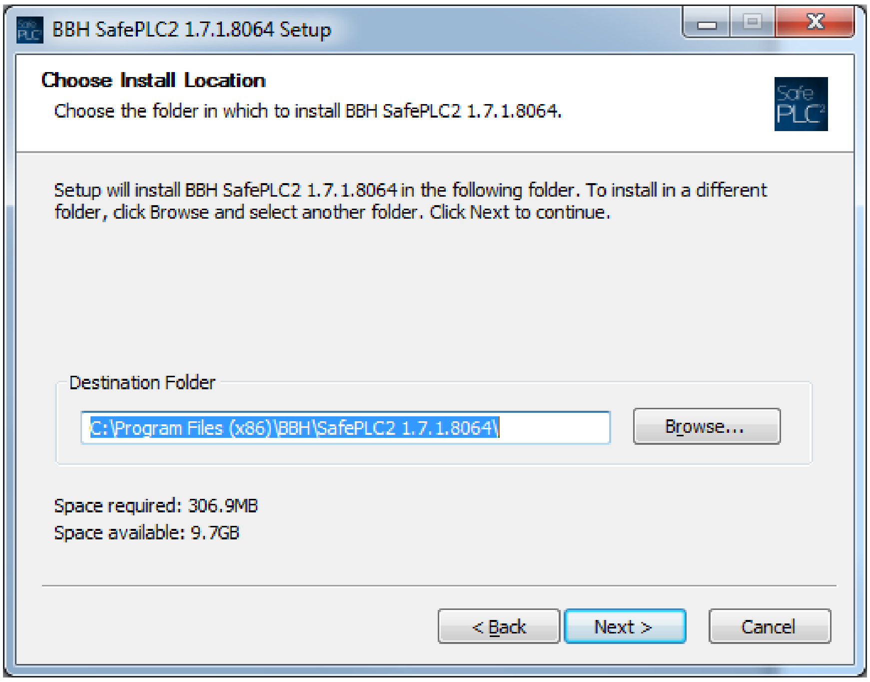
NOTICE:
This window only appears during the first installation. For the next time, the language is stored, and the first window to appear is the window with the license agreement. In the window of the installation language, only the language for the installation is set, not the language of the user interface of ***SafePLC²***.

After selecting the language, click on "OK", to continue the installation. When you click on "Cancel", the installation is completed, without installing the program. When you click on "OK", the next window with the license agreement appears.

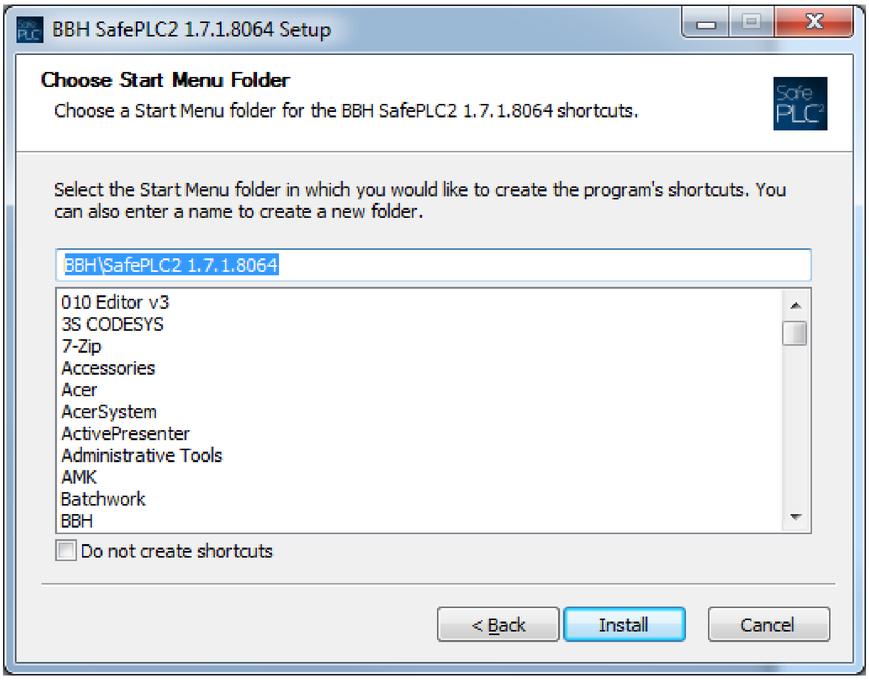


To continue the installation, click on "I Agree ".

If you do not agree to the license agreement, click on "Cancel". The installation is completed without installing the program.
After you have clicked on "I Agree", a window appears where you can select the target folder where the program shall be installed.

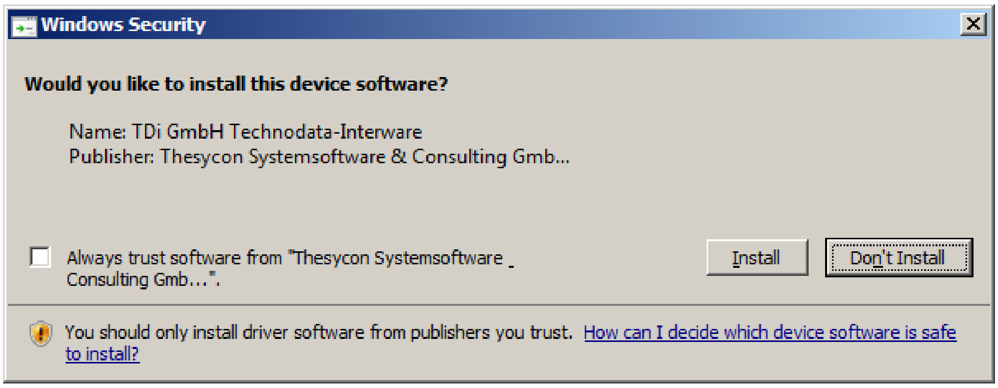


When you click on "Next", a window appears where you can select the start folder for quick access to the BBH SafePLC² program. It is also possible to create a quick access in the start menu. If you choose this possibility, a symbol to start the program appears on the computer's desktop.



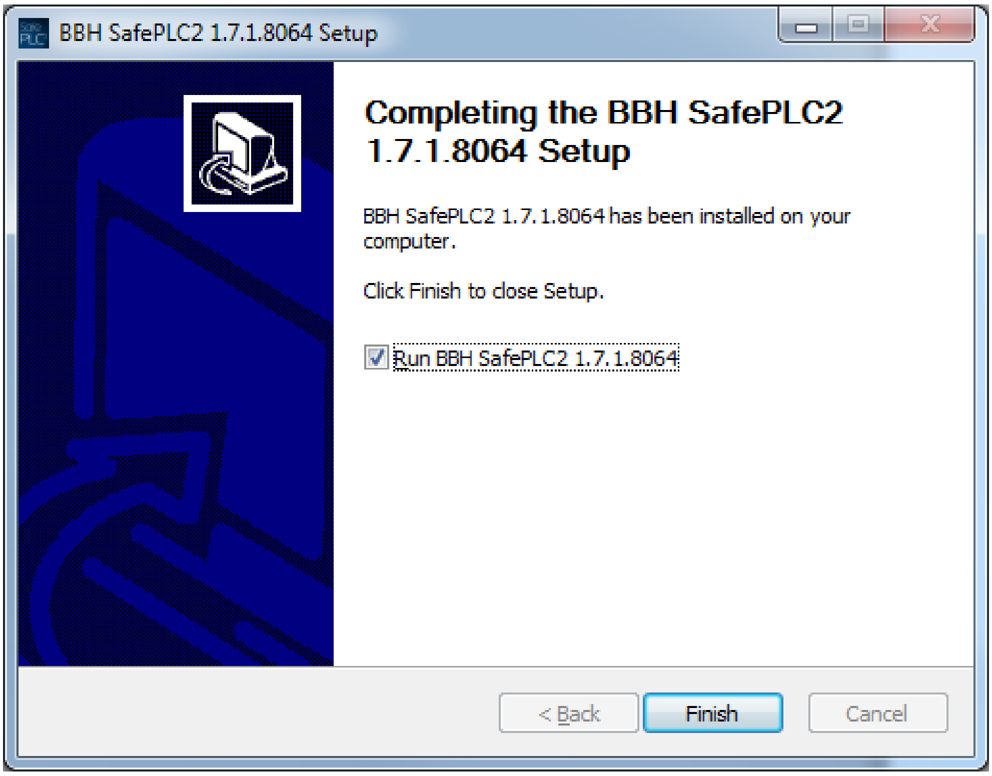
After you have klicked on "Install", the installation starts.

When a Windows Security window appears during installation, click on "Install", to install the Matrix-USB driver (Dongle driver) and the FTDI CDM driver (RS485 - USB) fort he connection between PC and PLC to transfer programmes from **SafePLC²** to the PLC hardware.



Tip: To accelerate the installation of the USB drivers, click on "Do not look for Windows updates"during the installation of the USB drivers

After the installation is completed, the following window appears:

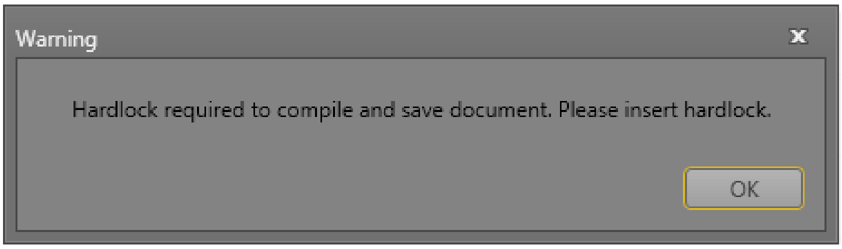


By placing the check mark in the box, you can choose if the program shall be started immediately or later. If the check mark is placed, after you have clicked on "Finish", the dialogue window closes after you have clicked on it, and the program starts.

During installation, the desktop symbol to start the program is created. With this symbol, you can always start the program. If during installation, a quick access has been created in the start menu folder, the program can also be started in this folder via quick access.

4.2.3. Hardlock

For the correct functioning of **SafePLC^2**, you need a hardlock. If you start the program without the hardlock, the following message appears:



Click on "OK ", and insert the hardlock into the USB port. The hardlock is recognized, and all functions of **SafePLC^2** can be used. If you remove the hardlock while working with **SafePLC^2**, functions get lost, and you can neither compile nor store created programs. When you reconnect the hardlock to the USB port, all functions are available again.

4.2.4. Uninstalling

To uninstall **SafePLC^2**, you can either use the quick access in the start menu or the function "Uninstall a program" on the Windows Control Panel.
If you want to reinstall the program and at the same time change the installation language, you must delete the registration key "Installation language" under the branching HKEY_CURRENT_USER\Software\BBH\SafePLC2.

4.2.5. Running Application

To start the program, doubleclick on the desktop, or start from the start menu.

INFORMATION:
if your setup file is market "user registration" after the start, "Login dialog" appears, and you can only use the program, after you have entered both user name and password.

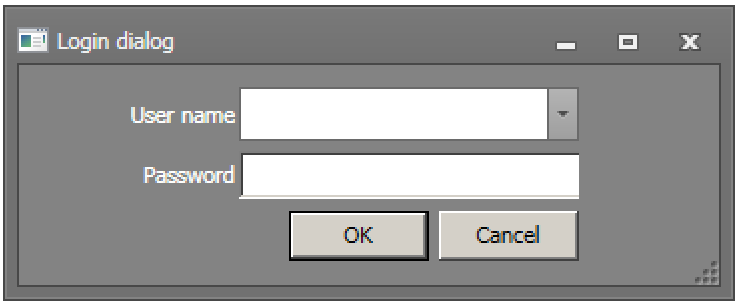


Figure 3 user registration

4.3. User interface

4.3.1. Main window

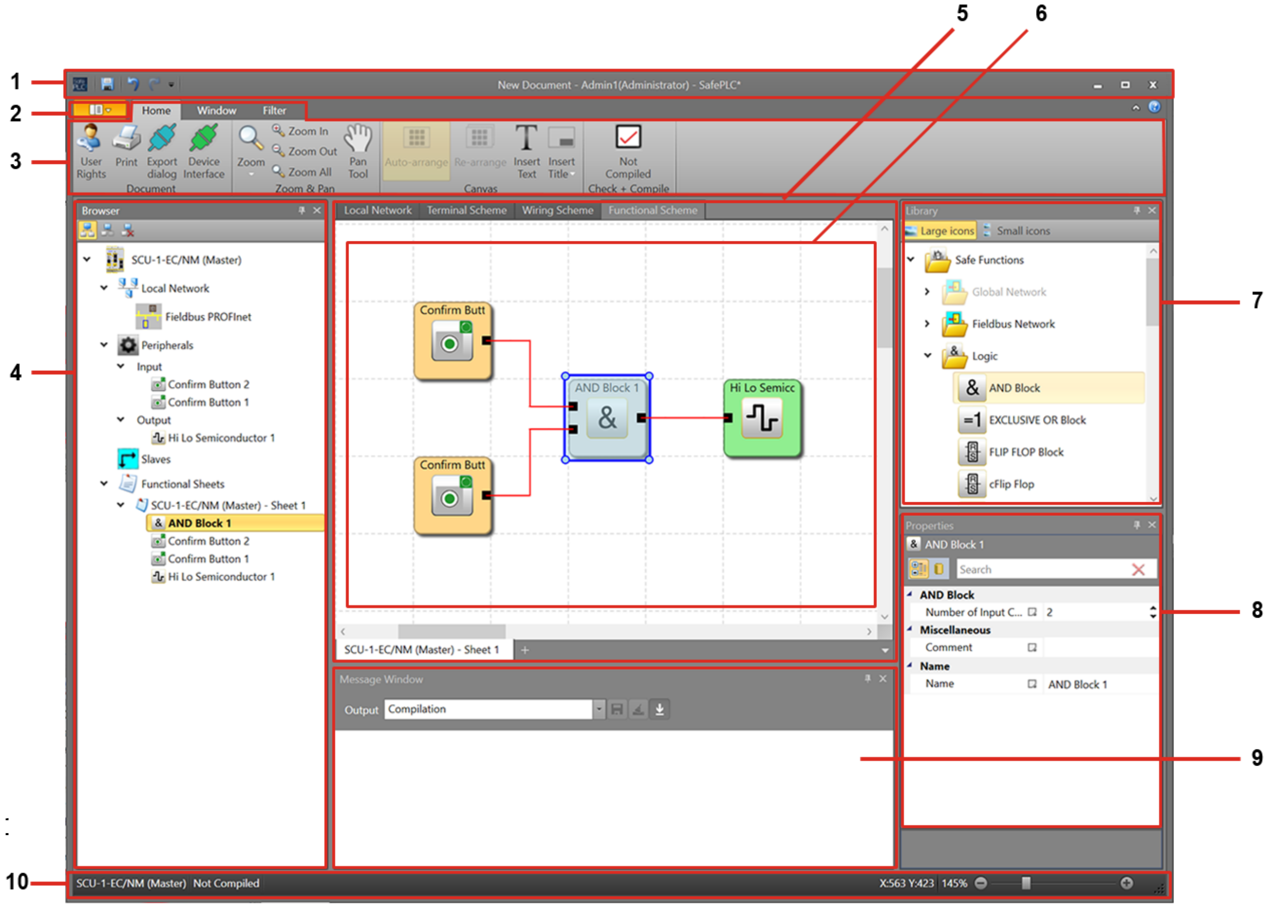


Figure 4 structure of the application window

The application window is the standard window of SafePLC². Via the window function surface, the window's size can be adapted, and the window can be minimized, maximized or closed. The window is divided in the following elements:

1. Title bar with tool bar for quick access
2. Main menu of SafePLC² surface
3. Ribbon menu (tabs: Home, Window, Filter with groups)
4. Browser
5. Control via document tabs with **plan tabs** (top) and **sheet-tabs** (bottom)
6. Work surface (Canvas)
7. Library (library window)
8. Properties (properties window)
9. Message window
10. Status bar

4.3.2. Customizing the main window

4.3.2.1. Layout reset

You can reset the layout of the application to default by clicking "Reset layout" in the register tab "Window".

Note, that this command deletes the user layout. The deleted user layout cannot be restored.

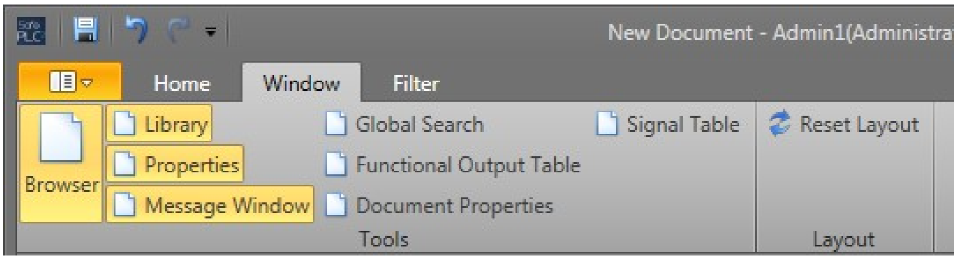


Figure 5 "window" tab, setting of the user layout via group "tools" and "Reset layout"

4.3.2.2. Docking

By docking, the application's layout can be personalized. Every section (except "Plans" and "Sheets" as well document properties) can be pulled out of the application window, and can be dropped in a different section, or in a register group.

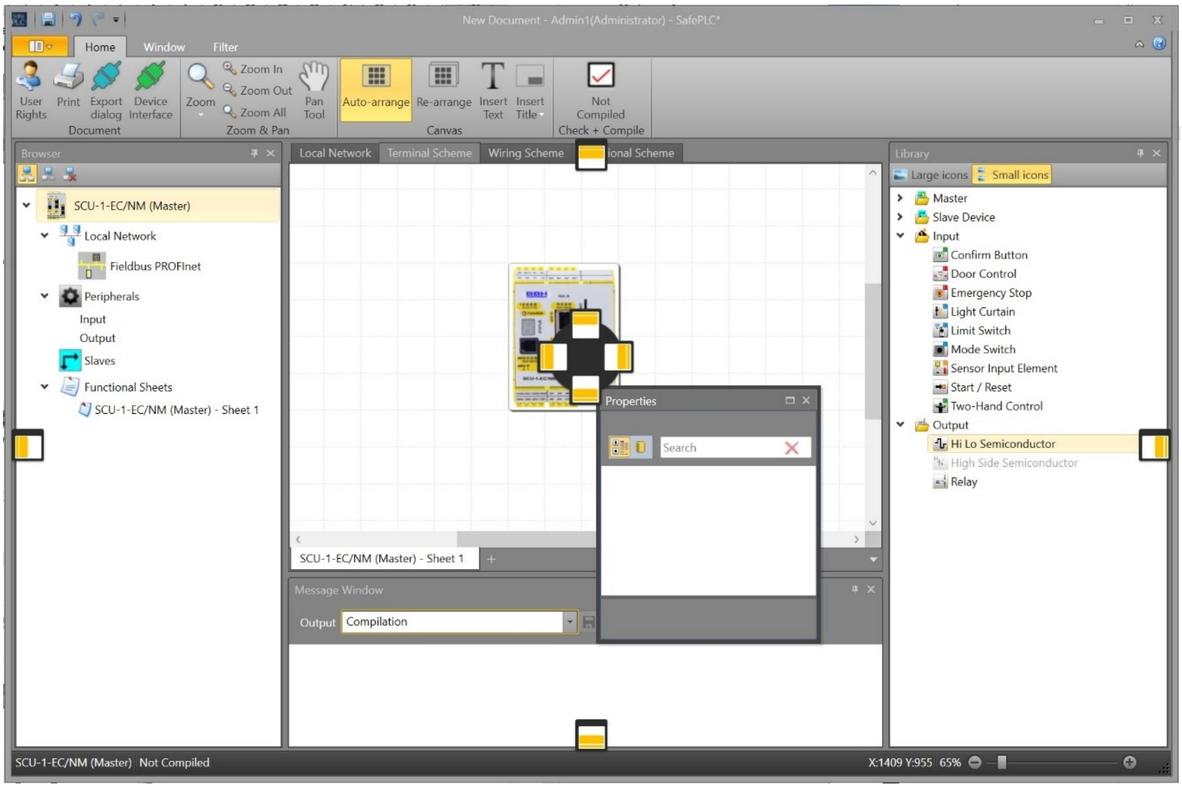


Figure 6 Personalize the application window by docking

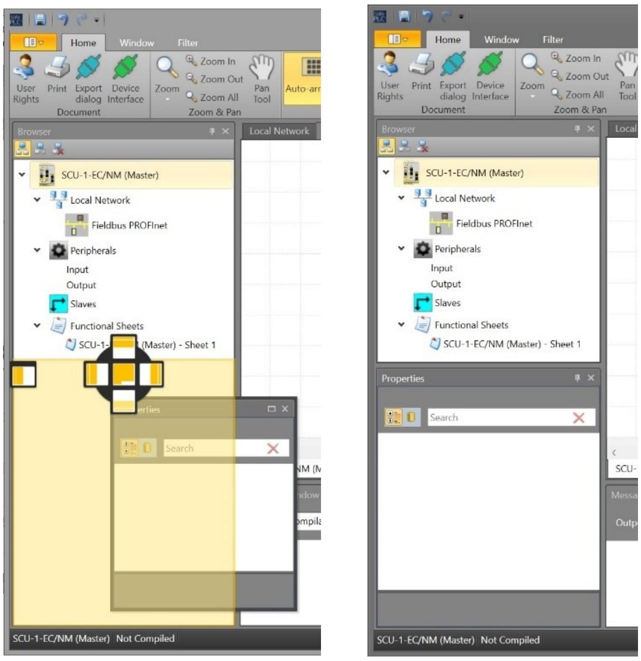


Figure 7 Docking, example: properties (properties window) on left side under "browser"

4.3.2.3. Automatic hide

Every section containing the symbol for automatic hide can be transferred to the background automatically. The user can switch off automatic hide, and can make the section visible again by again clicking on the symbol for automatic hide .

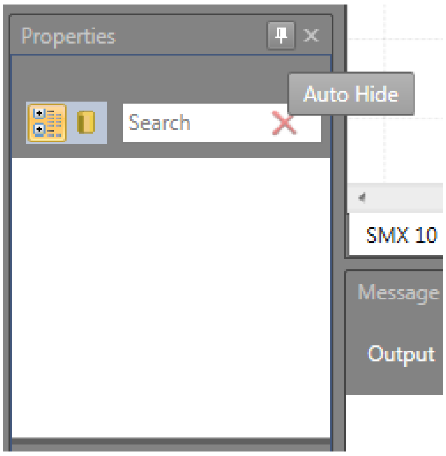
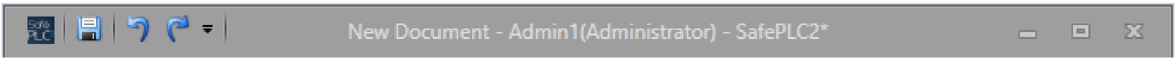


Figure 8 Automatic hide

4.3.3. Title bar



The title bar in the upper section of the application contains application icons, buttons for quick access, and the title of the application. The buttons for quick access are "Save"  (via keyboard with Ctrl+S),"Cancel"  (Strg+Z), and "Repeat"  (Strg+Y).

The application title 

consists of the current document name, the currently registered user and the application name with an asterisk, which indicates that at least one change has been effected that has not been saved.

By clicking on the application menu, the context menu of the application window can be called. The context menu contains the familiar window functions.

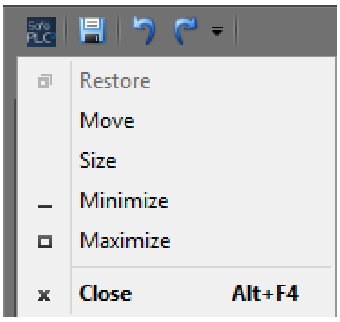


Figure 9 context menu of appliaction window

4.3.4. Main menu

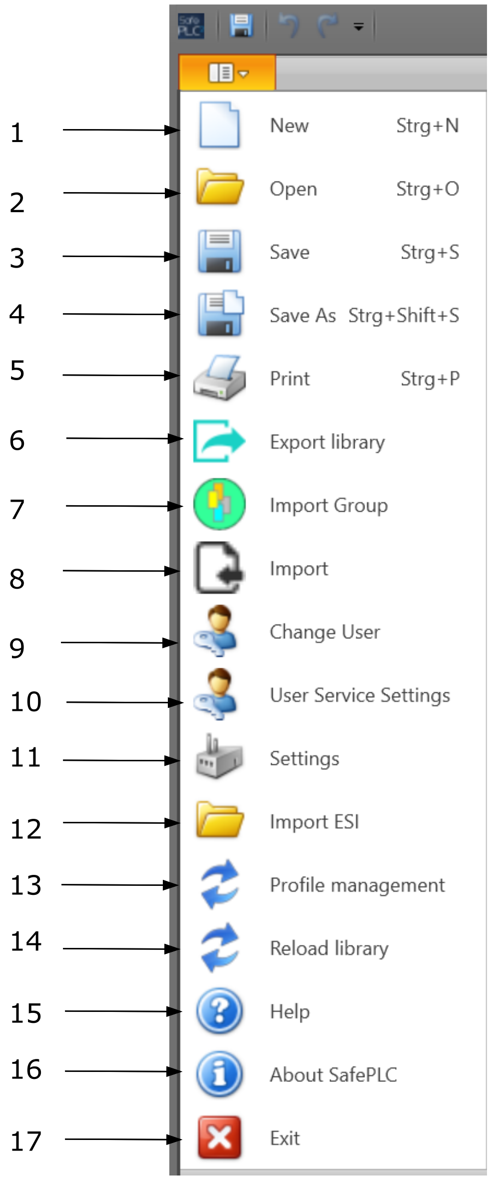


Figure 10 SafePLC2, main menu: basic document functions and application functions

Via the start menu, basic document functions and application functions can be called, e. new document, save document, print, user management, settings etc.
„Recent files" shows the documents used last (The last document is shown first).

1 New
Creates a new empty project. When a new project is opened, the program asks the user, if the changes in the old document shall be saved.

2 Open
Opens either an existing SafePLC^2 document or a document with a complete library. If a new document is opened, the program asks the user if the changes in the old document shall be saved.

3 Save
Saves the document at the selected place. In case of a new project, a window with extended storage options “Save as”.

4 Save as
Saves a document, and selects the document’s name, type and storage location. If the folder contains a document with the same parameters, the program asks the user if the document shall be replaced.

5 Print
Shows the printing options. You find the functional description under “4.3.16 Print”.

6 Export library
Export of a SafePLC^2 library to a *.splib file.

7 Group import
Group import function
The import includes the verification of the sensor configuration and the already existing resources in the function plan.

8 Import
Import of a SafePLC^2 libray from a *.pak file

9 Change user
This command appears after setting the check mark in the user service setting.
With this command a user can log in and log out.
Included are the users: administrator, programmer and user.
****** is not indicated separately. ******

10 User Service Settings
Here, users can be changed. It appears then additionally in the main menu "Change user”.

11 Settings
In the “Settings” window, application settings can be changed (cf. chapter “4.3.17 Settings”).

12 Import ESI
Import function to importing a device description file for one of FSoE slave modules.

13 Profile management
Management of the data profiles of the FSoE slave modules.

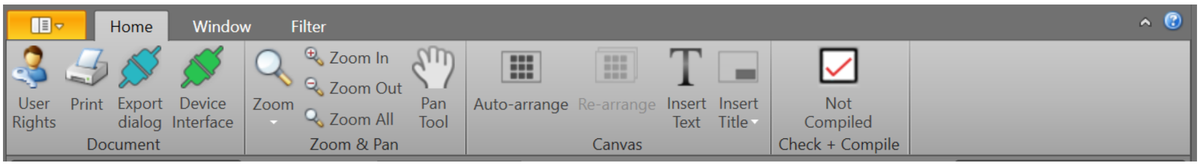
14 Reloaded Library
Libraries update. After importing a library, it must be reloaded.

15 Help
Opens the help window of SafePLC².

16 About SafePLC²
"About SafePLC²" gives short information about the Windows system and about the creation of the application. Also, compiling information is shown.

17 Quit
Closes the whole program.

4.3.5. Menu of the ribbon



The menu of the ribbon is part of the main window and consists of several register tabs.

The user can switch within the ribbon mode (minimize, maximize) by clicking on the  button in the menu's upper right-hand corner. If the menu ribbon was minimized, the user must click on the corresponding tab. This tab opens automatically, and closes again if it is no longer clicked on. The user can also call the tabs by pressing the Alt-button and the QuickInfo-button shown on the menu ribbon.

Afterwards, again QuickInfo buttons appear beside every function in the menu of the ribbon.

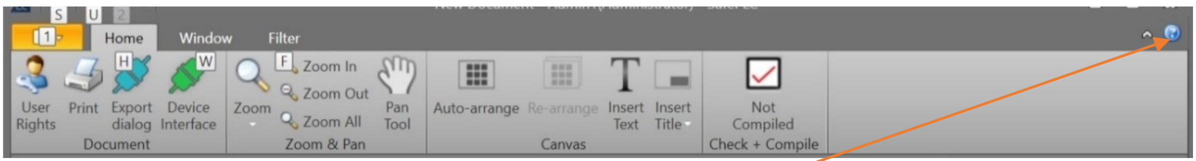
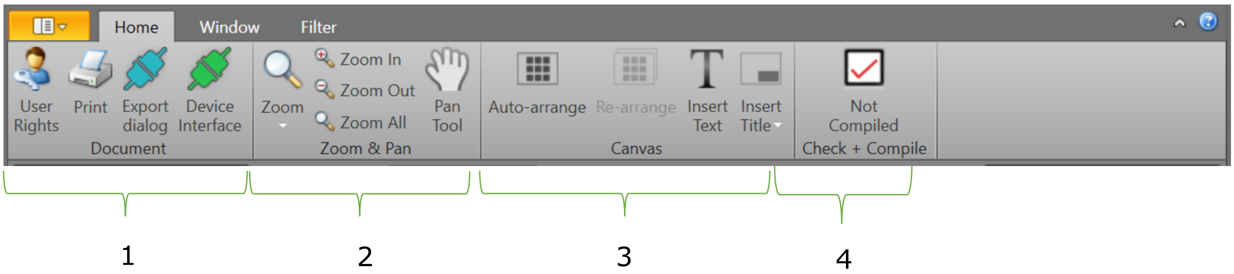


Figure 11 menu of the ribbon, quick info buttons

Beside the switching mode, the "Help"  button is situated. With the "Help" button, the help dialogue window is called.

4.3.5.1. Home



1 Document

- "**User rights**" shows a window where the user rights can be fixed.
- "**Print"** shows the printer menu.
- "**Export dialogue"** shows the export dialogue.
- "**Device interface**" shows the device dialogue.

2 Zoom and pan

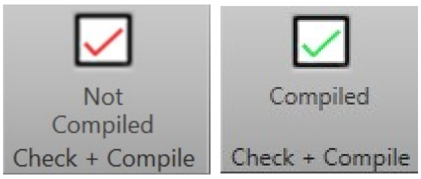
- The dropdown list "**Zoom**"gives you access to defined zoom values.
- The "**Zoom In**"**[Enlarge]** button increases the current zoom by 25 %.
- The "**Zoom Out**"**[Reduce]** button decreases the current zoom by 25 %.
- The "**Zoom all**" **[Whole page]** button can maximize the range to the page size.
- The "**Pan Tool**" serves to switch to the panning mode. When the function is enabled, the user can pan the range either with the left mouse button or with the middle mouse button.

3 Work surface (Canvas)

- The switch button "**Auto-arrange**"**[Arrange automatically]** serves to switch within the mode for automatic arrangement. If the function is enabled, elements are arranged automatically. Not all plans support automatic arrangement.
- The "**Re-arrange** " button serves to rearrange elements immediately. This arrangement is not saved. Not all plans support this automatic arrangement.
- The "**Insert text**" **button** serves to activate the filing of text.When the function is enabled, the user can file text on the working surface by clicking the left mouse button.
- The dropdown menu "**Insert title**" lists all available titles. By clicking on the selected title, the selected title is immediately inserted in all sheets of the whole document.

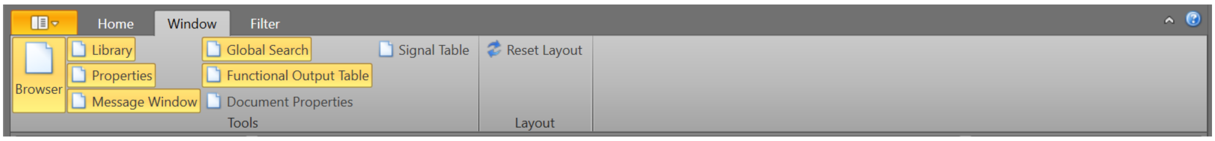
4 Check + compile

- Pressing the “**Compile**” **button** compiles the current document. After the compilation has been successfully completed, a green hook appears in the text box “**Compiled** “ . Otherwise,
  “**Not compiled**“ is displayed together with a red hook.



- With the “**block**“ switch button, documents can be blocked or released. If a document is blocked, the user cannot edit the document. However, the user can select elements and cange between plans or sheets.

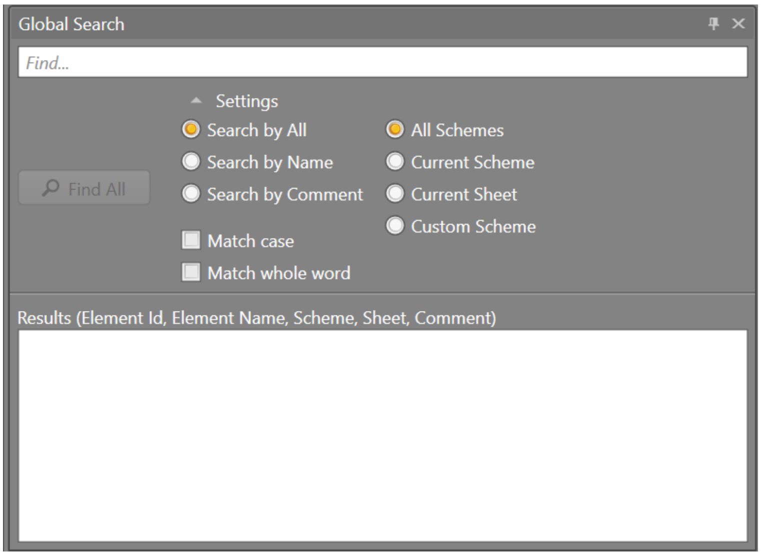
4.3.5.2. Window



Via the corresponding switch button, the user can show or hide windows.

- The “**Browser**“ switch button, serves to switch on or off the browser window on the user interface.
- The “**Library** “ switch button serves to switch on or off the library window on the user interface.
- The “**Properties** “ switch button serves to switch on or off the Properties window.
- The “**Message window**” switch button serves to switch on or off the Message window.

- The "**Global search** " switch button serves to enable the window for global search. For further information concerning this function, cf. chapter 4.3.15.



- The "**Functional Output Table**" displays the following table (cf. illustration).

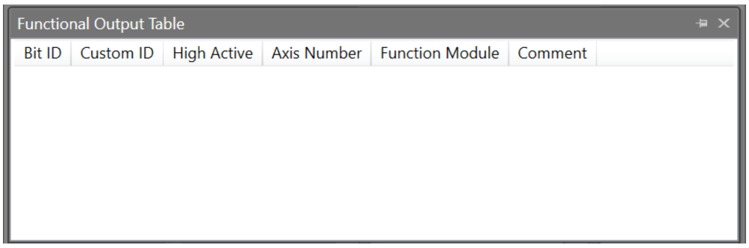


Figure 12 Functional Output table

- The "**Document properties**" switch button serves to display the window for document administration.

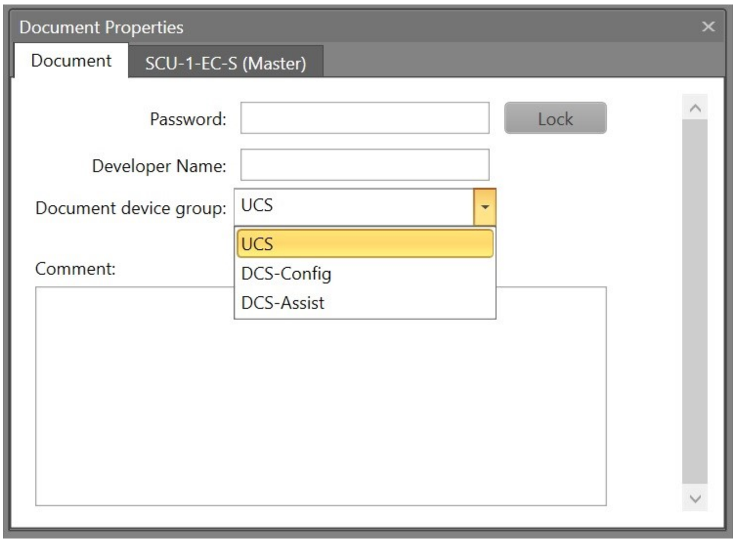


Figure 13 Window "Document properties"

The window contains the document tabs and the device tabs.

The document tab consists of:

- Password – With this password the project is locked and can be unlocked again
- Lock – can be locked and unlocked. If a document is locked, the user cannot edit the document. However, the user can select elements and switch between plans or sheets.
- Developer's name – name of the responsible programmer / developer.
- Document device group – The device group included in the programming. and can be selected here if several groups are available for selection.
- Comment – Via this input field, descriptive text can be entered. For example, this input field serves to enter program or parameter changes during the operating period of the currently used device.

The Master device tab, e.g. SCU-1-EC (Master), consist of information fields and connection settings.

- Device information – For further information concerning the edited fields, cf. chapter 4.7 "Validation report".

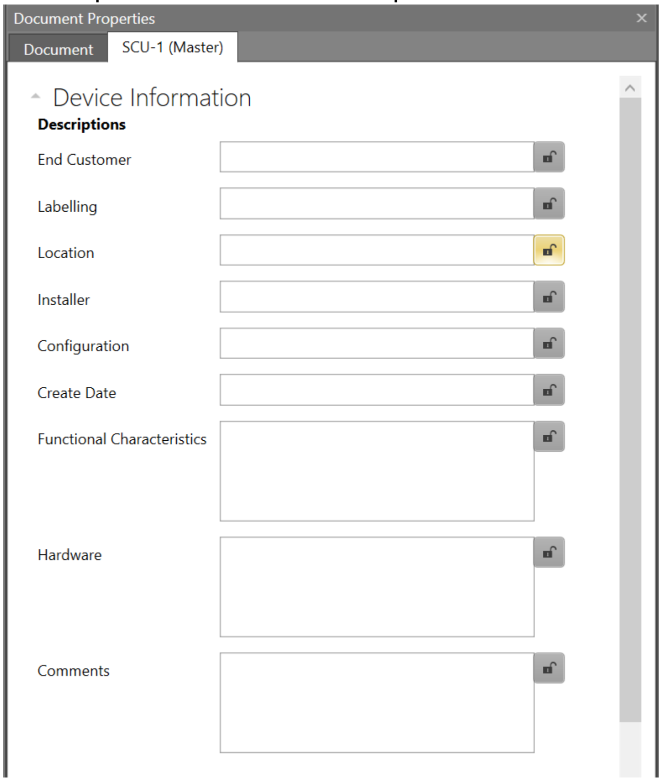


Figure 14 Window "Document properties" Device informations

- Connection settings – For further information, cf. chapter 4.6.10 "Transferring the program to the device"

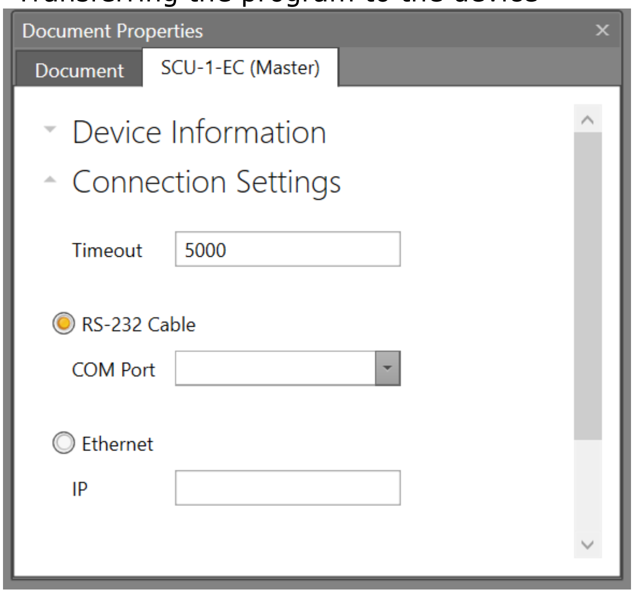


Figure 15 Window "Document properties" Connection settings

- Report settings – Select the format (PDF or Excel) for the report to be produced.

- The following window is shown or hidden with the "**Signal table**":

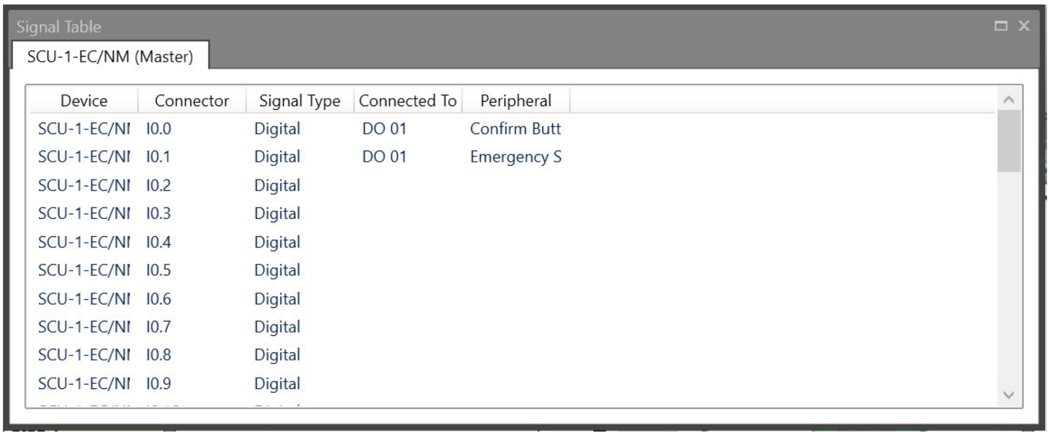
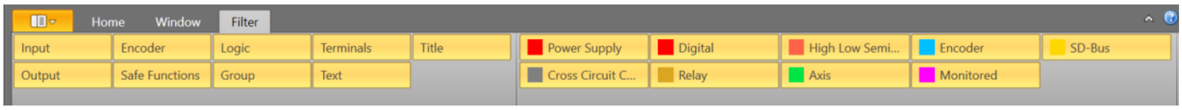


Figure 16 window "signal table"

- With the switch button “**Reset layout**”, the application’s layout is rest to default.

4.3.5.3. Filters



Via the filter settings the legibility of the work surface can be granted by hiding the desired element types or connection types. It is possible to select from two filter categories: elements and connection. Every group contains several filters. If the function is disabled, the filter elements (or the connection) of a certain filter are no longer displayed on the work surface.

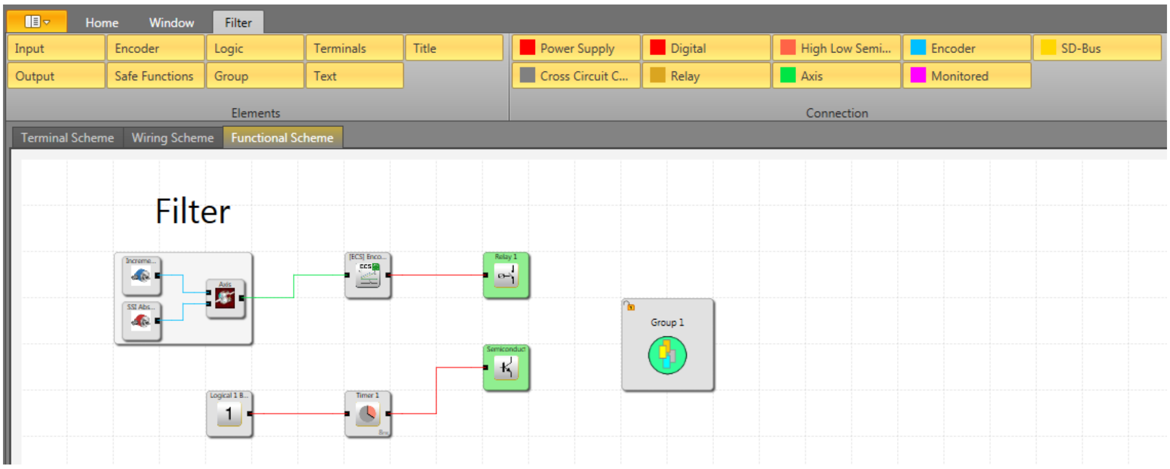


Figure 17 view functional scheme, all filters activated

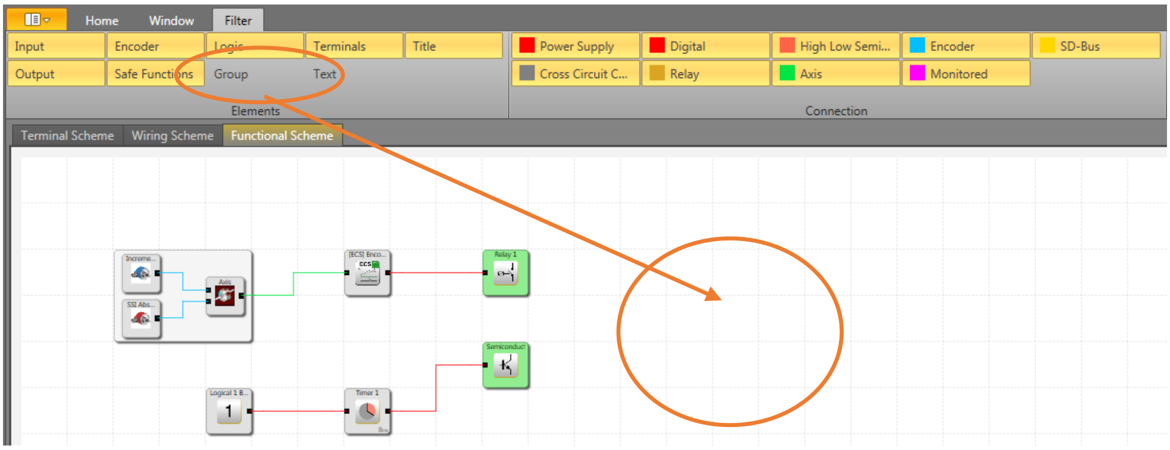
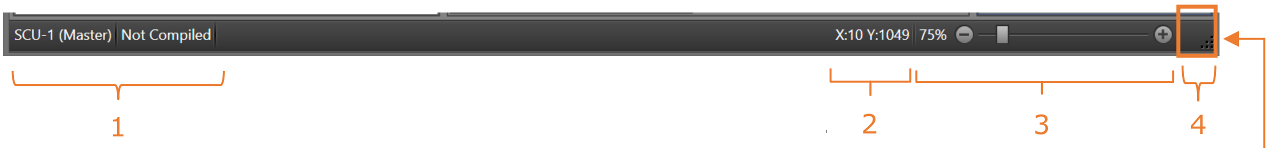


Figure 18 Functional scheme, deactivation of filter "group" and "Text"

4.3.6. Status bar

The status bar is situated in the main window. On the left side it contains the Master device’s name and the action mode, and on the right side it contains the cursor position on the work surface, the zoom control for the work surface and the symbol for size change. The symbol for size change indicates that the user can change the size of the main window. The action mode displays the current action the user performs on the work surface.



1. Name of master device, edit mode: compiled/not compiled
2. Cursor position
3. Zoom controller
4. Icon to change the window size

4.3.7. Mouse and keyboard commands

4.3.7.1. Mouse-independent actions

Left mouse click on a function block: selection / deselection of the corresponding block.

NOTICE:

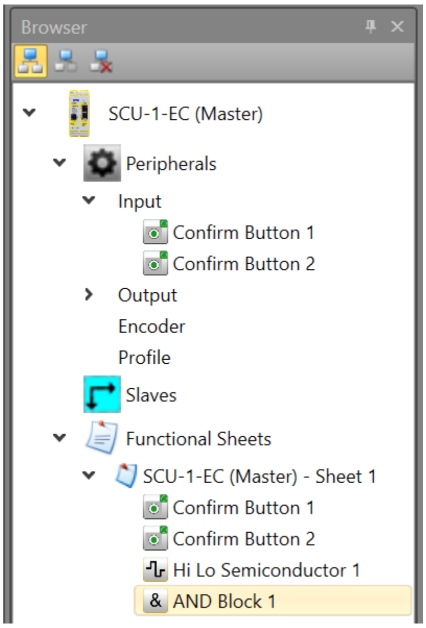
Several selections can be made either by pressing the Shift – button (adding a block to the selection) or via the Ctrl-button (removing of a block from the selection).

- Cursor above the block or the connection: emphasis of the block or of the connection
- “Shift“ button + left mouse button on a function block: adds a block to the selection
- Ctrl + left mouse button on a function block: removes a block from the selection
- Delete button: deletes the elements of the current selection, including their connections
- Right mouse button on object: display of context menu
- Left mouse button on connection: emphasis of the current connecting line.
- Scrolling with the mouse wheel. Scroll up /down through the work surface
- Move middle mouse button and mouse: panning of work surface
- ”Shift” button + scrolling with mouse wheel: scroll to the left / to the right throught the work surfaces
- Ctrl + scrolling with mouse wheel: dynamic zooming in the work surface
- Hold left mouse button and move cursor: moving an element on the work surface

4.3.7.2. Keyboard commands

- Ctrl + N: New document
- Ctrl + O: Open a document
- Ctrl + S: Save document
- Ctrl + Shift button + S: Save document as
- Strg + P: Print
- Ctrl + R: Open documents used most recently
- Ctrl + Z: Undo
- Ctrl + A: Select all
- Ctrl + Del: Delete
- Ctrl + C: Copy selected elements
- Ctrl + X: Cut selected elements
- Ctrl + V: Paste selected elements
- Esc: Cancel
- Backspace key: Remove previous connection point while drawing a connection
- Ctrl + F: Calling of the global search
- Ctrl + F: Display of the search elements (only if message window is active)
- "Shift" key + F11: Creation of a new sheet
- Ctrl + tab key: Shifting between plans
- F1: Display help concerning SafePLC² (In the individual windows, the help opens in the corresponding chapter).

4.3.8. Browser (library window)



The browser section gives a survey of the whole document. Devices, elements and function sheets are displayed as nodes in a tree structure. The library section automatically adjusts its content to the element currently displayed in the browser. Every node in the browser can be faded in or hidden. The user can rename every node – either by a double click on the node or via the context menu. Several selections can be made by holding the Ctrl button or the "Shift" button. The selection in the browser is adjusted to the selection on the work surface.
If a function plan has been selected, and if inputs, outputs or sensor elements are inserted in the function plan, these elements are marked with a green arrow. The green arrow indicates that these elements can be drawn into the function plan and that they can be inserted there.

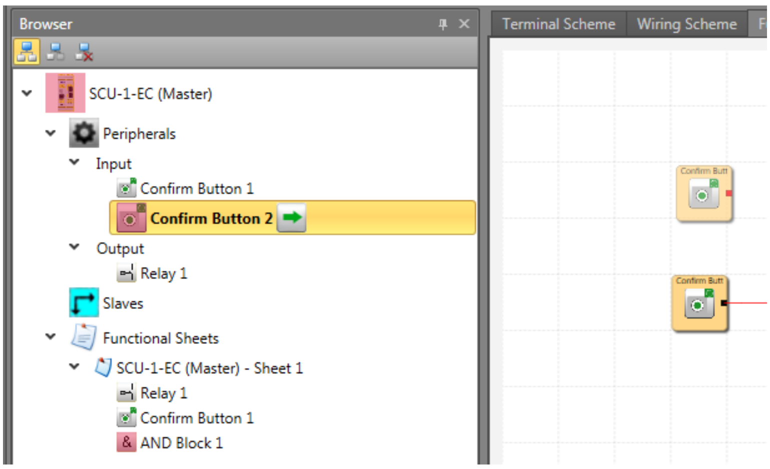


Figure 19 Inserting an output element in the functional scheme using the example of "B confirmation button1".

By pulling the elements and dropping them at the desired place, the user can change the sequence of inferior elements within a superior node.

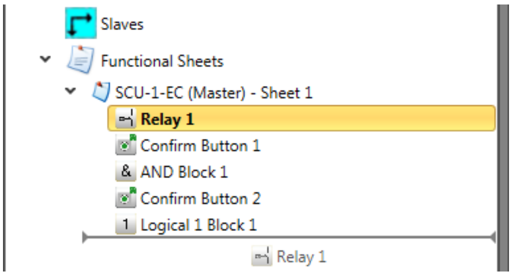


Figure 20 Moving an element within a parent node by dragging and dropping using the "Relay 1" example

4.3.9. Control via document tabs

SafePLC² contains the following plans:

Plan tabs:

- Global network
- Local network
- Terminal scheme
- Wiring scheme
- Functional scheme
- FastChannel
- Groups

Sheet tabs:

- One sheet for each device

The plans and sheets are represented by individual tabs in the document window. By navigating via the document tab, the user can switch between plans and sheets. Plan registers are situated at the top of the window, and sheets are situated at the bottom of the window. Please observe that by default only the register tabs, the terminal scheme, the wiring scheme and the functional scheme are visible. The other tabs (networks, groups, SD-bus-groups) are displayed only under particular circumstances. Every sheet belongs to a device. If a Slave device is selected in the browser, only the plans and the sheets of this Slave device are visible. To display the sheets of other devices, the user must select the devices in the browser.

NOTICE:
With the shortcut "Ctrl + tab key" you can switch between the plans.

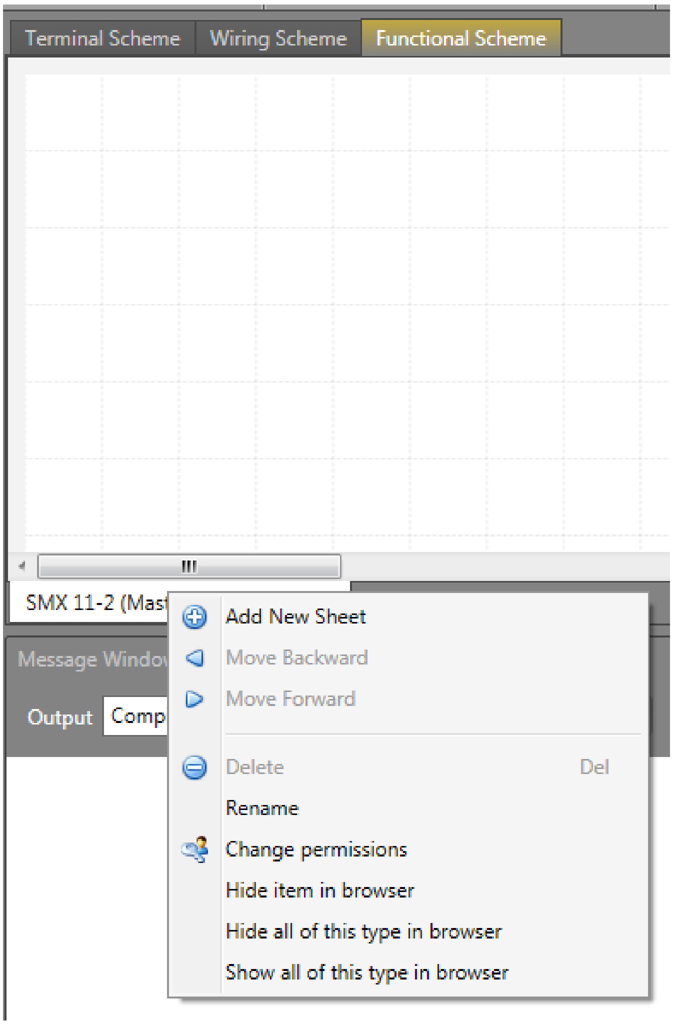
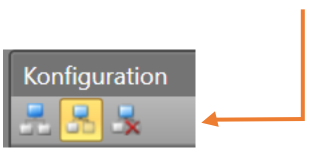


Figure 21 Context menu sheet

- **Add new sheet -** adds a new sheet to the current plan.
- **Move Backward [Shift to the left]-** shifts the current sheet one position to the left.
- **Move Forward [Shift to the right] -** shifts the current sheet one position tot he right.
- **Delete -** deletes the current sheet. This command is not available if only one sheet exists.
- **Rename -** renames the current sheet.
- **Change permissions -** shows the permissions dialog.
- **Hide item in browser -** the selected elements are hidden in the tree view (view filter must be selected in the browser)



- **Hide/show all of this type in browser -** If this option is selected, the upper menu of the configuration can hide, show or reset these elements in configuration

4.3.10. Plan types

In the document window, the plans and sheets are presented by individual tabs.

4.3.10.1. Terminal Scheme

The section "terminal scheme" shows a simplified schema of selected devices and peripheral devices of the SCU system and also has an editor function where the user can add, remove, replace and move objects. These actions are used to edit the project.

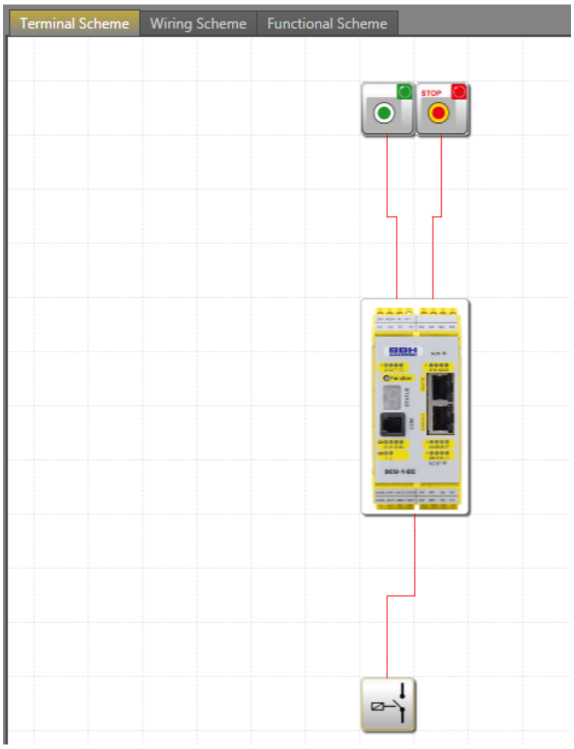


Figure 22 View "Terminal scheme"

<u>Terminal scheme</u>: If function blocks are inserted into the connetion diagram, the elements are automatically coupled with the device. If several devices are indicated in the connection diagram, the user must add peripheral devices to the corresponding device. Otherwise, the dialogue for device selection appears.

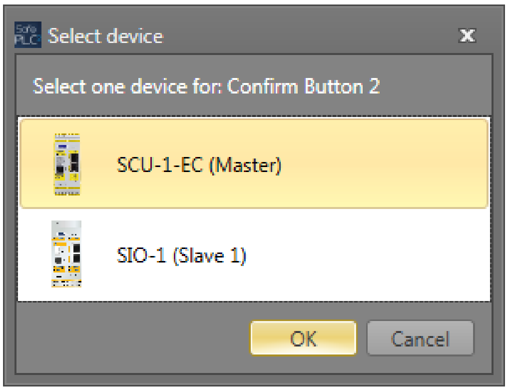


Figure 23 "Select device" window

4.3.10.2. Wiring scheme

Under "Wiring scheme" the assignments of the external connections in an SCU system to the selected sensors and actuators are shown. When a new project is created (Menu > New...), here, all possible inputs and outputs as well as additional sensor interfaces (encoder, analog sensors) are displayed.

NOTE:
Although automatic arrangement is enabled, in some cases it may happen that connections are displayed unfavourably. However, this does not influence the functioning! If the corresponding element is moved, the connection wiring is redrawn and possibly appears in a clearer way.

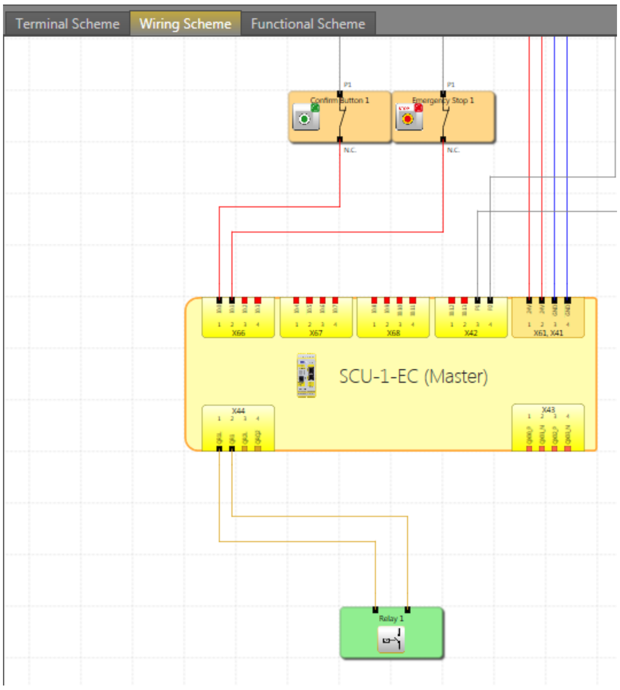


Figure 24 View "wiring scheme"

24 V: This line shows a permanent tension of 24 V DC. The SCU module requires a 24 V DC power supply.

GND: This line shows the permanently installed mass and is a (relatively) constant potential that can serve as a reference for other potentials.

T1/T2: Wiring of the clock outputs T1/T2.

NOTICE:
In this view no logical elements may be defined. The corresponding commands are available in the function plan.

4.3.10.3. Functional scheme

In the function block diagram, the connection is made between input, control, output and logical blocks.

This means that the output connectors of the input elements correspond to the input data of the function block diagram. Accordingly, the input connectors of the output elements must be seen as the output data of the function block diagram.

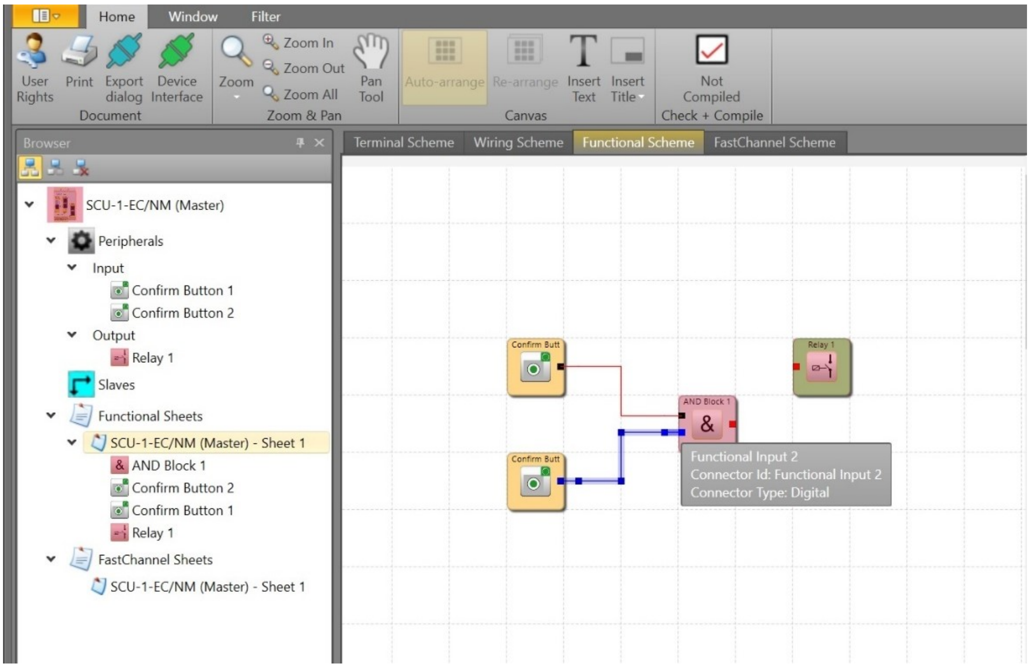


Figure 25 view Functional scheme, marked Functional Input

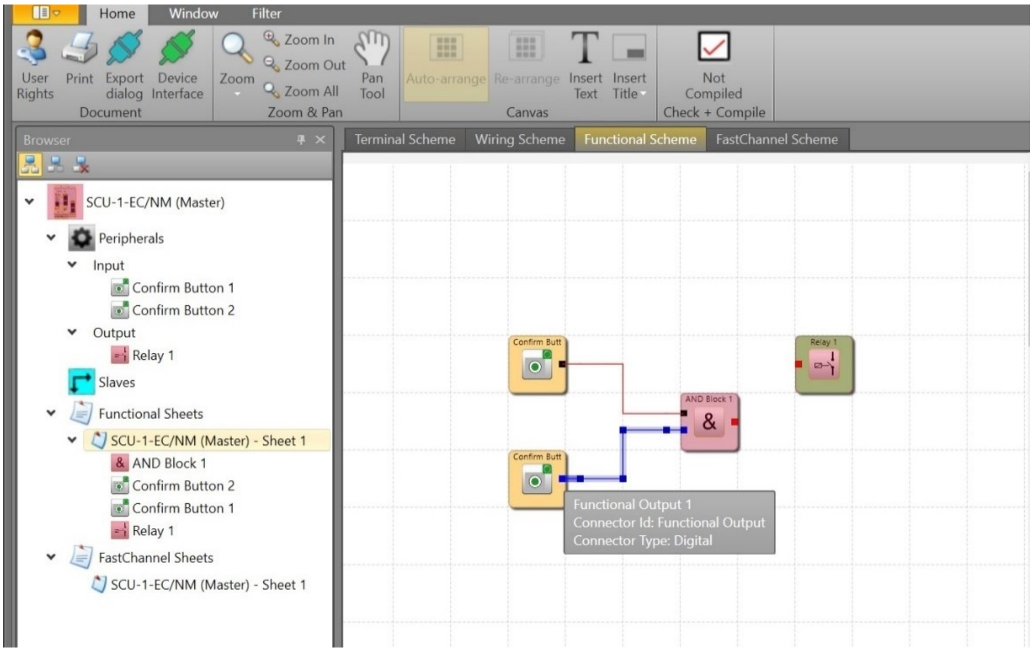


Figure 26 View functional scheme, marked Functional Output

To develop a clearly structured function block diagram, so-called conections can be defined. These constitute a determined connection between input connectors and output connectors.

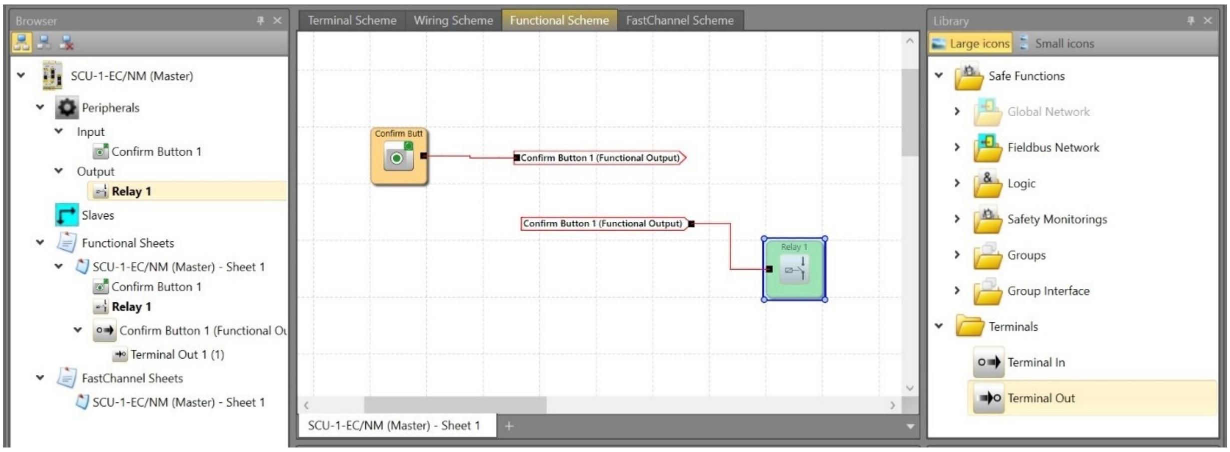


Figure 27 View functional scheme, structured layout of the function block diagram by using connectors

One or several flag output components (output terminal) can be determined for one flag setting block (input terminal) (Cf. the chapter “Connections”).

Tip: Use the comment field in the window “Eigenschaften” [Properties] to enter the connecting points. This information simplifies the use of additional connecting point outputs. This contributes to a clear arrangement.

4.3.10.3.1. Groups

“Groups‟ contains one sheet per group for every group block from the function plan. After tha group block has been created, this function is available in the function plan. For further information concerning the creation of groups cf. chapter “4.12.5.8 Groups”.

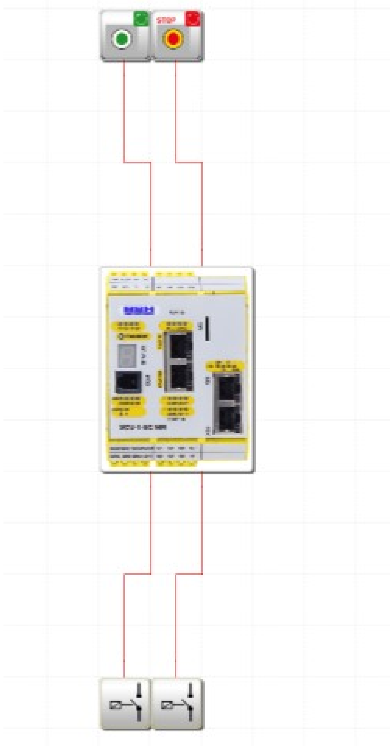
4.3.10.4. Global Network

The survey “Global Network” shows all networks. Master connections and connections with Slaves, Fieldbus groups and SD-Bus groups are shown. For further information cf. chapter “4.11 Networks”.

4.3.10.5. Local Network

The other networks are device-independent and have their own diagrams that are displayed for the corresponding device. This diagram is called “Local Network”.

4.3.11. Work surface (Canvas)



The work surface is the basic drawing tool. Every sheet is displayed in a proper work surface. Helpful tips:

- You can pan the work surface by pressing the middle mouse button, even if the panning mode is not active.
- The zoom can be changed by scrolling with the mouse wheel while pressing the Ctrl key. Using the mouse wheel and pressing the "Shift" key, the user can scroll horizontally through the work surface.
- Every work surface saves its own zoom factor.

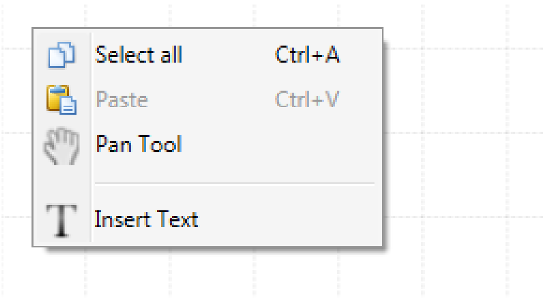
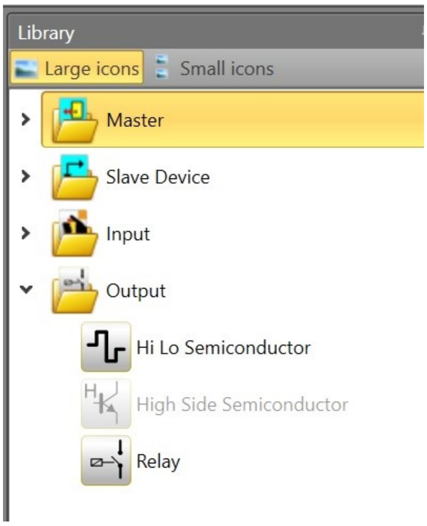


Figure 28 Context menu of the work surfaces

- With **Select all** all elements and connections or a work surface are selected.
- With **Paste** all elements and connections from the clipboard are inserted (if available).
- With the **Pan Tool**, you can switch within the Pan mode.
- With **Insert text** the text at the current position of the cursor be inserted into the work surface.

4.3.12. Library window



The window "Library" is the main tool for the user to insert elements into a document. The window "Library" consists of drop-down folders. Every folder contains one element or several elements. Folders and elements are filtered in the browser or in the work surface. With the buttons "Large icons" and "Small icons" situated at the top of the window, you can switch between the view sizes of the elements.

To insert an element into a document, the user must drag the element from the window "Library" to the work surface and drop it there.

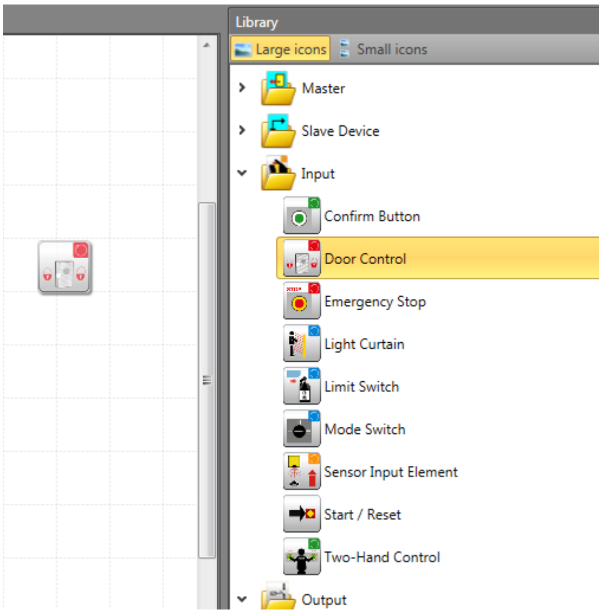
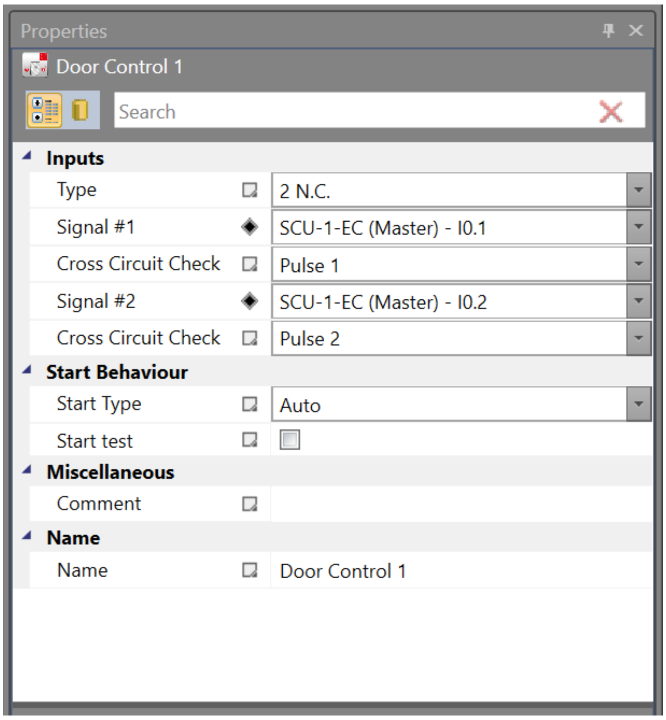


Figure 29 Dragging an item from the library window

4.3.13. Properties window



The window "Properties" serves to change the property values of elements. The content of the property window is updated automatically and shows the currently selected element. By default, the properties are grouped in categories. To reverse this setting, the user must click on the "Uncategorized" button  .

With "Search element" 

, the desired property can be found fast and easily.

If the property value is set to the default value, a white symbol appears beside the name of the property. If a value is set that does not correspond to the default value, the symbol is black. After clicking on that symbol, the user can reset the value to default.

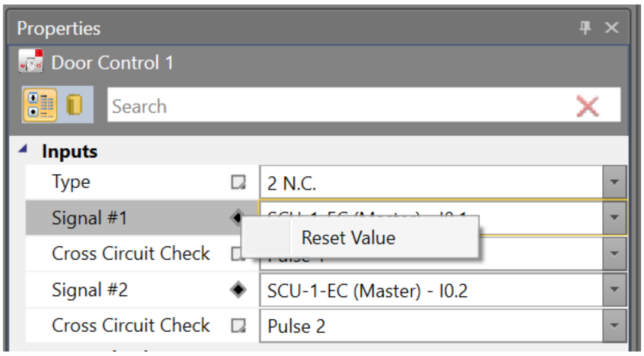


Figure 30 Properties window with context menu

In the bottom list of the window “Properties” additional information about the currently selected properties is displayed. (if available)

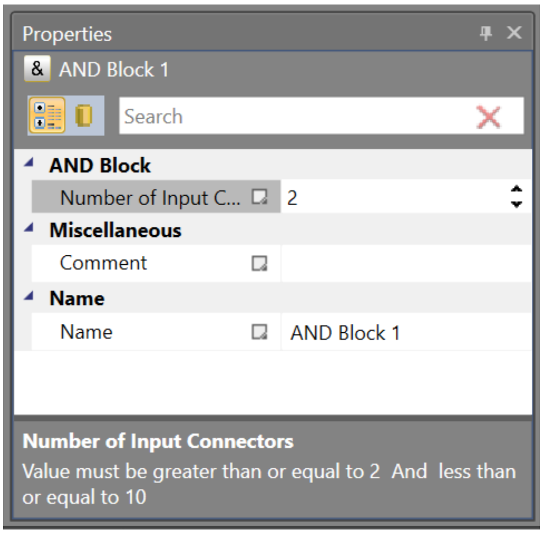


Figure 31 Properties" window with information about the currently selected property (Example “AND Block 1”)

In the window “Properties” you can show the properties of a selected object and change them. In this section different editing fields are displayed, depending on the requirements. These editing fields include input fields drop-down lists, and links to individual software dialogues. You open the window “Properties” by clicking on the “Properties” button in the “Window” tab of the menu ribbon.

Tip: The comment field can contain several lines. With the “Enter” key, you proceed to the next line.

4.3.13.1. Structure of the properties window

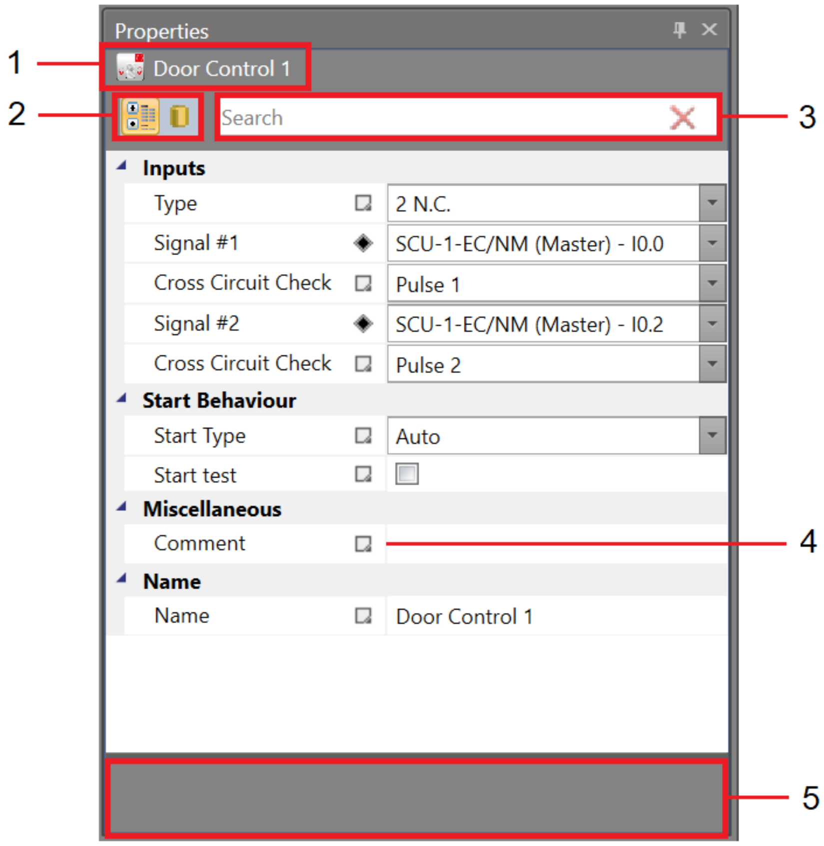


Figure 32 Strucutre Properties window

1. The name of the selected object.
2. Selection fields to change the order of the property list.
   - According to categories – lists all properties and property values fort he selected object according to the category. To reduce the number of properties you can hide categories.
3. Search field to filter properties according to the entered text.
4. Button to open the pop-up menu “Extended Properties”.
5. The description of the selected property.

4.3.13.2. "Extended Properties" menu

With "Erweiterte Optionen" [Extended Options], the user can execute property-specific commands.

- **Reset value** –The user can reset the selected property to a default value.

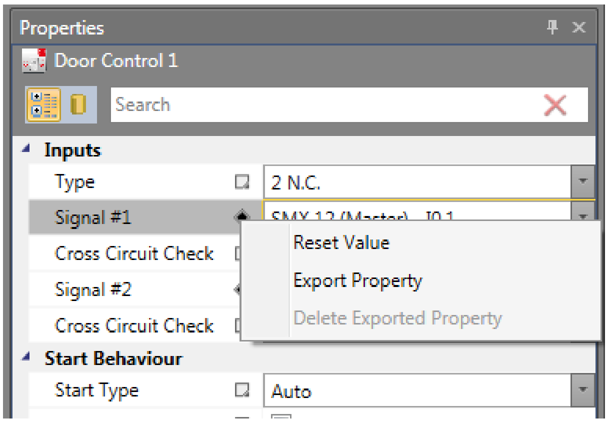


Figure 33 Property window with extended Options

4.3.13.3. Validation of properties

4.3.13.3.1. Input validation

The input validation controls if an entered value is situated within the section specified by the properties, and if this value contains invalid characters. If the value is invalid, a red rectangle is drawn around the editor.

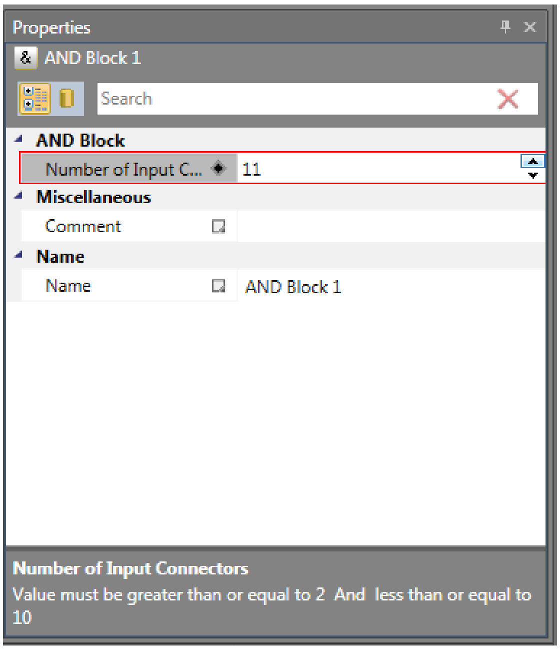


Figure 34 Value of the property Number of Input Connectors is not within the range

4.3.13.3.2. Value validation

The value validation controls if an entered value fulfills the conditions defined by other properties. If the value is invalid, the editor's background turns red.

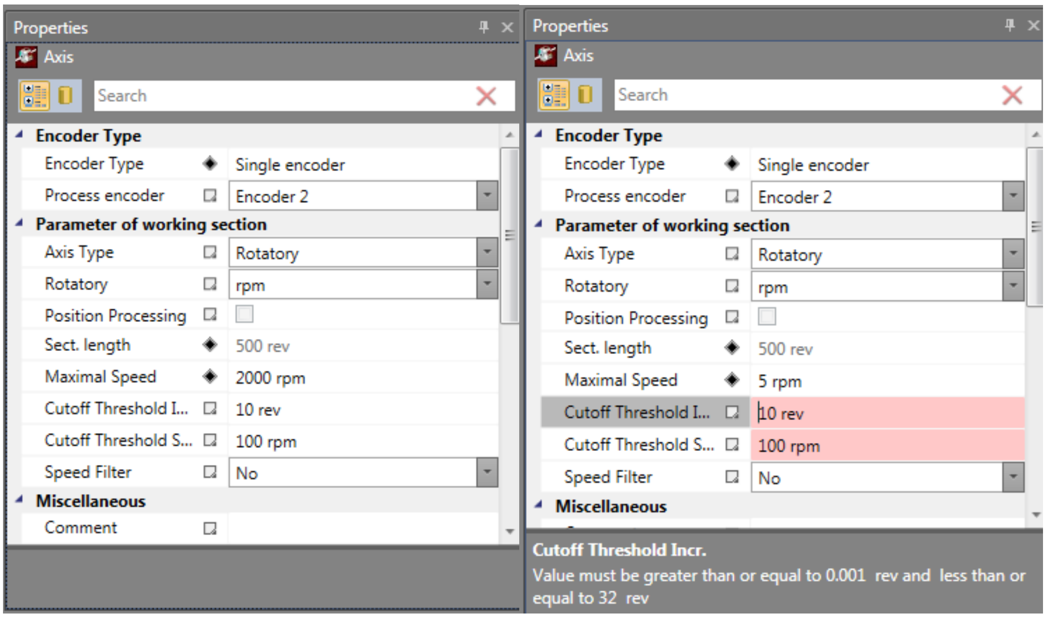


Figure 35 Example of a value validation. After the property "maximum speed " has been changed in 5, the property "Cutoff threshold" is invalid

4.3.13.3.3. Adaption

With the special validation function "Adaption ", a property value can be evaluated. If the value does not comply with the function's restrictions, the function updates the value, and a message window appears, giving the reason why the value was invalid.

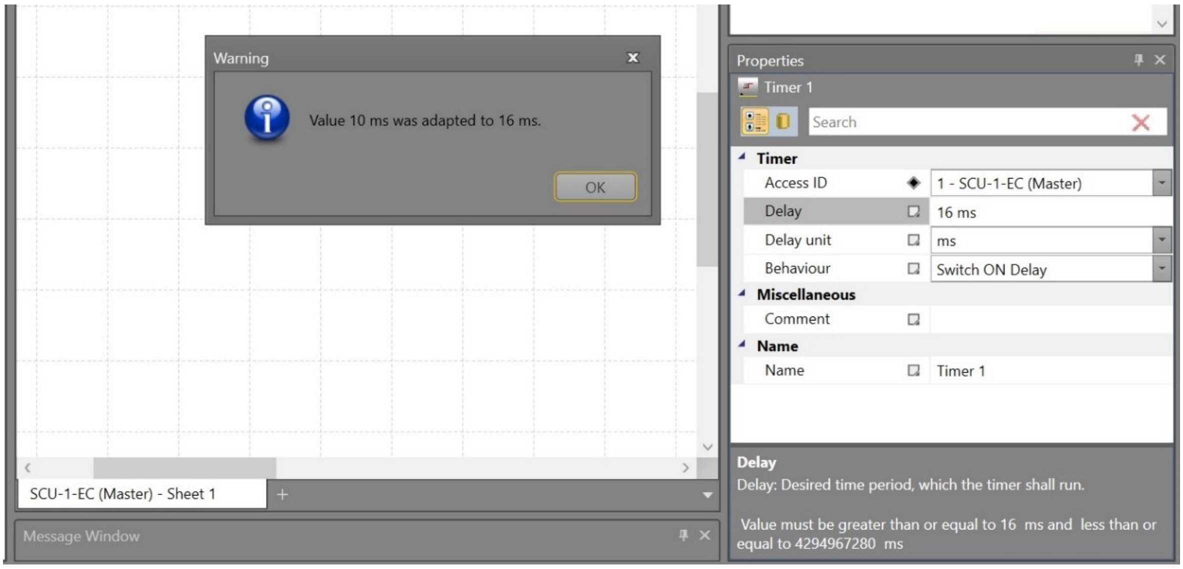


Figure 36 Example of an adaption: After the property value of "Switch ON Delay"has been changed in 10, the value has been analysed by the adaptation function and has been changed in 16.

4.3.14. Message window

The message window serves to output status messages and error messages, to display results of the evaluation of the function plan. Furthermore, the message window is an important instrument to check function block data within their respective contexts.

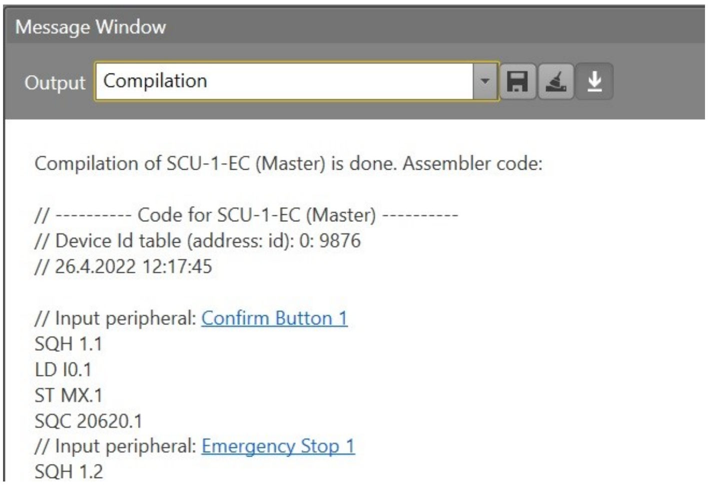


Figure 37 view message window with example output "Compilation"

- **Selection of message information via output**
  Status and error messages are output via **"Application"**.
  "Compilation" - Control of the function block data within their context.
  "Device interface" - display of results of the evaluation of the function block diagram

- **Quickly jump to an element**
  By clicking on the coloured block IDs in the message window, it is possilbe to jump to an element. The working surface scrolls to the desired position, and the element is visible.

- **Search field**
  Use Ctrl + F to open the search field. Note that this key combination only functions if the message window is active. The search field can also be called via the context menu.

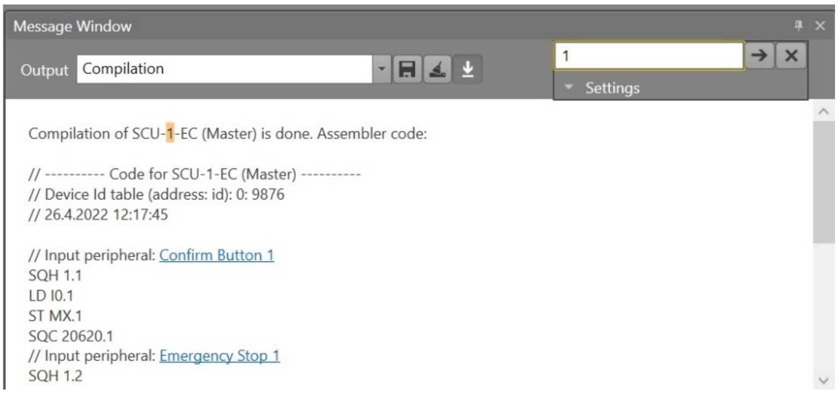


Figure 38 Message window with search field

Via the search field, the user can search the compiled code. To show the next match, either click on the "Search"  button or use the F3 key.By clicking on "Settings", additional settings can be displayed or hidden. By selecting the boxes, the usercan determine the search method.

4.3.14.1. Context menu in the message window

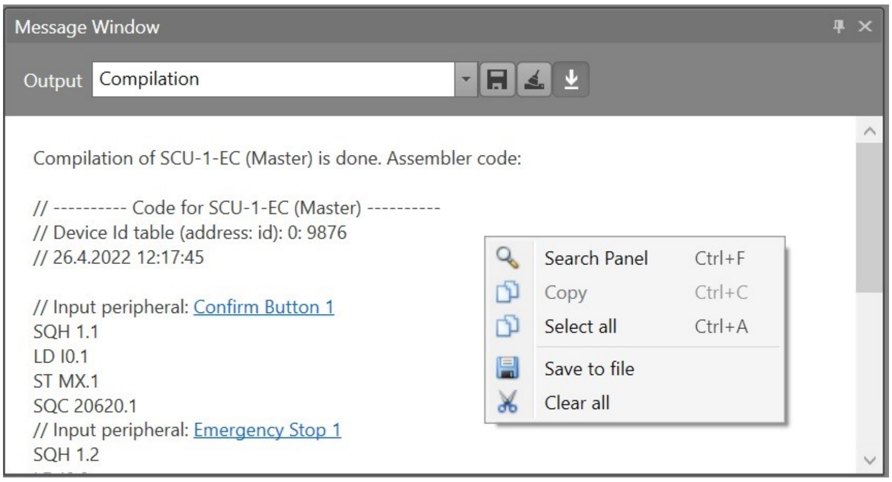


Figure 39 message window with context menu

"Search Panel" – shows or hides the search field.
"Copy" – copies the selected text in the clipboard. The text is available vor inserting.
"Select all" –selects the whole text.
"Clear all" –deletes the whole text.

4.3.15. Global search

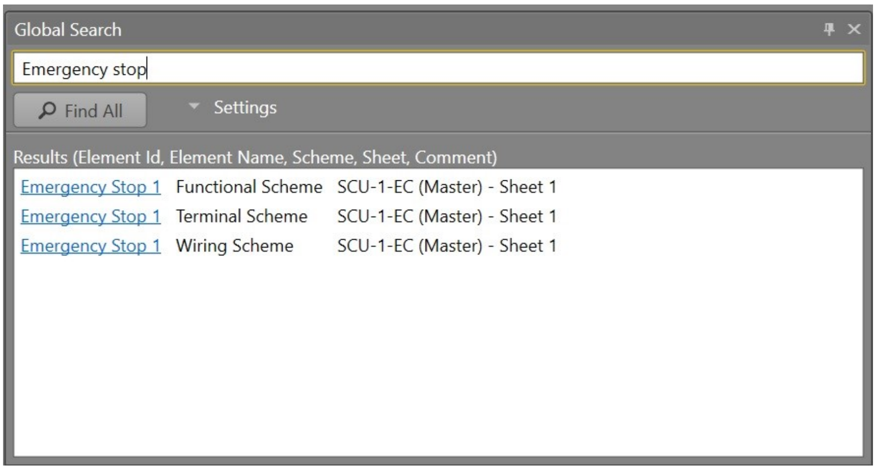


Figure 40 "Global search"

Global Search is an important search instrument. The text entered into the search field is searched for on the basis of settings. To show all hits of the desired text, either click on the "Find All" button  or press Enter.

4.3.15.1. Search settings

By default, the search settings are hidden. To show the settings, click on "Settings"

.

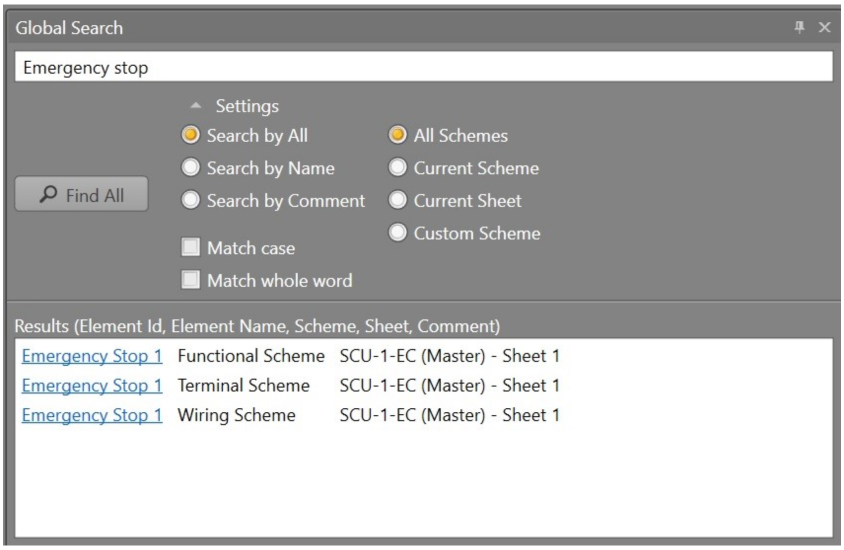


Figure 41 Define search settings

Beside the familiar settings, e. g. „Match case"or "Match whole word", the global search can also search for ID name or comment. Additionally, it can be fixed from which plans results shall be displayed.

4.3.15.2. Quickly jump to an element

If you click on the block-ID, the program immediately jumps to this block.

4.3.16. Print

“Print“ prints plans. The printer can be selected and the printer’s properties can be set. You can set the number of copies to be printed and select the pages to be printed.

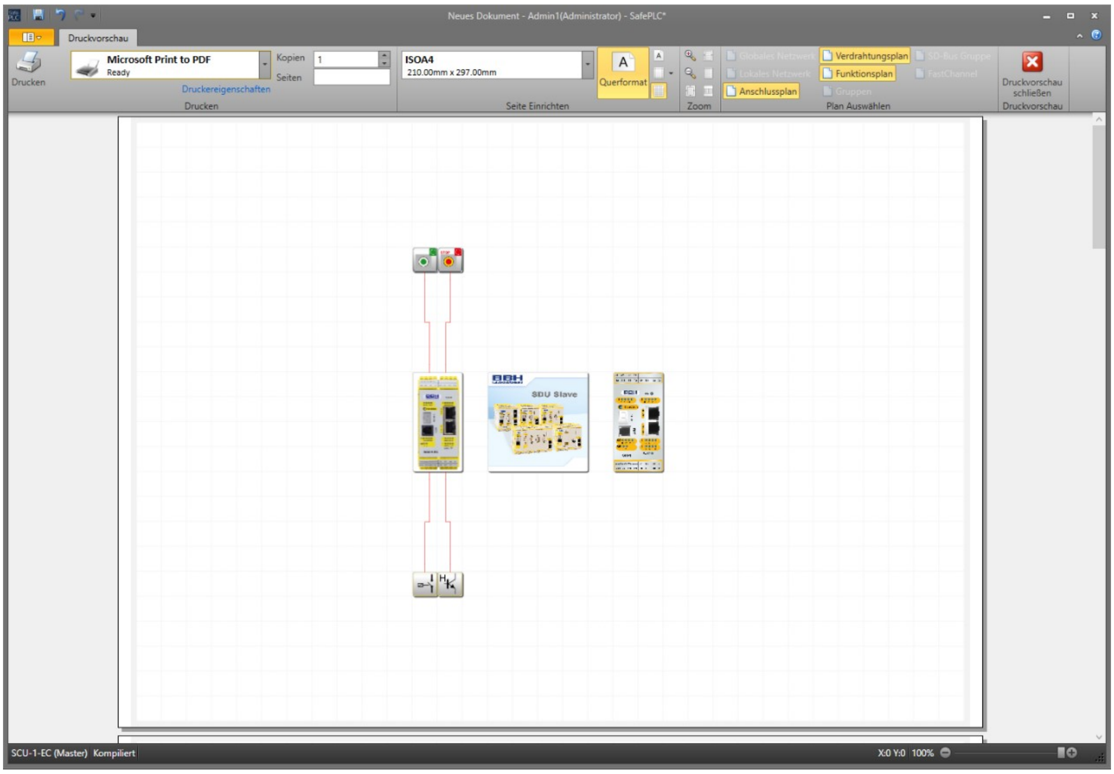


Figure 42 Tab “Print Preview”

Menu group “Page setup“:

With the “Print Grid“ button Page size, orientation, (landscape or portrait), margins (left, right, top, bottom) and the grid on the print can be switched on or off.

Menu group “Zoom“:

Zoom In – enlarges the content in the preview window (+10 %).
Zoom Out – reduces the content in the preview windowster (-10 %).
Actaul size – sets the size of the content at 100 %.
Page Width – shows the page in full width.
Whole Page – shows the whole page in the preview window.
Two Pages – shows two pages simultaneously.

Menu group “Scheme selection“:

Button Terminal scheme – sets if terminal scheme is printed or not.
Wiring scheme – sets if wiring scheme is printed or not.
Functional scheme – sets if functional scheme is printed or not.
Close Print Preview – closes the print preview window.

To print one of the above plans or diagrams from a document:

1. Click “Print” in the “Start” menu (or select the print command in the Quick Access Toolbar by using **Ctrl+P**).
2. Before printing, check the print preview of the terminal scheme, of the wiring scheme or of the functional scheme.
3. Select the printer from the list showing the operational printers.
4. Set the number of copies and pages.
5. In “Page Setup‟ you can select the page size, the orientation and the margins. You can also switch on or off the grid on the print. For further printer properties click on “Printer Properties”.
6. Click on the “Print” button 

. If you want to edit a plan or a diagram, or if you want to continue working, close the print preview 

.
7. Under “Zoom‟ you can set the size of the print preview.

4.3.17. Settings

In the Settings Window the user can change the application settings. To change the category, click on the desired tab on the left side of the program.

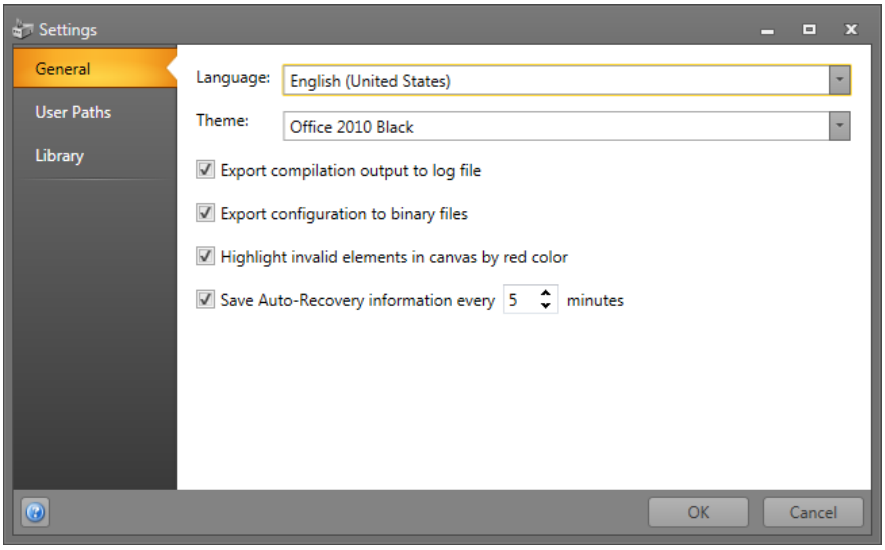


Figure 43 "General" tab in "settings" user window

General information:

Language selection – Setting English, German

Theme – Setting the color scheme of the surface (light or dark).

Check-on or Off – Settings and actiivation of the following functionalities:

- saving of the compilation result in a log file
- saving of the configuration in a binary file
- Invalid configured canvas elements are highlighted in **red** in the canvas work surface to enable validation in the canvas.
- Setting Auto Recovery Information save every x minutes

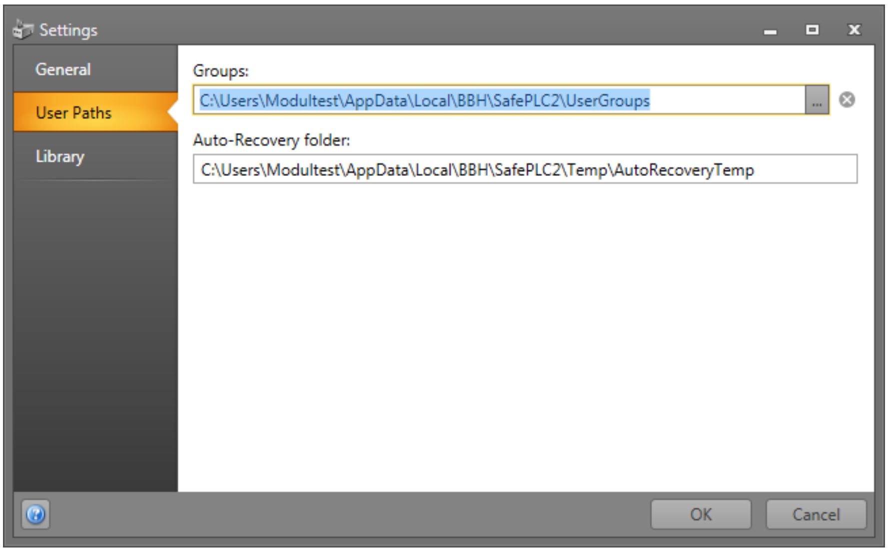


Figure 44 "User Path" tab in the "Settings" user window

User Path: save group target and Auto-Recovery file

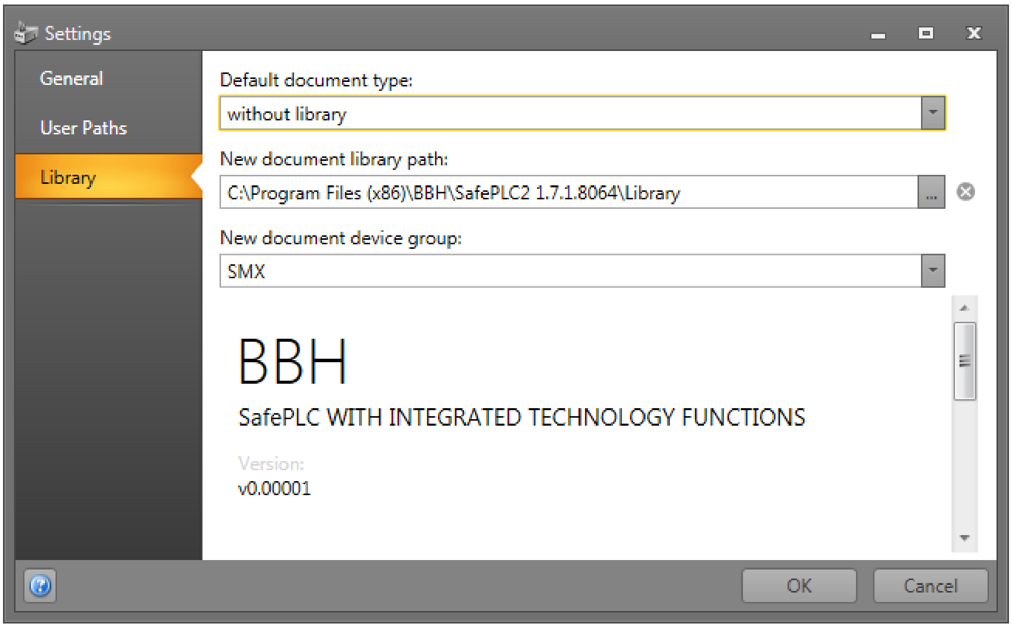


Figure 45 "Lbrary" tab in "Settings" user window

Library: Setting of document type, path and device series

Standard document type:
Differences can be made when selecting the document:

- without library *.spl2:
  Very small file size, program can be opened with same or newer SafePLC² with same range of function.

- With library *.spl2l:
  Very large file size, program can be opened with same or newer SafePLC² regardless of range of function.

NOTICE:
Here, all necessary libraries are saved. Thus, functional uniformity is given, although possibly in a new version the function has been changed.

Storage location of the new library – Setting the storage location of the new library file * .splib.

Device group of the new document – In case of several device groups, the device group can be preselected, e.g. in case of different series.

NOTICE:
Please note that changing the device group requires a restart of the application.

4.3.18. Auto Recovery

SafePLC^2 has an integrated Auto-Recovery function. This function saves the document process of the open file within a customizable fixed interval (1 to 60 minutes). If the program closes unexpectedly, e. g. during a power failure or an unexpected crash, the files can be restored. This ***SafePLC^2*** function saves the document process in the temporary file directory whose path is indicated on the tab “User Paths”.

Restarting *SafePLC^2*

The crash enables the user to store, to delete or to skip the autorecovery to make the selection at the next start of ***SafePLC^2***. The Auto Restore function allows the user to save, delete or skip the temporarily saved document to make the selection the next time the SafePLC2 is started.If ***SafePLC^2*** is closed in the normal way, no data are saved.

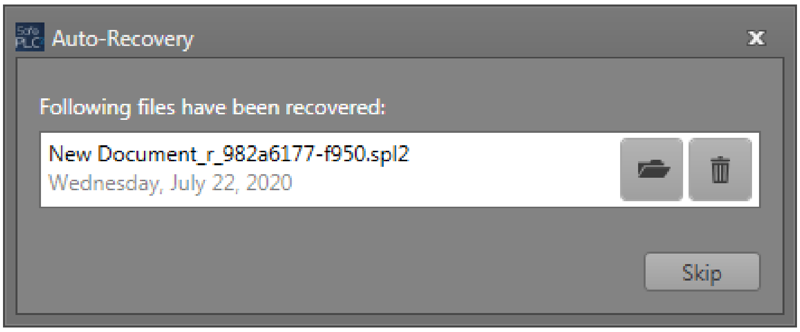


Figure 46 Dialog window "Auto-recovery"

Open File  – This allows to continue the process with the selected recovery file. At the next restart of ***SafePLC^2*** other stored recovery files survive.

Delete File  – Delete recovery file and continue program with empty document. If only one recovery file exists, a next selection is not necessary.

Skip 

– Skips the recovery selection and continues the program with an empty document. Recovery selection holds files for next start.

4.3.19. Information concerning the program

Figure 47 information window about SafePLC²

Under "About SafePLC²" short information is given about the Windows system, the application development, and about compilation. Below, typically, sales information and the web address are indicated.

4.3.20. Exit

The user exits the SafePLC² programming software with the "Exit" or "X" button.

4.3.21. Window "User Rights Dialog"

This window is only accessible for administrators. The The window opens after a click on the "User rights" button under the "Start" ribbon tab. The "Manage user rights" window consists of the tabs "Users" and "Groups".

In the window User Rights dialog the administrator can change the user rights for every object in the plan. The program works with three specific rights that are valid for each object:

1. Read Permission: The properties of an object can be read.
2. Write Permission: The properties of an object can be changed.
3. Linking Permission: An object can be linked to another object.

NOTICE: Administrators are standard users of the programming software SafePLC² and have all permissions.

4.3.21.1. Tab "Users"

In the tab "Users" the administrator can change the rights for the other users.

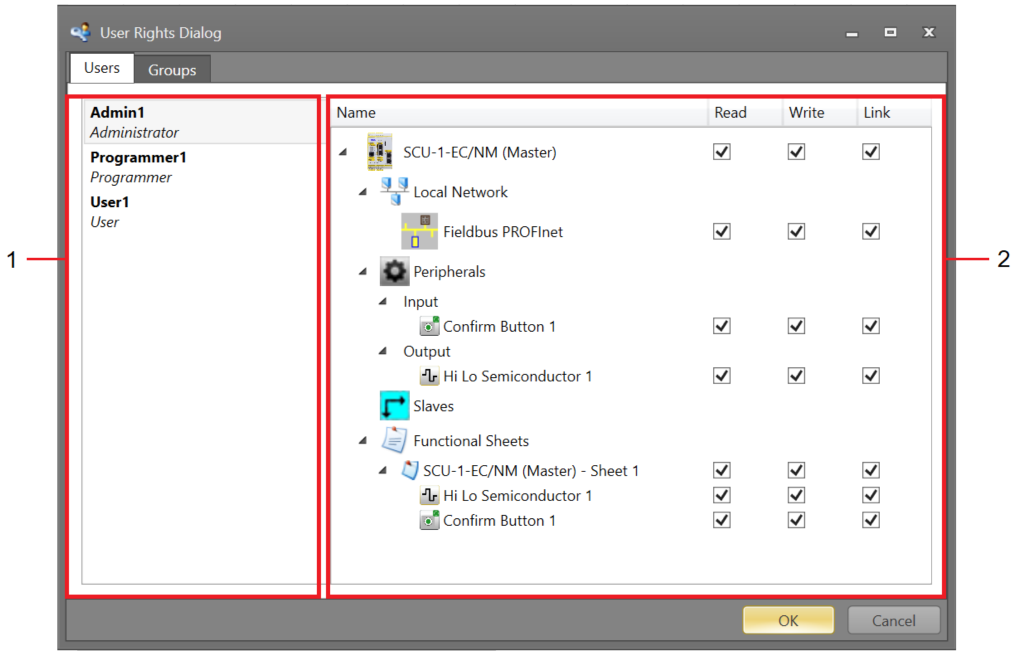


Figure 48 Tab "Users" in the window "User Rights Dialog".

1. List of users with the name of the user group to which they belong (e. g. administrator). If the default user rights are not cancelled, every user has default rights he or she assumes from the user group.

2. List of elements and of rights of the current selected user.

NOTICE:
You can change the rights for every block or every group separately by clicking the right mouse button on the block or on the group and selecting "Change permissions".

4.3.21.2. Tab "Groups"

With the tab "Groups" the administrator can change the standard rights of individual user groups.

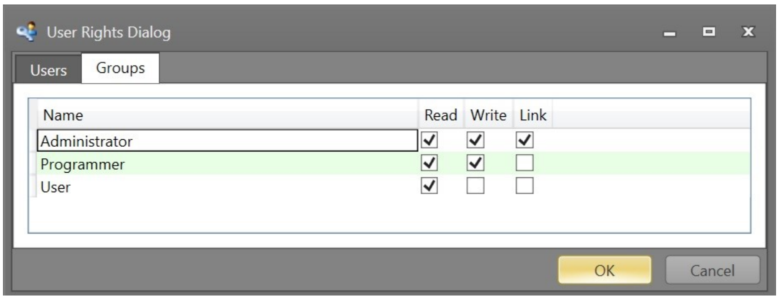


Figure 49 Tab "Groups" in the window „User Rights Dialog".

4.4. Process

The program ***SafePLC²*** is a graphics-oriented software to create an PLC-based monitoring program for an SCU system. With the device, it is possible to reliable monitor drive motors. The process described below, has proven most successful for SCU-devices, but it is not compulsory.

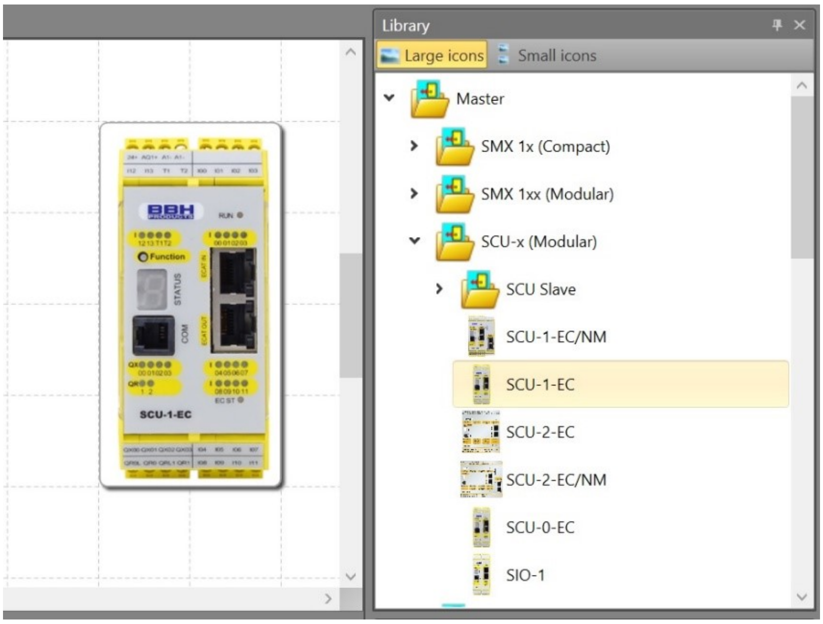
General information:
The program requires the write permission and the read permission of the user registered on the PC used for programming. The application can be run as an administrator. If this is not the case, access to some restricted folders cannot be granted. Missing access authorisation can either have unwanted effects on troubleshooting in the functional scheme, or it can cause problems if logic diagrams are saved in directories with restricted rights.

With administrator rights, there are no restrictions on modifying the project in SafePLC².

4.4.1. General workflow

Drag a symbol from the library or from menu option in the work surface to insert it into the selected plan or diagram. If this is possible, the element automatically adds a block in the work surface. The proposed working steps correspond to the reflections that should be made when planning a safety-relevant monitoring system for a drive axle.

- **"Drag & Drop"**



The easiest way to insert a block or a device is "Drag & Drop".

The basic steps for "Drag & Drop" are:

1. Move the pointing device to the object.

2. To "grab" the object, press and hold the mouse key or the key of the pointing device, With the Esc key the process is reversed.

3. Use the pointing device to "drag" the object to the desired place.

4. By releasing the button you can "drop" the object.

To create an application, proceed as follows:

1. Select the device type to be programmed:

After ***SafePLC²*** has been started or after a new terminal scheme has been created, an empty work surface appears. All available devices are in the library. By clicking on the appropriate module, inter alia, the description window shows the following files: programming interface, digital I/O, outputs, inputs etc. The desired module can be added with "Drag & Drop".

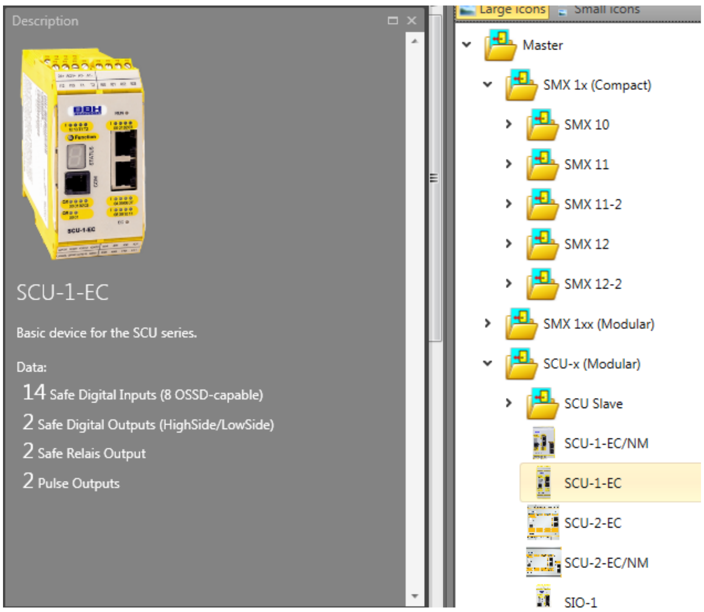


Figure 50 Description of the device

To continue the process, a master device must be selected.

Add a Slave device:
If a Master device has been added in the terminal scheme, a Slave device must be selected in the browser tree to add a secondary device. Otherwise, the Master device is replaced. Before replacing the Master device, the program shows a warning message.

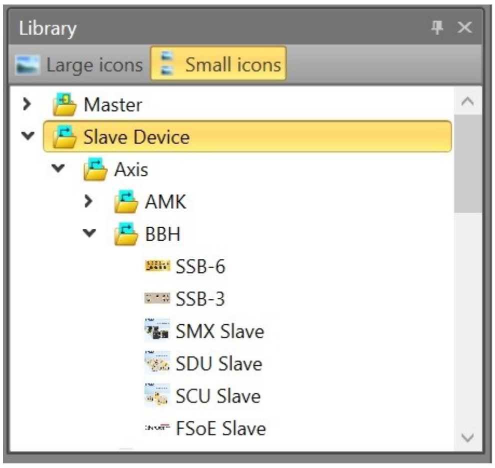


Figure 51 Selection of an slave device

NOTICE:
Due to the resources involved, and because of their administration in the programme environment, is not recommendable to change the type of equipment later on.

The following Properties window can be used for every input cluster to assign a name and to select the parameter "Cycle time". 16/24/32 ms can be selected for the "Cycle time" setting.

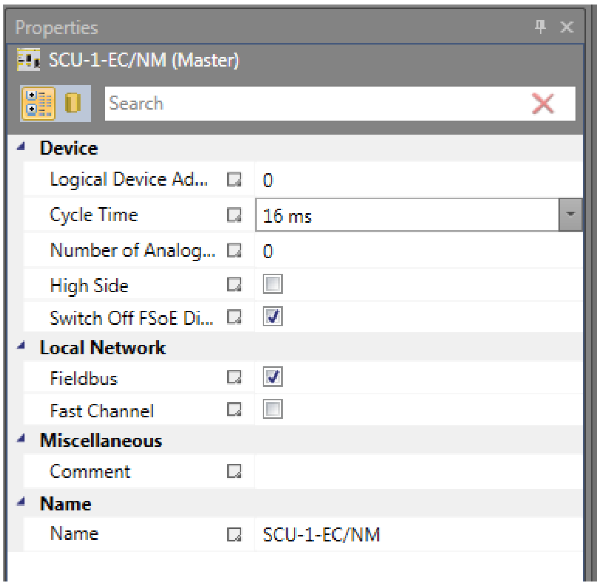


Figure 52 Property window of the device

2. Fixing the peripheral devices in the terminal scheme

Under "Terminal scheme" a simplified scheme with selected devices, encoders, inputs and outputs of the SCU-system is shown. The necessary modules are linked automatically after they have been inserted.

The following process is recommended:

1. Select the corresponding type of peripheral devices (master or slave) in the browser tree of the browser.
2. Select a Input or Output block from the library.
3. In case of slave modules with the monitoring of speed and position, the definition of encoders and their parameters is necessary.

NOTICE: A red symbol indicates a missing setting

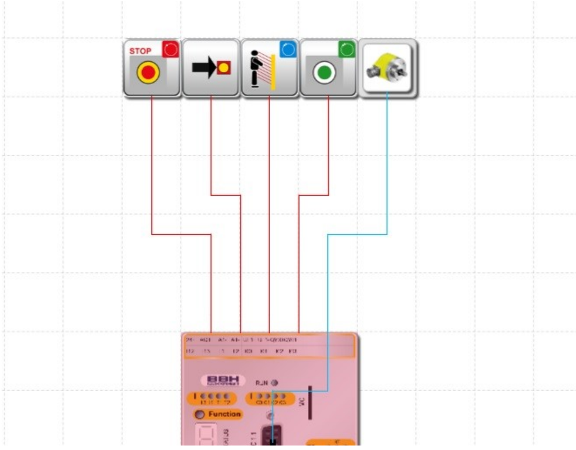


Figure 53 Confirm button with missing setting (red)

In case of modules with analogue processing, the interfaces must be set.
The selection of input and of peripheral modules (confirm button, door control, Emergency stop, light curatin etc.) is made via the "Input" folder in the library.

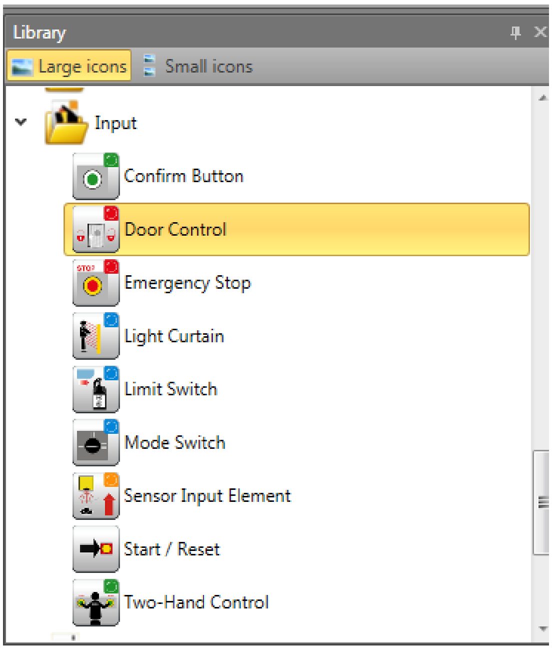


Figure 54 Selection of an input element via the library

Add the necessary output modules (semiconductor, relay etc.) in the same way.
The wiring scheme shows the connections between sensors and actuators with the indicated connectors. After the necessary peripheral devices have been selected, they are connected to the system.

NOTICE:
When inserting the input or output block elements, it is not necessary to connect them manually to the device, as is the case with the function blocks in the functional scheme. These connections are made automatically in the terminal/wiring scheme

3. Definition of peripheral devices in the functional scheme

The functional scheme shows the logic modules and their interconnections. Peripheral devices that have not yet been incorporated into a functional scheme are marked with a green arrow. The arrow indicates that these elements can be dragged and into the functional scheme and can be inserted.

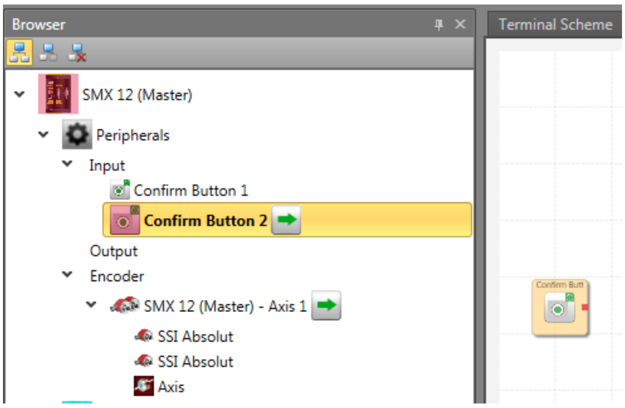


Figure 55 Inserting the input module

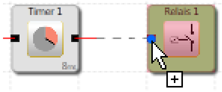
4. Definition of monitoring functions and logic modules in the functional scheme

The functional scheme shows the logic modules and their interconnections.

Programming of the functional scheme by:

- Logic elements and processing elements.
- Timers, Flipflops (triggering elements) and terminal blocks.
- Monitoring modules for drive monitoring (Drive monitoring only possible if the corresponding sensors have been defined).

After their selection, the necessary modules are interconnected.



For this purpose, drag the cursor over a "Start connector". Use the left mouse button to click on the start connector. By doubleclicking, the start connector is connected to a "target connector". For further information, cf. the chapter 4.6.7 "Circuit".

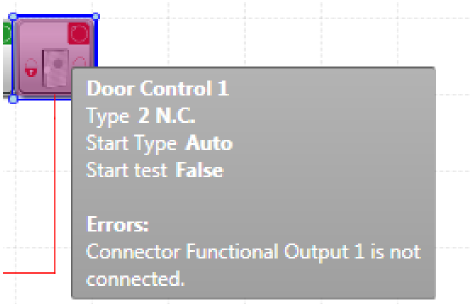


Figure 56 information window

5. Compilation of the monitoring program

After programming has finished, the function plan is compiled and is converted into a machine-readable format.

This process consists of the following steps:

- Check for open connectors in the functional scheme
- Check of the limit conditions for the monitoring functions
- Check if the pulse numbers are correctly distributed via the cross-connections
- Creation of a transferable OP programs for the module

6. Program transfer to the safety module by clicking on the device interface

After the dialogue window of the device interface has been opened, the software automatically compiles the program. The program transmission process consists of:

- Setting the COM-output or setting of the IP-address
- Transmission of the automatic program (IL programming code).
- Password entry. Corresponds to the serial number of the safety module (only if TCP/IP-connection is used)
- Test of the program in the safety module.
- Creation of a validation report and validation of the configuration

4.4.2. Network plan

Before the network is installed and programmed, it is recommended to create a plan of the FSoE network.

The following steps are necessary to create the plan of the FSoE network:

1. Determination of the FSoE Master and of its directly connected elements (e. g. IOs, encoder)
2. Determination of the FSoE Slaves and of its directly connected elements
3. **Determination of the EtherCAT Master** (-> SCU)
4. **Determination of the EtherCAT Slaves**
5. Determination of the structure of the FSoE network
   (The structure of the FSoE network is equal to the structure of the EtherCAT network.)

6. Determination of the network parameters
   - Module adress (unambiguous status number in the network; the assignment takes place automatically after inserting the devices in SafePLC².)
   - Adresses (Master = 0, Slave x –consecutively numbered)
   - Connection ID (freely selectable, unambiguous status number in the protocol- determines the communication sequence)

After the network structure and its parameters have been determined, the network can be programmed via SafePLC².

For parameter determination, please also use the installation manual of SCU and module data files, the ratings' data files and the specifications of the manufacturer.
To program the network, the programming manual SafePLC² must be used or considered.

Below, only those steps supplementary to the SafePLC² programming manual are described that are necessary considering the SafePLC²programming manual to map the network in the program SafePLC²' and to program the modules Master SCU and Slaves (SDU, SSB,…) – also those produced by external companies.

⚠ **WARNING**
Malfunctions of the FSoE network can cause loss or impairment of safety functions.

NOTICE:
Incorrectly assigned or incorrectly set parameters can cause network malfunctions (e. g. loss of input data or delay of input data).

By means of this example "hall door", the operation and the parametrization of the SCU-series shall be cexplained:

<u>Example:</u> hall door
Our example concerns two neighbouring halls (hall 13 and hall 15) where the FSoE Slaves manage data and transfer data to the FSoE Master situated remotely in hall 12.
Between hall 13 and hall 15, an automatic gate is situated, whose position is monitored via an (nonsafe) sine-cosine encoder (axle) and position switches (digital switches via digital input). Furthermore, the gate may only be released if the central entrance boom is closed (external data via EtherCAT from Keba-Slave).
The FSoE Slaves in the halls enable the operating personnel to request the opening of the door and, of course to reques a stop via emergency stop .
Partially, the modules are integrated in the switch cabinets situated in the different halls. The switch cabinets are interconnected via EtherCAT cables.
In hall 13 a light curtain is installed to cause a disconnection (collision prevention)

<u>Sketch</u>:

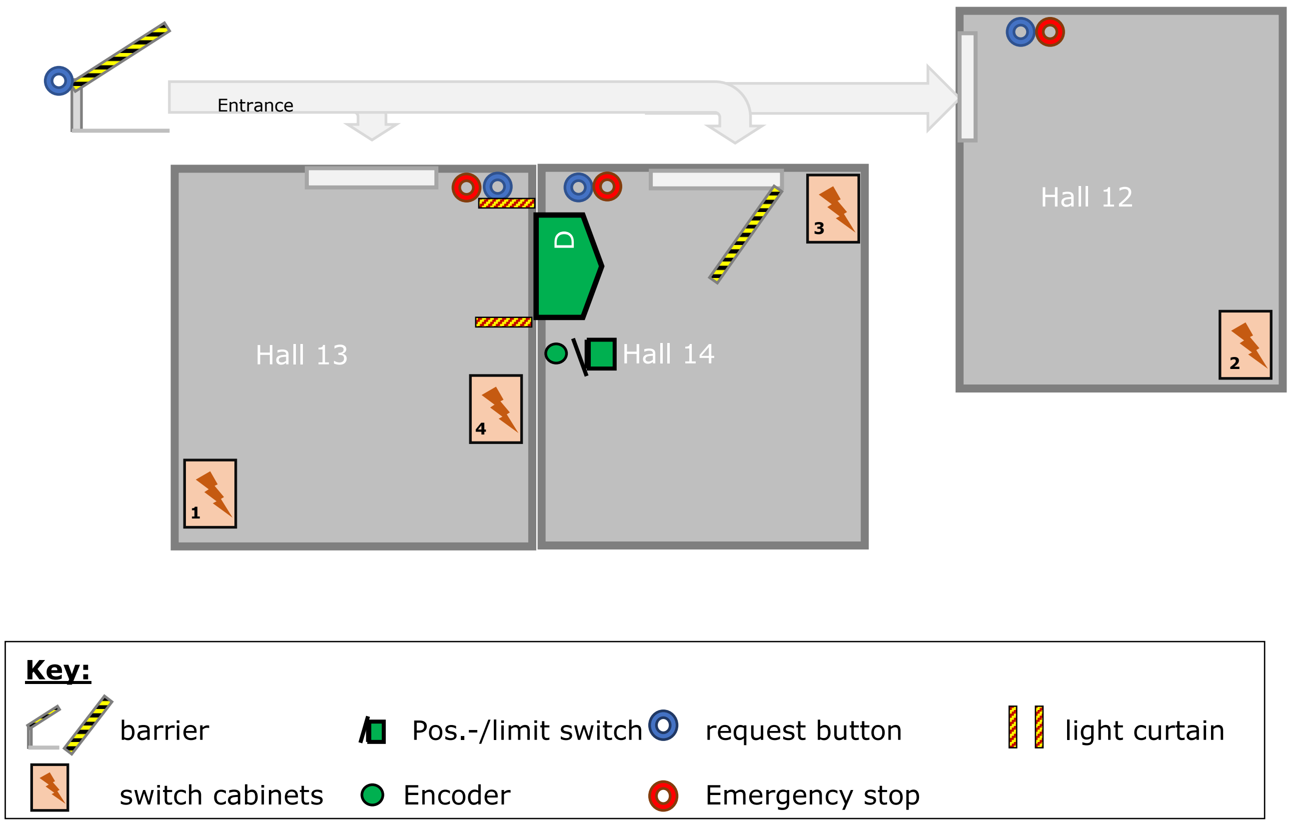


Figure 57: example - sketch

Furthrmore, the barriers only clear the way (via the Slave outputs)in the best-case scenario. The data transfer between Master and Slaves is effected via FSoE.

FSoE—network plan:

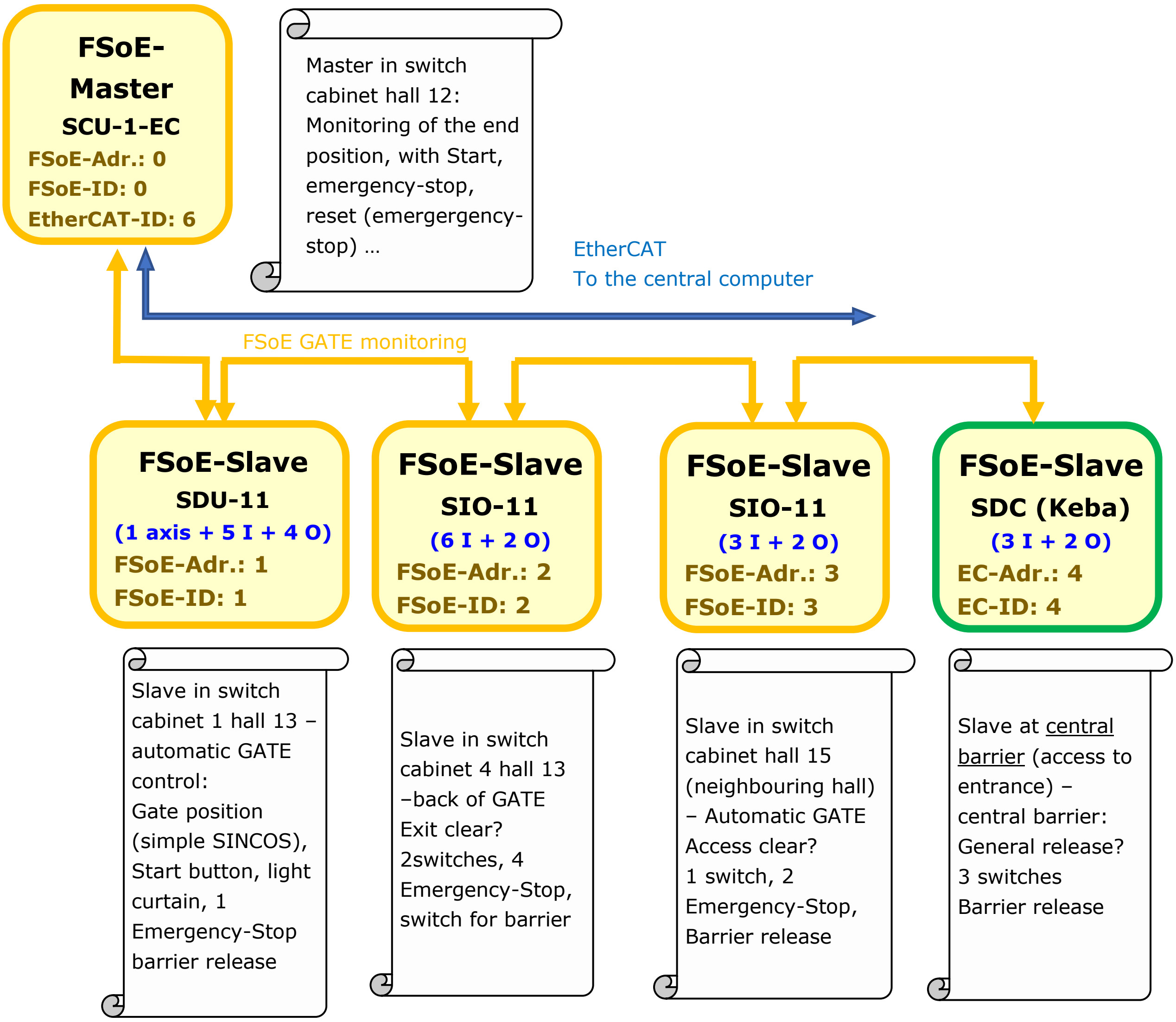


Figure 58: FSoE-Plan

4.4.3. Selecting the units / network components

After the start of the program SafePLC², first the suitable variant of the SCU-x-EC/x must be selected from the "Library" – by dragging it into the main window (example: SCU-1-EC) – tab "terminal scheme".

Via the property window (bottom right), the parameters of the selected units (e. g. FSoE diagnosis, connection ID, etc.) can be determined.

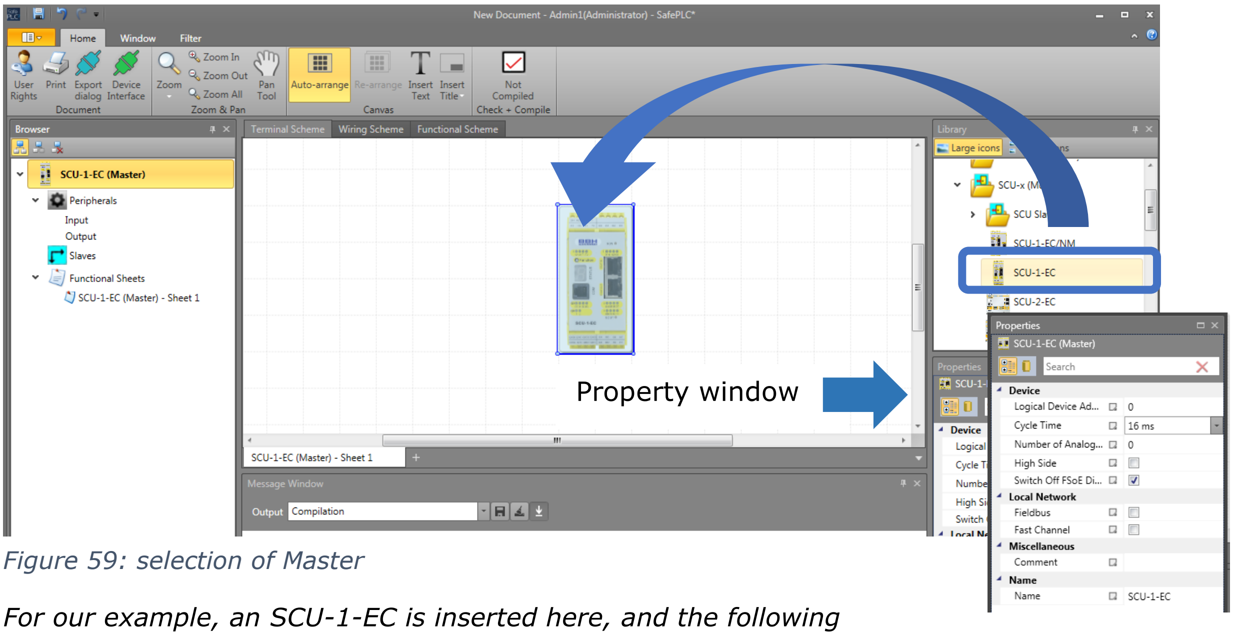


Figure 59: selection of Master

For our example, an SCU-1-EC is inserted here, and the following parameters are entered:

Logical device adress:	0	(selectable – see plan)
Cycle time:	16ms	(from the data sheet)
Local network:	EtherCAT	(EtherCAT to the central computer)
Serial number:	4	(back of units)

After the insert of the FSoE-Master units, automatically the selection folder for Slave units (FSoE and EtherCAT) is shown (Before the folder is hidden).
Afterwards, the Slave units for axle monitoring and IOs can be added. In the same window also the inputs and the outputs of the unit can be selected. Here, no difference is made between FSoE and EtherCAT. – The differenciation is made via parameters.

Slaves the read-in of axle data:

- SSB... (Safe Sensor Box – reading in of encoder data of 6 encoders BBH)
- SDU… (Safe Drive Unit – reading in of encoder data of one encoder BBH)
- Incremental Encoder Electronic GmbH (– read in encoder data of one encoder)
- SDC (– read in external encoder data (option encoder) Keba)

Slaves for IOs:

- Beckoff EL … (IO unit by Beckoff)
- SIO-… (Safe IO – input data and output data BBH)
- SDC (– read external IO data Keba)

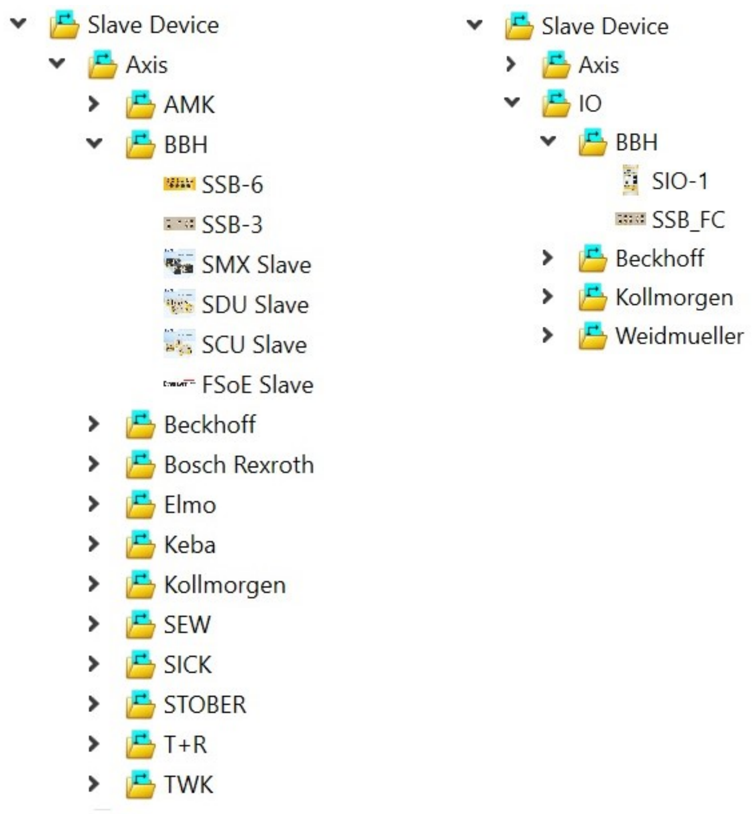


Figure 60: selection of axis and IO slaves

NOTICE: These units are predefined and are available at the installation of the SafePLC² . Other units can be inserted by the user (see below).In the wiring scheme, a proper sheet is created for each incorporated unit.

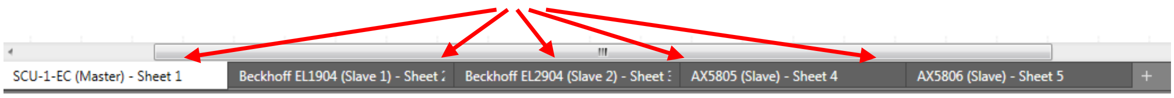


Figure 61: One sheet per unit

According to the example, the following Slave units are inserted:

- *SDU - automatic gate – switch cabinet 1, hall 13 – unit by BBH*
- *SIO - back of gate – switch cabinet 4, hall 13 – unit by BBH*
- *SIO - automatic gate – switch cabinet, hall 15 – unit by BBH*
- *SDC - Read-In of datafrom central barrier – unit by Keba*

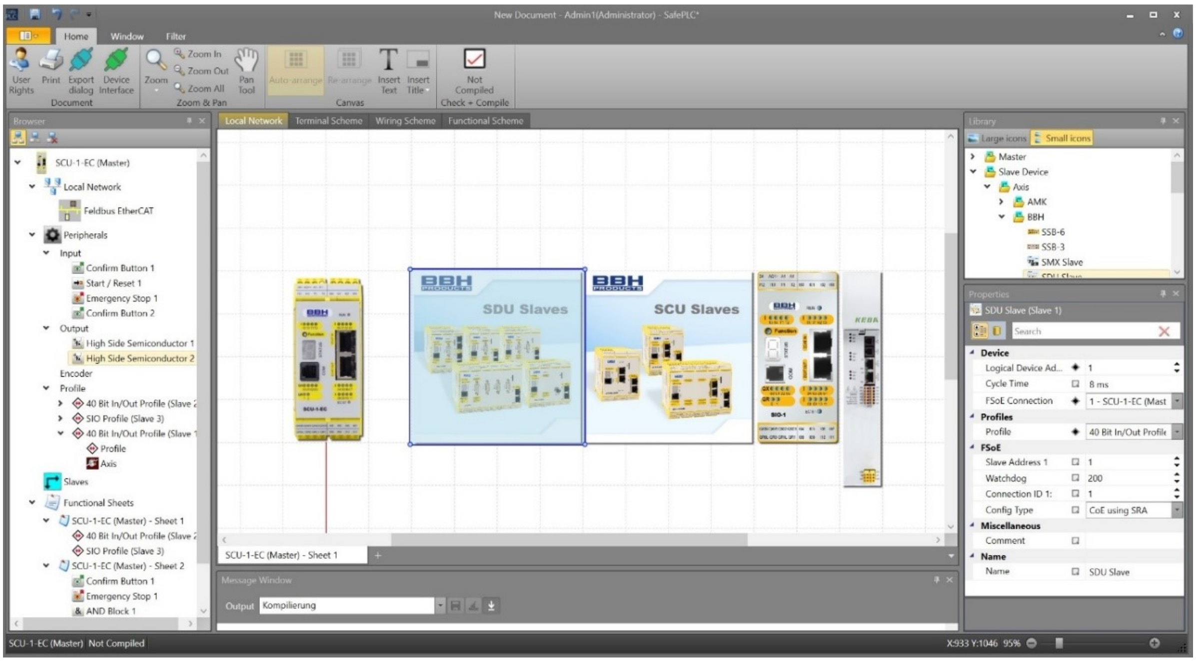
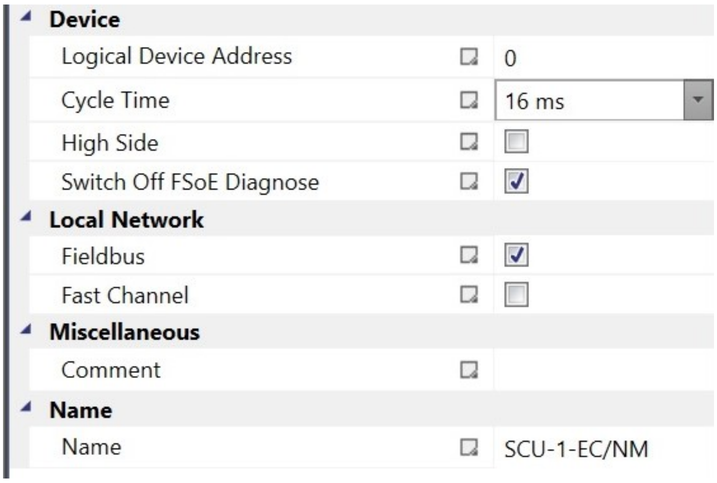


Figure 62: selection of units

4.4.4. FSoE settings of the units / the network options

The FSoE-Master is automatically set to FSoE and has the preset address 0 (cannot be changed).

In the "Properties" window of the master device it is possible to activate or deactivate the "Switch off FSoE diagnostics".



"Switch off FSoE Diagnose" means that FSoE connection errors <u>do not trigger</u> an alarm message on the module.

4.4.5. Optional Fieldbus interface

With the SCU-x-EC/NM variants an additional Fieldbus interface can be configured.

1. First, the setting „Local Network" must be selected in the unit.

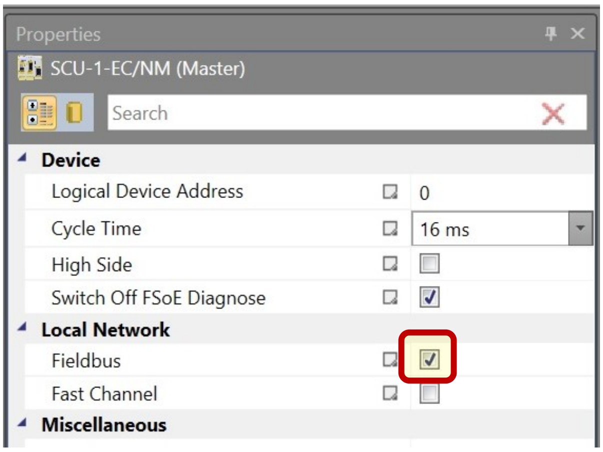


Figure 63: network selection

2. Afterwards, in the browser window "Local network" the symbol "Feldbus EtherCAT" can be clicked on to change the settings in the „Properties" window. In the "Properties" window, the additional "Network Type" can be selected.

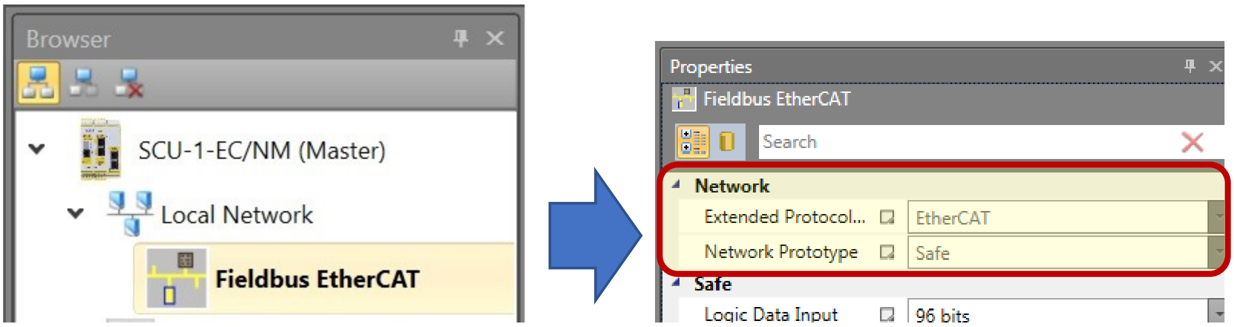


Figure 64: EtherCAT Network

3. In case of inserted Slaves, the FSoE properties must/can be set in the FSoE section in the window "Properties".For this purpose, the respective Slave unit must be selected /clicked on again with the mouse.

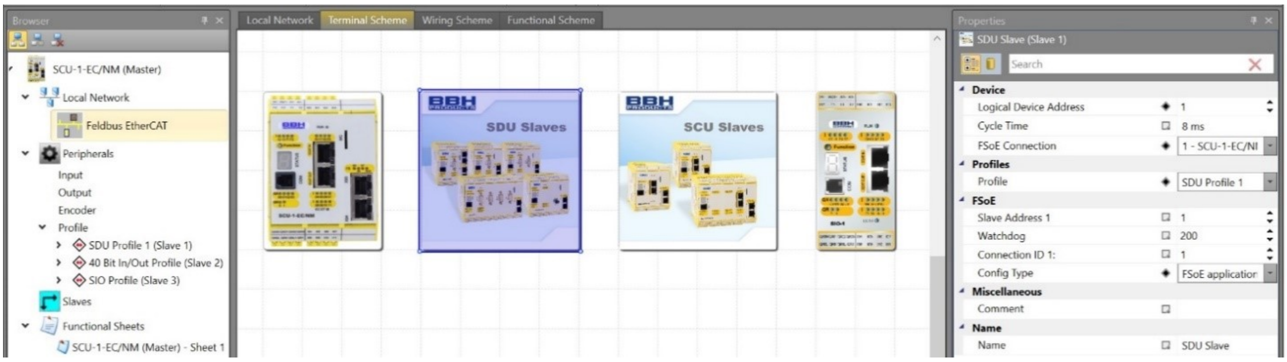


Figure 65 FSoE settings of slave devices

The following data can be set:

- Slave address (number)
  →To determine the Slave ranking in the FSoE network.
- Watchdog time (im ms)
  →To determine the max. permissible delay per unit.
- Connection-ID (ID)
  →To determine the communication assignment (protocol).
- Profile
  →To determine the FSoE profile
- Configuration type (if available)
  →To determine the configuration type – selectable "FSoE application parameter" or "CSOE using SRA"

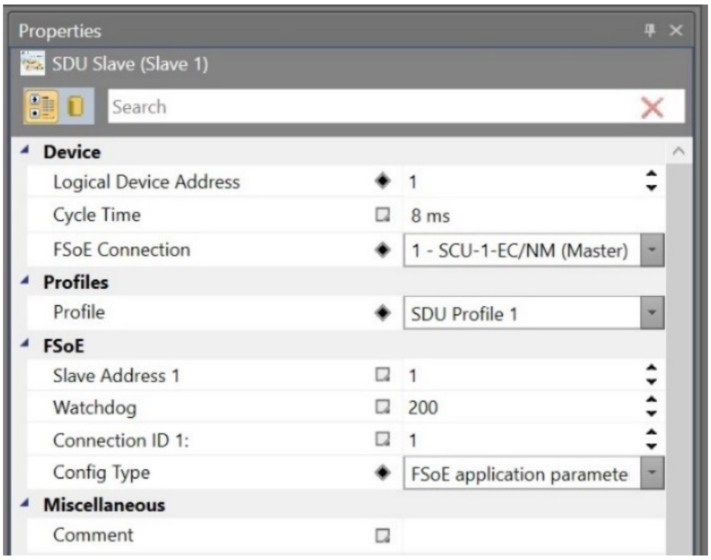


Figure 66: FSoE options for Slaves

The following properties apply to the units in our example:

	SDU	SIO1	SIO2	SDC	Comment
Logical device addr.:	*1*	*2*	*3*	*4*	*as fixed in plan*
Cycle time:	*8ms*	*8 ms*	*8 ms*	*8ms*	*as per Master cycle*
FSoE-Slave-addr. and -ID:	*1*	*2*	*3*	*4*	*as fixed in plan*
FSoE-Watchdog:	*200 ms*	*200 ms*	*200 ms*	*200 ms*	

4.4.6. Determining the inputs and the outputs of the units

After the units are selected, their inputs and their outputs are determined in the SafePLC² – tab "Wiring scheme".

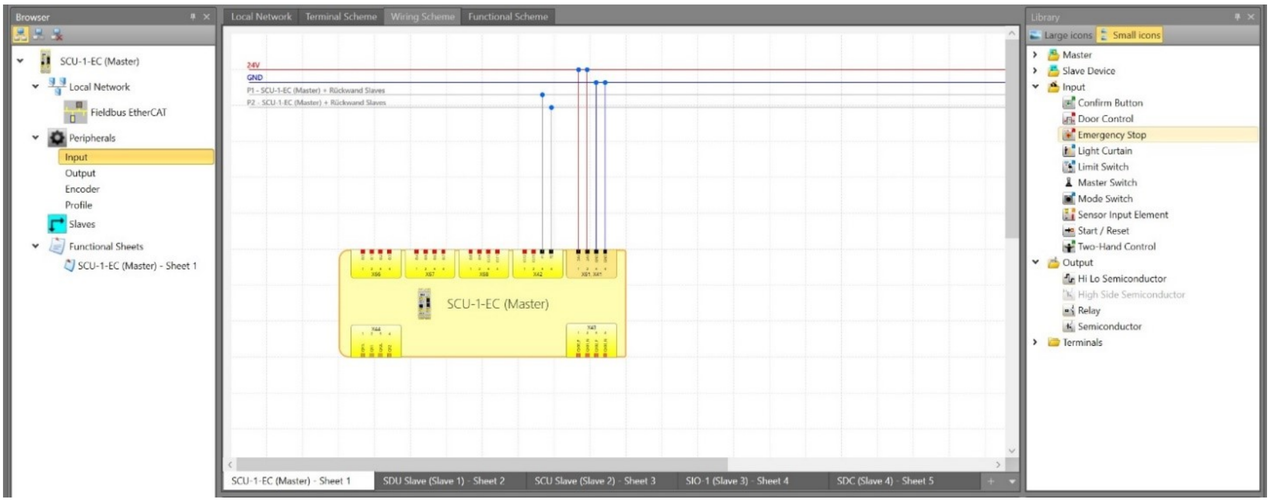


Figure 67: Insert IO element

By selecting it and dragging it into the main window, an input element or an output element from "Library" (upper right window) is incorporated.
Thereby, via the query window it is quarried to which unit this element shall be connected (automatically).

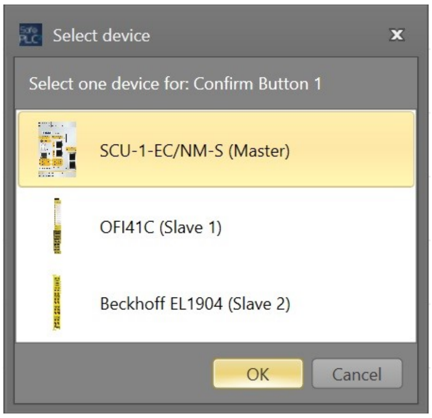


Figure 68: additional example for assignment to a unit

The connection from the element to the input / the output of the units in the wiring scheme takes place automatically.

NOTICE:
This assignment by SafePLC² must be implemented accordingly by the wiring. The unit is programmed accordingly and changes into the error mode if necessary.

For our example we had determined the inputs and the outputs. Now, the inputs and the outputs are set:

- *Master with Start button and Help button, Emergency-Stop button, Reset, 2 outputs*

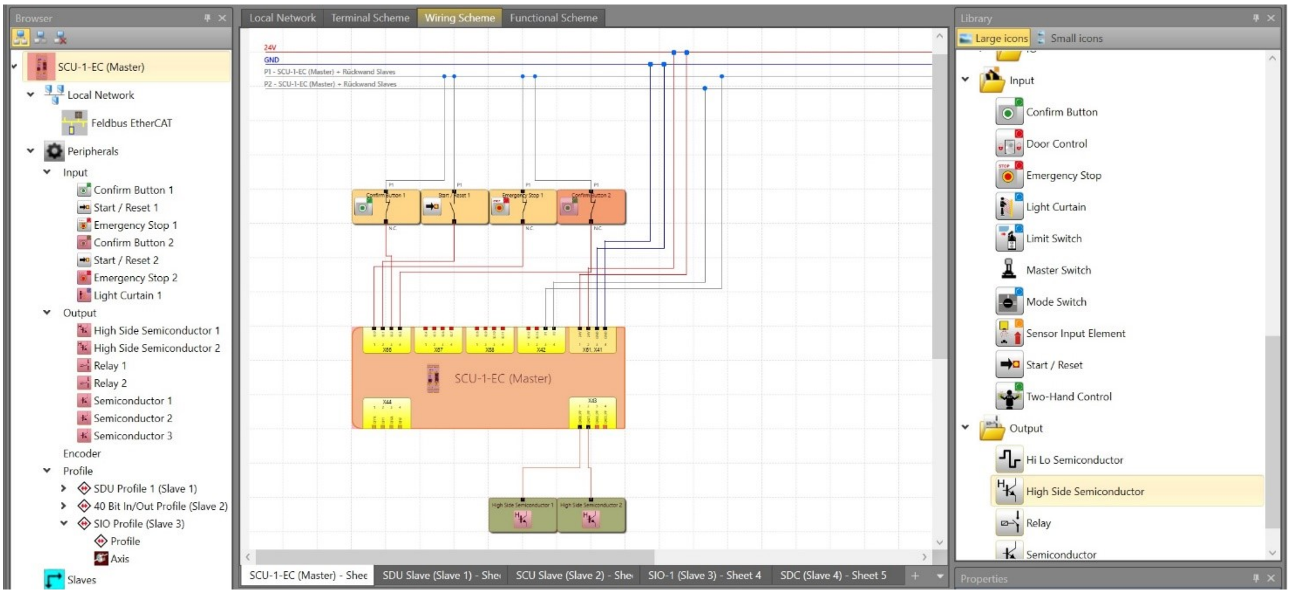


Figure 69: IO-elemente inserted (Master)

The assignment must be done separately for each module including all slaves.

ATTENTION:
Inputs and Outputs of slave device cannot be linked in the main configuration!
A new document must be created for this purpose. (description see chapter 4.4.7)

4.4.7. Slave Configuration

The assignment of inputs and outputs must be made for each slave separately in an new document – here Slave 1:

Before you can configure the inputs and outputs of the slave, you have to click the respective slave device again with the mouse in FSoE master configuration. The properties window appears and a profile can be selected.

Figure 70 Slave profile setting in FSoE master configuration

The profiles now appear in the library window (left). Drag the respective profile of the slave into the functional scheme.

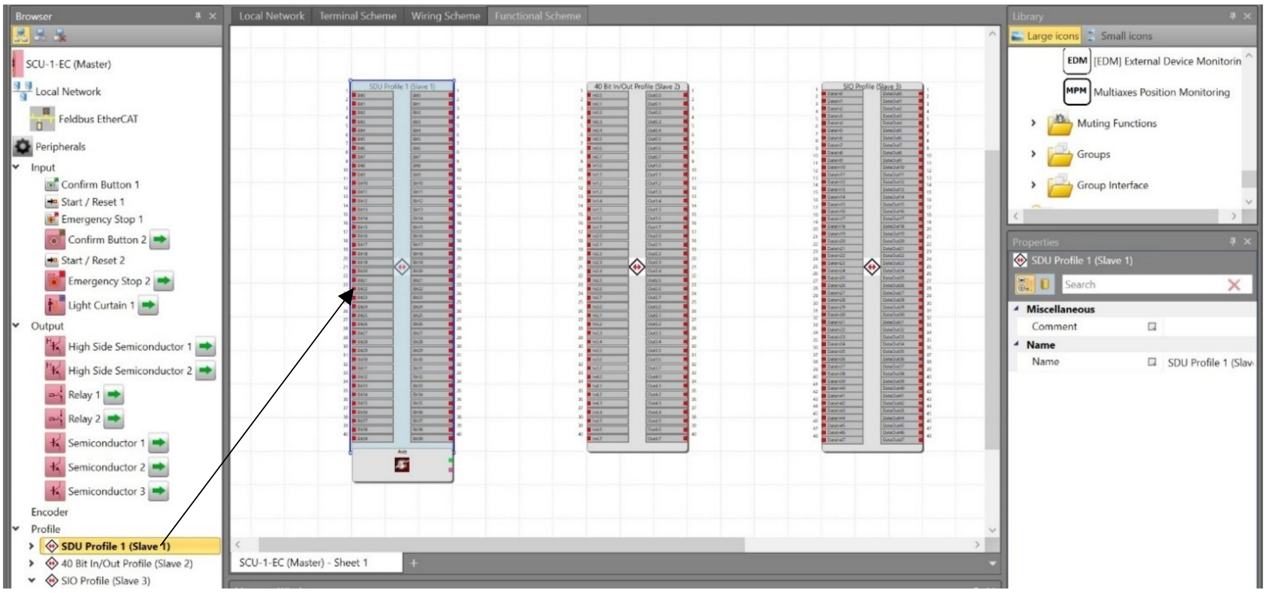


Figure 71 slave profiles in functional scheme

NOTICE:

For the slave modules SMX, SDU and SCU the assignment of the inputs and outputs is done separately. (For the further procedure see chapter 4.4.7.1) For slaves that cannot be freely configured, the configuration can still be executed in the master configuration.

4.4.7.1. Creating a slave profile

After creating the slave profiles in the FSoE master configuration, open a new document. Add a slave device as master (Master>SCU-x (Modular)> SDU-Slave ...) and define the input and output elements.

- Our example:
  Slave 1 with Emergency-Stop, Start/Reset button, light curtain + 4 outputs (relays + semiconductors for 2 switch cabinets with each one signal lamp).

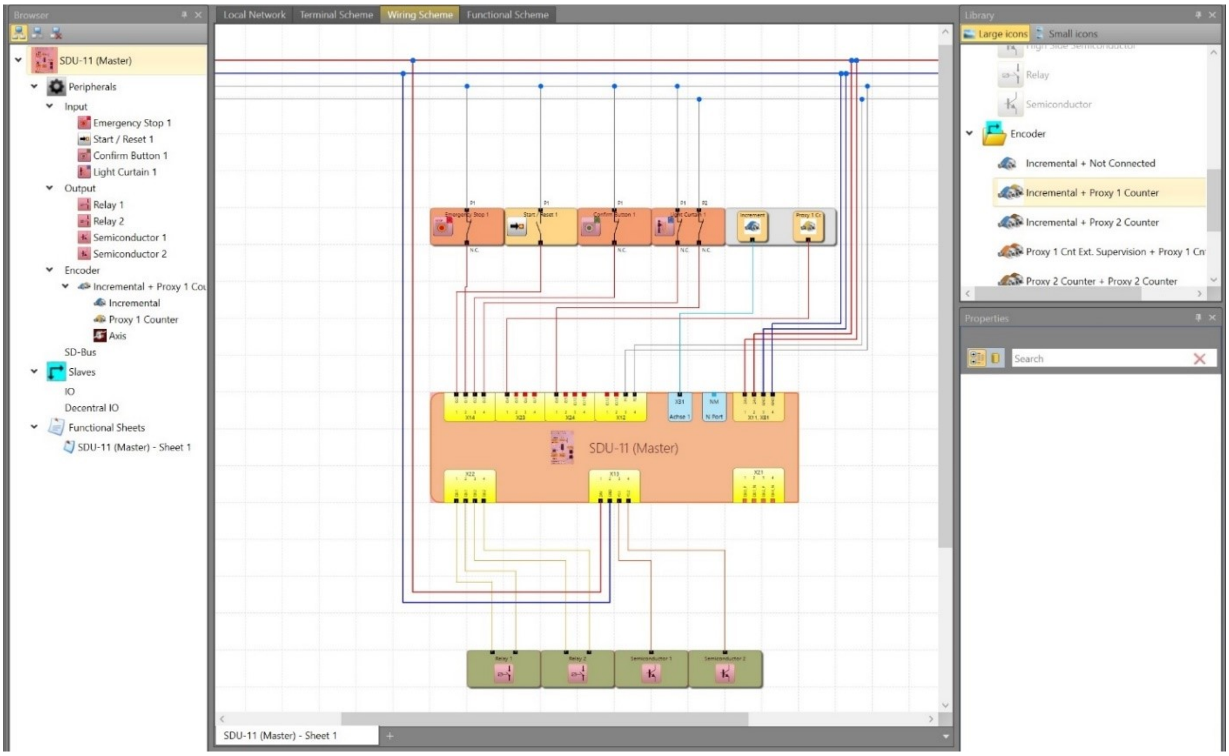


Figure 72 new document> wiring scheme> IO elements inserted (Slave 1)

The (safety) functions can be determined in the tab "Functional scheme".
(see chapter 4.5)

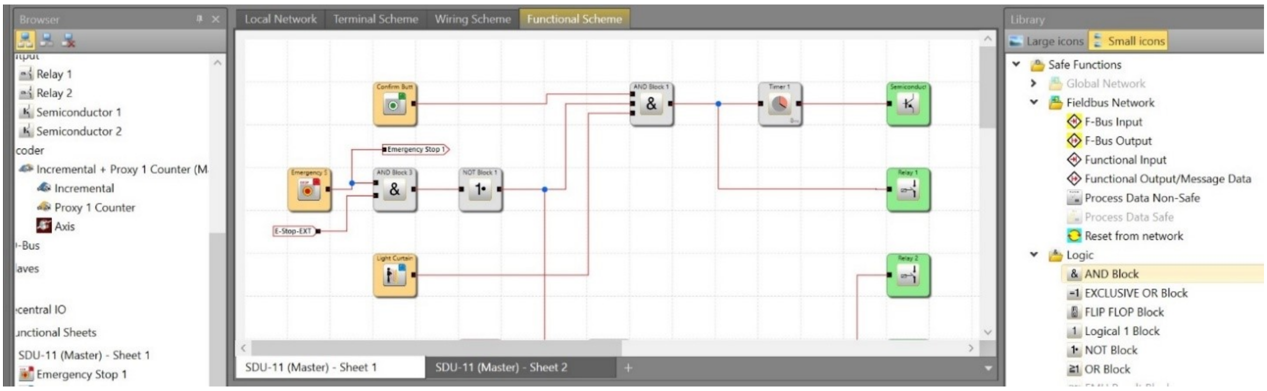


Figure 73 Determining the safety functions of slave 1

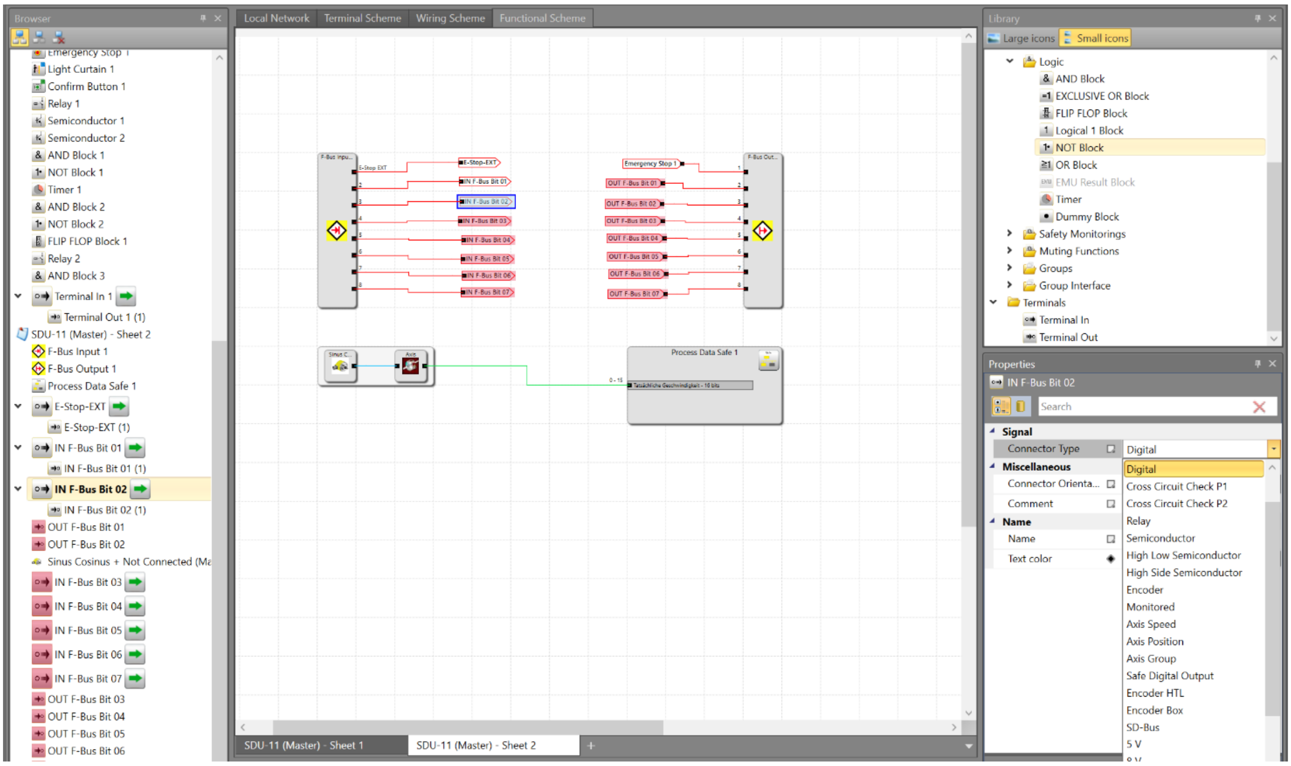


Figure 74 Creating the safe F-bus inputs and outputs in FSoE Slave (Slave 1)

Up to 12 bytes can be flexibly created in the F-bus profile.

NOTICE:
The data width refers to the slave profile from the FSoE master configuration and the number of available bytes is limited to 40 bits IN/OUT.

The connections of the elements are also adopted and displayed in the terminal scheme:

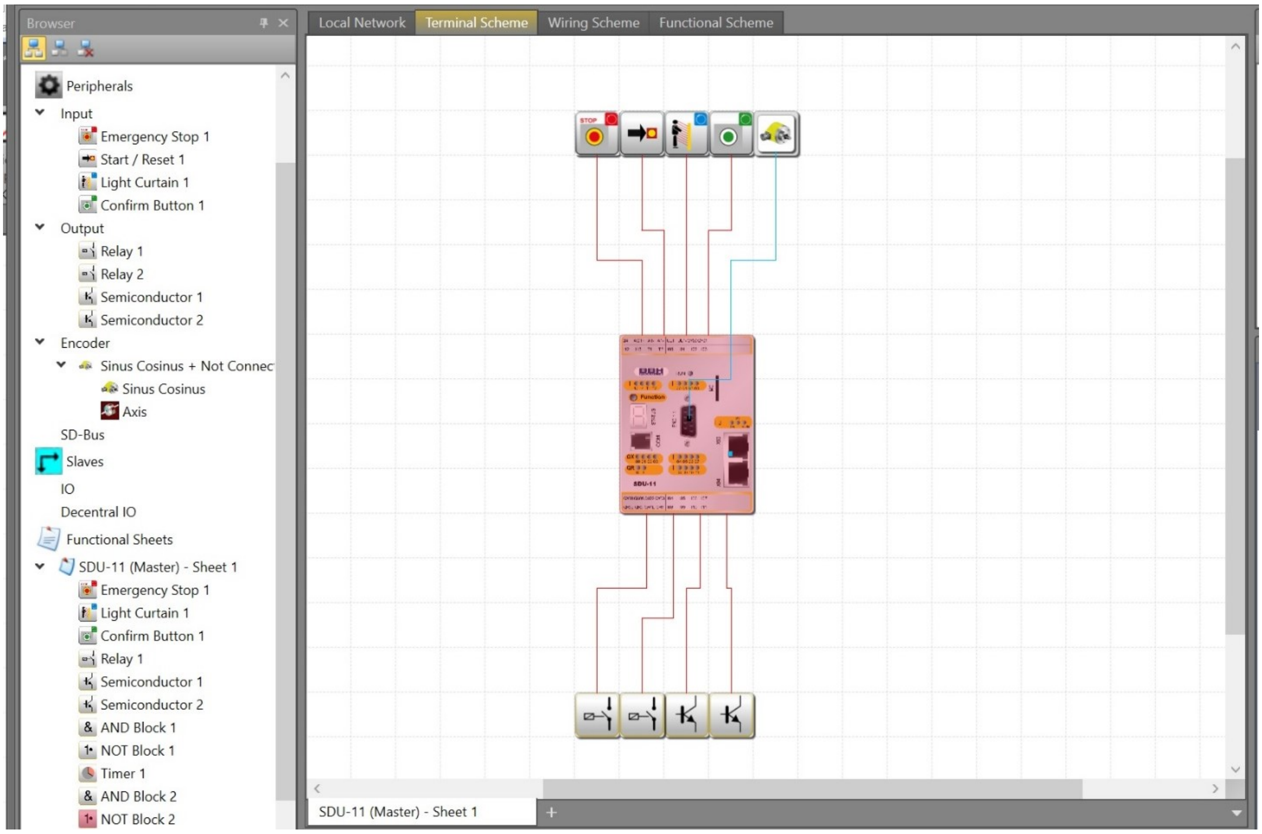


Figure 75 Terminal scheme with elements – slave 1

NOTICE:
Units marked in red still have elements that have not been used yet.
The elements marked in red in the left browser window have not been used yet.

NOTICE:
All available elements (marked in red) in the"browser" window must be used / incorporated. If elements are unused, the compilation in SafePLC² creates an error with a corresponding error message.

The other Slaves are assigned accordingly:

- *In our example:*
  Slave2: 2 buttons (Start) + 4 x Emergency-Stop + 2 switches (barrier + lamp)
  Slave3: 1 button (Start) + 2 x Emergency-Stop + 2 switches (relays to interrupt other functions)

For units with an encoder connection, also these encoder connections must be given. For this purpose, an encoder version can be selected from the library in the "Encoder" folder.

- *Our example:*
  sine cosine + not connected

4.4.7.2. Configuration of slaves in FSoE master

When the slave profile has been created, the FSoE outputs and inputs of the slave profile can now be adapted in the FSoE master configuration. For this purpose, connection points are added to inputs and outputs of the profile. The naming of the connection points is composed of the block names of the profile. Optionally the name can be changed in the properties of the connectors.

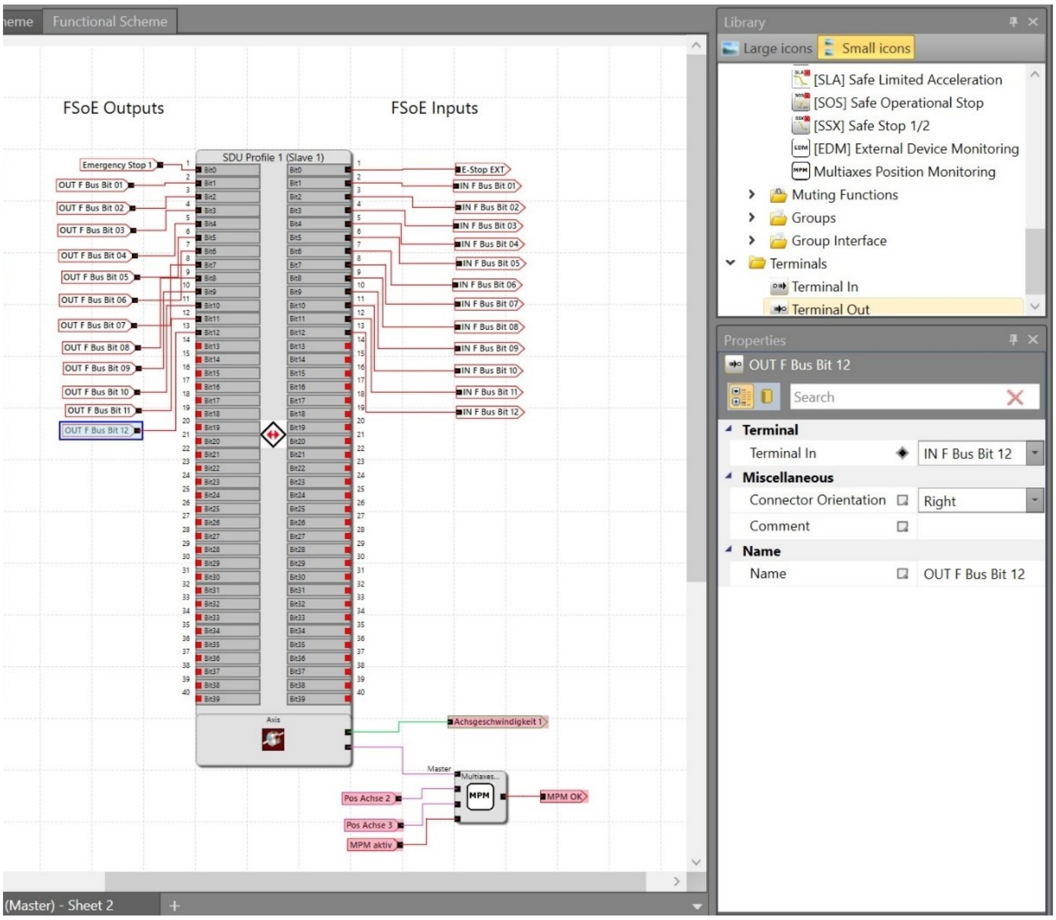


Figure 76 view of Slave profile (in FSoE Master configuration)

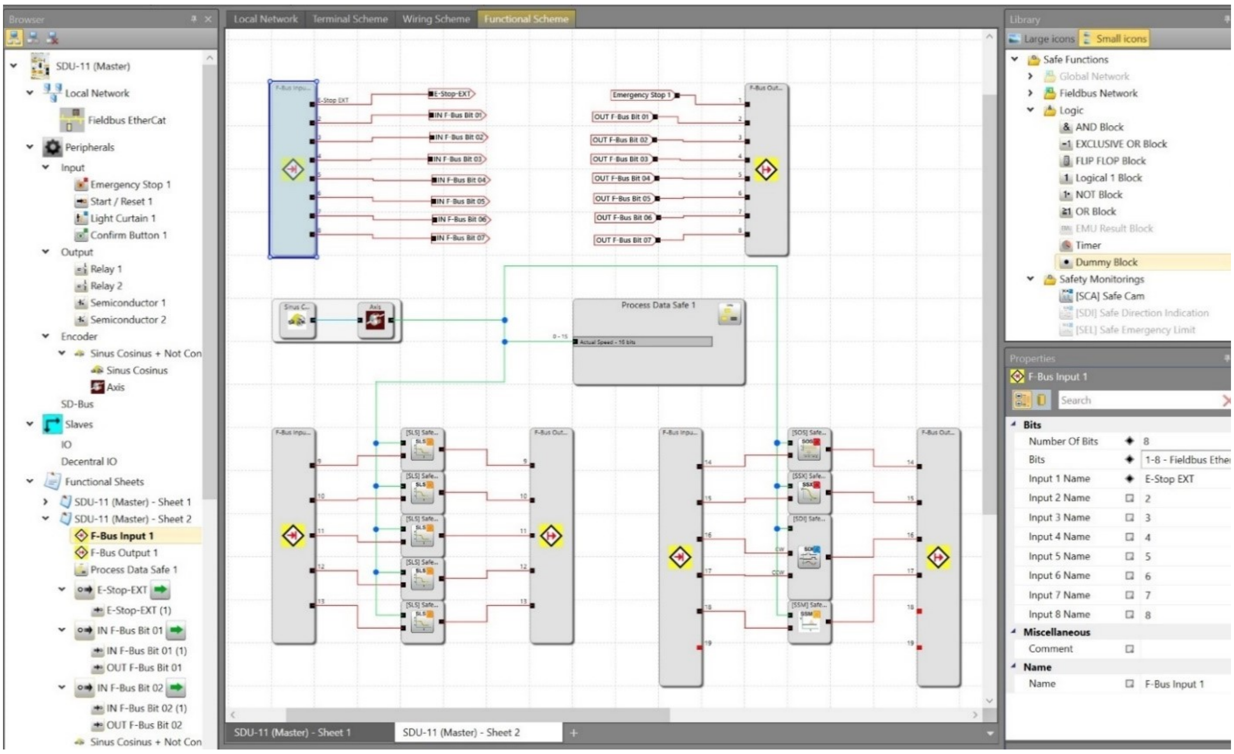


Figure 77 finished profile creation of FSoE inputs and outputs in FSoE slave document (Slave 1)

4.4.7.3. Slave profile types

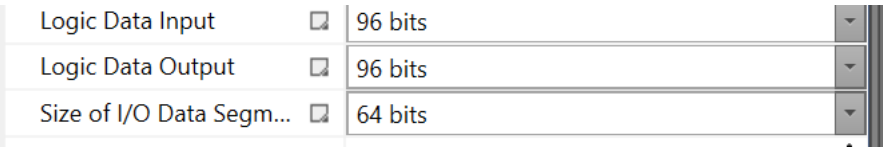
The slave profiles are configurable and depend on the structure of the respective application to be parameterized.

For the user specific FSoE slave modules, the profiles are fixed in the configuration and are based on the according ESI file.

For the SDU modules, the profiles can be configured depending on the application, the data size is set to 12 bytes by default.

NOTICE:
16 byte FSoE data is supported with certain FW versions (see release notes of the SDU modules).

The profile definition is done in the SDU module and can be configured by the user:



- Logic data
    - Configuration Size of binary data in byte steps (1 Byte to 12 Byte)
    - Configuration of respective input/output data

- Process data
    - Configuration Process data such as safe speed, safe position, etc. with data size 16 bit, 24 bit or 32 bit.
    - Process data only supported for output data.

After configuration, the user can export the created profile via the "Export Profile" button 

in the SDU configuration (*.pak) and import it in the SCU configuration (FSoE master document).
In the device property of the slave in the FSoE master document (SCU configuration) the created profile is now displayed and can be selceted. (see Figure 78)

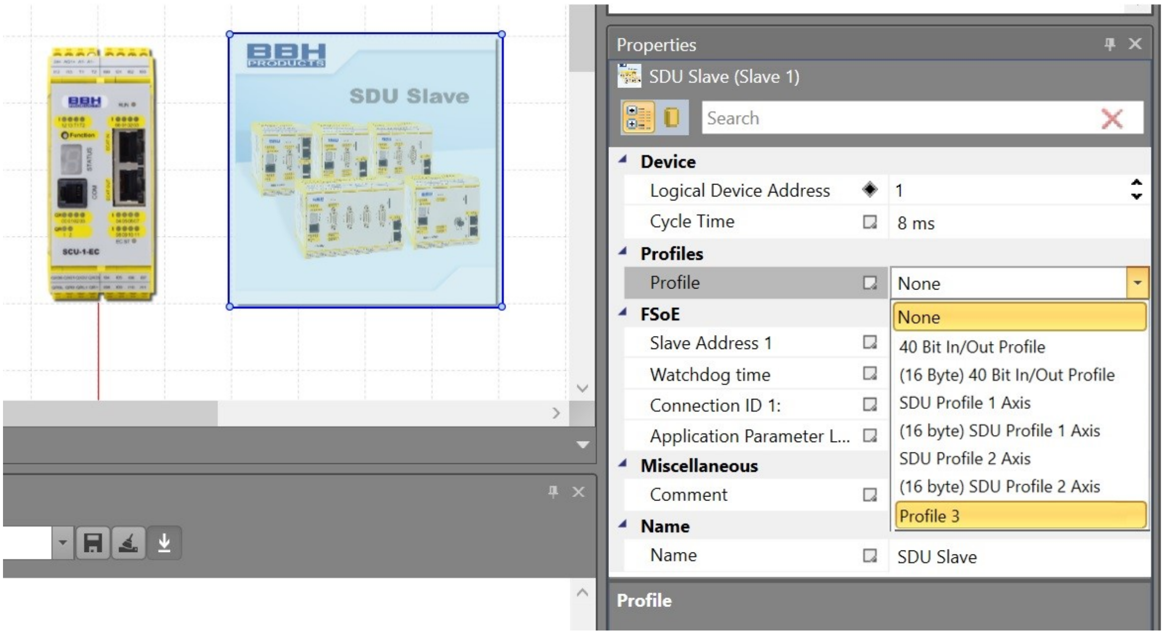


Figure 78 FSoE master document, Selection of the created slave profile (example SDU slave)

NOTICE:
Profile for 16 byte F data are only available from FW 5.1.3.x and NetX V1.15.

The profile can then be inserted and used in the functional scheme.
Depending on the profile, the SDU profile contains additional connections for axis position and speed.

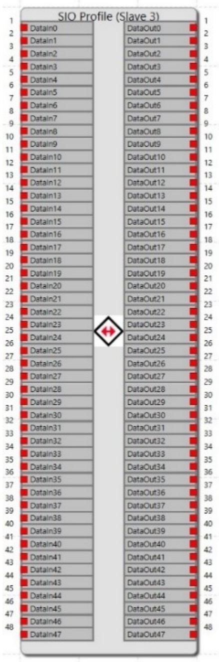


Figure 79 SIO profile (Function plan)

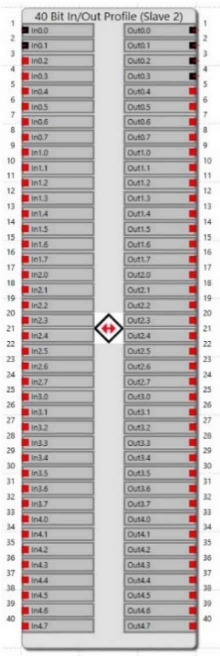


Figure 80 40 Bit In/Out profile (Function plan)

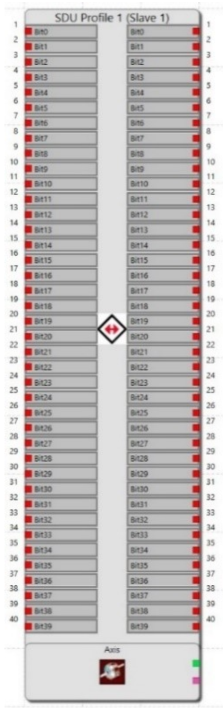


Figure 81 SDU profile (Function plan)

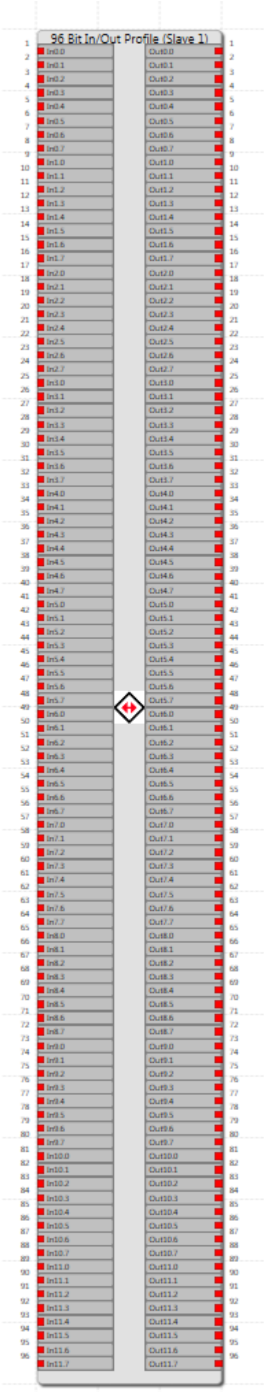


Figure 82 96 Bit In/Out profile

4.5. Determining the functions

The elements inserted in the wiring scheme are shown in the left browser window which can also be used in the functional scheme.

In the tab "Functional scheme" the (safety) functions can be determined.
These functions can be entered according to the SafePLC² programming manual.

For entering the functions, all inputs and outputs of all units in the window "Browser" (left) can be selected. The safety functions and the logic functions can be taken from the window "Library" (top right).

The options of the selected element can be determined in the property window (bottom right).

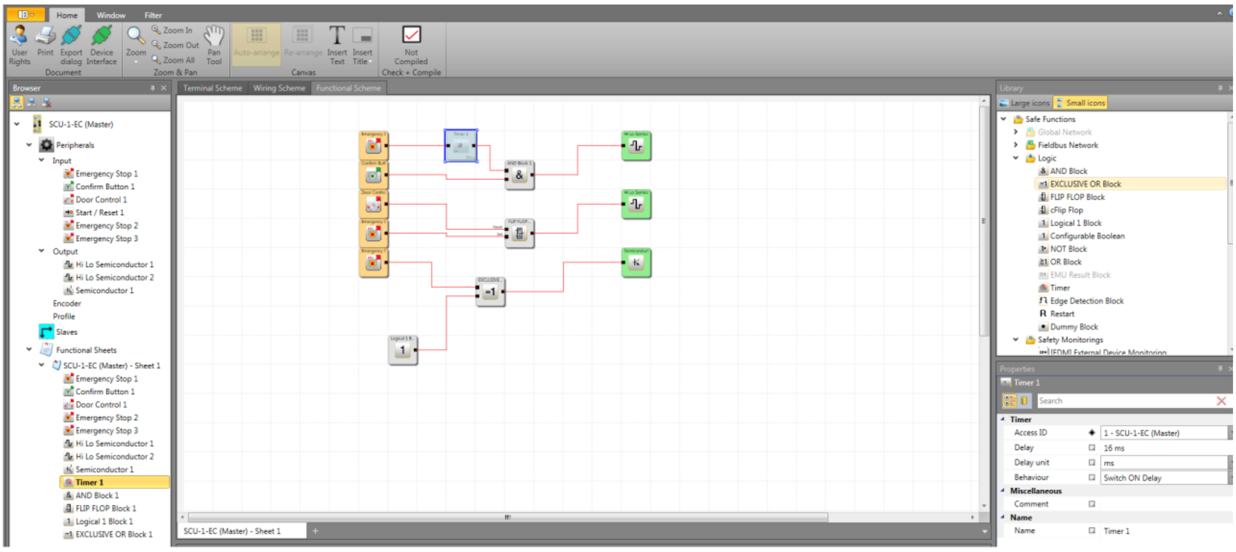


Figure 83 Safety functions

After all functions have been entered, and all elements are used, the program can be compiled, and it can be transferred into the Master units.
Afterwards, the SCU can be started.

4.6. Embedding of existing Slaves from external companies

4.6.1. General information

Slaves already included in the program SafePLC² are incorporated according to the following simplified procedure: These Slave-units are included in the library and can be integrated according to the chapter "Selecting the units / network components" just like the units manufactured by BBH. Via drag & drop, the units are drawn from the "Library" in the window "Terminal Scheme".

Together with the corresponding picture, the selected units are shown in "Local Network" and in the "Terminal Scheme".

SDC **EL1904 EL2904** **AX5805** **AX5806**

Figure 84 Library-Slaves SDC (Keba), EL1904, EL2904, AX5805, AX5806 (Beckhoff)

4.6.2. Adjusting the Slaves

The Slave units are selected and integrated according to the chapter 4.4.3 "Selecting the units / network components".

4.6.2.1. SDC (Keba)

The wiring scheme shows the SDC-Slave without contact points for inputs and outputs. The order ("Slave" + logical device adress number) is shown in brackets. The axes of the SDU can <u>only be parameterised via "Properties"</u>. Up to 3 axles can be selected.

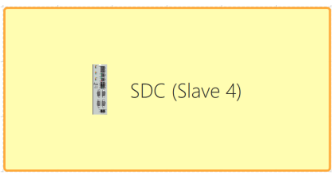


Figure 85 SDC-symbol wiring scheme

1. For the Slave-unit SDC a module address must be indicated. The module address must be taken from the FSoE-network plan.

The number of axles can be specified in the field "Axles".

Afterwards, the Profiles of the different axes can be selected in the field "Profiles" .

The profiles to be selected can be imported before.

For every axle x individual parameters can be determined:

- FSoE-addresses via the field "Slave Address x"
- Connection-ID via the field "Connection ID x"
- CRC via the field "SRA CRC x"

The Watchdog time in the FSoE-network can only be determined equally for all axes of the SDC. – The SDC is valid per unit in the network.

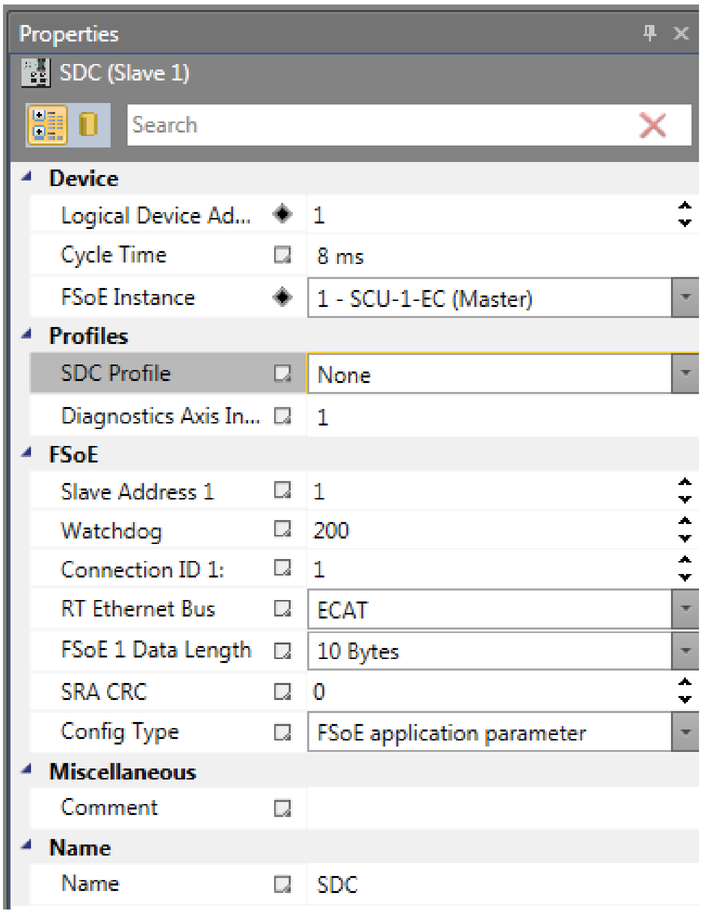


Figure 86 SDC-parameters

2. The encoder settings for the respective / selected axles must be set as follows:

   - By clicking on the desired axis symbol, the properties can be shown in the window "Properties".

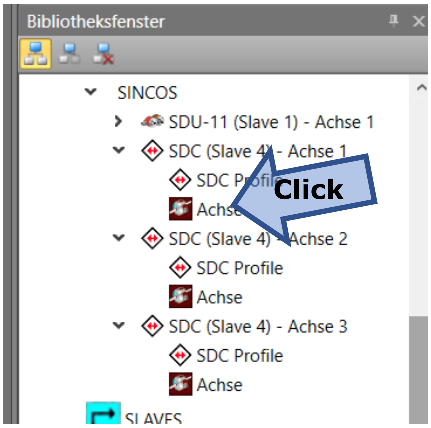


Figure 87 SDC axes

- The properties of the selected axis can be changed via the window "Properties" (bottom right).

  The settings must be set according to the data sheet of the encoder.
  These are e.g. type, scaling, direction, transmission ratio, position and speed limits, switch-off thresholds, acceleration, etc.

3. After all encoder parameters and all network parameters have been set, the Slave-unit SDC is integrated.

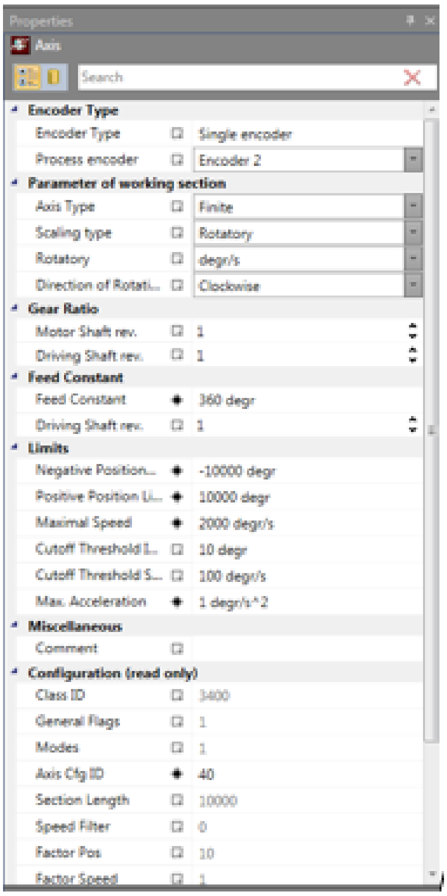


Figure 88 Properties of the SDU axle(s)

4.6.2.2. EL1904, EL2904, AX5805, AX5806 (Beckhoff)

After one or more Slave-units, type EL1904, EL2904, AX5805, AX5806 have been integrated, these Slave-units are shown in the Windows “Local Network” and “Terminal Scheme”.

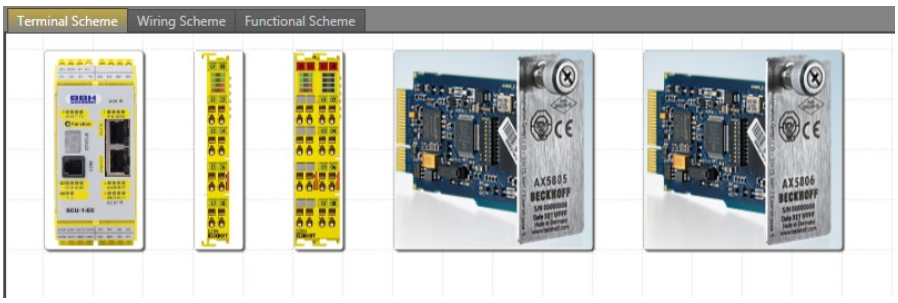


Figure 89: EL…. und AX…. In the terminal scheme

In the wiring scheme, one sheet per inserted unit is created.

According to the inserted Slave-unit, the wiring scheme offers contact points for inputs or outputs.

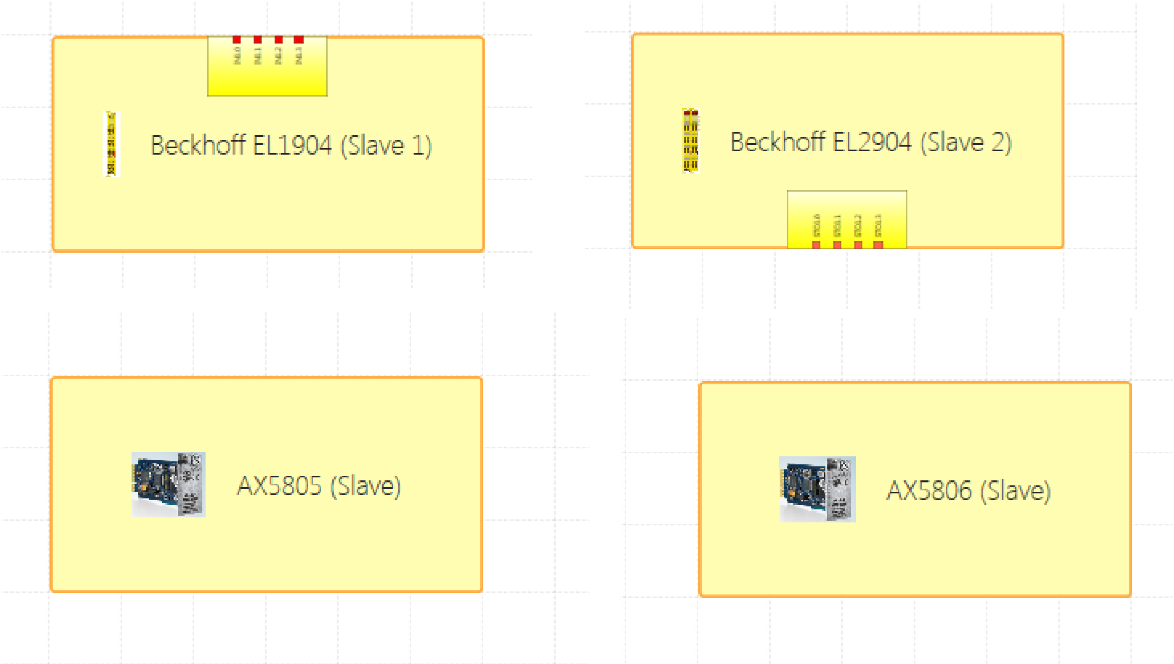


Figure 90: IO-Slave-units EL x904 and Axle-Slave units AX580x in the wiring scheme

In the wiring scheme the inputs and the outputs of the Slave-units EL x904 can be wired accordingly.

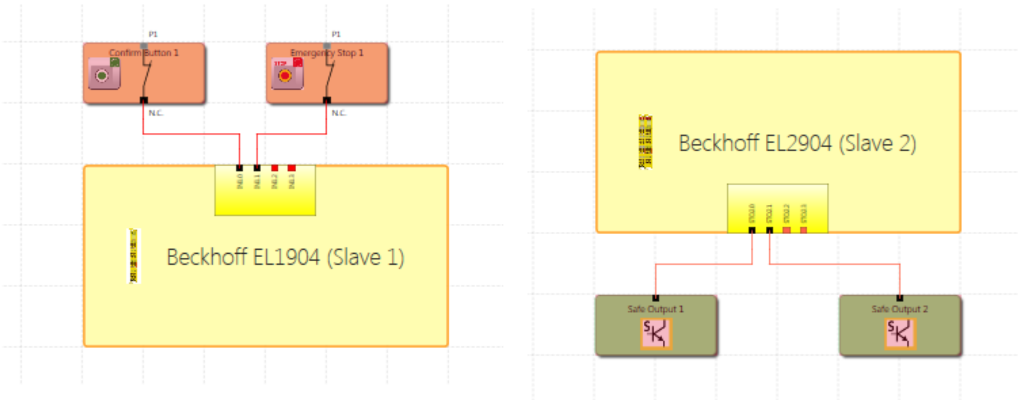


Figure 91: IO-Slave-units EL x904 with ternimal assignment

The properties of the respective units can be selceted in the window „ Properties “. Example of properties: modul address (according to network map), cycle time (longest period of all units in the network), unit address (according to local network) , Slave address (according to network plan – indicates the order in the network), Watchdog time (max. telegram duration per unit), connection-ID (ID for the sequence of the messages) etc.

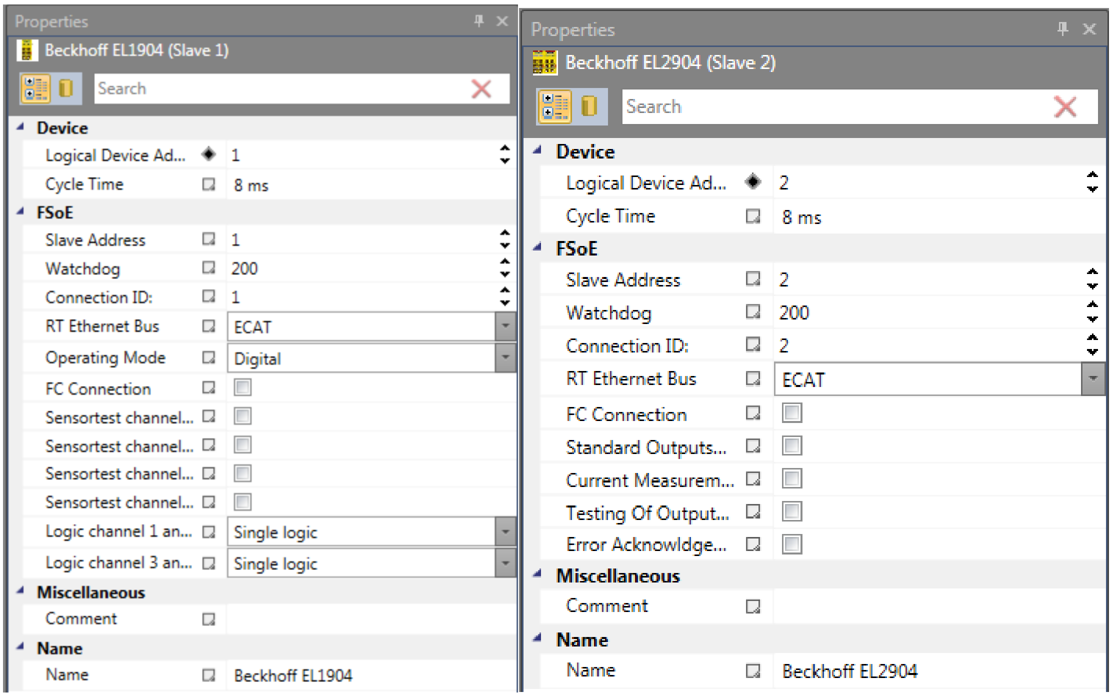


Figure 92: Properties of the IO-Slave units EL x904

The axis of the Slave unit AX580x can only be parameterised via "Properties". It directly evaluates the information coming from the inverter.

The properties of the axis-Slave units are identical, and are more manageable because the connected axis is selected automatically:

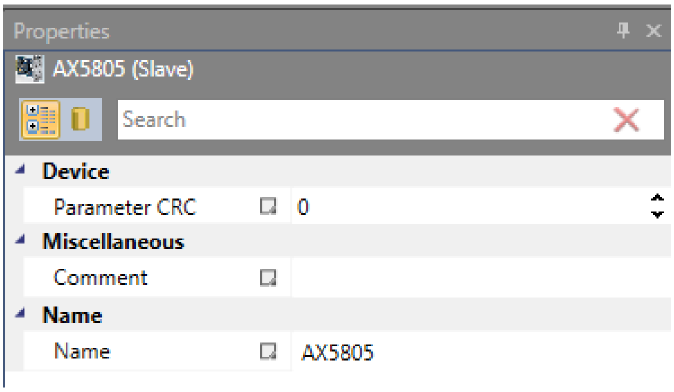


Figure 93: Properties of the axle-Slave-units AX 580x

After the Slave units have been parametrised, they are available in the network.

4.6.3. Embedding not pre-defined Slaves

- ➔ This option will be available in the later versions of the manual.
- ➔ To embed not pre-defined Slaves, please contact the manufacturer

4.6.4. Adding input elements

The input elements create the digital connection between one or more connected sensors or further subordinate switching devices in the ***system***. Except the selection switch to select the operating mode, every input element provides one logic output signal "0" or "1" for further processing in the PLC.
The input elements are automatically added to the terminal scheme or to the wiring scheme where they are processed. In the functional scheme, the input elements are inserted from the browser.
The resource management of the function block elements of the SCU system manages the available elements whose number can be restricted.
If no further elements are available during the programming of the terminal scheme, no blocks are available to add the corresponding modules or function blocks. The available blocks are displayed in the library. These resources can be released again by deleting the corresponding function blocks. To delete a block, select the block from the browser or from the work surface. Click on it with the right mouse button, and select "Delete" or press "Del".

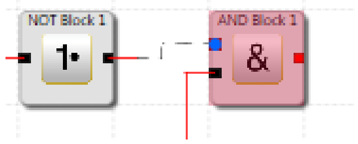
4.6.5. Inserting output elements

The output elements create the digital connection between one or more external circuit in the ***system***. This element of the functional scheme has direct influence on the drive to be monitored, Furthermore, it can be determined, how external switching devices shall be monitored. Every output element is controlled by a logic input signal "0" or "1" via the functional scheme.
The output elements are added in the terminal scheme or in the wiring scheme. The output elements are edited in the functional scheme. In the terminal scheme, the output blocks are dragged from the browser and dropped.
By the automatic monitoring of the resources of the function blocks of the SCU module, only the available elements are enabled in the program. If no resources for the monitoring program are available in the safety module the commands to insert the corresponding components or function blocks is deenabled (The library functions are not available.). This is the case if p. ex. all digital outputs of SCU module are occupied. These resources can be released again by deleting the respective function blocks.

4.6.6. The logic modules

These modules are the basis for the creation of a program for the safety application. They enable the logical connection of the inputs with monitoring functions to the outputs. Only in the "Funktionsplan" [Functional scheme] view, it is possible to insert logic modules. Otherwise, the corresponding module commands are disabled. This applies if the resources for a module are already exhausted, e. g. after all timer modules have been inserted. You find a description of the logic modules in the chapter "Logic functions".

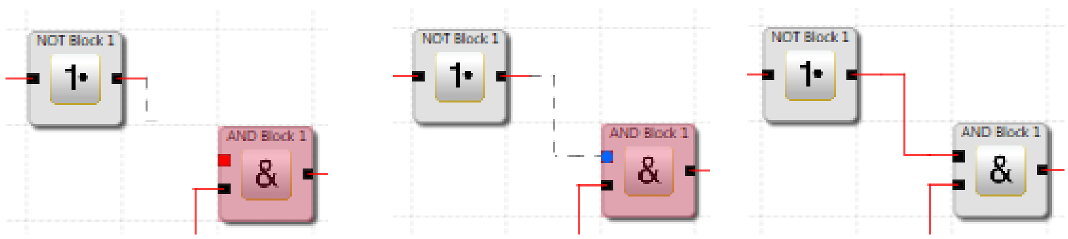
4.6.7. Circuit



The assignments in the functional scheme are achieved by linking the input connectors and the output connectors of the functional modules. If necessary, one module output can be connected with the inputs of other modules, but one module input may only be assigned once. Furthermore, for technical reasons certain module groups cannot be interconnected. In case of an invalid connection, the programme displays a corresponding warning.
Only rectangular control points can be created, i. e., the connecting lines always run horizontally or vertically.

<u>Creating the connection:</u>

1. Press the left mouse key to select a start connector.
2. The user can define the root and breakpoints by clicking in the free range.
3. With a second click the target connector is selected.
4. If "automatic arrange" has been enabled, the connection and the blocks are arranged automatically.



NOTICE:
Connections can only be selected with the mouse, and they can only be deleted with the Del-button.

Tip: If all connections of a module shall be deleted, the corresponding function block can be deleted. In this case, the connected terminals are automatically deleted.

The program automatically creates a new connection in the terminal scheme and in the wiring scheme. The program draws the connection by inserting additional junctions (break points) on the basis of a bisection algorithm.
The graphical representation can be adjusted and the general representation can be optimised by moving the function blocks (if automatic arrangemetn has been deenabled).
In complex diagrams, it may occur that a connecting line crosses a function block. This has no effects on the internal function of the connection.

NOTICE:
Not in all diagrams and charts the connection is created automatically.
The drawing of user-defined connecting lines is also supported. User-defined connecting lines remain unchanged until the shifting of a connected function block forces the recalculation of the controls points (cf. the button "Auto arrange/re-arrange").

Add segment / Add junction
To add a segment (junction/segment) to the connecting line press the right mouse button and select "Add segment" or "Add junction".

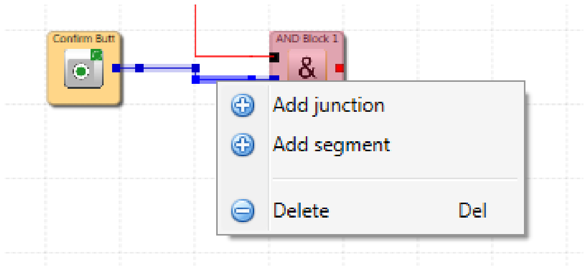


Figure 94 Add segment

To add a node, double click on the connecting line, or press the right mous button and select "Add junction".
By selecting the conection the user can define the connection and change every segment.

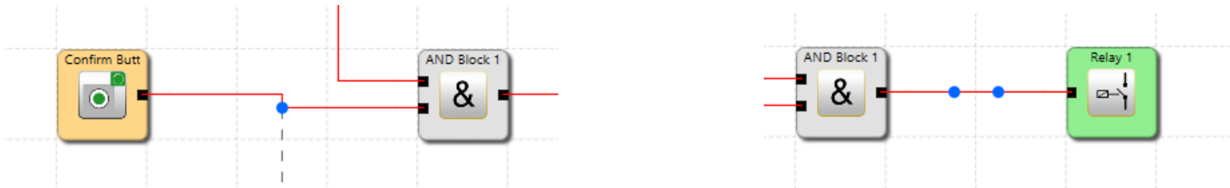


Figure 95 Added junction or segment in functional scheme

NOTICE:
If the segments of a connection are adjusted, they are automatically unified.
Junctions can only be entered with rectangular connecting lines, i. e., the connecting lines always run horizontally and vertically. The program connects the entered points until the drawing command is completed.

Tip: Visual corrections of the functional scheme should be carried out shortly before the blocking of the functional scheme. Then, the layout is complete, and the blocks no longer need to be moved.

4.6.8. Using function groups

Function groups connect several function blocks to a superordinate logical structure. This matching block group is created within the function group, and is connected via this block.

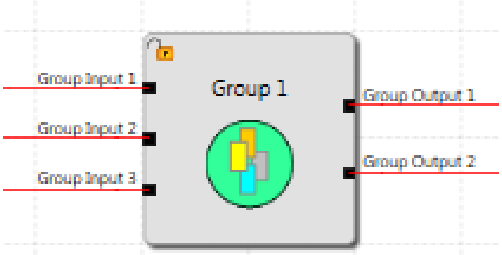


Figure 96 function group

By this grouping, the function block diagram obtains a clear structure. Via the Export function or the Import function, it allows the creation of an individual function library.

By clicking on the function group, another "Groups" tab appears, in which function blocks can be inserted.

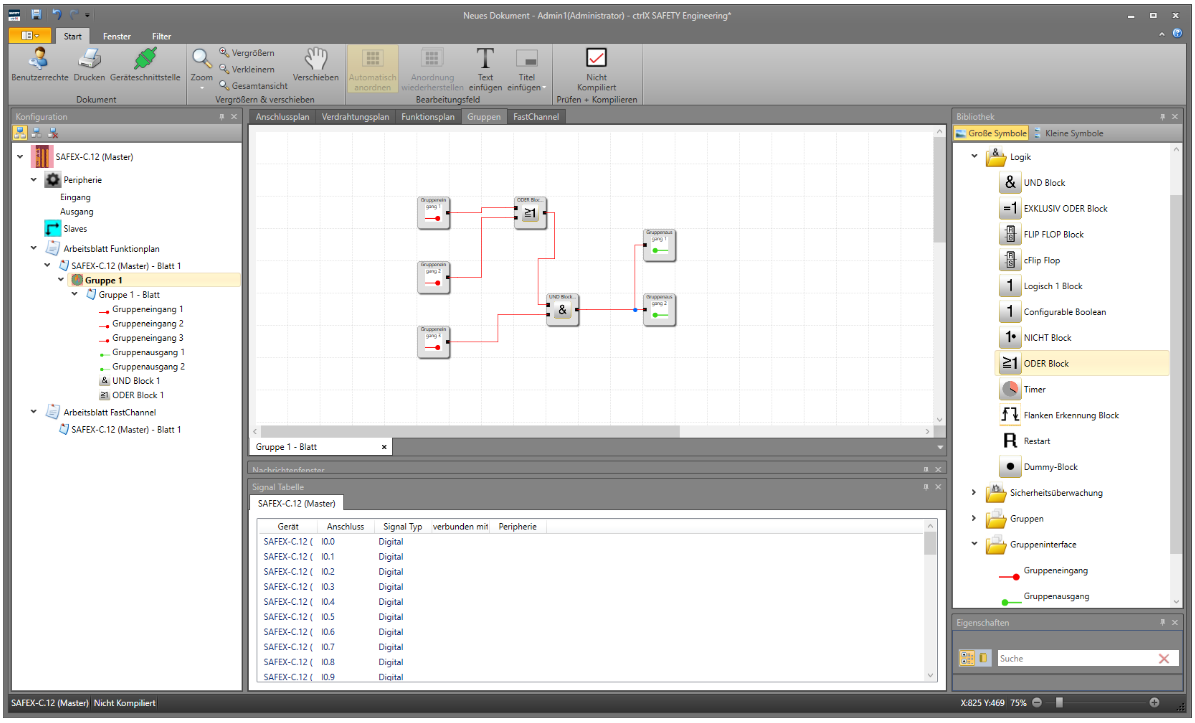


Figure 97 Function blocks in Tab „groups"

The connection between „functional scheme" and „Groups" is established via "Group input or Group output".

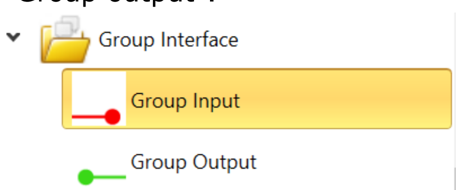
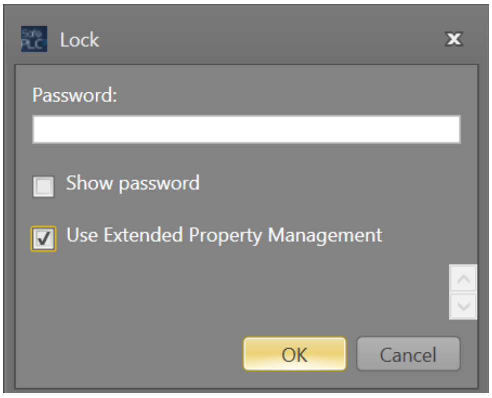


Figure 98 Folder „groups interface" in the library

After finishing the group, it can be locked for modifications by password. By right mouse click the option lock can be selected. After that, a window appears that requests a password entry. With this password the function group can be locked/unlocked.



4.6.8.1. Group versioning

It is possible to assign versions for groups. These are then displayed as a property for closed groups.

NOTICE: To display the versioning as a property, "Extended Property Management" must be selected.

The following steps are necessary:

1. add the version module to the "Groups" function block diagram.
2. define the version number in the property window that now appears.

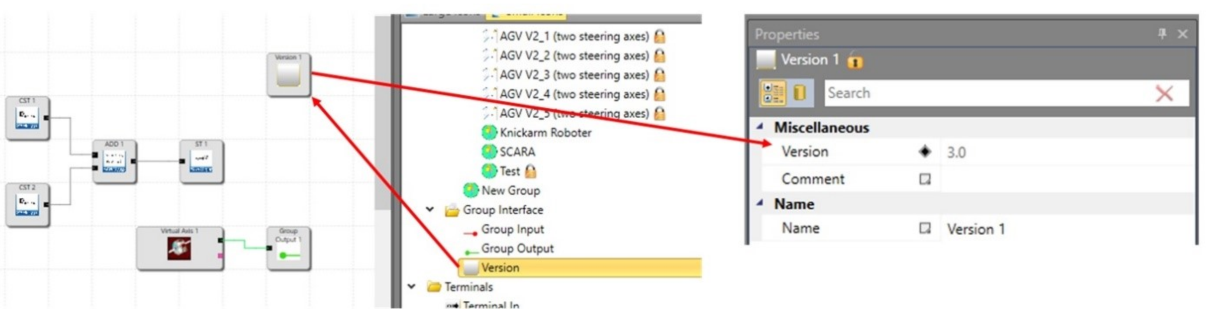


Figure 99 Function group, adding a "Version" block

3. When locking the function group, checkmarks must be set for "Extended Property Management" and the version of the version module.

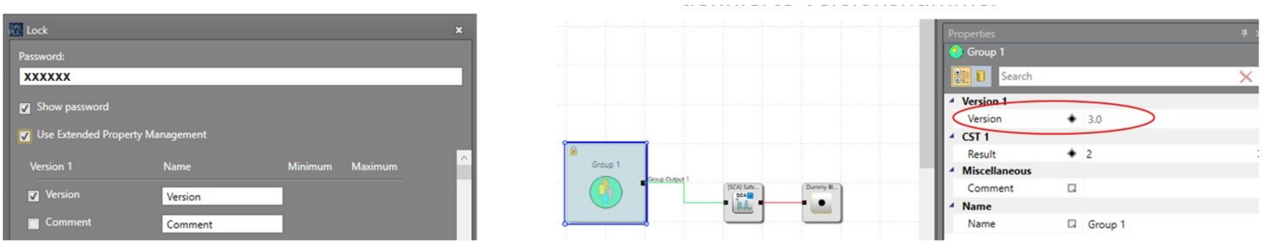


Figure 100 Function group, Setting during group lock

xxxxxx serves as a placeholder of the respective password

4.6.9. Creating a program

After the program has been completed, compiling can be started by activating the compiler . The results are shown in the message window which automatically appears during compilation. After the compiler has been called, the compilation undergoes the following steps.

Verifying open connections
SafePLC^2 ensures that all connections between function blocks can be opened. Connectors that are not connected are recognised as errors.

Verifying of non-referenced "Anschluss ein" [Connection on] and "Anschluss aus" [Connection off]
SafePLC^2 ensures that all terminal blocks entered in the functional scheme are used. Open references are recognised as errors.

Verifying the value ranges of the monitoring functions
Before creating the instruction list, the ***SafePLC^2*** checks if the parameters of the monitoring function lie within the range of the current encoder configuration. Otherwise, an unnoticed overflow of the range may occur if the encoder settings are changed with a monitoring function.

Creating an instruction list (IL)
The IL code that has been created on the basis of the function groups is displayed in the message window where it also can be verified. Code segments of the corresponding function blocks are identified via the corresponding block ID.

Creating the OP code
Generating of a machine-readable code for the system. Afterwards, this code is transmitted together with the parameters.

Message window

All results of the compilation are displayed in the message window. If errors are detected, the message window appears automatically.

Tip: Use the "Quick Jump" to jump directly to the respective block in the diagram by double clicking on the displayed *block ID* in the message window. Thereby, the corresponding function block can be easily identified in case of an error message.

CRC backup

After a successful compiler accomplishment, a total of three CRC signatures is created.

- CRC for equipment configuration: signature for the program and for the parameter data.
- Parameter CRC: signature for parameter data
- Program CRC: signature for the program

NOTICE: If an existing SafePLC² program is opened with a later version of SafePLC² , this program is ported. To grant the complete porting, an additional step is absolutely necessary.

Important:
This display only serves for information purposes and must not be used for safety-relevant documentation!

4.6.10. Transferring the program to the device

This section describes the transfer of data and of the program to a safety module. If the interface has been started (via the device's button 

), the device's interface toolbar appears. The toolbar contains connection tools and transmission tools. You find a description of the device in the chapter "Device Interface".
With "Connection Settings" the window for document administration opened where the tabs "Document" and "Document" and "Master Device" are situated.

In case of several devices, each device is stored in a different tab. In the "Document" tab, the user can add the developer's and enter a comment. The tab "Master Device" consists of device information and connection settings. This menu can also be called via the button "Document Properties" in the menu ribbon menu.

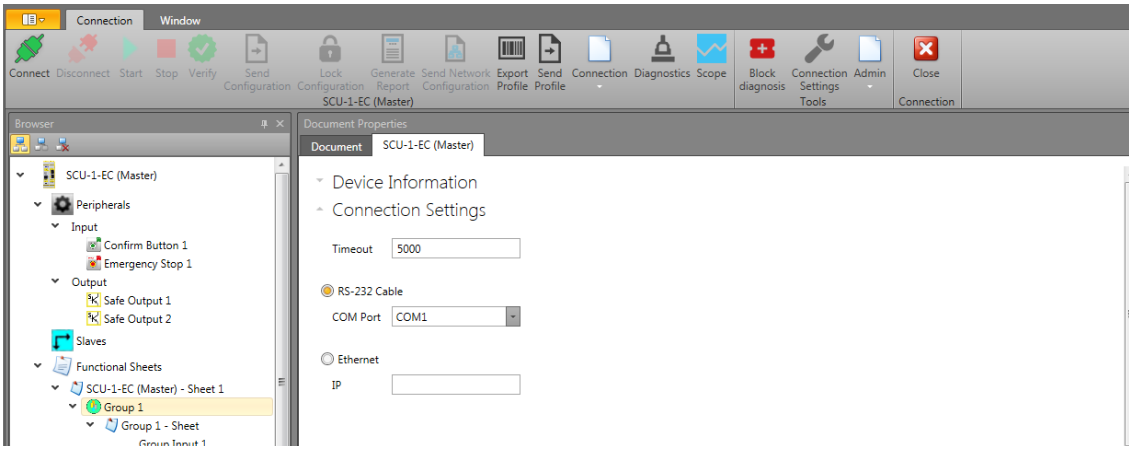


Figure 101 document properties, connection settings

More detailed current transmission statuses or errors that may have occurred are displayed in the message window. Due to restricted space, this window is not enabled automatically with every message. Thus, as much of the functional scheme as possible can be displayed in the diagnosis.

NOTICE:
The standard connection between PC and SCU system is based on an USB/RS485 interface.The corresponding driver is automatically installed with the installation of the SafePLC². If the driver is not installed automatically, it must be installed manually. You find the driver in the installation directory of the ***SafePLC²*** programming surface (directory RS485_USB_Treiber).

4.6.10.1. Connection settings:

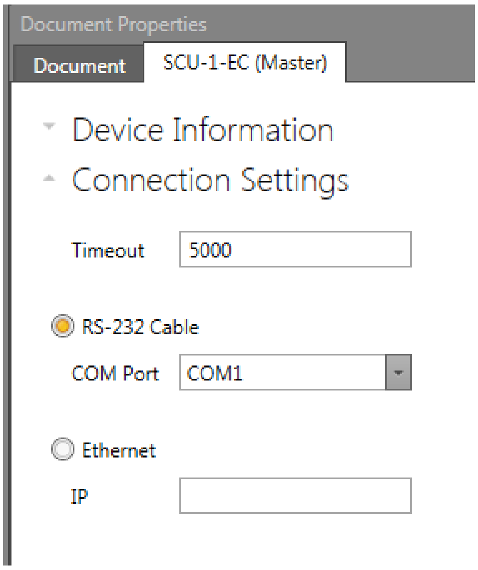


Figure 102 connection settings

Timeout
Communication timeout can be set in milliseconds.

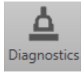
RS-232 Cable
The COM interface used by the Windows driver must be configured.

Ethernet (TCP/IP)
Can be Ethernet over EtherCAT including routing via controller or
Standard Ethernet directly to Ethernet interfaces.
The IP address must be configured.

<u>PC sided separation</u>
After 5 s at the latest, the system detects that the connection no longer exists and cannot be restored automatically if the connection shall be re-established.

<u>SCU-sided separation</u>
After 10 s at the latest, the system detects that the connection no longer exists. However, the connection is re-established automatically if the physical connection is re-established.

4.6.11. System and logic Diagnostics

After the device interface has been enabled, the "Diagnostics" button  is available. If you click on the „Diagnostics" button, the diagnosis window appears. The diagnosis cannot be performed simultaneously with the range function.

NOTICE:
A correct diagnosis requires an adaption of data between functional scheme and equipment configuration. A missing functional scheme or inconsistencies between the existing functional scheme and the equipment configuration only allow a restricted diagnosis. In this case, the tab "Diagnose of Function Block" is not available.

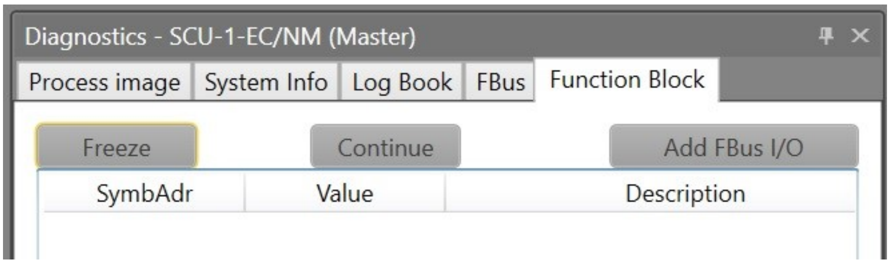


Figure 103 Diagnostics window of functional module

The diagnosis window consists of the following tabs:

Process image: Shows the statuses of all addresses of the input image and the output image in the SCU module. The CRC of the active configuration is displayed together with the status of the internal transmission meter. The meter reading increases with every transmission to the SCU module and, can serve as reference for documentation purposes.

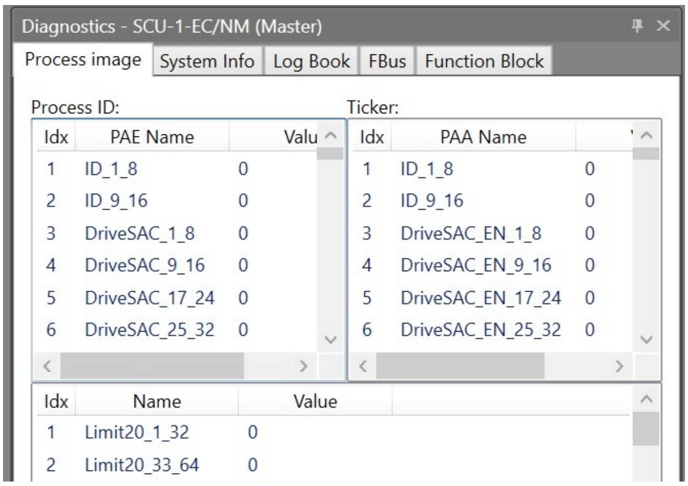


Figure 104 Diagnostics window, process image

System info: System information on the SCU module:

Parameter	Description
Configuration – CRC	CRC signature for the parameters
Program-CRC	CRC for the program
Transmission meter	Status of the internal transmission meter. The meter reading increases with every transmission to the SCU module,and can serve as reference for documentation purposes.
Serial number	Latest serial number of the equipment
Version number	Version number of the firmware
FPGA version	Latest FPGA version
HW version	Latest Hardware version

Logbook: Display of general information of the status of the SCU module as well as reading out the errors and diagnostics.

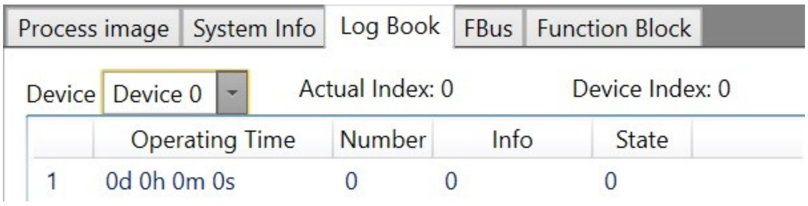


Figure 105 Diagnostics, logbook parameters

Parameters	Description
Operating Time	Operating hours counter in hour
Number	Error/info number [display in decimal]. Error/info number see error list
Info	Additional display [Value in decimal]
State	Status 1 – Fatal Error 2 – Alarm 3 – Info

Encoder position: shows the position values for encoder A and encoder B, transmitted by the encoders. You can mark the current position with the “Snapshot” button. The program shows the distance parameters from the registered position.

Encoder interface: displays the voltage difference of the output modules and the status of the input jumpers in the encoder interface. If one of the stress values is 0, the encoder is defective or not connected. The value of the input jumper must be interpreted differently.

With incremental encoders:

0: = jumper OK

1: = error

With SSI encoders:

0: = receiver operation

1: = SSI encoder operation

Alarm mute: displays the active functions for alarm mute.

Function block: allows the selective monitoring of previously selected function blocks. To select function blocks from the work surface for diagnosis, use the button „ Add FB I/O" or "Block diagnosis". Via these tab the logic states "0" or "1" can be displayed in the functional scheme.

4.6.11.1. Diagnosis process in the function block diagram

The most important prerequisite for diagnosis is to start the program. The "Start" button in the symbol bar "Connection" is grey.

4.6.11.1.1. Diagnosis in the work surface

The diagnosis in the work surface is only possible if the user has selected the tab „ Function blocks" in the dialogue window. After the tab "Function blocks " has been selected, diagnosis starts automatically.

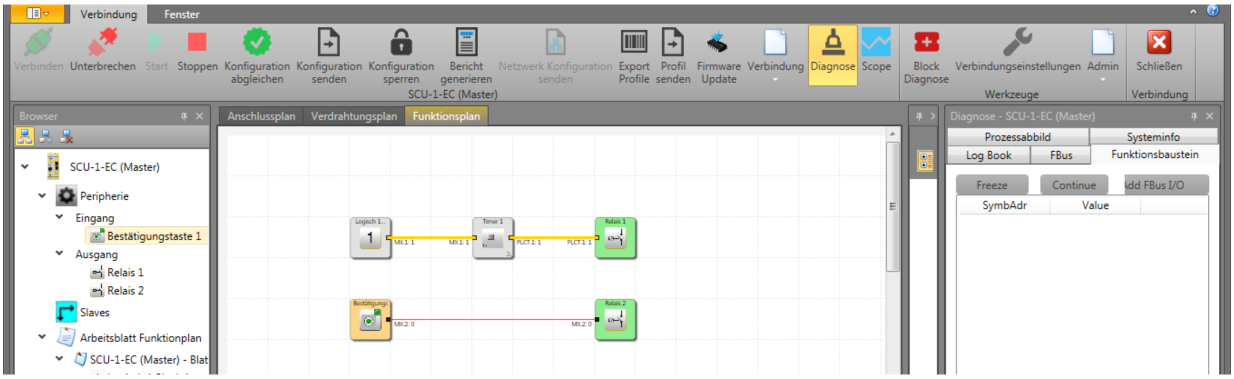


Figure 106 Diagnosis procedure in the work surface

When doing a diagnosis in the work surface, the current input statuses and the current output statuses are shown in the plan according to their logic state "0" (red line) or "1" (yellow line). The logic state is indicated in the work surface beside the connection-ID. When the tab "Function Block" in the diagnosis window is left, and another diagnosis type is opened, i. e., another tab (e. g. "Encoder Position") is opened, the diagnosis information is hidden from the work surface.

4.6.11.1.2. Diagnosis in the Function Block tab

It is possible to do a diagnosis for selected blocks.

Selection of the data to be displayed

In the "Function Block" tab it is possible to select function blocks whose status shall be monitored. Function blocks can be selected in the work surface.

After the selection, click on the "block Diagnosis" button  . After a click on the button, the blocks are transferred int o the monitoring list.

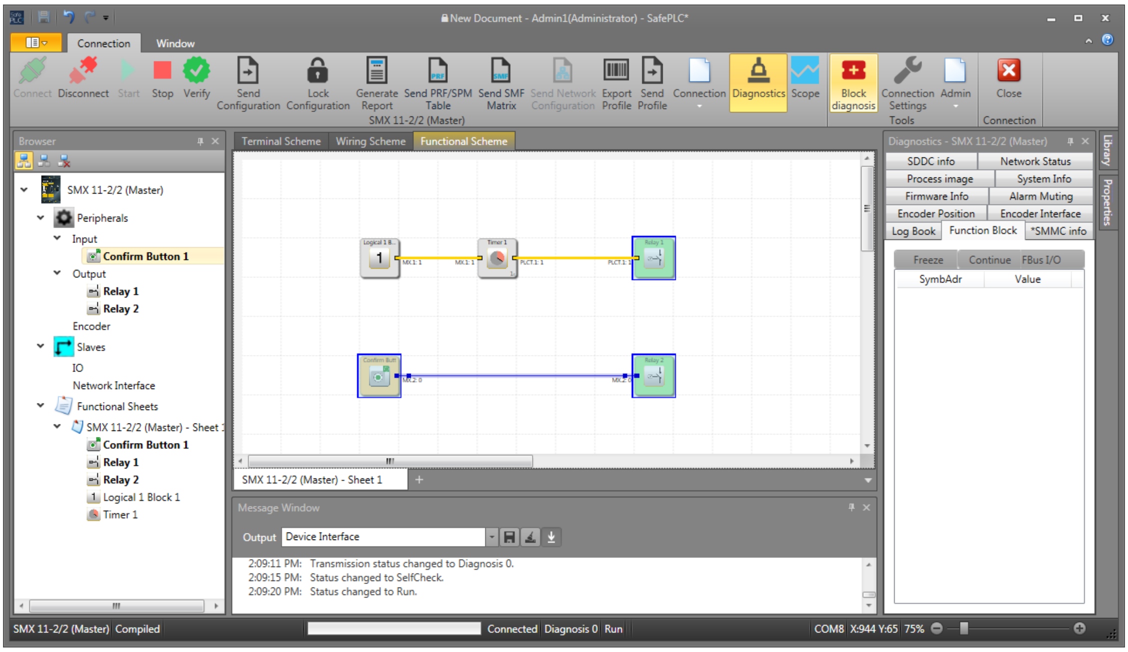


Figure 107 Selected blocks on the work surface

The monitoring list shows the symbol address, the logic value and the description of every added block. If the diagnosis of a function block is carried out, the current input statuses and the current output statuses are shown in the selected block according to their logic status "0" or "1".

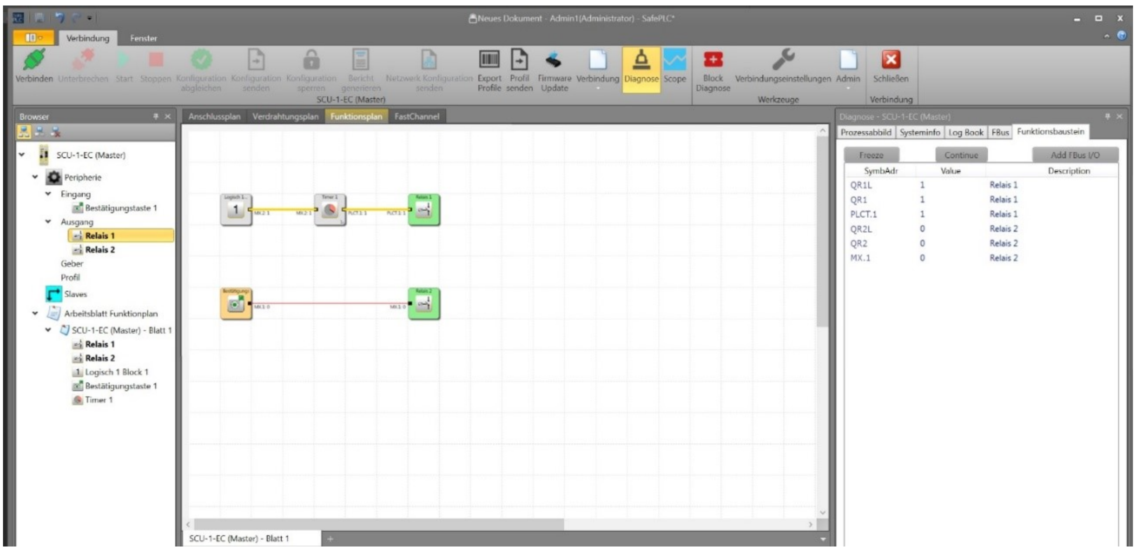


Figure 108 Display of the logical state of inputs and outputs in the selected block

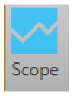
A block can be removed from the monitoring list by marking it and pressing the Del button.
By double clicking on a list entry, the corresponding function block is shown in the list.

NOTICE: The symbol addresses shown in the list are also used for compilation and in the validation report.

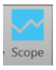
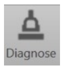
Tip: With the command "Select all" in the context menu (right mouse button) all files in the functional scheme can be selected.
Now, the selected files can be diagnosed if the information in the functional scheme matches with the functions in the actively connected SCU system.

NOTICE: The integrated trouble shooting function requires an intensive data transfer between the SCU system and ***SafePLC²***. This can cause a temporary delay in the display of data. Thus, recent status changes at module outputsmay not be detected.

ATTENTION:
If the SCU module is set into alarm mode, the process representation is no longer updated. The change of input levels is no longer valid, and is no longer displayed in the diagnosis.



4.6.12. Scope diagnostics

Setting the drive monitoring requires an exact knowledge of the process data from the point of view of the SCU system. Knowledge of the chronological sequence of speed, acceleration and position is paramount. Only in this way, correct threshold values and restrictive parameters can be set.
The range function is available in the dialogue window oft he device interface. You select the range monitoring by activating the "Scope"  button. If the diagnosis  button is enabled, the "Scope" button is deenabled immediately.

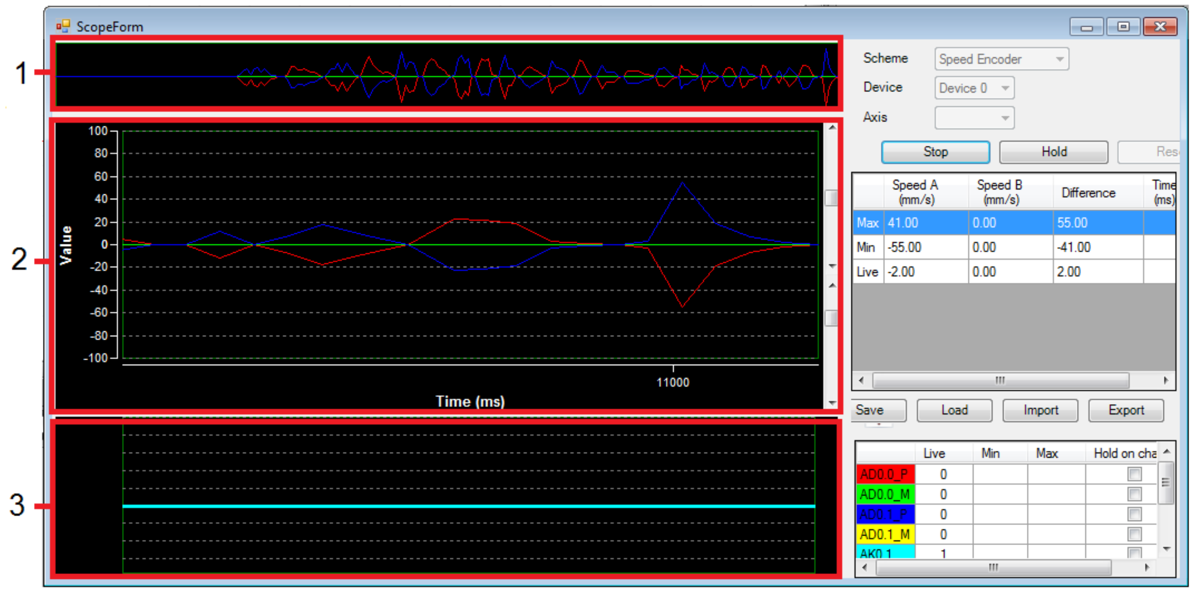


Figure 109 Scope, Range view in the device interface

1- Overview scroll bar

2- Main diagram window

3- Signal output window

All available graphic functions read the required process data ONLINE from the active SCU via the communication station for time-based display. Current values are displayed on the right side of the range monitoring, move to the left during the recording and then disappear from the left edge of the screen. Even if these data have disappeared from the visible window, they are saved in a cache memory. They can be made visible again by dragging the scroll bar beyond the main window.

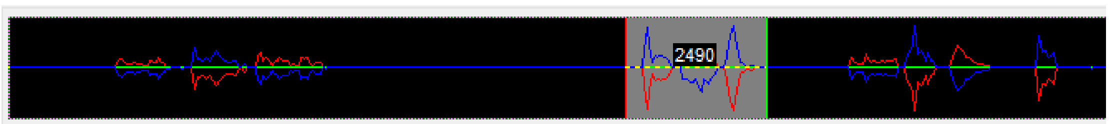


Figure 110 Overview of the scroll bar for the main diagram

Scheme: With the "Scheme" function, the current context for the desired visualization" can be selected. The context of the displayed curves changes according to the selection of the plan. To the curves a colour is assigned that is specified in the key.

The following selections are possible:

- Encoder data
- Speed encoder/sensor
- Position pass ***- currently not supported***
- Speed position
- SDI (Safe Direction)
- SLI (Safe Limited Increment)
- SEL (Safely EMERGENCY Limit)
- SRS ***- currently supported by Sensorbox***
- SRX ***- currently supported by Sensorbox***
- Data of block SSX (Safe Stop 1 / Safe Stop 2)
- SLP (Safe limited Position)
- SLS-Filter (Safe limited Speed)
- SCA-Filter (Safe CAM)
- SCA Position
- SLA (Safe limited Acceleration) ***- currently not supported***
- SOS (Safe Operating Stop)
- MPM1 (Multiposition Monitoring 1)
- MPM2 (Multiposition Monitoring 2)
- SWM

Depending on whether the encoder displays planned time values or position-dependent values, the X-axis shows either the expiring control time or the measuring length configured in the encoder. The Y values refer to the selected plan.
The plan cannot be changed during a current measuring.

Device: selection of the device.

Axis: If several identical functions are used, these can be selected and can be shown separately in this selection. The values of these measuring data are shown for the relevant cursor positions.

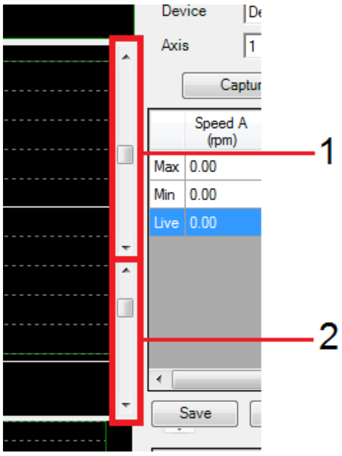


Figure 111 Scaling the diagram via the slide bars

By scaling the displayed diagram function, it is possible to adjust the Y values in the individual curves via the slide bars 1 or 2.

Slide bar 1: changes the visible range of the Y values in the diagram.

Slide bar 2: changes the displayed maximum range oft he Y values in the diagram.

Collection / Stop: starts or stops the recording.

Pause: Press the "Pause" button to stop the displayed values in the main diagram. The data are still available in the cache.

Reset: reset of diagram values and process data.

Tip: With a double click in the main diagram window, the cursor is inserted at this position. Thus, the cursor is added in the value table for optional measurements.

Hold on change: If the "Hold on change" button isactive, the recording stops 2 seconds after a slope change of the respective output (see above). This function allows longi-time recording and error analysis in the absence of the operator.

Save: If the range has been stopped, the current recording can be saved in afile. In the file, the range data are written as ASCII values. The índividual values have XML tags, so that the recording can be used for documentation purposes or for the analysis connected to encoder configuration. The data can also be shown with the Microsoft Explorer or with another XML viewer.

Load: With this control surface, a measurement saved under the range function of an XML file can be loaded in the range. In this case, the range window changes to the view mode. Due to possible differerences in encoder configuration of the displayed measurement for the current programme, and the resulting deviances in the scaling of the position values and of the speed values, the "Start" button and the plan selection list are deenabled after the data have been loaded for display. The measurements remain deenabled, until the range function is restarted.

Import: Import of a measurement from . ScpXml files.

Export: Export of the measurement of a selected output to an .ScpXml filei.

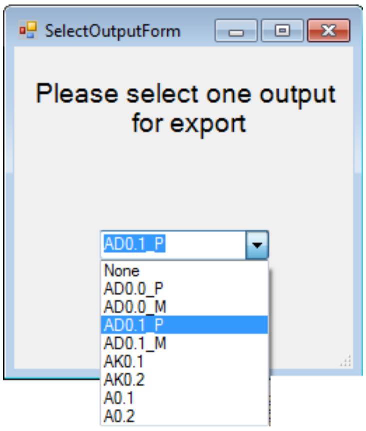


Figure 112 Selecting an output for export

4.6.12.1. Scope scheme

Encoder data

Functions	• Recording of scaled position values of system A and system B over the whole period. • Recording of velocity values over the period. • Axis 1...12 and Virtual axis 1...18 selection
Application	• Recording of the scaled position and velocity values depending on the set axis. • Analysis and behaviour of the encoder signal for diagnostic purposes (e. g. troubleshooting). • Acceleration behaviour and speed behaviour of the drive. • Detection of thresholds.
Output	• Position in [conf. unit] in yellow • Speed in [conf. unit] in green • Selectable SCU output in grey • Two cursor values –positionable ***Information:*** Optionally, the assigned colours can be adjusted.

Speed encoder

Function	• Recording of the current velocity over the period • Axis 1...12 and Virtual axis 1...18 selection
Application	• Recording of the scaled speed value depending on the set axis. • Analysis and behaviour of the encoder signal for diagnostic purposes (e. g. troubleshooting).
Output	• Speed in [rpm] in red • Selectable SCU output in grey • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

Position Pass1

Function	Currently not supported
Application	Currently not supported
Output	Currently not supported

Speed encoder

Function	Currently not supported
Application	Currently not supported
Output	Currently not supported

SDI

Function	• Recording monitoring function SDI • Selection of block ID (number see 4.12.5.3.1)
Application	• Recording of the current position and velocity value depending on the set ID. • Analysis functionality monitoring module SDI with the help of Enable (cw,ccw) and Status
Output	• Current position in [conf. unit] in red • Threshold in [conf. unit] in green • Current speed in [conf. unit] in blue • Activation of monitoring module in yellow • Status of monitoring module in gray • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SLI

Function	• Recording monitoring function SLI • Selection of block ID (number see 4.12.5.3.1)
Application	• Recording of the current position and velocity value depending on the set ID. • Analysis functionality monitoring module SLI with the help of Enable (cw,ccw) and Status
Output	• Current position in [conf. unit] in red • Threshold in [conf. unit] in green • Jog Step in [conf. unit] in blue • Activation of monitoring module in yellow • Status of monitoring module in gray • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SEL

Function	• Recording monitoring block SEL • Selection of block ID (number see 4.12.5.3.1)
Application	• The diagram shows the calculated braking distance and stop position. • Analysis functionality monitoring module SEL with the help of Enable (cw,ccw) and Status
Output	• Current position in [conf. unit] in red • Braking distance in [conf. unit] in green • Stop distance in [conf. unit] in blue • Speed in [conf. unit] in yellow • Acceleration in [conf. unit] in turquoise • Status SLI in grey • Two cursor values - positionable ***Information:*** Optionally, the assigned colours can be adjusted.

Data blocks SSX

Function	• Recording monitoring block SSX • Selection of block ID (number see 4.12.5.3.1)
Application	• When the SSX function is activated, the limit velocities and the current velocity are recorded and displayed over the period. • Analysis Functionality Monitoring module SSX current velocity to configured limit velocity.
Output	• Act. Speed in [conf. unit] in red • Limit speed in [conf. unit] in green • Acceleration in [conf. unit] in blue • Internal status SSX in blue • Result SSX in gray • Two cursor values - positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SLP

Function	• Recording monitoring block SLP • Selection of block ID (number see 4.12.5.3.1)
Application	• The diagram shows the calculated braking distance and stop position. • Analysis functionality monitoring module SLP with the help of Enable (cw,ccw) and Status
Output	• Current position in [conf. unit] in red • Conf. position in [conf. unit] in green • Speed in [conf. unit] in blue • Result SLP in yellow • Two cursor values - positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SLS

Function	• Recording monitoring block SLS • Selection of block ID (number see 4.12.5.3.1)
Application	• Display of the current speed to the set limit speed. • Analysis Functionality Monitoring module SLS
Output	• Speed in [conf. unit] in red • Threshold in [conf. unit] in green • Fault Distance in [conf. unit] in blue • Result of the function in yellow • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SCA

Function	• Recording speed monitoring, monitoring block SCA • Selection of block ID (number see 4.12.5.3.1)
Application	• The curve shows the current velocity in relation to the set speed limit • Analysis Functionality Monitoring module SCA
Output	• Speed in [conf. unit] in red • Threshold in [conf. unit] in green • Fault Distance in [conf. unit] in blue • Result of the function in yellow • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SCA Position

Function	• Recording position monitoring, monitoring block SCA • Selection of block ID (number see 4.12.5.3.1)
Application	• The curve shows the current position in relation to the set limit positions. • Analysis Functionality Monitoring module SCA
Output	• Position in [conf. unit] in red • Threshold1 in [conf. unit] in green • Threshold2 in [conf. unit] in blue • Result of the function in yellow • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SOS

Function	• Recording Standstill monitoring Monitoring module SOS. • Selection of block ID (number see 4.12.5.3.1)
Application	• Recording of the current position and velocity value at the configured threshold depending on the set ID. • Analysis Functionality Monitoring module SOS
Output	• Current position in [conf. unit] in red • Threshold1 in [conf. unit] in green • Velocity in [conf. unit] in blue • Activation SOS [conf. unit] in yellow • Result of the function in gray • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SLA

Function	Currently not supported
Application	Currently not supported
Output	Currently not supported

MPM1

Function	• Recording of monitoring module MPM ID1 • Axis selection for slave position
Application	• Recording of the current master position value to the slave position value. • Analysis Functionality Monitoring module MPM ID1.
Output	• Master position in [conf. unit] in red • Slave position in [conf. unit] in green • Tolerance in [conf. unit] in blue • Result of the function in gray • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

MPM2

Function	• Recording of monitoring module MPM ID2 • Axis selection for slave position
Application	• Recording of the current master position value to the slave position value. • Analysis Functionality Monitoring module MPM ID2
Output	• Master position in [conf. unit] in red • Slave position in [conf. unit] in green • Tolerance in [conf. unit] in blue • Result of the function in gray • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

SWM

Function	• Recording Standstill monitoring Monitoring module SWM • Selection of block ID (number see 4.12.5.3.1)
Application	• Recording of the current position values in X, Y and Z direction for evaluation SWM. • Analysis Functionality Monitoring module SWM
Output	• Position X in [conf. unit] in red • Position Y in [conf. unit] in green • Position Z in [conf. unit] in blue • Result of the function in yellow • Two cursor values – positionable ***Information:*** Optionally, the assigned colours can be adjusted.

4.6.12.2. Procedure of a measuring in the range

After the range window has been opened, it still stands on “Stop“, i.e., no cyclical process data from the SCU system are read. To carry out a measurement that is faultless to the greatest extent, you should proceed as described below.

NOTICE:
All Internet based and all LAN based programs running in the background should be closed before the measuring.

4.6.12.3. Preparing measurement

Select the desired measuring scheme. During a speed-oriented measurement, the running control-time of the SCU module is shown on the X-axis. The running control-time is a continuously rising meter for the system control signs of the SCU module. The measurement data are constantly updated and are stored in the cache. The recording memory is aprox. 15 minutes.
Measurement automatically stops when the cache is full. The preceding measurement is automatically saved as “ScopeTempData.ScpXml“.
During position-oriented measurement, the configured measuring range of the set axis is shown on the X-axis.

NOTICE:
When the scheme is changed, recorded data from the preceding measurements get lost. When the window size is changed, the displayed data must be newly scaled.

The list of the different measurement plans (schemes) can be found in chapter 4.6.12.1)

4.6.12.4. Starting measurement

The button “Capture” to start measurement is only available with an active connection to the SCU system. After this button has been clicked on, the data are transferred cyclically into the cache, and are shown in the diagram from left to right. An active recording can be stopped with the “Stop” button.

4.6.12.5. Stopping measurement and indicating data

After measurement is finished, the data can be analysed by appropriately moving the slider bar.

4.7. Validation report

SafePLC² uses the validation function („Device interface →Generate report") to produce a configuration report for equipment configuration. This function is only available with an active connection to an SCU system. This function can also be enabled by selecting the field "Generate validation report "in the toolbar.
In the window "Document Properties" information and description can be entered or edited. After generation, these are shown in the validation report

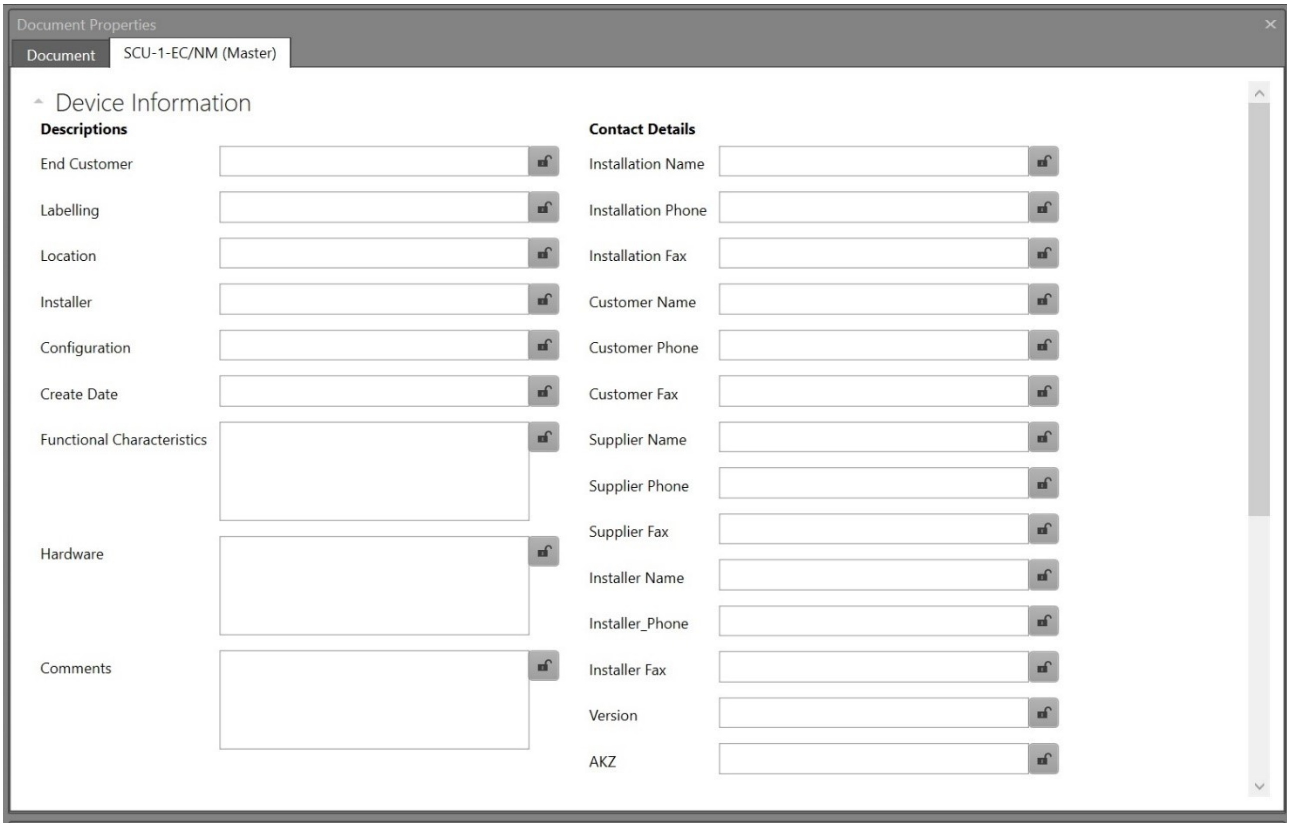


Figure 113 Fields containing device information for the validation report

Every field has a blocking function.

The report is created as a PDF file.
The report is saved in a file and can be edited afterwards.

ATTENTION:
The printed file serves as template for the safety-relevant check!

NOTICE: The report can be created only after saving the configuration file (functional scheme).

The created file (*.pdf`) has the same name and is situated in the same list as the corresponding functional scheme.

4.7.1. Editing steps

1. Step: Editing the report's header

In the header, the following fields can be edited.

End customer:	customer's name
Labeling:	project name
Configuration:	configuration name
Comments:	helpful comments
Verification:	indication of the auditors of the project
Creation date:	date, when the report was created

2. Step: fill in contact person and installation description

AKZ:	plant identification
Version:	document version
Installer Name:	commisioning of the equipment
Installer (phone):	phone number
Installer (fax):	Fax number
Installation Name:	xxx
Installation (phone)	phone number
Installation (fax):	fax number
Customer Name:	operator of the equipment
Customer (phone:	phone number
Customer (fax):	fax number
Supplier Name:	manufacturer of the device / the equipment
Supplier (phone):	phone number
Supplier (fax):	fax number
Installation designation:	Project manager
(Installation) Location:	Project location
End customer:	Name of customer
Configuration:	Configuration name
Creation date:	Date when the report was created
Function characteristics:	Application description
Comments:	Helpful comments, e.g. file name of the function plan
Hardware:	Code designation of the equipment

3. Step: individual check of the system components

This section contains control boxes that should be marked if the given information is correct.

Visual check for personal injuries and correct fastening:

Component documentation is available:

Visual check for deviations from the installation guideline:

Device type:	Enter device type, e. g. SCU-1-EC, SCU-2-EC/NM etc.
Serial number:	Serial number of safety module (label)
CRC device configuration:	Signature for the program and for the parameter data
CRC program:	Signature for the program
Extension device:	Description of extended devices
Transmission meter:	This field can also be edited.
Number of axles:	Total number of axles

Functional test:

To create the validation report, the correct programme data and the correct parameter data must be loaded!
The test engineer must revalidate all configured data in the printed report by verifying the programmed functions on the equipment/the device.
All set limit values of the monitoring function used must be checked for correctness. The reaction times mentioned in the installation manual SCU must be observed.

A successfully completed validation should be completed by clicking on the button "Lock validation ".

NOTICE:
If a new configuration is loaded into the SCU/SDU system, the system LED lights *YELLOW* if the SCU/SDU system is operated correctly. This signals an unvalidated application! If the button "Lock validation" is enabled during an active connection to the module, the LED flashes *GREEN*.

4.8. User management

Via the user management, functional scheme can be locked against unintentional or unauthorised changes. The access to function blocks in the current functional scheme can be deenabled or enabled. This means, that in a disabled functional scheme all menu options and toolbars for adding function blocks are grey (= disnabled).

Furthermore, in function blocks that have already been added, the parameters can't be changed any more.

For unlocking, a password is necessary. In this case, the configured values and the functional modules of a disabled paln can be displayed but they cannot be modified. This function ensures that unauthorized persons can't modify the functional scheme.

NOTICE:
Functional schemes can only be unlocked with the password that has also been used to deactivate the plan. A disabled functional scheme can no longer be compiled. However, access to the SCU-x-EC/x module is still possible.

4.9. Device interface

The device interface is displayed in the device window. This window contains advanced communication options, e. g., program transmission, diagnosis and range monitoring by connected SCU devices. If the device interface is opened, the programme automatically starts the compilation. The window contains the instruments of the device interface.

Symbols in the device interface:

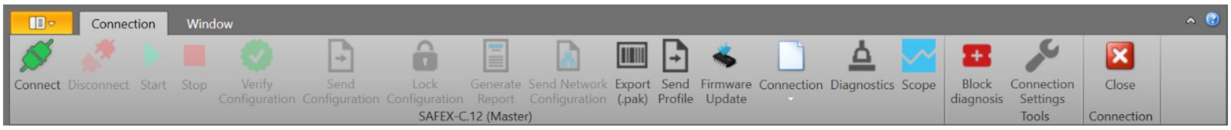


Figure 114 Symbols in the device interface – not connected

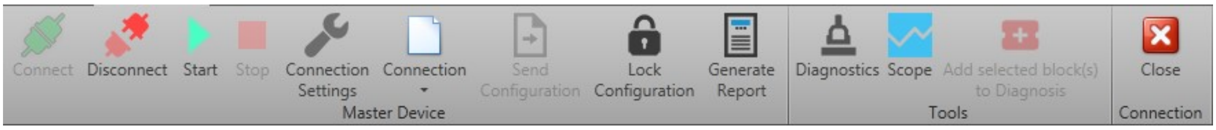


Figure 115 Symbols – connected

Connect: starts the connection with the SCU system.

Disconnect: disconnects an active connection

Start: Starts the programme sequence in the "Connected" mode.

Stop: Stops the programme sequence in the "Connected" mode.

Verify Configuration: Compares the application of the SafePLC² to that on the SCU system by means of the CRC.

Send configuration: Transmits the configuration of the function block diagram to the SCU system. This is only possible in the "Stop" mode.

Lock configuration: After every configuration data transfer to an SCU system, these data are marked "Not validated ". The basis group signals this by the status LED flashing yellow. With the "Lock configuration" command access to the configuration data can be disabled in the basis block. This is indicated by the status LED flashing permanent green.

Generate (validation) report: Creates a PDF file of the SCU configuration for the connected device. The text file lists the parameters of the configured modules and of the IL program. The printout must be confirmed and approved by TÜV according to the relevant guidelines.

Send network configuration: Transmits the set network parameters (not necessary for all modules).

Export Profile: Creates a profile from an FSoE Slave.

Send Profile: Transmits a profile to an FSoE Slave.

Connection: Sends or reads the current configuration to/ into the file. This function is disabled if the diagnostics window or the range window is open.

Diagnostics: Opens the diagnosticswindow; cf. chapter "Diagnostics".

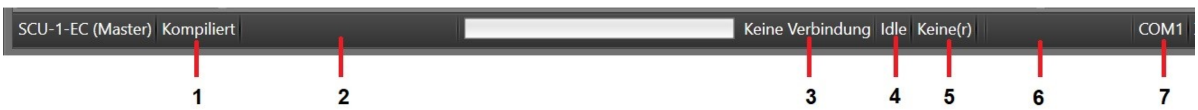
Scope: Opens the scope window. This enables online diagnosis of various process data.

Block Diagnostics: With this button selected elements in the device diagnostics window can be added the the "Function Block" tab. This tab is only activated if the device window is displayed.

Connection Settings: Opens the document properties with the connection settings. To create a connection with an SCU system, the transmission parameters must be set accordingly.

Close: Closes the device interface.

Status bar of the device interface:



1) Compilation display
   a. **Compiled** – Current file has been compiled
   b. **Not compiled** – Current file has not been compiled
2) Progress
   a. **None** – Grey background indicates that no configuration process is available
   b. **Send configuration** – Transmission of the configuration of the functional scheme to the SCU module.
   c. **Read configuration** – Reading out of the current device configuration by the SCU
3) Connection status with display

a. **Connected** – Active connection to the COM interface of an SCU monitoring unit

b. **Disconnected** – No active connection

4) Program status

a. **Idle** – Program has completed all tasks of the control

b. **Upload** – Program uploads into the SCU system

c. **Binary download** – Program downloads the configuration from the device

d. **Diagnostics** – Program uses diagnostics instruments in the "Diagnostics" tab.

e. **Range** –The program monitors the time-dependent processes of speed, acceleration and position in the "Range" tab.

5) Device status

a. **Stop** – Stops the transmitted program

b. **Carry out** – Starts the transmitted program

c. **Initialise** –Program initialises the device

d. **None** – No connected device (only status "Disconnected")

6) Alarm status: only in case of an alarm

a. **Alarm** – case of alarm with number of errors

7) Connected COM port

NOTICE:
The diagnosis function is described in the chapter "Diagnostics". For further information about the range, c. f. the chapter "Range monitoring".

4.10. Export window

In the export window parameters and configurations can be exported. After the click on the button, the diagnostics window appears to establish a connection between the PC and the SCU unit.

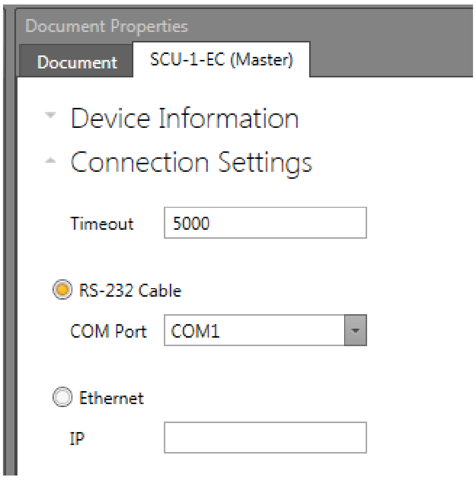


Figure 116 connection settings in document properties window

NOTICE: The connection settings are described in chapter 4.6.10.1.
After the connection has been established, and the "OK" button has been pressed, the main window with the control survaces for parameter export appears.

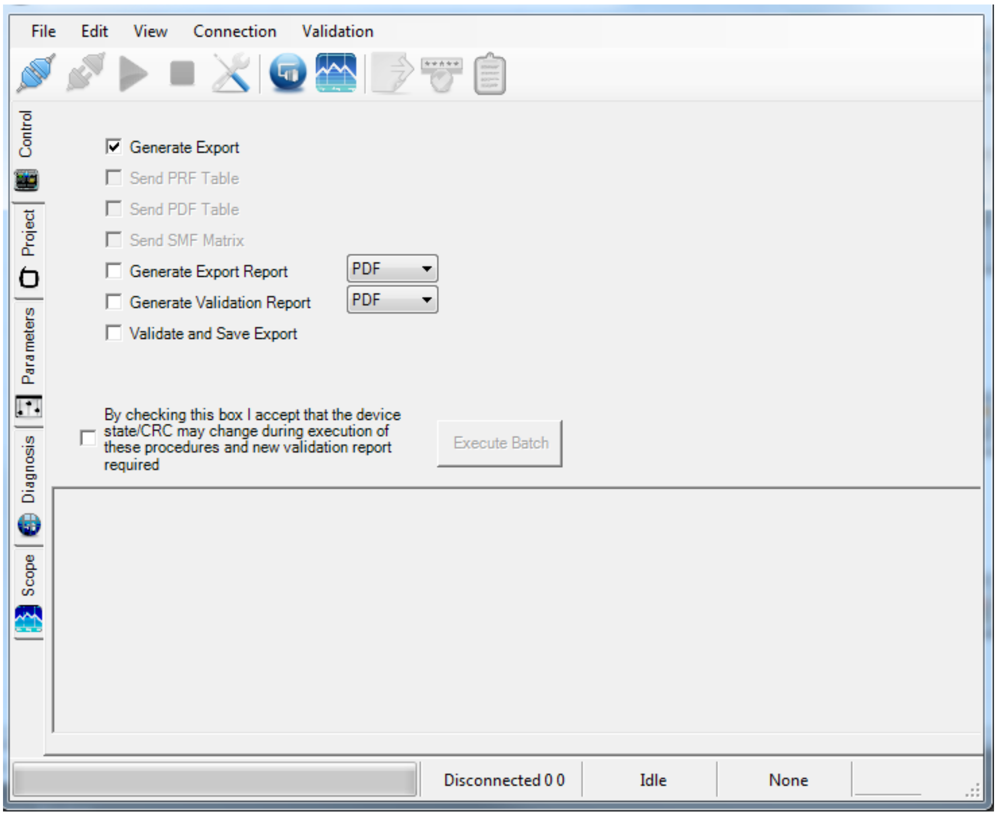


Figure 117 Export window

The following menus are available: **File, Edit, View, Connection** and **Validation**.

Commands in the “File”:

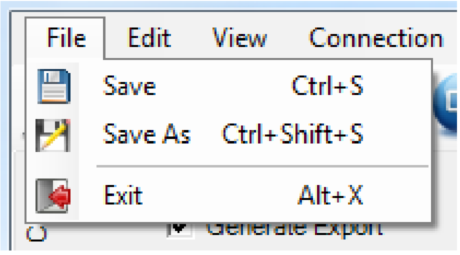


Figure 118 parameter export, menu "file"

Save– Saves export parameters.

Save as – Indicates how data are exported. They can be exported as single files or as project containers (PMT-package files). The project container can be protected by a password. To protect the file with a password, mark the field “Schutz aktiviern” [Enable Protection], and enter the password.

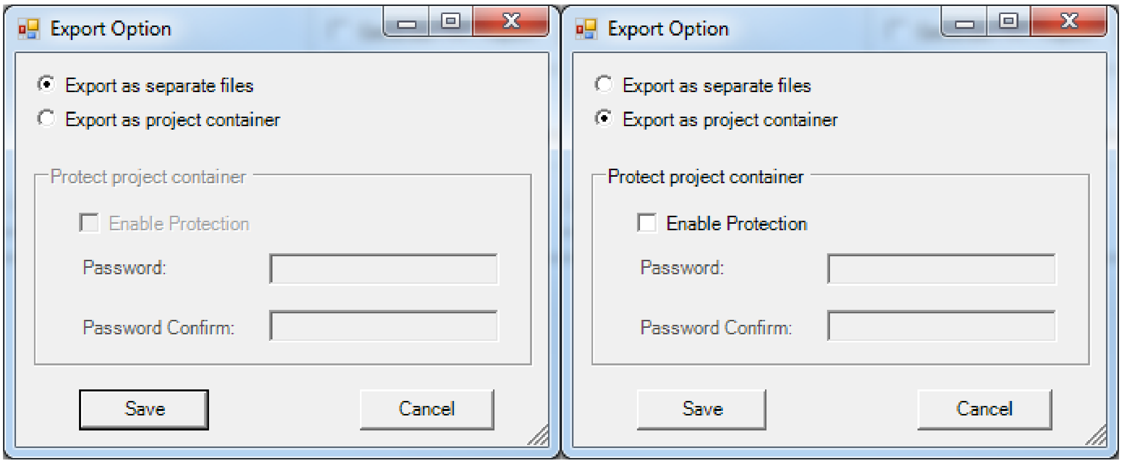


Figure 119 „Export option“ window

Exit– Close the parameter export window.

Commands in the "Edit" menu:

The commands in this menu are intended for work in the "Parameters" tab.

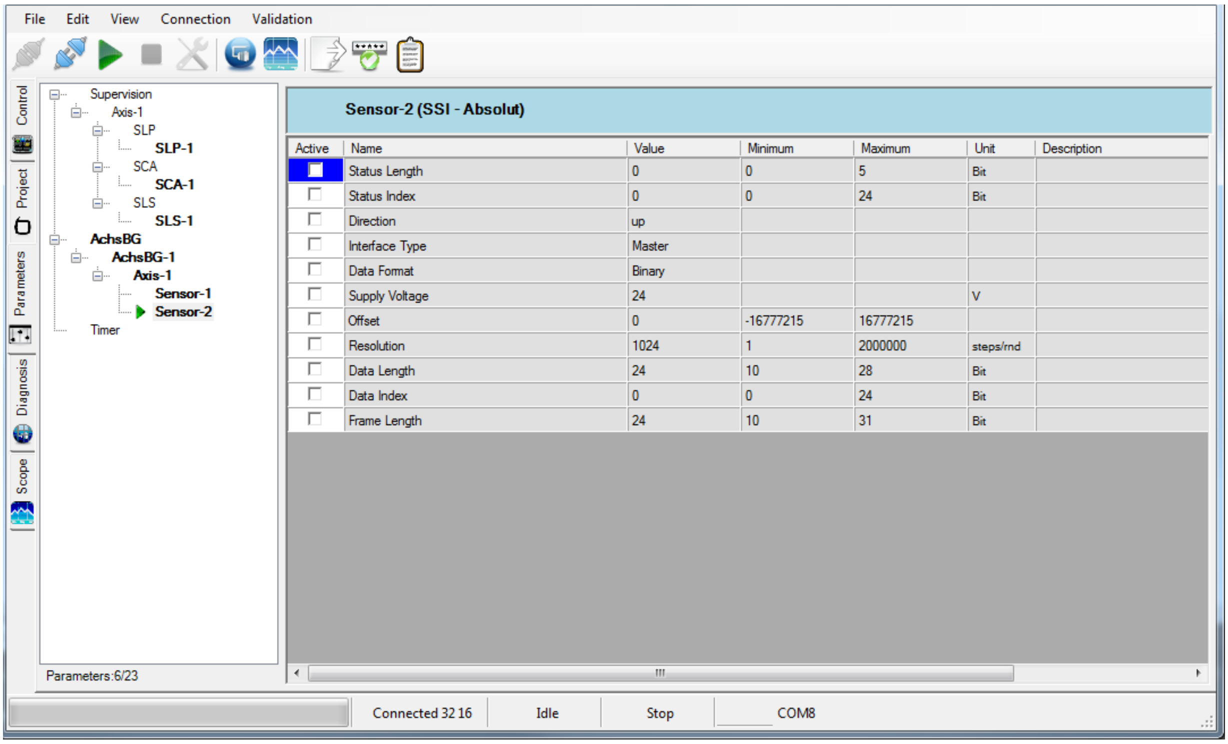


Figure 120 Export window, view tab "Parameters"

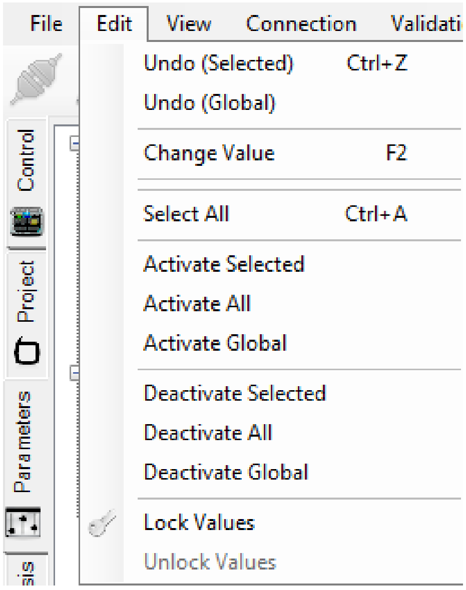


Figure 121 parameter export, menu "edit"

Undo (Selected) – Resets the selected value to the default value.

Undo (Globa)l – Resets all parameter changes to the default values.

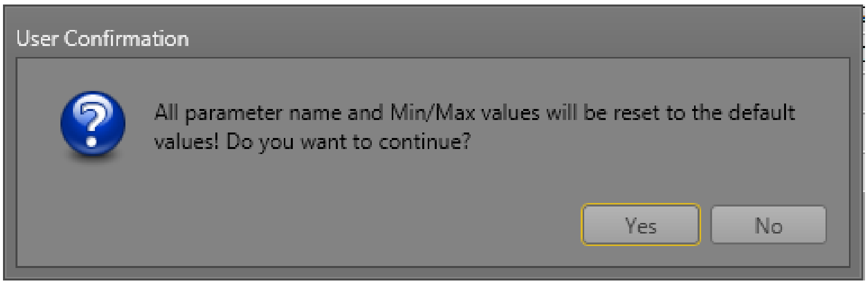


Figure 122 User confirmation before resetting parameter values

Change value – Change of the selected value. Value change is also achieved by double clicking on the value with the left mouse button.

Select All– Selects all parameters in the “Parameter” tab for a selected element, e. g. encoder.

Activate Selected – Enables the selected parameters (sets) in the “Parameter” tab.

Activate All – Enables all parameters (sets) for a selected element, e. g. encoder.

Activate Global – Enables all parameters (sets) for all elements used.

Deactivate Selected – Disables the selected parameters (sets) in the “Parameter“ tab.

Deactivate All – Disables all parameters (series) for a selected element, e.g. encoder.

Deactivate Global – Disables all parameters (series) for all parameters used.

Lock Values – Locks the selected value. A dialogue window appears to enter a password. The blocked values are blocked for use in other programmes.

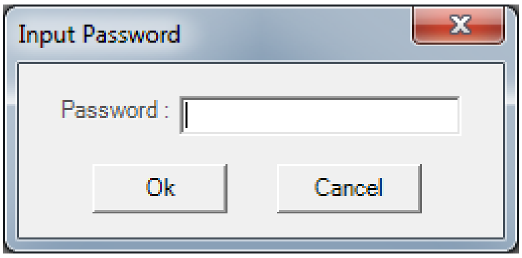


Figure 123 After locking the values, the "Enter password" dialog box appears.

Unblock Values – Unblocks the value that has been blocked before by the “Sperren” [Block] command. You are not asked for a password because in this environment you are the administrator who has set the password.

Commands in the "View" menu:

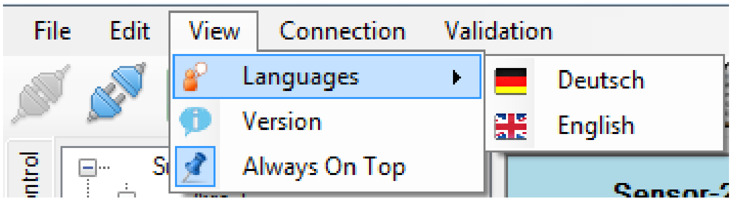


Figure 124 parameter export, menu "view"

Languages– Changes the language for the user surface and the names of the parameters in the "Parameter" tab. (English / German)

Version – Shows information concerning the export version of the parameters.

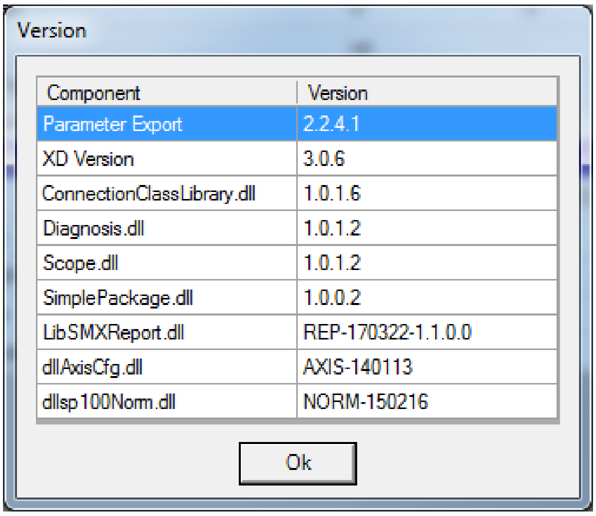


Figure 125 export versions of parameters

Always in the foreground – display of export window always in the foreground.

Commands in the "Connection" menu:

The surface of this menu depends on whether the SCU is connected or not.

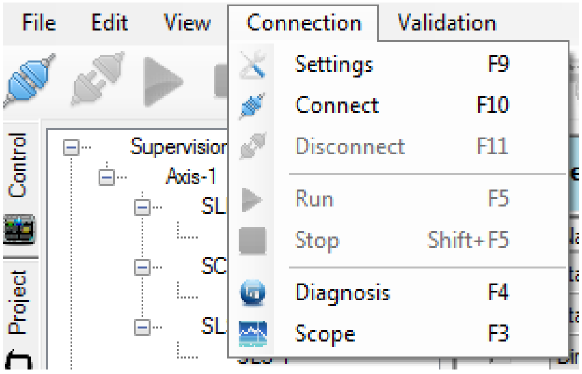


Figure 126 Surface, if SCU disconnected

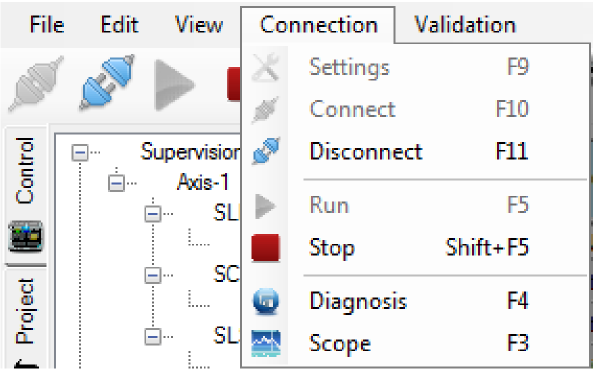


Figure 127 Surface if SCU is connected and runs.

Settings – Opens the window for the connection window. To establish a connection in an SCU system, the transition parameters must be set accordingly

Connect – Starts the connection to the SCU system.

Disconnect – Disconnects Trennt die aktive Verbindung.

Run – Starts the programme sequence in the "Connected" mode.

Stop – Stops the programme in the "Connected" mode.

Commands in the “Validation” menu:

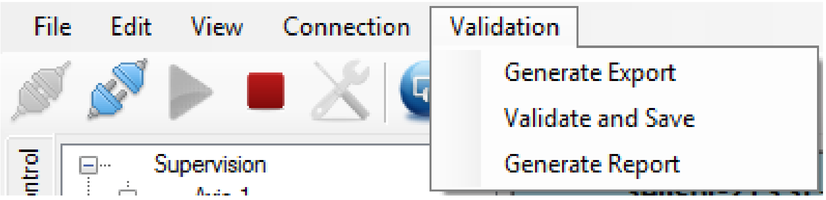


Figure 128 parameter export, menu "Validation"

Genrate export – This function connects two commands: “Create Export” and “Create Export Report“.

Validate and save – validating and saving of parameters.

Generate report – creates a PDF file or an Excel file of the current SCU configuration for the connected device. The text file lists the parameters of the configured modules and of the IL program. The print must be approved and released within the TÜV certification and according to the necessary guidelines.

These functions can also be activated in the control surface by selecting the corresponding control boxes. Cf. the illustration below.

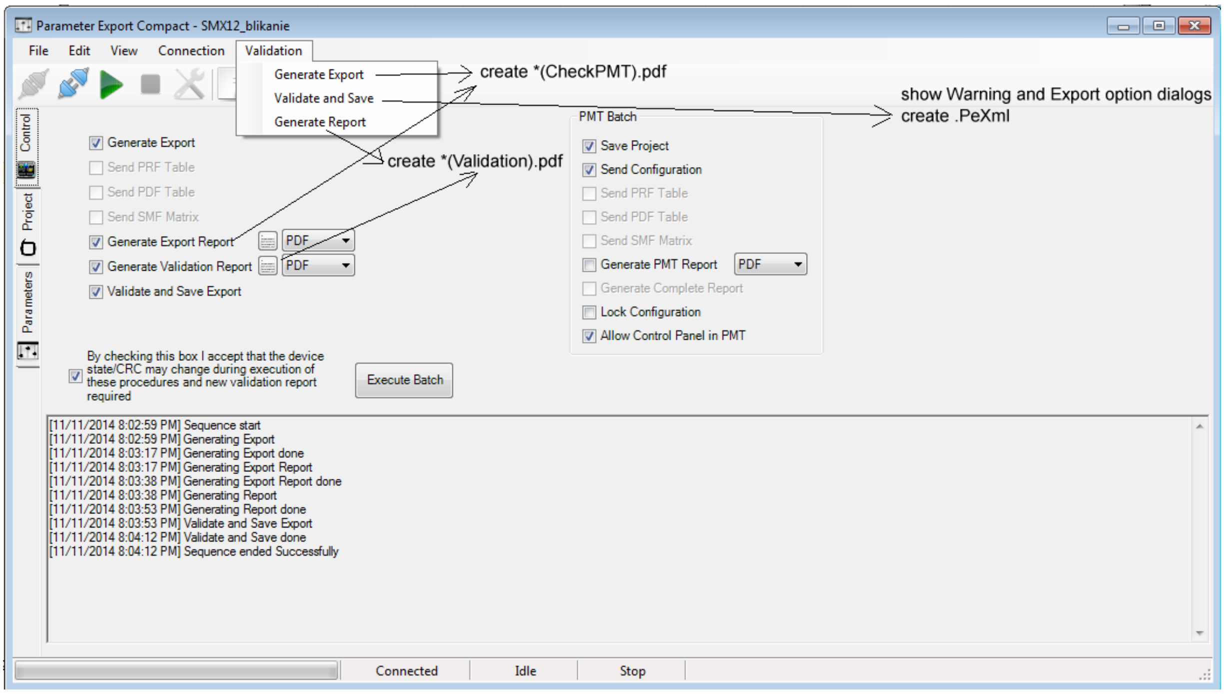


Figure 129 activated functions by selecting the checkboxes

Symbols for parameter export:

Figure 130 symbols in the device interface – not connected

Figure 131 symbols connected

Connect: starts the connection with the SCU system.

Disconnect: disconnects an active connection.

Execute: starts the program sequence in the “Connected” mode.

Stop: Stops the program sequence in the “Connected” mode.

Settings: opens the window for the connection settings. To establish a connection to an SCU system, the transmission parameters must be set accordingly.

Create export: creates an export.

Validate and save: validates and saves parameters.

Create report: creates a PDF file or an Excel file of the connected device. The text file lists the parameters of the configured modules and of the AWL program.The print must be approved and released within the TÜV certification and according to the necessary guidelines.

"Control" tab

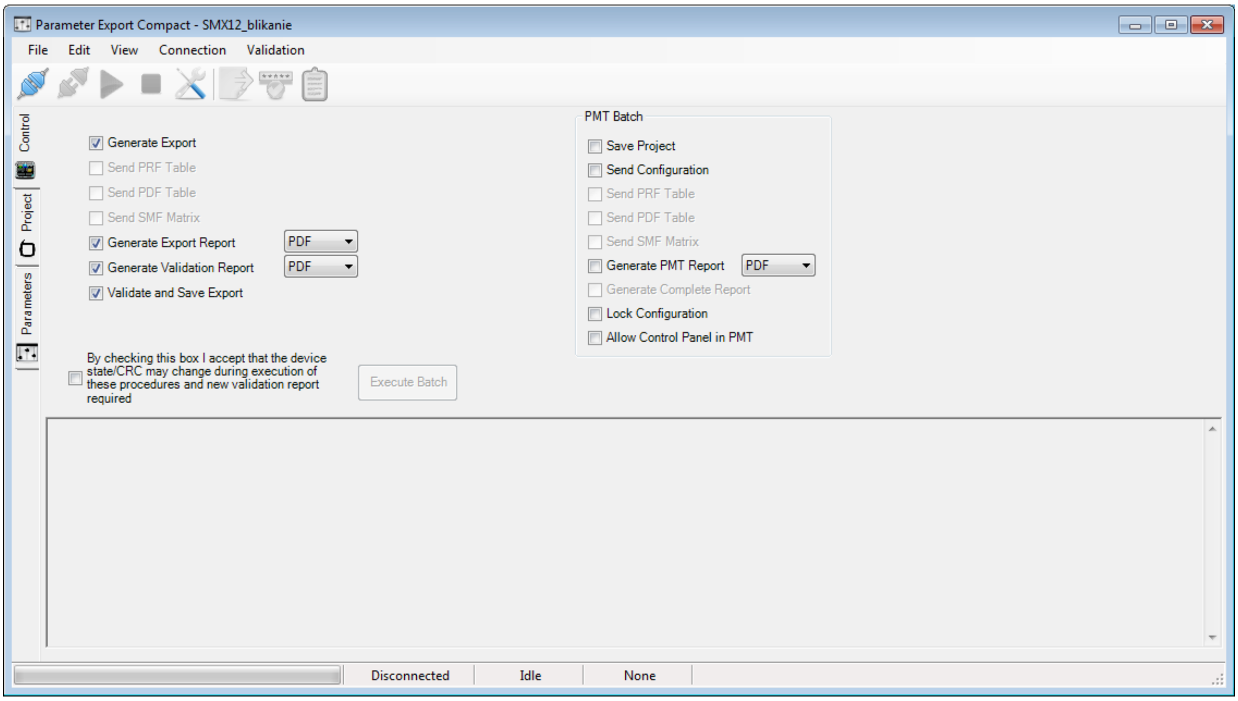


Figure 132 "control" tab

Send PRF chart (**P**osition **R**eference **F**unction): transmits all files necessary for using the PRF function, e. g. the position chart. For further information about the PRF function, c.f. chapters 10.3.3.7 and "TD-37350-820-11-xxF PRF description of the application

Send SMF matrix (**S**afe **M**atrix **F**unction): transmits position data of the coordinate matrix. For further information about the SMF function, cf. "TD-37350-820-11-xxF SMF Überwachungsfunktion SMF"[TD-37350-820-22-xxF SMF control function].

SafePMT
Another instrument for setting, cf. "HB-37350-820-21-xxF-EN manual SafePMT ".

The “Project” tab

In this tab, it is possible to fill in text boxes and to export this information with the exported parameters. These boxes can also be blocked. Blocked boxes cannot be edited after having been exported to and opened in another program.

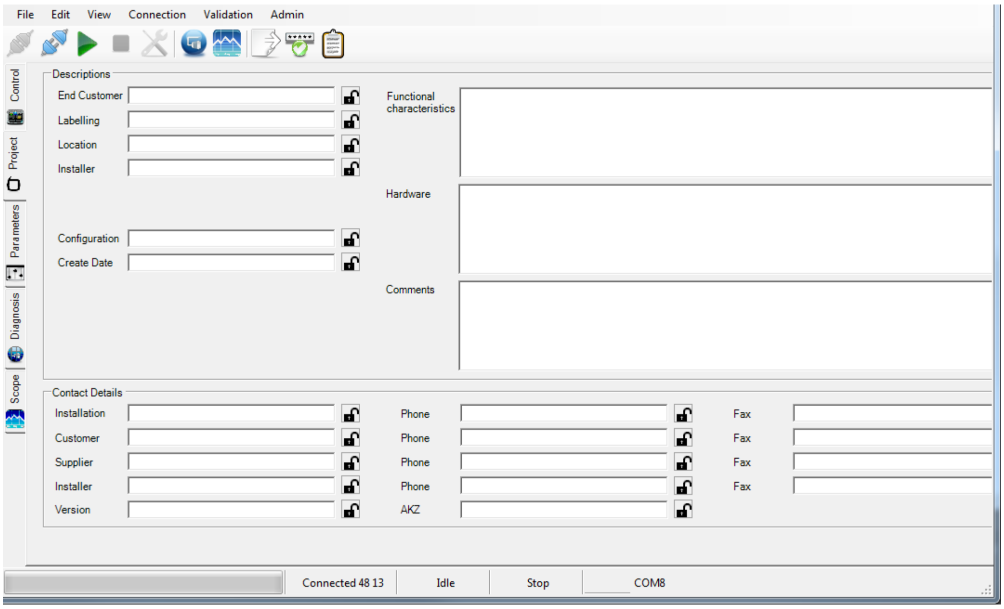


Figure 133 „project“ tab

The “Parameters” tab

All parameters and their values can be shown. After the parameters have been activated, they can be edited. To edit the parameters, use the command in the “Edit” menu or the mouse buttons.

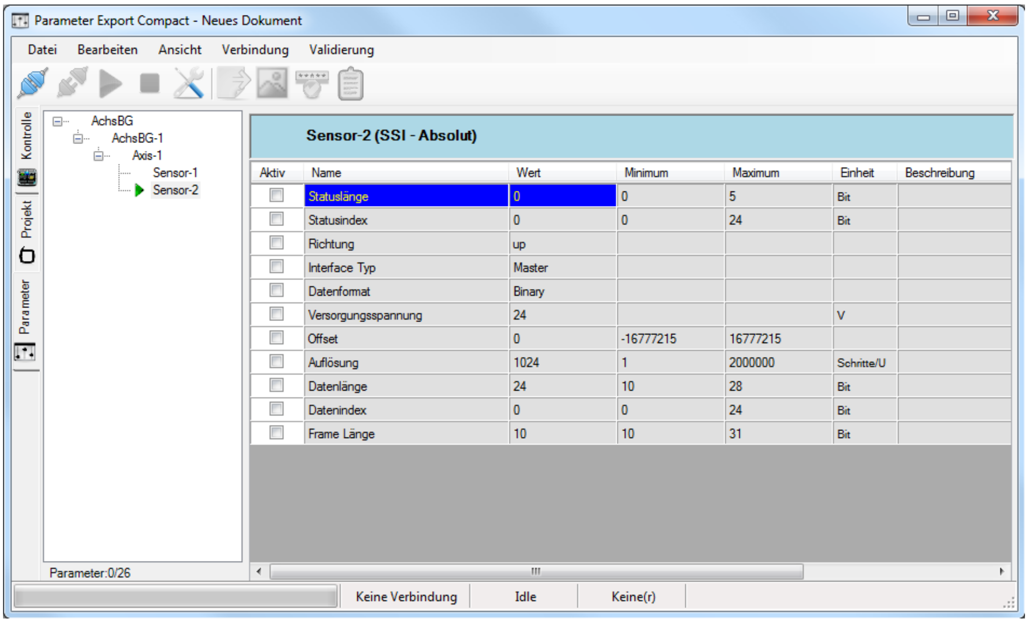


Figure 134 "Paramters" tab

4.11. Networks

4.11.1. Master to Master (SMMC)

4.11.1.1. Description

Global network with an SMMC Safe-Master-Master-Communication. The minimum are 2 Masters; the maximum are 4 Masters.

4.11.1.2. Creation

The user must connect a Master that supports SMMC.
If the user connects a second Master that supports SMMC, the following window appears:

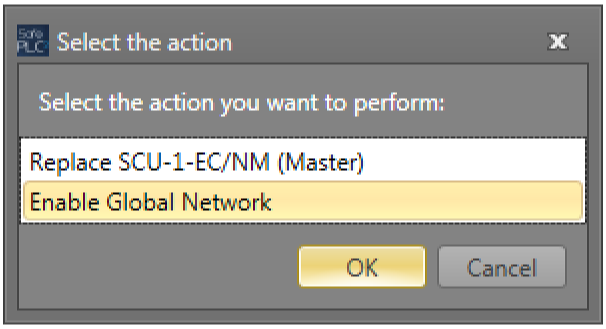
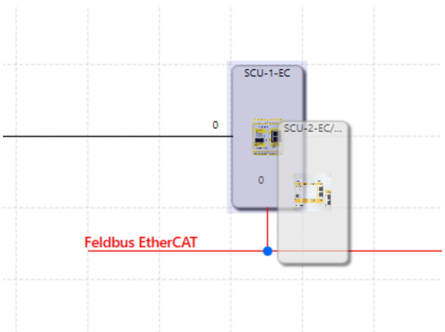


Figure 135 Select "Enable Global Network"

Select “Enable Global Network”, and confirm with “OK“. The second Master is added, and the tab “Global Network” appears.

After an SMMC network with at least 2 Master devices has been created, the window does not appear any more when the next Master device with SMMC support is added, and the device is automatically added to the global network (up to 4 devices).

If you want to replace a Master device by another device, you must drag the new device from the library exactly over the symbol of the device you want to replace. The mouse pointer must point to the symbol of the device you want to replace (cf. illustration below).



Disable – automatically when a Master is deleted and only one Master is left.

View
under “Global network”

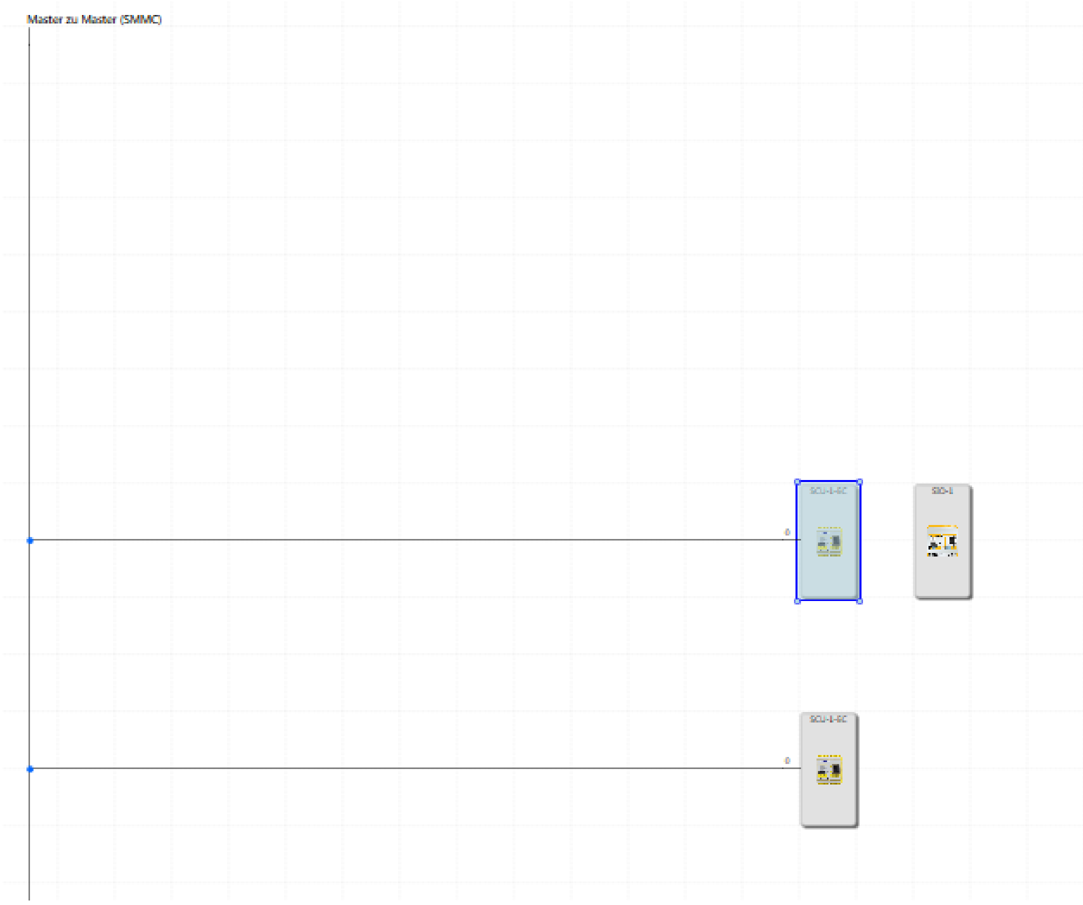


Figure 136 View "global network", SMMC

4.11.1.3. Configuration

Joint configuration

If the user clicks on the SMMC line in the global network

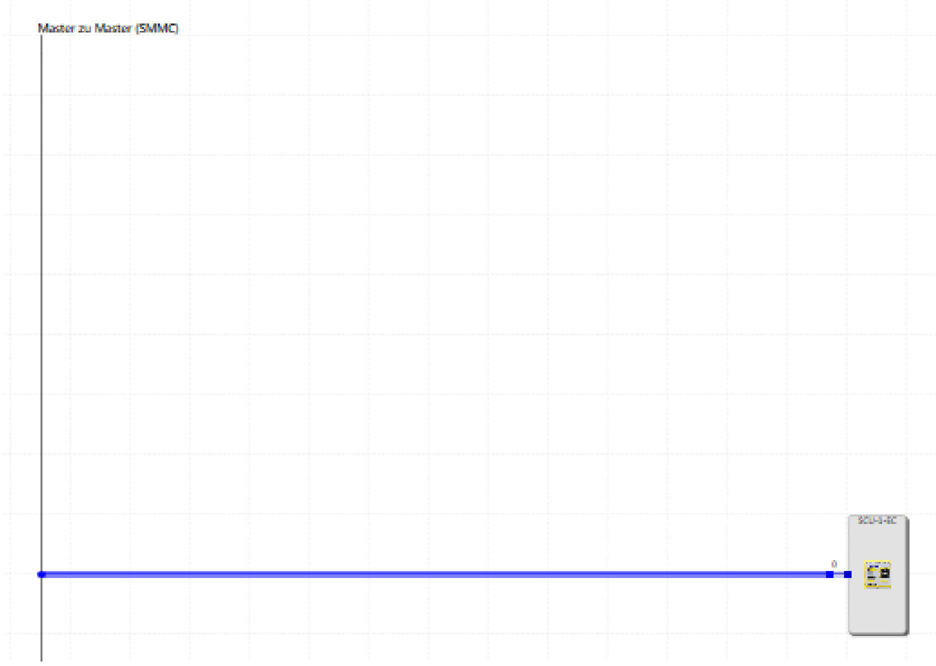


Figure 137 "global network", SMMC-line

or if he selects SMMC in the browser,

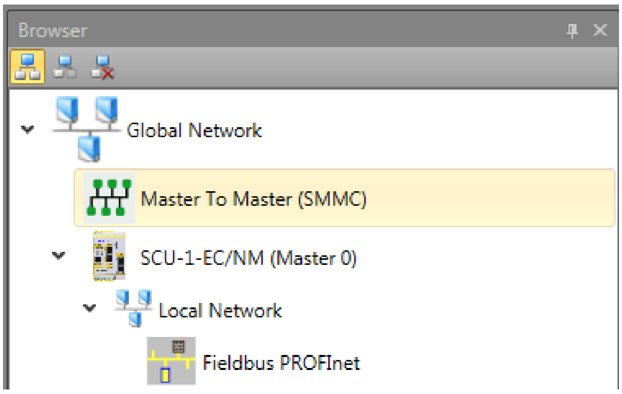


Figure 138 The SMMC properties in the browser window.

The SMMC properties appear in the properties window.

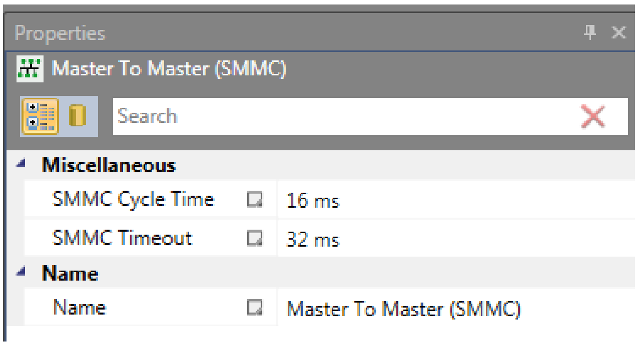


Figure 139 Properties window, master property

SMMC cycle time	Cycle time of the SMMC communication in [ms] [16ms…2048ms]
SMMC Timeout	Timeout time of the SMMC communikation in [ms] [32ms…4096ms]
Name	Free input possible for users

<u>Individual Master configuration</u>

After a click on every Master device in the SMMC network, the properties with which these devices can be configured individually appear in the properties window.

Figure 140 Tab "global network"

SMMC serial number – entry field. Here, the serial number of the selected device must be entered.

SMMC Master adress –This is the address of the device in the SMMC network. The SMMC Master device has the address 0. The ranking in the plan and the Master address are interconnected. The first device (from top to bottom) is the SMMC Master, and has the addres 0. The second device has the address 1, the third device has the address 2, and the last device has the address 3. If the user changes the ranking of the devices in the global network via Drag & Drop, the Master address changes according to the above principle (first device = SMMC Master address 0, etc.).

By clicking on the Master device in the browser with the right mouse button, this device can be selected as SMMC Master via “set as SMMC master”. After this device has been selected as SMMC Master, it appears in first position in the global network, and the other devices are pushed downward, and the addresses of the Masters change.

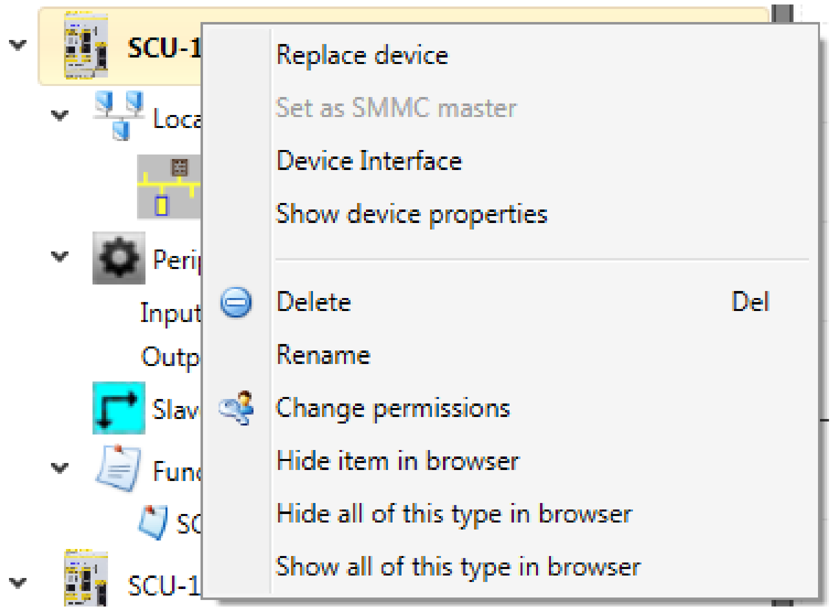


Figure 141 Browser, device set as SMMC master via browser window

4.11.1.4. Use

Every device can write 16 bit as output on SMMC. By the connection with SMMC, these bits are defined as "Terminal out".

<u>SMMC Terminal Out "SMMC Output connection "</u>

Every device can write 16 bit as output on SMMC. By the connection to SMMC; these bits are defined as "SMMC Terminal Out" [SMMC output connection].

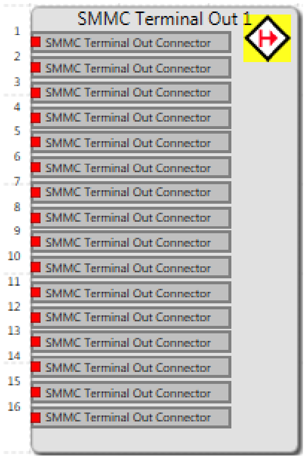


Figure 142 SMMC Terminal Out

SMMC Terminal In "SMMC input connection"

Every device can read bits from other devices, and it can read its own bits.

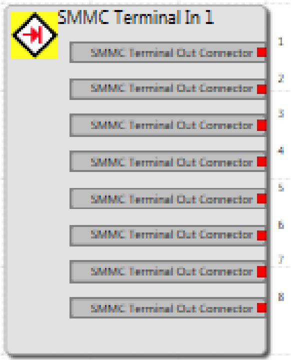


Figure 143 SMMC Terminal In

A restricted number of configurable common bits for every device and this Master can be assigned to logic as SMMC "Terminal Out", and can later on be used in the functional layouts of other Masters in these Masters' logic.
These elements of the bit group SMMC "Terminal Out" connection are available in the library as elements that can be used individually for every Master for input configuration in the functional scheme. Afterwards, an instance can be generated as SMMC "Terminal In" connection. This instance can be used as bridge in the functional scheme of other devices. The connection behaves like a normal output – with the exception that the corresponding output must be inserted into every Master's functional scheme, and that it can be assigned to the logic of the respective functional scheme. The SMMC "Terminal In" connection is available after the user has configured the corresponding SMMC "Terminal Out connection in the library of the functional scheme.

4.11.2. Fieldbus

4.11.2.1. Description

Fieldbus is the name of a group of industrial computer network protocols for real-time transmission control. Fieldbus is standardised in the IEC 61158 standard.

Fieldbus network protocol:

Non-safe networks

- PROFINET (currently only safe data)
- EtherCAT (currently only safe data)
- TCP/IP

Safe networks

- PROFIsafe
- FSoE Slave

4.11.2.2. Creation

If the device supports fieldbus, a “fieldbus” field is shown in the property window.
By marking the corresponding check box, the fieldbus can be switched on or off.

Figure 144 properties window "local network"

If a fiedlbus is enabled for a device, a network line is created for this purpose. The fieldbus network communicates with the subordinate PLC via individual connections. Furthermore the "Fieldbus EtherCAT" object appears in the configuration tree

View

In the local network:

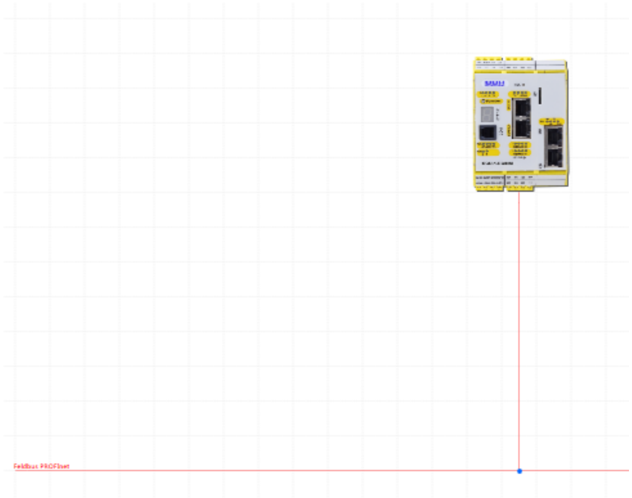


Figure 145 view local network

This view shows a connection of the device with the superordinate PLC. In the local network, the fieldbus is activated separately for each device.
The use of the fieldbus can be safe, non-safe or both. The use can be selected in the properities window. The design of line for the fieldbus changes according to the selected use.

Safe use is indicated by a solid red line.

Non-safe use is indicated by a dashed line:

If both possibilities are used, this is inidicated by a double orange-blue line:

4.11.2.3. Configuration

Figure 146 Fieldbus interface configuration in master device

The properties of this network are configurable when user clicks on the fieldbus line or selecting the fieldbus in the browser through the Property grid with defined properties in library.

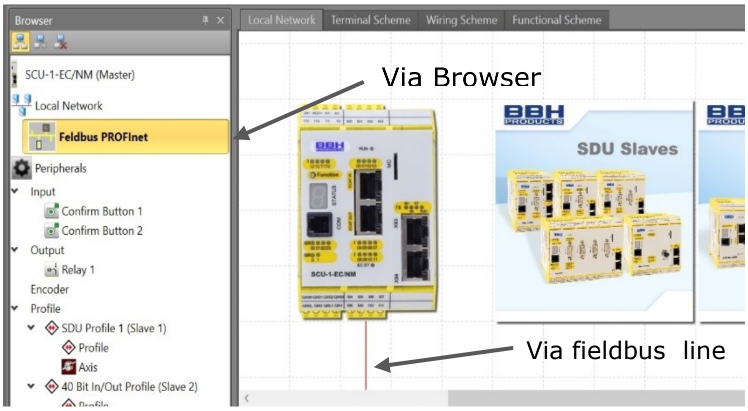


Figure 147 Fieldbus configuration in project

The information over this bus has a fixed size for transmission (e. g. 96 bit). This information is shared between “process data” like speed, position which each one can be defined as byte, int16, int24 or int32 or even brand defined types. The other part can also be used for the transmission of the logic information. The way that this network is configures is depend on the pre-defined profiles.

The properties window for the different protocols is shown below.

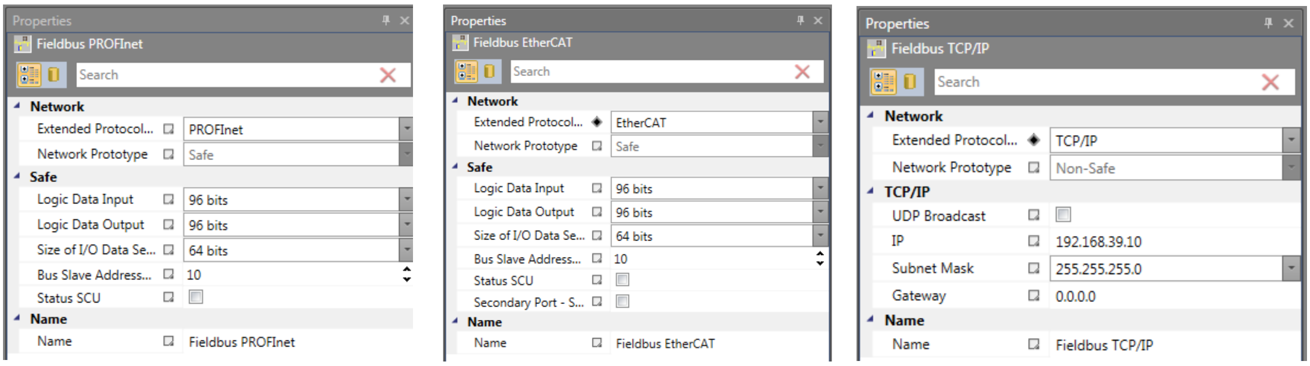


Figure 148 overview network protocols

Network typ – PROFINET, EtherCAT und TCP/IP can be selected for SCU devices.

Network use – possible selections: safe or safe and non-safe.
Safe and non-safe network connection is only possible with SDU devices.

4.11.2.3.1. Non-safe use

TCP/IP is currently available for **non-safe** use, which can be used as a communication interface for SafePLC².

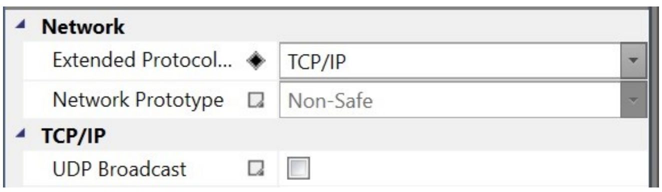


Figure 149 activated non-safe use

<u>Logical data inputs and outputs with fixed bits are available</u>.

Functional input - a bit can be used only once. Each block can configure the number of inputs from 1-32. It is possible to set the range of inputs. The non-safe input is possible via the functional input, but it is not allowed to use a non-safe input directly. It is possible to activate a non-safe input via another safe input. Therefore, a functional input has an additional input connector for each non-safe input, which ensures the use of the non-safe input. This checks that this additional input connector is connected to a safe input module. The connection with logic 1 or another logic device is not permitted.

Figure 150 Function block "Functional Input" (in library)

This Function block has one input and one output. The input is to be connected with a signal of the safety control. The input signal of the function block is linked with a logical AND to the functional input signal.

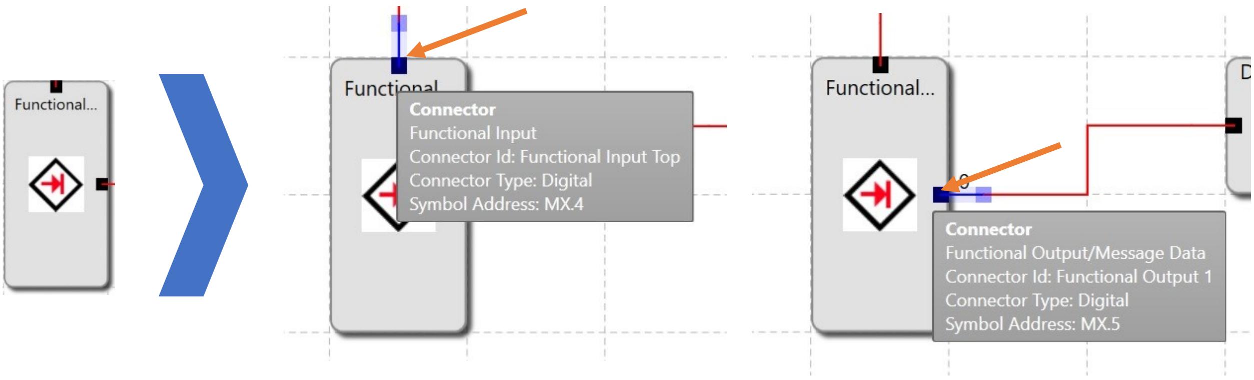


Figure 151 Function block "Functional Input" (functional scheme)

The structure of the block ensures a forced AND-operation with a signal of the safety control. A typical arrangement results as follows:

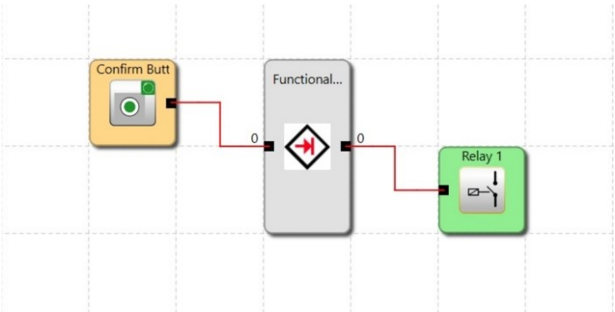


Figure 152 example of a connected „functional input"

A maximum of 136 functional inputs can be used.

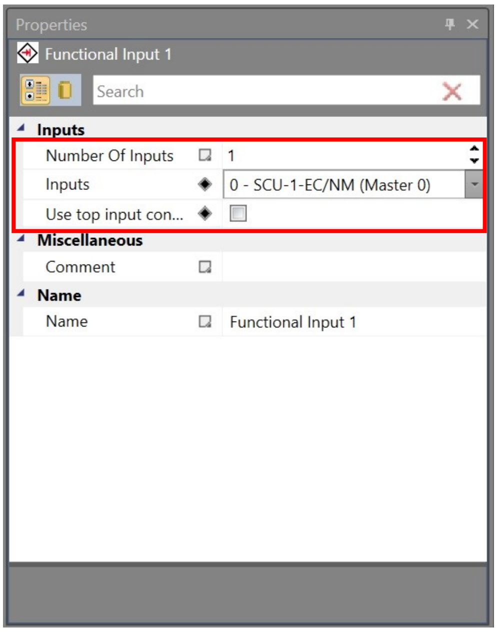


Figure 153 properties window – Functional Input

The following settings are possible:

- Inputs – Input selection on the master device
- "use top input connector" – visual setting of the input and output connection of the functional input

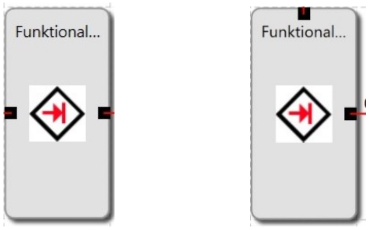
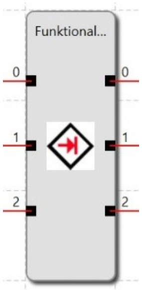
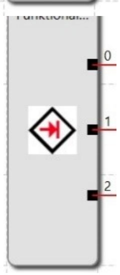


*Figure 154 Functional input: *use top input connector-function , with variant inputs left (inactive) or top (active)*



***use top input connector: inactive**
If the *Use top input connector function has not been selected for "Inputs", an individual enable must/cannot be used for each functional input.



***use top input connector: active**
If at "Inputs" a check mark was set at *use top input connector, then a collective enable must/can be used for each functional input.

Diagnostics data are only transmitted via the EtherCAT connection (FSoE Master).
The data can be parametrised via the functional outputs.
A maximum of 136 functional outputs can be configured. The configuration of process data is not possible.

Functional output - a bit can be used only once. Each block can configure the number of outputs from 1 - 136.

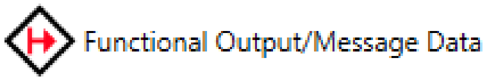


Figure 155 function block „ Functional Output/ Message Data" in Library window

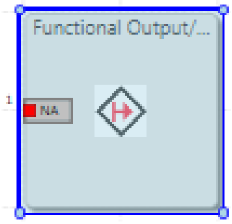


Figure 156 function block "functional output/message data"(functional scheme)

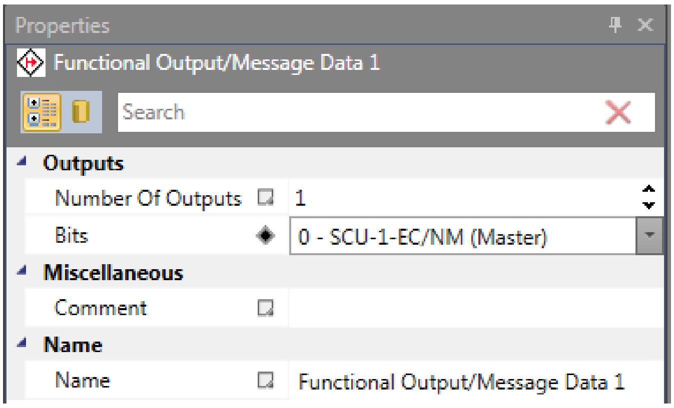


Figure 157 properties window " functional output/message data"

In one block also more than one functional output can be configured.
The illustration shows a functional output with 3 outputs.
Each connector can be configured.

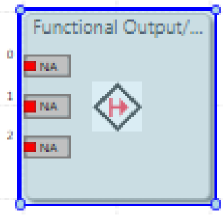


Figure 158 functional output with 3 outputs

Properties window of a functional output

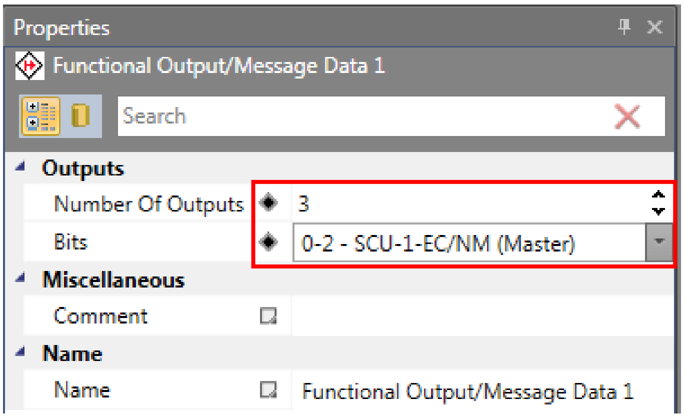


Figure 159 Properties window of a functional output, with 3 outputs

After selecting the functional output connector, the properties of the selected connector appear in the properties grid. The connector output can be set as follows:

NA - not activated (both boxes are unchecked).
A - activated
A/H - activated with "High active”

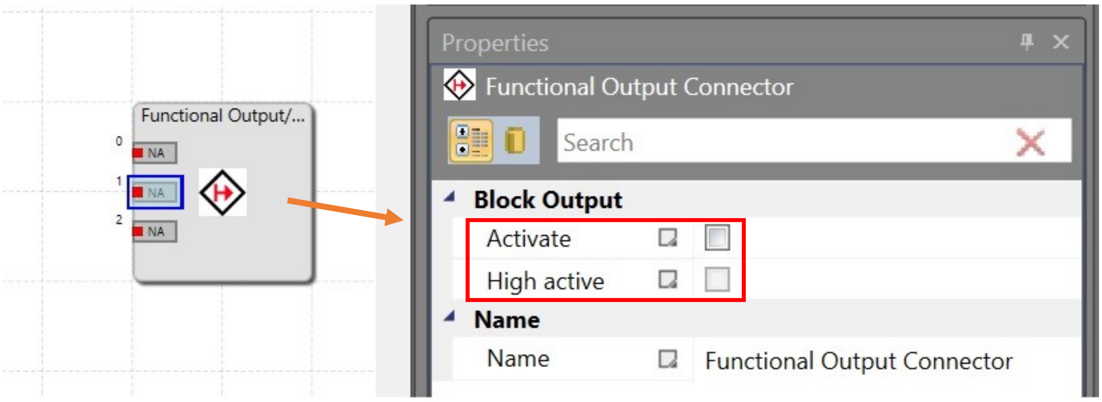


Figure 160 Functional output, Block output settings

4.11.2.3.2. Safe use

Currently, PROFIsafe Slave and FSoE Slave can be selected for **safe** use.

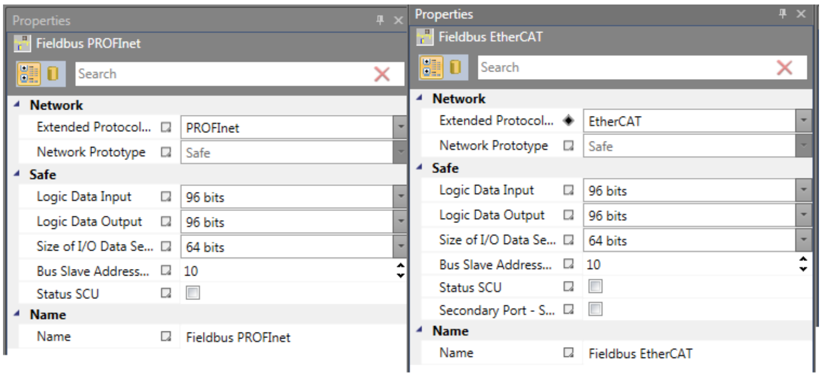


Figure 161 Selection of safe network connections

The following settings are possible:

- Logical data input – settings range from 8 bit to 96 bit.
- Logical data output – same range as for logical data input (8 bis96 Bit).

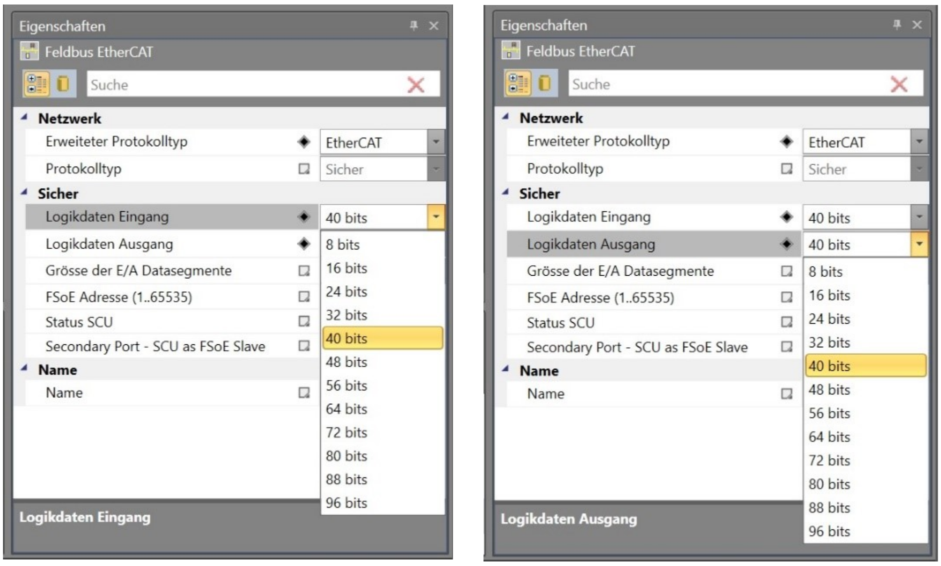


Figure 162 Settings of logical inputs and outputs

The values “fieldbus input” and “fieldbus output” indicate the number of bits used by the safe connections.

- F-Bus Input and F-Bus Output
- Safe connections “F-Bus Input” and “F-Bus Output” – each bit can only be used once. For every block, the number of bits can be set within the range of 1 to 32. The bits used can also be set.

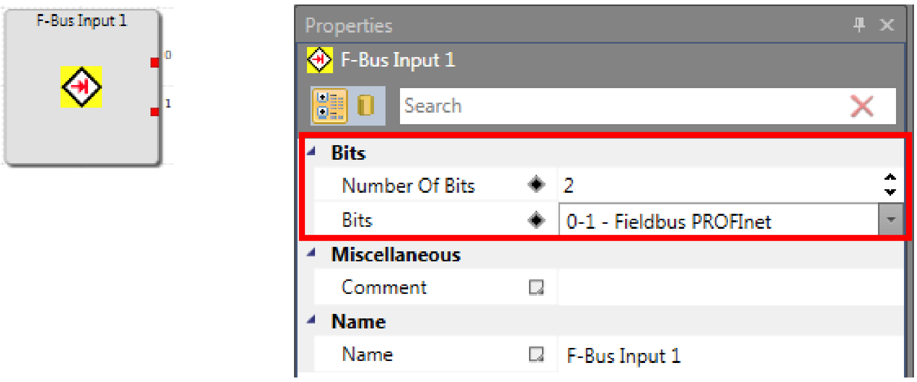


Figure 163 F-Bus input: function block and properties window

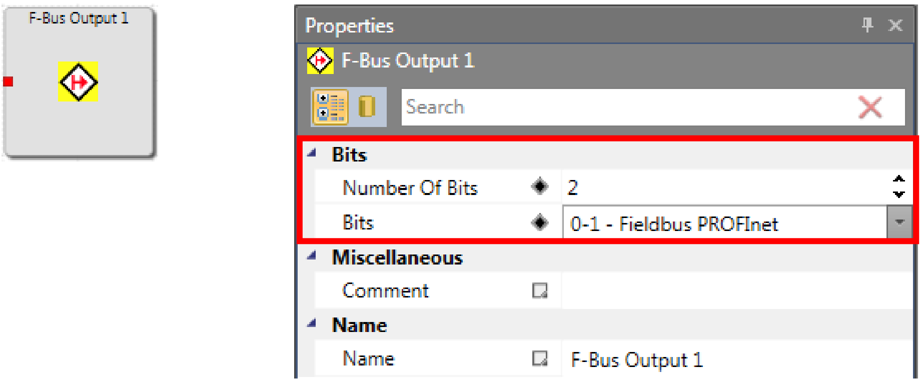


Figure 164 F-Bus Output: function block and properties window

<u>Process data outputs for the SCU</u>

Currently, only the status can be transmitted as a process data word.

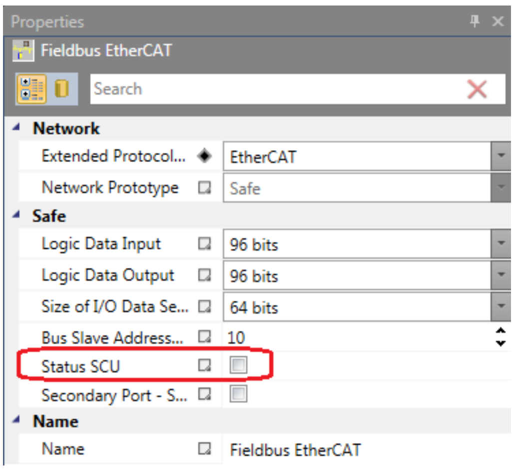


Figure 165 properties window "Fieldbus EtherCAT", Device status

Reset via network

If safe communication is used, a device reset (alarm messages) can be configured via safe communication.

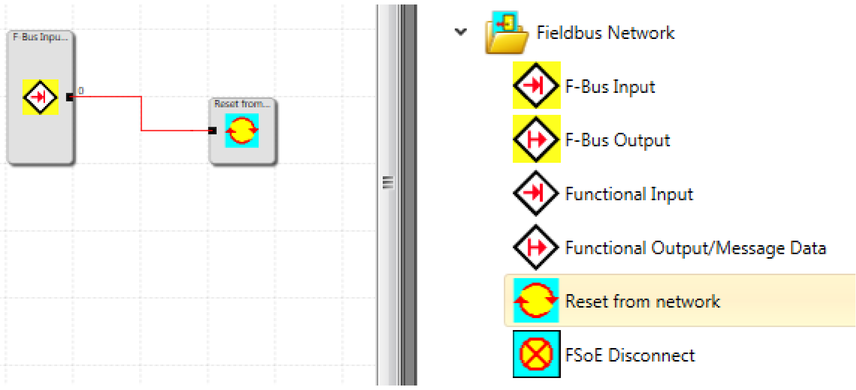
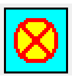


Figure 166 reset from network

This function block enables an alarm reset via the FBus slave.
For this purpose, the block must be linked to an FBus input block on which the reset signal is transmitted.

A maximum of 1 reset block is available.

NOTICE: Bit "0" cannot be configured/used for reset via network.

 **(FDB) FSoE Disconnect Block** – Secure disconnection of an FSoE connection

Function: The "Disconnect Block" is used to temporarily disable the FSoE connection.
The recognition of which device must be deactivated is set via the FSoE address.
This function avoids an error status in the master when a slave module is switched off/disconnected.
More information see chapter 4.12.5.3.1 „Overview of safety modules"

4.11.3. Fastchannel

4.11.3.1. Description

In the SCU FSoE I/O slave modules with a maximum process data width of 16 bit can be operated in the Fastchannel mode.

The processing time of a Fastchannel connection is indicated in the installation manual SCU- Master and Slaves.

4.11.3.2. Creating a FastChannel connection

In the Master device, a FastChannel connection can be activated.

After the activation of the FastChannel connection, a FastChannel scheme is displayed (FastChannelDE), where FastChannel data can be concatenated via IL

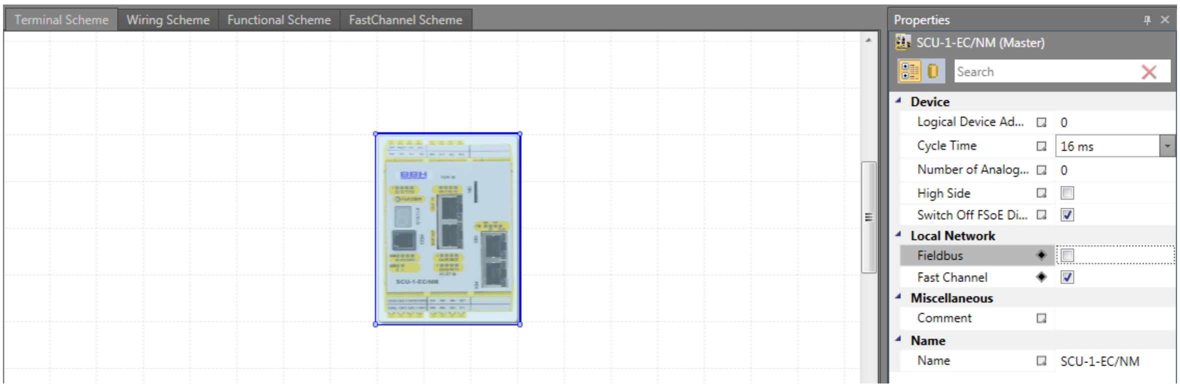


Figure 167 FastChannel scheme

4.11.3.3. Device configuration in Fastchannel scheme

The Slave units used in the Fastchannel must be configured. To configure the Slave units, "FC Connection" must be activated.

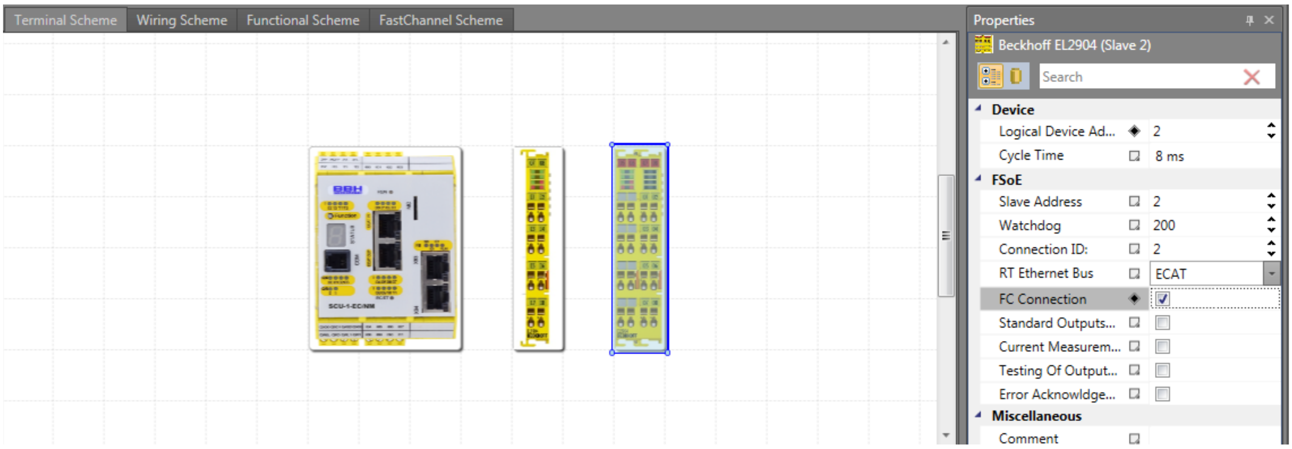


Figure 168 FastChannel scheme, FC-connection activated in slave unit properties window

NOTICE:

- Slave units that are configured with Fastchannel can only be used in the Fastchannel scheme.
- Mixed operation with Slave units with / without Fastchannel connection is possible. Observe that the units with Fastchannel are configured within the first 8 module addresses.
- A maximum 4 units can be configured in the Fastchannel.

4.11.3.4. User program

The processing in the Fastchannel functional scheme is similar to the "Standard" functional scheme. There are restrictions with respect to program size and with respect to the library.

Logical program concatenation between "Standard functional scheme" and the Fastchannel functional scheme takes place via the function blocks FastChannel Merker Input and FastChannel Merker Output. This data is exchanged between the two functional schemes once per device cycle.

Attention: Program concatenations must consider the reaction time of the "Standard" functional scheme.

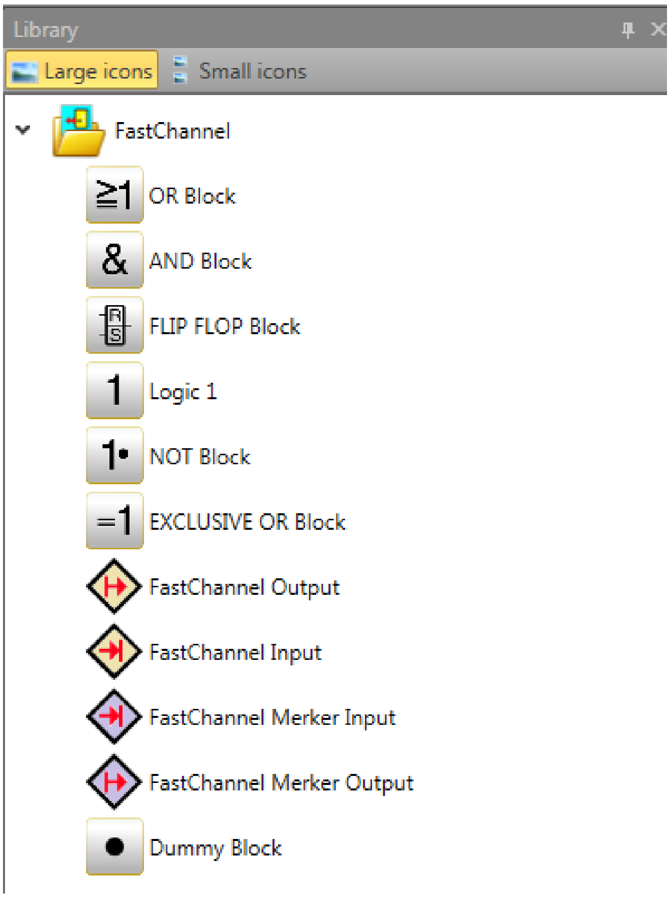


Figure 169 Library window "FastChannel"

FastChannel Output
FSoE Slave output bit

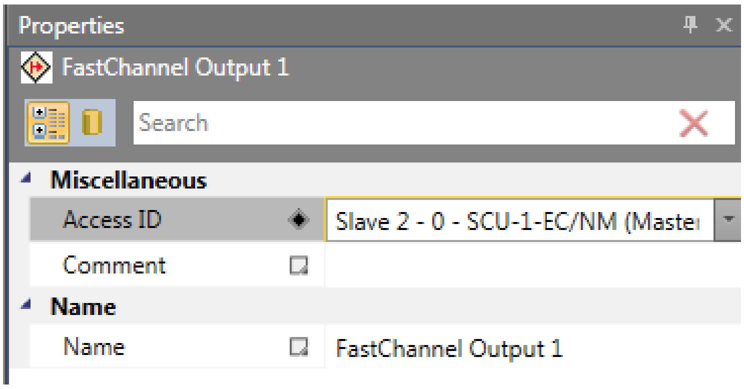


Figure 170 Access ID: selection of Slave number and bit number

FastChannel Input
Reading of FSoE Slave input bit

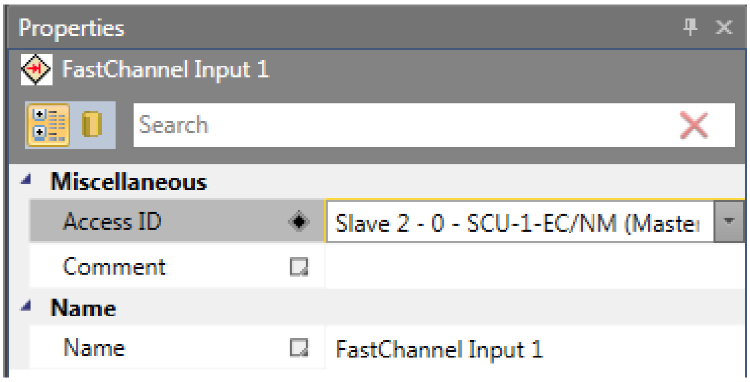


Figure 171 Access ID: selection of Slave number and bit number merker

FastChannel Merker Output
Flagbit for use in the "Standard" functional scheme

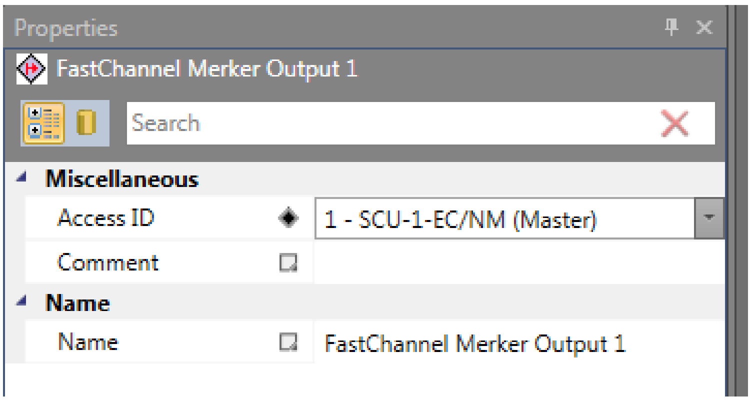


Figure 172 Access ID: selection of bit number of flag

FastChannel Merker Input
Reading of flag bit from "Standard" functional scheme to FastChannel functional scheme

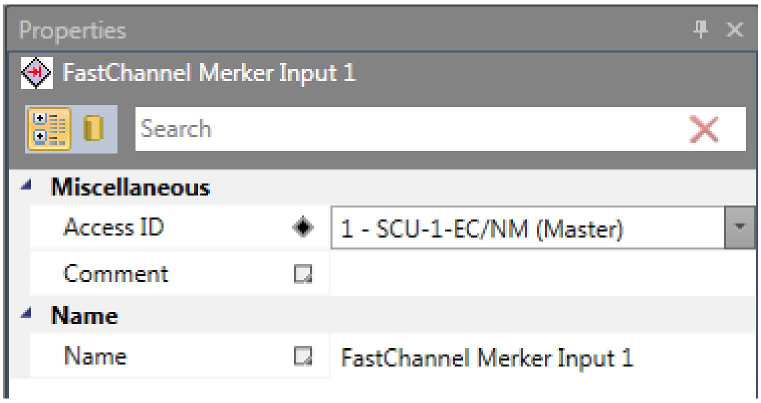
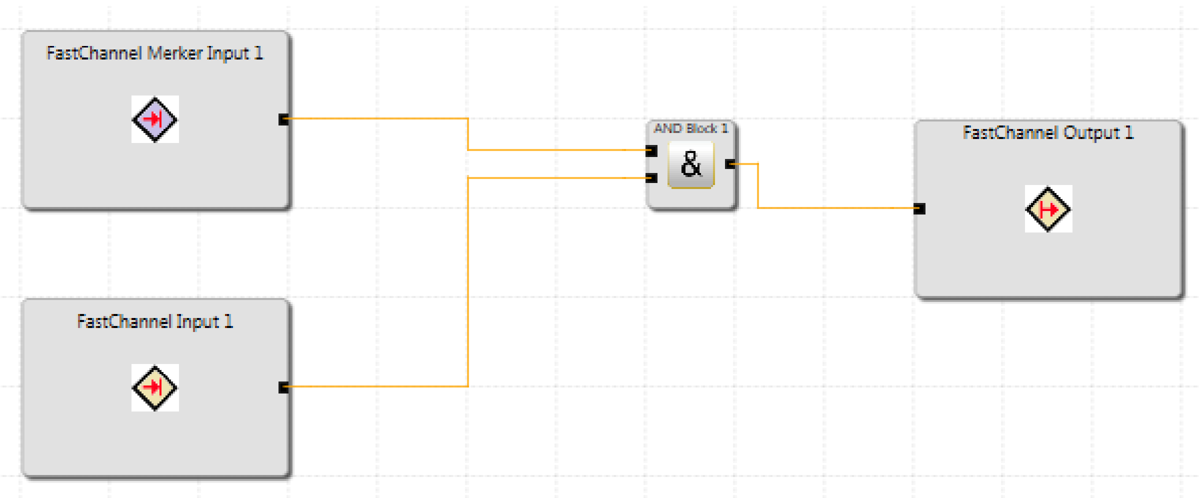


Figure 173 Access ID: selection of bit number of flag

Program example:

Enabling of output on Slave device via "Standard" functional scheme (standard reaction time) and disabling via FastChannel processing (FastChannel response time)



4.12. Content of the library

The library offers all available blocks to create the desired function block diagrams.
Only the elements that can be used in the selected plan are displayed.
The blocks can be added with Drag&Drop in the plan view, and they can be edited in the properties window.

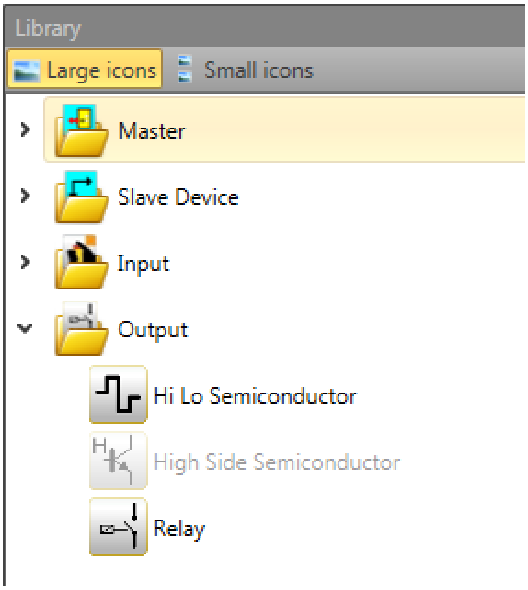


Figure 174 View of the library – terminal scheme selected

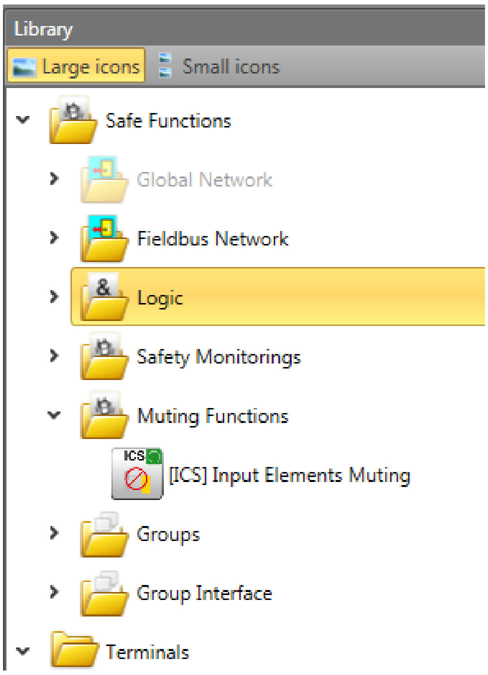


Figure 175 View of the library – functional scheme selected

The resource control of the block elements of the SCU system manages the available elements. The number of available elements can be restricted.

Through the automatic monitoring of resources of the block elements of the SCU module, only available elements are enabled in the program. Above all, this concerns the time-monitored peripheral devices. Some blocks depend on other blocks and are available only if these blocks are already present in the plan.
If no resources (memory) for the monitoring program are available in the SCU module, the components or function blocks are displayed grayed out in the library view.
This is the case if, e. g., all digital connections of an SCU module are occupied, or if all timer modules have been used.
By deleting the corresponding function blocks, these resources can be released again.

4.12.1. Device modules

4.12.1.1. Master devices

The Master device is the basic module for programming. Compact or modular series of the Master devices on the basis of the Slave module (extensions) can be used.

For the compact series, only I/O extensions can be configured.
For the modular series, I/O extensions or axle extensions up to the maximum number of Slave devices permitted by the Master device can be used.

A SafePLC^2 document can contain programmes for several Master different Master devices. The Master devices with this capability can communicate with each other via the SMMC network.

NOTICE: In the modular series, I/O devices can be configured to the maximum number of Slave devices.

4.12.1.2. Slave devices

A Slave device is an extension module offering more I/O connections or allowing the control of more axles.

There are two types of Slave devices:
I/O extensions enlarge the number of inputs and outputs.
Axle extension modules can be used to monitor additional axes. The axle extension modules also provide additional inputs and additional outputs.
If the I/O extensions or the axle extensions are connected with their Master via the SDDC network, they are shown in the document browser of the local I/O or in the local axis folder

4.12.1.3. Peripheral devices

Peripheral devices are external blocks which are connected to the inputs / outputs of the SCU module, and provide input signals / output signals.
They can be inserted in the terminal scheme or in the wiring scheme, where they are automatically connected with the corresponding connections of the SCU devices.
After the peripheral devices have been inserted, the corresponding function block is created. (In the functional scheme) Together with other function blocks, this function block can be used to configure the desired function of the system.

4.12.2. Input elements

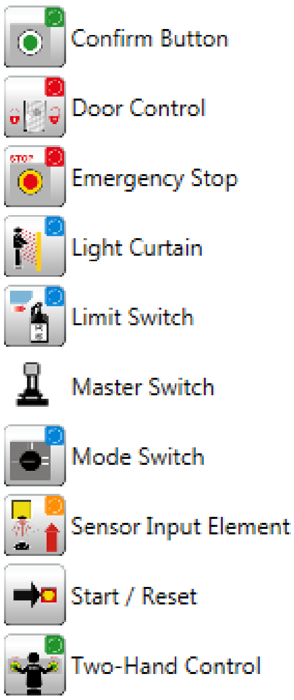


Figure 176 List of input blocks

Input elements create the digital connection between one or more connected sensors or other subordinate switching devices in the SCU system. They provide data about the operating status of the plant that is monitored by the SCU module. These components, which, from the point of view of the SCU module, are situated outside the device, can be added or configured only in the terminal scheme or in the wiring scheme. Except the operation mode switch, every input element provides a logic output signal "0" or "1" for further processing in the PLC.
The units are structures according to their application, and according to the type of the input signal. Thus, a targeted resource monitoring of the SCU module is possible.
The input blocks are structured according to their application (e. g. "Confirm" button).

The following paragraphs contain details about this type (e. g. the "Confirm" button).

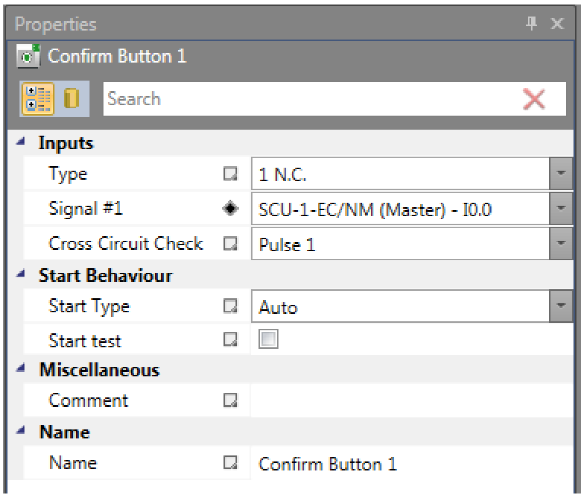


Figure 177 properties of the confirm button

NOTICE:
The configuration of the input block has significant effects on the performance level. (cf. Installation manual SCU – Master and Slaves).

Inputs that are not used are always assigned to Impulse 1 (standard configuration). Nevertheless, in the configuration report inputs that are not used, are listed with the standard configuration.

The configuration of the digital inputs is always based on the same process:

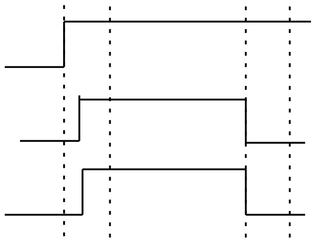
- **Switch type**
  Switch type of the component that is connected to the SCU module. The number of corresponding input signals and the monitoring behaviour of the SCU module vary with the selection.
  In case of time-monitored switching elements, another signal change must take place at t = 3 s after the first signal. If this does not happen, a malfunction is detected.

- **Signal n°**
  Assigned number of the external signal to the digital input of the SCU module. This selection list shows the identifiers of the input signal of the SCU module (e. g. "E1") that have not yet been used. The identifiers are assigned by the user. A double allocation of the input signals is prohibited. If the resources of the SCU module are nearly exhausted, and if the selection of the switch type required too many input signals, the selection list remains empty. In this case, a switch type with fewer connections must be used.

- **Test of the cross-connection (cross-circuit check)**
  Source of the input signal used. Two signal impulses, impulse1 and impulse2, are available. As an alternative, the option "OFF" can be selected. To grant the reliable monitoring of short-circuits and line breaks in the SCU module, neighbouring inputs must have different impulse numbers. Otherwise, a warning is emitted.

- **Start behaviour**
  This setting determines the behaviour of the peripheral devices during a start or a reset of the system.

- **Automatic system**
  Due to this preset startup type, the SCU module can be booted without a user feedback.

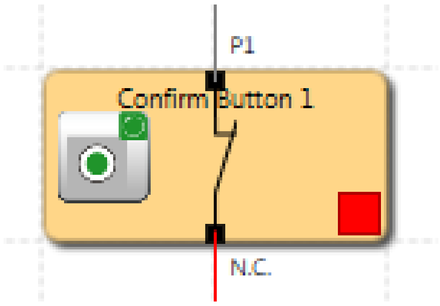
Startup type	Function	Plan
automatic start	Automatic start after reset of equipment. The output of the input element switches to “1” if the safety circuit is closed / active according to the definition of the switch type.	 [Device Run/start-up] [Switching function] [Output]

<u>Monitoring</u>
Release of the monitored input element with a falling edge of the corresponding monitoring input. This is always necessary if the monitored input element is to be switched.

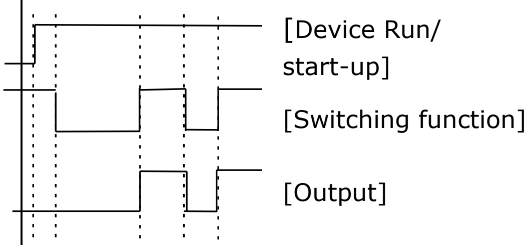
<u>Example</u>: start of a drive only after confirmation by the operating personnel.
With a monitored startup type, an additional connector is provided to connect the startup element. Here, continuous behaviour for the monitoring of the input element during the startup phase can be configured.

- **Start test**
  Manual start after equipment reset or after interruption of the preset safety circuit, including the test of the connected monitoring equipment. The monitoring equipment must trigger once in monitoring direction, and it must switch on again afterwards. Afterwards, normal operation follows. This one-time triggering of the input element during the start (or during reset) of monitored equipment warrants the functioning of the input element at the time of start. A start test can be carried out for all elements except the selection switch for the operation mode.

  An activated start test is indicated by a red rectangle around an inserted function block.



Every input block can carry out an automatic function test (= start test). In total, two switch elements can be configured for the start test.

Startup type	Function	AWL	Plan
Start test	Manual start after a restart or a reset of an alarm, including the test of the connected monitoring equipment. The monitoring equipment must trigger once in monitoring direction, and must then switch on again. Afterwards, normal operation follows. E1: switching function y1: auxiliary flag	LD E1 ST MX.y1 LD NOT MX.y1 ST MEAA_EN.1 LD MX.y1 ST MEAA_EN.2 LD MEA.1 AND MX.y1 ST MX.2	

- **Comment**
  Text to be shown on the block. You can enter your own comment.

4.12.2.1. List of input elements

The "Confirm" button

Switch type	Description	Comment
1 (1 N.C.)	1 opening contact	Activation switch (standard)
2 (1 N.O.)	1 closing contact	Activation switch (standard)
3 (2 N.C.)	2 opening contacts	Activation switch with larger number of requests
4 (2 time-monitoredd N.C.'s)	2 opening contacts (time-monitored)	Activation switch (monitored)

Emergency stop

Switch type	Description	Comment
1 (1 N.C.)	1 opening contact	Emergency stop (standard)
3 (2 N.C.'s)	2 opening contacts	Energency stop with higher requirements
4 (2 N.C.'s; time-monitored)	2 opening contacts (time-monitored)	Emergency stop (monitored)

Door control

Switch type	Description	Comment
3 (2 N.C.)	2 opening contacts	Door monitoring with higher requirements
4 (2 N.C. time-monitored)	2 opening contacts (time-monitored)	Door monitoring (monitored)
5 (1 N.O. 1 N.C.)	1 closing contact + 1 opening contact	Door monitoring with higher requirements
6 (1 N.O. 1 N.C. time-monitored)	1 closing contact + 1 opening contact (time-monitored)	Door monitor (monitored)
7 (2 N.O. 2 N.C.)	2 closing contacts + 2 opening contacts	Door monitoring with higher requirements
8 (2 N.O. 2 N.C. time-monitored)	2 closing contacts + 2 opening contacts (time-monitored)	Door monitor (monitored)
9 (3 N.C.)	3 opening contacts	Door monitoring with higher requirements
10 (3 N.C. time-monitored)	3 opening contacts (time monitored)	Door monitoring (monitored)

Two-hand control

Switch type	Description	Comment
11 (2 toggles)	2 closing contacts + 2 opening contacts	Two-hand switch with higher requirements; type III C
12 (2 N.O.)	2 closing contacts	Two-hand switch (monitored); type III A

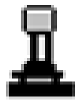
NOTICE: These input elements cause a fixed impulse assignment which the user cannot influence. No fault evaluation! No time monitoring when changing to the inactive state.

Limit switch

Switch type	Description	Comment
1 (1 N.C.)	1 opening contact	Activation switch (standard)
2 (1 N.O)	1 closing contacts	
3 (2 N.C.)	2 opening contacts	Activation switch with higher requirements
4 (2 N.C. (time-controlled)	2 opening contacts (time-monitored)	Activation switch (monitored)

Master switch

Switch type is not supported with this device type



Light curtain

Switch type	Description	Comment
3 (2 N.C.)	2 opening contacts	Light barrier with higher requirements
4 (2 N.C. (time-monitored)	2 opening contacts (time-monitored)	Light barrier (monitored)
5 (1 N.O. 1 N.C.)	1 closing contact + 1 opening contact	Light barrier with higher requirements
6 (1 N.O. 1 N.C. (time monitored)	1 closing contact + 1 opening contact (time-monitored)	Light barrier (monitored)

Operation mode switch

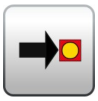
Switch type	Description	Comment
13 (N.C. N.O.)	Selection witch opening contact / closing contact	Selection switch (monitored)
14 (3 Phases)	Selection switch 3 steps	Selection switch (monitored)
15 (4 Phases)	Selection switch 4 steps	Selection switch (monitored)

NOTICE: If the status of the switch is changed, the SafePLC² program warrants that the outputs of the module are disabled. (Attention: standard 60204-Part1-Section 9.2.3).

Sensor

Switch type	Description	Comment
1 (1 N.C.)	1 opening contactt	Sensor input (standard)
2 (1 N.O.)	1 closing contacts	Sensor input (standard)
3 (2 N.C.)	2 opening contacts	Sensor input with higher requirements
4 (2 N.C. (time-monitored)	2 opening contacts (time-controlled)	Sensor input (time-monitored)
5 (1 N.O. 1 N.C.)	1 closing contact + 1 opening contact (time-monitored)	Sensor input monitored

Start element / reset element



This input element offers an extended monitoring function and the possibility to reset an alarm that has occurred.

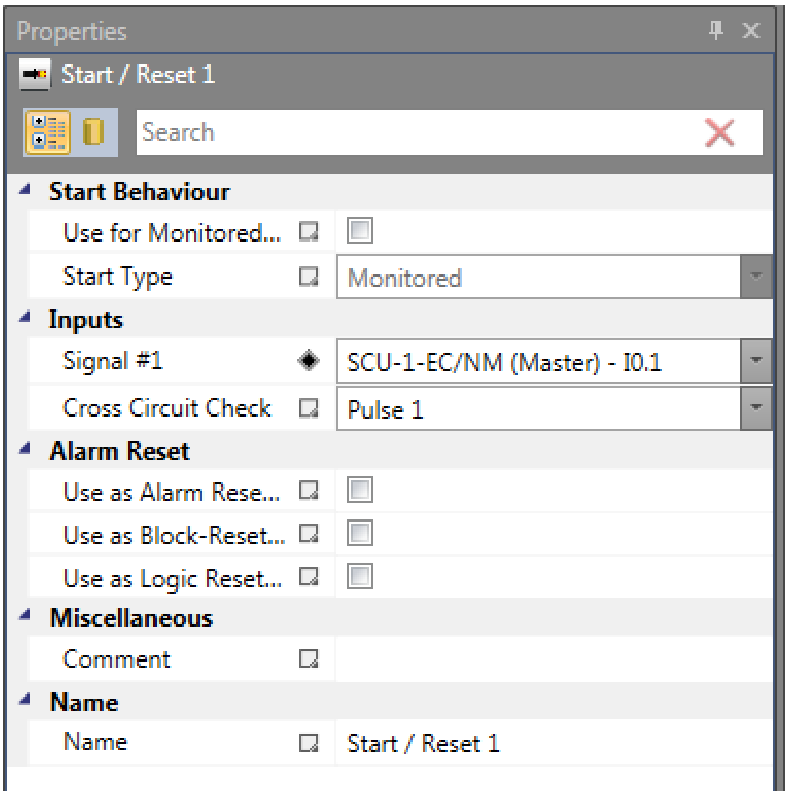
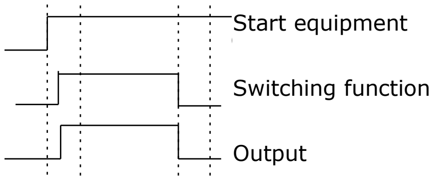
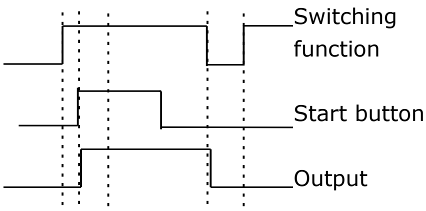
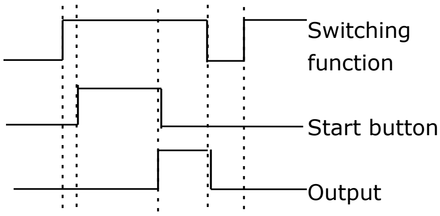


Figure 178 Properties of the start element / the reset element

Use of the monitored start

With the start monitoring being enabled, automatically a special AWL code segment for the monitoring of an assigned input segment during a restart or during the reset of an alarm of the equipment to be monitored/ the alarm tob e monitored is created.
This function-relevant test of a peripheral element (e. g. triggering of an emergency stop) shall ensure the function during the start of the equipment.

Startup type	Function	AWL	Plan
Auto	Automatic start after the reset of the equipment or after the input has been enabled. The output of the input element changes to "1"when the safety circuit is closed / active according to the definition of the switch type.		
Manual start (by hand)	Manual start after the reset of the equipment. This output of the input element changes to "1" when the safety circuit is closed / active according to the definition of this switch type, and the start button has been pressed once. The output changes to "0" after the safety circuit is open E1: switch function E2: start button M.(X1): auxiliary flag 1	LD E.1 AND E.2 S M.(X1) LD NOT E.1 R M.(X1) LD M.(X1) AND E.1 ST IE.X	
Start (monitored)	Manual start after equipment reset with monitoring of start circuit for static 1-signal. The output of the input element changes to "1" if the safety circuit is closed according to the switch type, and the start button has been pressed and released once. The output changes to "0", after the safety circuit is open. E1: switch function E2: start button M.(X1): auxiliary flag 1 M.(X2): auxiliary flag 2	LD E.1 AND E.2 S M.(X1) LD NOT E.1 R M.(X1) LD M.(X1) AND E.1 AND NOT E.2 S M.(X2) LD NOT E.1 R M.(X2) LD M.(X2) AND E.1 ST IE.X	

4.12.2.2. Listing of the startup types via the "confirm button" button

The input of the start element to be monitored must be connected with the output of the input element by the description "Start element". Various elements can be monitored.

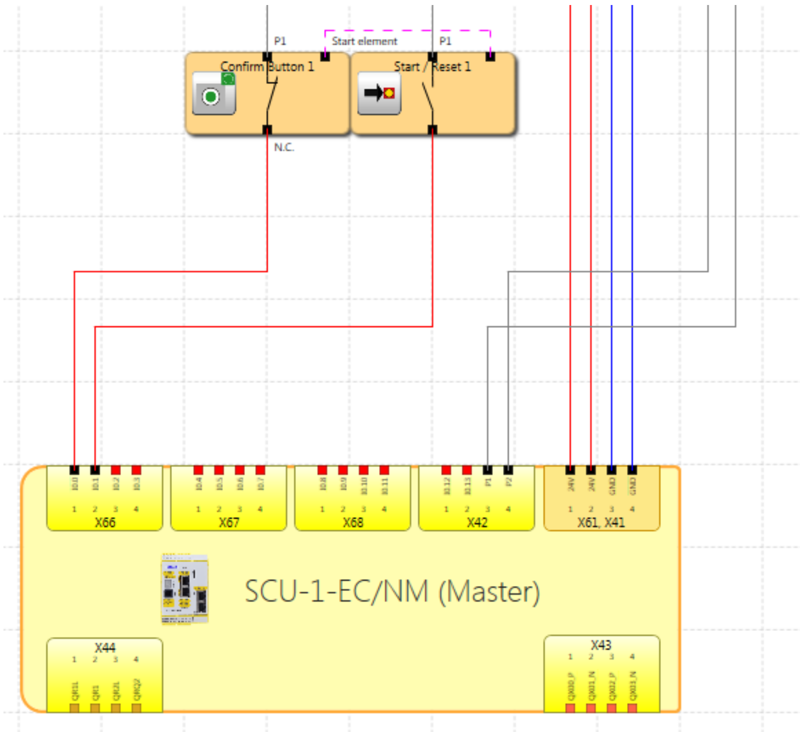


Figure 179 Start block / reset block connected with monitored start

NOTICE: When the corresponding input element is edited, the connection to the start element is deleted, and cannot be restored any more. After the connection has been deleted, it must be manually completed.

- **Input: signal nº: 1**
  Like with the input elements, this selection list is used to determine the input in the SCU module, with which the button for the start element shall be connected. Internally, this input is restricted to the assignation to a basic module (I0.1 to I0.14).

- **Use as "alarm reset (normally open contact) "**
  If this option is set, the corresponding button can be used to reset (to quit) a device alarm that can occur during operation. Thus, the user is not forced to reset an occurring error with the "Function" button at the SCU module. A special programme code is created, but in case of an alarm reset this input is directly processed by the SCU module.

  NOTICE: If a reset element is used, no monitoring of the cross-connection can be processed for this input. In this case, the test of the cross-connection is set to "AUS" ["OFF"].

In case of error messages of the type "Fatal Error ", the SCU module must be restarted. The input of the alarm reset can be operated with 24 V continuous voltage and is triggered by the edge.

- **Use as Block-Reset (normal contact)**
  Up to 6 Block-Reset elements can be inserted. The Block-Reset elements are used to reset the monitoring functions (e. g. SLS, SOS, etc.).
  For the resettability of monitoring functions, cf. Overview of safety modules.

  NOTICE: For time based functions, the condition T ≤ 3sec is only possible using the block reset. For more information refer to the SCU installation manual -Master and Slaves, cf. 11.2.2 "Reset-Timing"

- **Use as logic reset (normally open contact)**
  With this option, the rest function / the confirmation is available in the functional scheme for further processing. In this case, the output of the function block is created automatically, and can be used to be processed with a logic function. With this option the reset function / the confirmation function in the functional scheme is available for further processing. In this case, the output of the function block is created automatically and can be used for the connection with a logic function. Normally, this logic reset signal is used to confirm RS-FlipFlops. This option is designed for the case that an occurring SCA error is always pendant on the RS module, and that thiss error can only be reset by pressing the RESET button on the RS module.

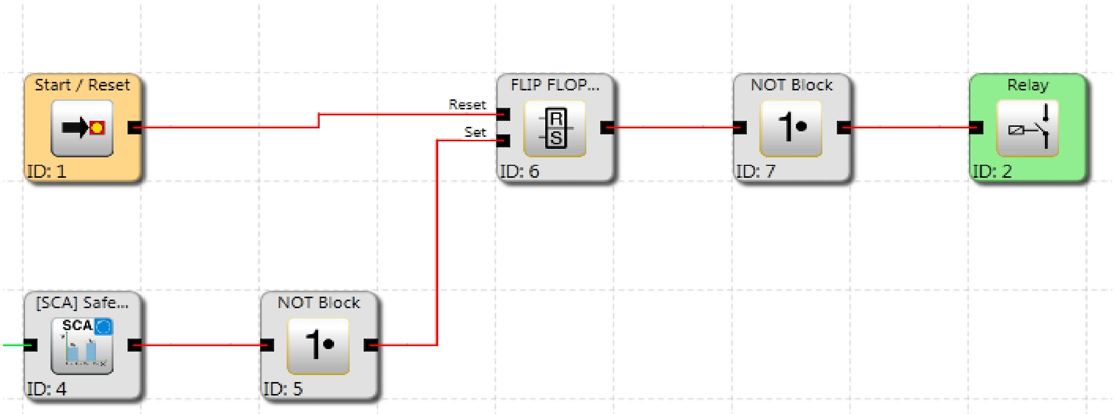


Figure 180 Start / reset to save and reset of errors of the SCA-Module via RS-FlipFlop

Switch type	Comment	Classifcation category	classification SIL
1 closing contact	Alarm reset Standard (evaluation of the edge)	--	--
1 closing contact	Logic reset Standard	category 3	SIL 2
1 closing contact	Start monitoring Standard (optional function)	--	--

4.12.3. Output blocks

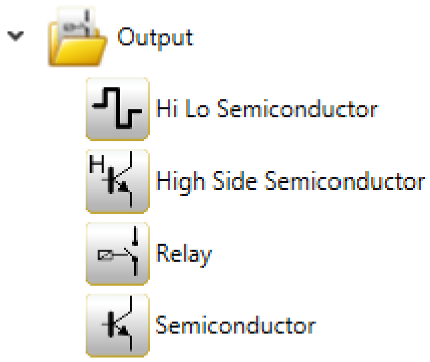


Figure 181 List of output elements

The output blocks create the digital connection between one or several external switching circuits in the SCU system. Every block is triggered by a logic input signal "0" or "1" via the functional scheme.

NOTICE:
EMU monitoring
EMU monitoring is supported by the SDU slave modules.
With the device variants SCU and SIO, EMU cannot be used.
This function is deactivated in the SafePLC² interface and is displayed grayed out.
The functionality can be realised via monitoring block EDM.

4.12.3.1. EMU Monitoring

The multiplication of contacts and power normally requires additional switching devices, which are triggered through the outputs of the **SCU system** (SDU Slave modules). EMU monitoring realizes the "safety relay" function by processing an external feedback circuit. Applications with higher safety requirements (Cat. 4 of EN 954-1) among others require functional monitoring for these swichting devices. For this purpose, the switchgear must be equipped with positively driven auxiliary contacts. For details, refer to "SCU installation Manual – Master an Slaves", chapter. 10.3.5.2 "Wiring examples for safe digital outputs IOs". Contacts to be monitored are switched in series and are closed when in idle state. It is verified wether all contacts are closed when the output is not switched on and open in swichted on state. Time related expectations can be parameterized. The same source as for the inputs are also used for the supply of the contacts to be monitored.
The contacts to be monitored must be supplied via fixed, assigned cycle series.

NOTICE: Details to this subject can be found in the examples of the circuits in the installation SCU manual (chapter 10.3.5.2.).

For a more detailed description of the EDM function, refer to chapter 4.12.5.3.1 "Overview of safety modules".

4.12.3.2. List of output elements

HiLo semiconductor

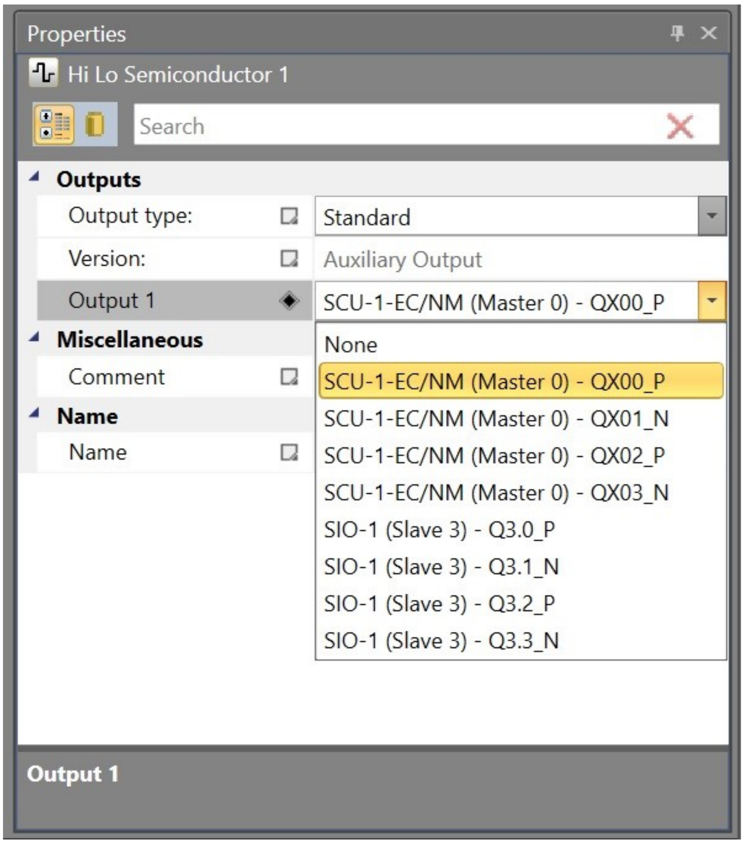


Figure 182 Properties window "HiLo semiconductor"

Output type

Standard: "HISIDE"(= P-switch) or "LOSIDE" (= N-switching) can be selected as standard outputs. The use of simple standard outputs is not suitable for safety outputs.

Redundant: This option obliges a combination "HISIDE" outputs and "LOSIDE" outputs as auxiliary output or safety output.

Output as auxiliary or safety output
HiLo semiconductors can be used individually (as standard outputs) or grouped (as safety outputs) (for details see the installation manual).
For exact contact monitoring, see the chapter "EMU Monitoring".

Highside semiconductor

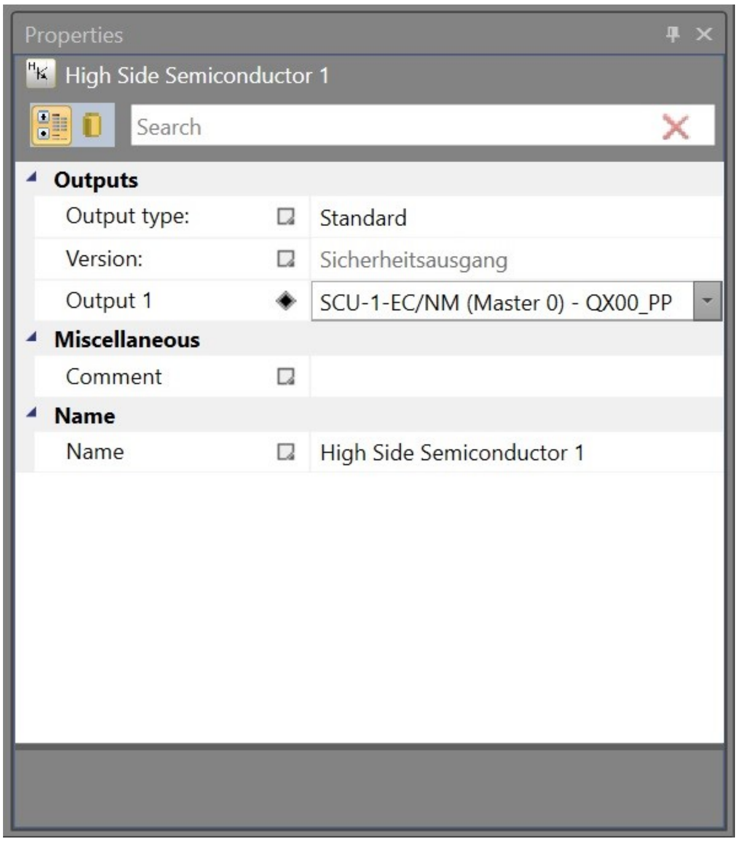


Figure 183 Properties window "Highside semiconductor"

Relay

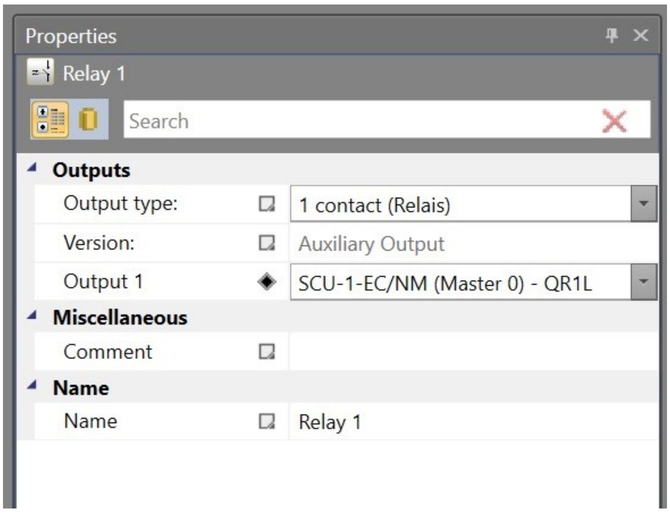


Figure 184 Properties window "Relay"

Output type

Standard: 2 simple relays (K1 to K2) can be evaluated independently.

Redundant: Two relay outputs are combined, and are always interconnected.

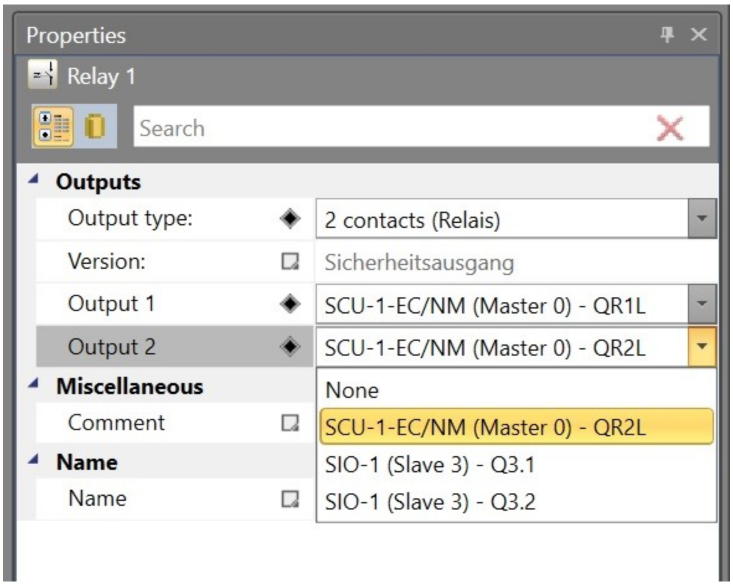


Figure 185 Properties windwow with 2 relays

NOTICE:
Observe the explanatíons in the installation manual when inserting a relay in safety applications.

For EMU monitoring, c. f. the chapter "EMU Monitoring".

Semiconductor

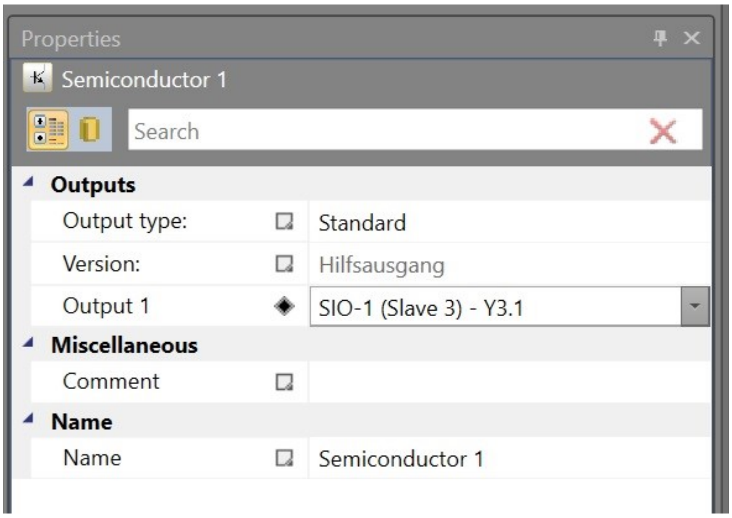
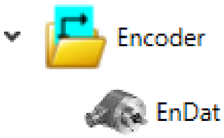


Figure 186 Output as auxiliary outputs

Certain semiconductor outputs can only be used as auxiliary outputs. Thus, they are not suitable for safety applications. (For details, see the installlation manual SCU.)

4.12.4. Encoder combinations

Encoder configurations are configured in the Slave modules. Via FSoE, they are transferred to the SCU modules as safe position ans safe velocity.
The library lists the available encoders.



In the library window of the terminal scheme, you can select between different manufacturers of axis modules.
The configuration of the axis modules of the SDU devices and the third-party devices is described further below. To be able to use the axes, a profile must be selected for the corresponding assemblies in the device property.

4.12.4.1. SDU device configuration

In the SDU module, the encoder combination can be selected in the device properties. Up to two axes can be parameterized according to the module type.

None
HTL EX + Not Connected
Incremental + Not Connected
Incremental + Proxy 1 Counter
Incremental + Proxy 2 Counter
Incremental + Incremental EX
Incremental + HTL EX
Incremental + Resolver EX
Incremental + Sinus Cosinus EX
Incremental + SSI Absolut EX
Proxy 2 Counter + SSI Absolut EX
Proxy 1 Cnt Ext. Supervision + Proxy 1 Cnt Ext. Supervision
Resolver EX + Not Connected
Sinus Cosinus + Not Connected
Sinus Cosinus + Proxy 2 Counter
Sinus Cosinus + Incremental EX
Sinus Cosinus + HTL EX
Sinus Cosinus + Resolver EX
Sinus Cosinus + SSI Absolut EX
Sinus Cosinus EX + Not Connected
SSI Absolut + Proxy 2 Counter
SSI Absolut + HTL EX
SSI Absolut + Resolver EX
SSI Absolut + Sinus Cosinus EX

NOTICE:
To use the safe position or speed, suitable encoder combinations must be selected.
The setting of the encoders must always refer to a common axis. If the two encoders are connected to different mechanical positions and these positions are connected e.g. with an intermediate gear, the measuring range must be set to one of the two encoder positions and the transmission ratio must be taken into account for the other encoder.

4.12.4.1.1. Encoder type

Selection of the function type of the encoder:

- **Incremental encoder**
  Position and speed are determined via pulses/distance.
- **SIN / COS**
  Position and speed are determined via sine and cosine/distance.
- **Absolute encoder**
  Absolute encoder, i.e. the position is determined absolutely and remanently. When activating position processing in the axis range, the "Offset" input field can be activated additionally.
- **Proximity switch 1Z**
  Position and speed are recorded by a pulse counter.
- **Proximity switch 2Z 90**
  Position and speed are recorded by two pulse counters.
- **Analog encoder**
  Scaling of the applied analog sensor signals is possible via an analog encoder.
- **Not connected**
  No secondary encoder

NOTICE: For position monitoring, at least one of the two encoders must be designed as an absolute encoder. If neither of the two sensors is an absolute encoder, the fields for the position inputs in all other input masks of the monitoring function are deactivated.
If an absolute encoder has been selected, the system displays the data format range in the properties for further selection.
In the case of an incremental encoder, the pulse multiplication takes place in the device. The resolution of an encoder must always be specified in pulses per revolution (PPR) via the "Resolution" calculation button. The multiplication depends on the set encoder configuration and runs internally automatically. Further information can be found in the installation manual.

4.12.4.1.2. Setting the encoder‘s range

Configuration only concerns the control software of the encoder. For correct functioning, an advanced hardware setting of the encoder interface. For details concerning this subject, c. f. the installation manual of the SCU.

<u>Properties of the encoder range</u>: Parameters depend on the encoder type.

Parameter	Description	Value
Direction	Selection of sensor counting direction	Up/Down
Supply voltage	Power supply encoder	8 V, 10 V, 12 V, 20 V, 24 V
Resolution (i) (i)= double-click Open calculation dialog	Encoder resolution in relation to the measuring axis in the predefined context (linear or rotary)	1 – 2 000 000 Incr/1000 or Inkr/U
Shift (i) (i)= double click Open calculation	Shift value for the position encoder. Usable if position processing is activated.	0 – 268435455 Inkr
Encoder type (SinCos EX)	Activation of high-resolution mode for slow counting of SinCos encoder	Normal -> no high resolution HighRes -> high resolution
SSI Interface (Absolute encoder)		
Interface type	SSi Version	SSI-Masterclock, SSI-Listener
Data format	Format of position data	Binär, Graycode
Frame length	Length total SSI frame	10 – 31 Bit
Data length	Length SSI data starting at MSB. In this data field e.g. no status bits are allowed (only SSI data)	10 – 28 Bit
Data index	Start index for bit information of encoder data.	Integer value: Bit position starting at LSB
Status length	Length status information (e.g. error bit, status bits)	Integer value: Length starting at LSB
Status index	Index in which status information (bit index) is listed	Integer value: Bit position starting at LSB
“Mask Err" status	Not used	
“Mask Def” status	Not used	
Resolver type (Resolver)		

Form factor	Form factor of the resolver	Out, Sine, Triangle
Resolver ratio	Resolver ratio	2:1, 3:2, 4:1, Pattern1 (Amplitude control: Off), Pattern2 (frequency control: Off), Pattern3 (frequency and amplitude control: Off)
Pole pairs	Number of pole pairs	1-8 pole pairs
Interface type	Resolver type	Master, Listener
Listener frequency	Frequency in listener mode	6 kHz – 12 kHz, 14 kHz, 16 kHz

Configuration (read only): Displayed result data related to the currently used encoders.

Column name	Meaning
Class-ID	Eindeutige ID of the encoder configuration
General flags	BIT-coded assignment D0: 1= encoder input is enabled
Modi	BIT-coded assignment for SSI interface, data format and direction of rotation D0: 1= SSI listener 0= SSI standard D1: 1= SSI binary 0= SSI GrayCode D2: 1= rising 0= falling D3: not used D4: 1= WCS
EXT-Modi	BIT-coded assignment for Encoder D0: 1= 5 V D1: 1= 12 V D2: 1= 24 V
V_Standardization	Standardization value for speed (internal calculation value)
PosStandardization	Standardization value for position (internal calculation value)
ShiftvalPos	Integer exponent for base 2. Internal calculation value for position standardization
ShiftvalSpeed	Integer exponent for base 2. Internal calculation value for speed standardization.
Shift	Shift between encoder value and position in the measuring section.
Resolution	Resolution of the encoder with respect to the measuring axis in steps/m or steps/rev.
FilterTime	Not used
Data length	Field with the data width in the encoder interface
Cycle period	Indication of the cycle duration of the PSCBR module.

V_max	Maximum speed that can be entered for setting the monitoring dialogs. Is defined via "encoder dialog maximum speed" * factor 1.5.
V_minused	Internal minimum speed for standardization calculation
V_min	Minimum speed that can be entered for setting of the monitoring dialogs.
Messlänge	Entered measuring length.
Pos_Minused	Internal minimum position for standardization calculation
Pos_min	Maximum position that can be entered for setting the monitoring dialogs.

4.12.4.1.3. Axis configuration

Encoder Type	
Encoder Type	Redundant encoder
Process encoder	Encoder 2
Parameter of working section	
Axis Type	Rotatory
Rotatory	rpm
Position Processing	☐
Sect. length	500 rev
Maximal Speed	2000 rpm
Cutoff Threshold Incr.	10 rev
Cutoff Threshold Speed	100 rpm
Speed Filter	No
Miscellaneous	
Comment	
Configuration (read only)	
Class ID	3400
General Flags	1
Modes	0
Axis Cfg ID	65
Section Length	500
Speed Filter	0
Factor Pos	1000
Factor Speed	1
Maximal Speed	2000
Cutoff Threshold Pos	10000
Cutoff Threshold Speed	100
Unit	5
Name	
Name	Axis

<u>Working range setting:</u>

Linear: The measuring range has a linear characteristic. The unit of position in this case is "mm" and the speed can be specified in "mm/s" or "m/s".

Rotatory: The measuring range has a rotating characteristic, i.e. the movement is rotating. The position is specified in "mgrd" or "revolutions", the speed in "mgrd/s", "revolutions/s" or "revolutions/min".

Position processing eneable: Processing of an absolute measuring range. This function can only be selected if an absolute encoder has been set before! If position processing is enabled, all monitoring functions of the position are also enabled.

Measuring (Section) length: Specification of the maximum measuring length of the position in mm, m or mgrd, revolutions. When position processing is activated, the application must always move within the limits of the set measuring length. Any actual position outside the defined measuring length will cause an alarm of the PSCBR axis.

Another Sect. Length: Not supported

Maximal speed: Specifies the maximum speed of the reference axis in the currently selected unit. The maximum allowed speed describes the highest speed that can be reached as maximum with the current technological system configuration. The maximum value that can be reached by the axis to be monitored should be entered here. This can possibly be a theoretical maximum speed of the current application. The set value does not refer to the safety shutdown (e.g. shutdown via SLS), but to the fail-safe, i.e. the resistance of encoders or the resistance of the mechanical conditions. Exceeding this value triggers an alarm with cutoff/alarm status. This is not a planned cut-off due to a safety-relevant exceeding of the speed, but the fail-safe of the encoders or the mechanical conditions is endangered (encoder error, error in the electrical voltage converter, etc.), since this speed is normally not reached from a drive-technical point of view. If this case occurs, the PSCBR module goes into alarm state and switches off all outputs. This means that the maximum speed must always be above the Cutoff threshold speed of a safety function. The aim is to detect a fault on the safety axis with the help of the measuring systems. The value entered in this field changes both the dimension of the encoder resistance in relation to the increment Cutoff threshold limit and the speed Cutoff threshold limit. A higher maximum speed allows higher Cutoff threshold limits between the encoders. The maximum value should therefore not be selected too high, as otherwise the cutoff limits could be too high for the failure safety of the encoders among each other. The table with the property values of the configuration (read only) shows these calculated limits for the variables V_max, V_min.

Cutoff threshold limit: The cutoff limit specifies the permissible speed/position deviation between two detection channels/encoder channels. It can depend on the arrangement of the sensors and the maximum mechanical clearance (e.g. gear and spring rate) between the two measuring points. The smallest possible value at which monitoring is not yet triggered in normal operation should be selected taking into account the dynamic processes (e.g. load/play in the gearbox).

Speed filter: Average filter over the recorded speed values of the encoder for damping speed peaks at low resolution or deviation of the connected sensor.

When the filter is switched on, the set response time of the overall system increases by the set time. The filter affects the speed-relevant parameters of the monitoring modules.

Configuration (read only): Displayed result data in relation to the currently used encoders.

Colum name	Meaning
Class-ID	Unique ID of the axis configuration
General flag	BIT coded assignment D0: 1= axis input is enabled
Modes	BIT coded assignment for Position processing and type of measuring range D0: 1= position sprocessing activ 0= inactive D1: 1= linear 0= rotary
Axis CFG ID	Unique ID for both encoder configurations
Section length	Measuring length for the position from the main dialog
Factor Pos	Factor for position calculation (standardization)
Factor Speed	Factor for speed calculation (standardization)
Maximal Speed	Standardized maximum speed
Cutoff threshold Pos	Value of cut-off limit increasing, but not standardized
Cutoff threshold Speed	Value of cut-off limit for speed, but not standardized
Unit	Unit of the displayed values 1 = UNIT_MM 2 = UNIT_M 3 = UNIT_MDEG 4 = UNIT_REV_SEC 5 = UNIT_REV_MIN

NOTICE*:* The displayed values serve the technical support of the encoder configuration and are used for the standardized calculation in the PSCBR module!

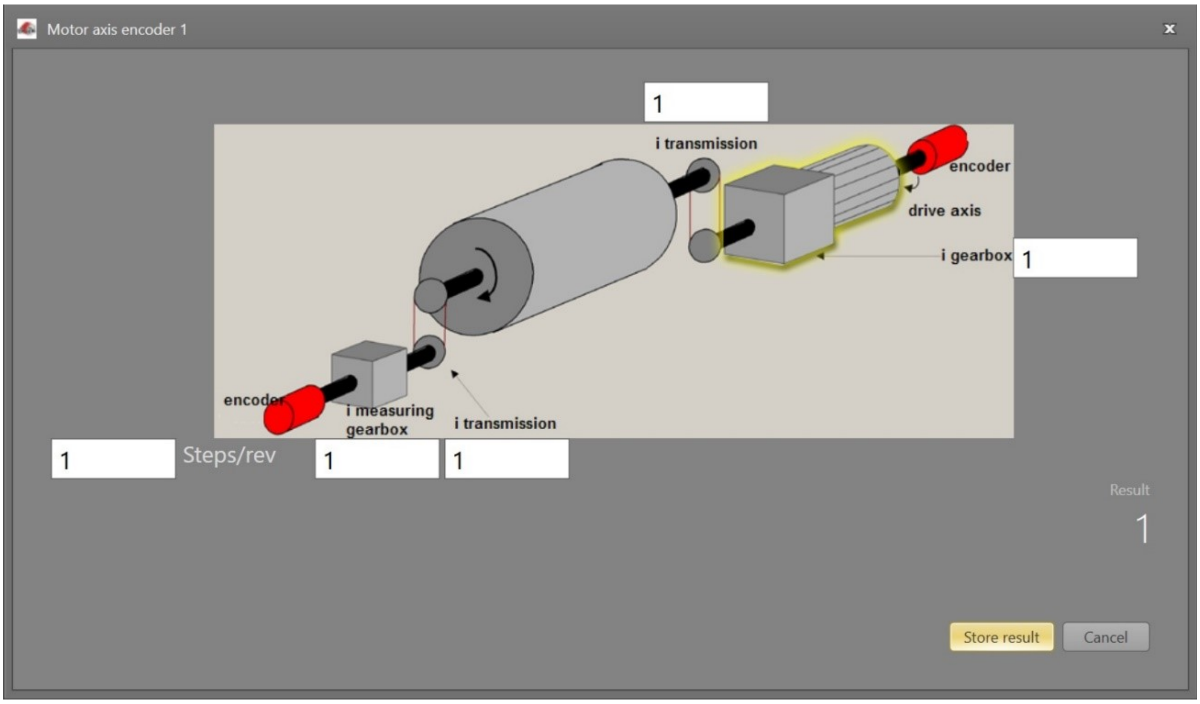
- Defining the characteristic of the measuring length as linear or rotary generally affects all position and speed inputs of the monitoring functions. This changes the input from mm, m or mm/s, m/s to mgrd, U or mgrd/s, U/s or rpm and vice versa.
- The maximum measuring length and the maximum speed must be defined. A missing or wrong entry can cause the monitoring functions to respond undesirably.

- In general, the first encoder has the function of a process sensor and the second encoder is a reference sensor. With the combination of absolute encoder/incremental encoder, the absolute encoder is always used as process sensor. If encoders with different resolutions are used, the encoder with the higher resolution must be configured as the process sensor.

4.12.4.1.4. Determination of the resolution

Definition of the resolution with respect to differently characterized measuring lengths. The values must always be entered via calculation button in the resolution of the encoder. Encoder resolution with respect to the measuring axis in the predefined context (linear or rotary). Input data for the definition must be saved for definition.

Rotating measuring lengths



Reference axis	Input values		Resolution with respect to measuring length
Input axis (process axis)	Encoder 1: Resolution Gb 1 In measuring gearbox in intermediate transmission	A_Gb1 in [steps/rev] I_MG I_VG	$Gb1 = I\_MG \cdot I\_VG \cdot A\_Gb1$
	Encoder 2: Resolution Gb 2 in gear box	A_Gb2 in [steps/rev] I_G	$Gb2 = I\_G \cdot I\_VA \cdot A\_Gb2$

	in intermediate transmission for drive	I_VA	
Drive axis	Encoder 1: Resolution Gb 1 in measuring gearbox in intermediate transmission ∅ measuring gearbox in gearbox in intermediate transmission	 A_Gb1 in [steps/rev] I_MG I_VG D_MR in [mm] I_G I_VA	$Gb1 = \frac{I\_MG \cdot I\_VG \cdot A\_Gb1}{I\_G \cdot I\_VA}$

<u>Input Example 1:</u>
In a manufacturing device, the speed of certain manual processes must be monitored for a safety-reduced value and the direction of standstill and movement. The motion to be actively monitored is a rotary motion. The drive works with an electric motor with an integrated motor feedback system and an intermediate transmission.

<u>Selection of the module or module</u>
Selection of the encoder type:
No monitoring of positions required -> absolute encoders are not required, speed recording with incremental encoders is sufficient.

Determination of the measuring length:
The axis of rotation of the manufacturing device is selected as the reference axis. The following parameters are selected:

- rotating
- Measuring length unknown
- Reference axis is rotation axis => designation = mgrd

Setting the parameters for encoder 1:
Encoder 1 is directly connected to the output axis of the gearbox = load axis. An encoder with the data: Pulse generator A/B track, 5000 pulses/revolution is used..

The following parameters are selected:

- Encoder type: Incremental encoder
- Resolution:

Encoder 1:	
Resolution Gb 1	5000 [Steps/rev]
in measuring gearbox	1
in intermediate transmission	1

Gb1 = I_MG · I_VG · A_Gb1 = 1 · 1 · 5000 = 5000;

Seeting the parameters for encoder 2:
The existing motor feedback system is used as encoder 2. The motor is connected to the rotary axis of the manufacturing device via an intermediate transmission.

The encoder interface is connected to the pulse outputs of the voltage converter. The sensor data are as follows: Hiperface, 1024 I/rev. According to the data sheet of the voltage converter manufacturer, the sine/cosine tracks of the Hiperface encoder are outputs in the form of pulses -> emulated pulse output of the voltage converter = pulse generator, A/B track, 1024 I/rev.

The following parameters are selected:

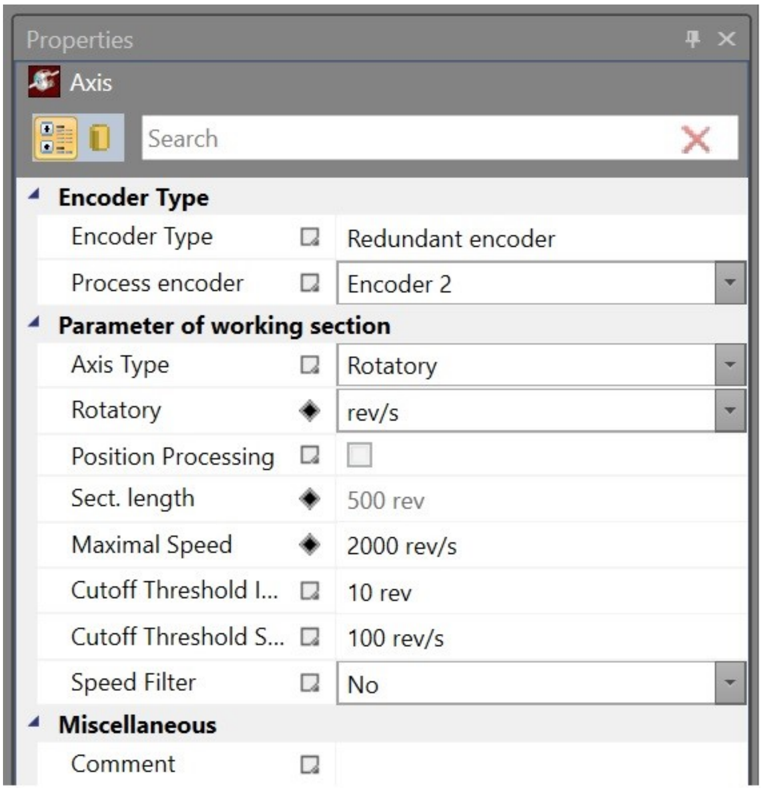
- Encoder type: Incremental encoder
- Resolution:

Encoder 2:	
Resolution Gb2	1024 [Steps/rev]
in gearbox	350
in intermediate transmission for drive	1

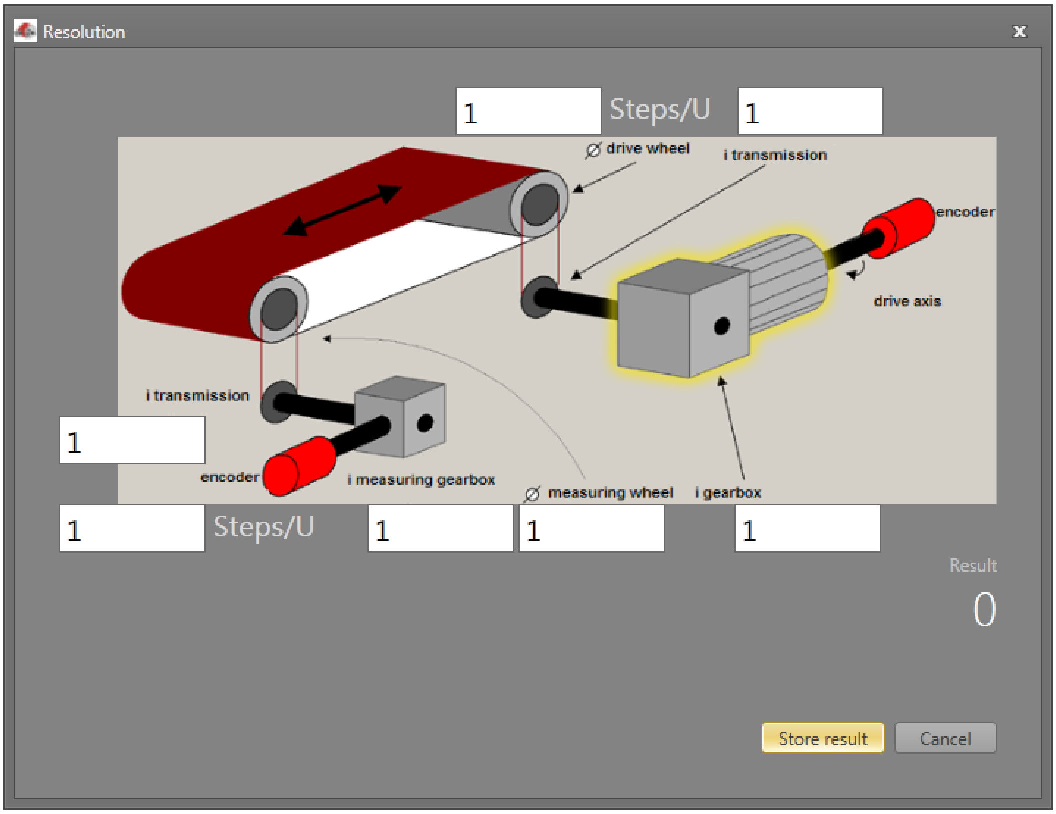
Gb2 = I_G · I_VA · A_Gb2 = 1024 · 350 · 1 = 35840;

Determination of the maximum speed:
The maximum speed of the output axis results from the maximum motor speed. In rev/s with respect to the load axis and with Nmax = 1500 rpm, this gives (1500 [rpm] / 60 [s]) / 350 = 0, converted to mgrd/s this gives 0.07142 [1/s] * 360 *10^3 [mgrd] = 25 714 [mgrd/s].

Input of the maximum deviation:
The empirical measurement gives a maximum difference between the two measurement points of 80 mgrd. A value of 100 mgrd is selected.



Linear measuring length



Reference axis	Input value		Resolution with respect to measuring length
Input axis (Process axis)	Encoder: Resolution Gb 1 in measuring gearbox in intermediate transmission ∅ measuring wheel	A_Gb1 in [Steps/rev] I_MG I_VG D_MR in [mm]	$Gb1 = \frac{1000}{D\_MR \cdot \pi} \cdot I\_MG \cdot I\_VG \cdot A\_Gb1$
	Encoder 2: Resolution Gb 2 in gearbox in intermediate transmission for drive ∅ drive wheel	A_Gb2 in [Steps/rev] I_G, I_VA, D_AR in [mm]	$Gb2 = \frac{1000}{D\_AR \cdot \pi} \cdot I\_G \cdot I\_VA \cdot A\_Gb2$

Motor axis	Encoder 1: Resolution Gb 1 in Measuring gear in ntermediate transmission ∅ measuring gear in gearbox in intermediate transmission for drive ∅ drive wheel	A_Gb1 in [Steps/rev] I_MG I_VG D_MR in [mm] I_G I_VA D_AR in [mm]	$$Gb1 = \frac{\frac{1000}{D\_MR \cdot \pi} \cdot I\_MG \cdot I\_VG \cdot A\_Gb1}{\frac{1000}{D\_AR \cdot \pi} \cdot I\_G \cdot I\_VA \cdot A}$$

Input example 2:

On a production machine, access to the work area is to be ensured at certain points on the main input axis for manual input or settings. The drive remains active in this position and is only monitored for standstill. The limits of the working area are variable and are to be monitored electronically in a safety-related mode instead of the mechanical safety limit switch. The movement to be actively monitored is a linear movement. An absolute encoder is connected to this main drive axis of the linear length measuring system. The drive functions with an electric motor with an integrated motor feedback system and an intermediate transmission. The output shaft of the intermediate transmission is connected to a drive wheel with ∅ 31.83 mm (= 100 mm circumference).

Selection of the module

Selection of encoder type:

Monitoring of the positions is required -> absolute encoder is required, incremental detection and reference switch are sufficient for the second encoder.

Determination of the measuring length parameters:

The main axis of the plant is selected as the reference axis.
The following parameters are selected:

- Linear
- Measuring length = 600 mm
- Reference axis is rotation axis => designation = mm

Setting the parameters for encoder 1:

Encoder 1 is connected directly to the drive axis. Absolute encoder SSI, 4096 steps/rev is used.

The following parameters are selected:

- Encoder type: Absolute encoder
- Data format: SSI
- Resolution:

Encoder 1:	
Resolution Gb 1	4096 [Steps/rev]
in measuring gear	1
in intermediate transmission	1
∅ drive wheel	31,83

$$Gb1 = \frac{1000}{D\_MR \cdot \pi} \cdot I\_MG \cdot I\_VG \cdot A\_Gb1 = \frac{1000}{31{,}83 \cdot \pi} \cdot 1 \cdot 1 \cdot 4096 = 40960$$

Defining the parameters for encoder 2:
The existing motor feedback system is used as encoder 2. The motor is connected to the drive wheel via an intermediate transmission. The ratio of the gear is 4.51 times the Ø of the drive wheel 31.831 mm.
The encoder interface is connected to the pulse outputs of the voltage converter. The encoder data are as follows: Hiperface, 1024 I/rev. According to the data sheet of the voltage converter manufacturer, the sine/cosine tracks of the Hiperface encoder are outputs in the form of pulses -> emulated pulse output of the voltage converter = pulse generator, A/B track, 1024 I/rev.

The following parameters are selected:

- Encoder type: Incremental encoder
- Resolution:

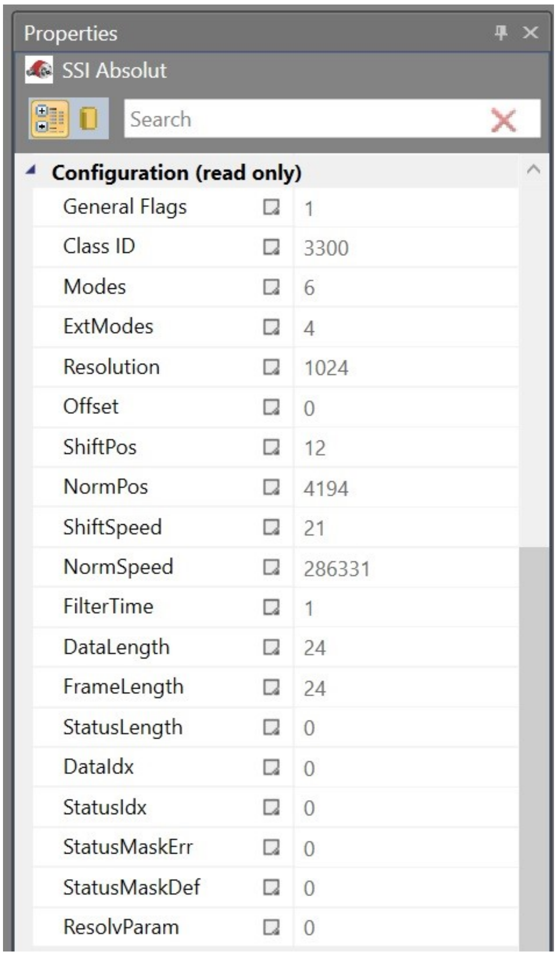
Encoder 1:	
Resolution Gb 2	1024 [Steps/rev]
in gearbox	4,51
in intermediate transmission	1
∅ drive wheel	31,83

$$Gb2 = \frac{1000}{D\_AR \cdot \pi} \cdot I\_G \cdot I\_AV \cdot A\_Gb2 = \frac{1000}{31{,}83 \cdot \pi} \cdot 4{,}51 \cdot 1 \cdot 1024 = 46182$$

Determination of the maximum speed:
The maximum speed of the output axis results from the maximum motor speed. In rev/s with respect to the load axis and with Nmax = 1500 rpm
the result is (1500 [rpm] / 60 [s]) * 0.012 [m] = 0.3 [m/s] = 300 [mm/s].

Input of the maximum deviation:
The empirical measurement results in a maximum difference of < 1 mm between both sensor points on the motor axis and the motion axis. The selected value is 1 mm.

The received information about the encoder configuration:



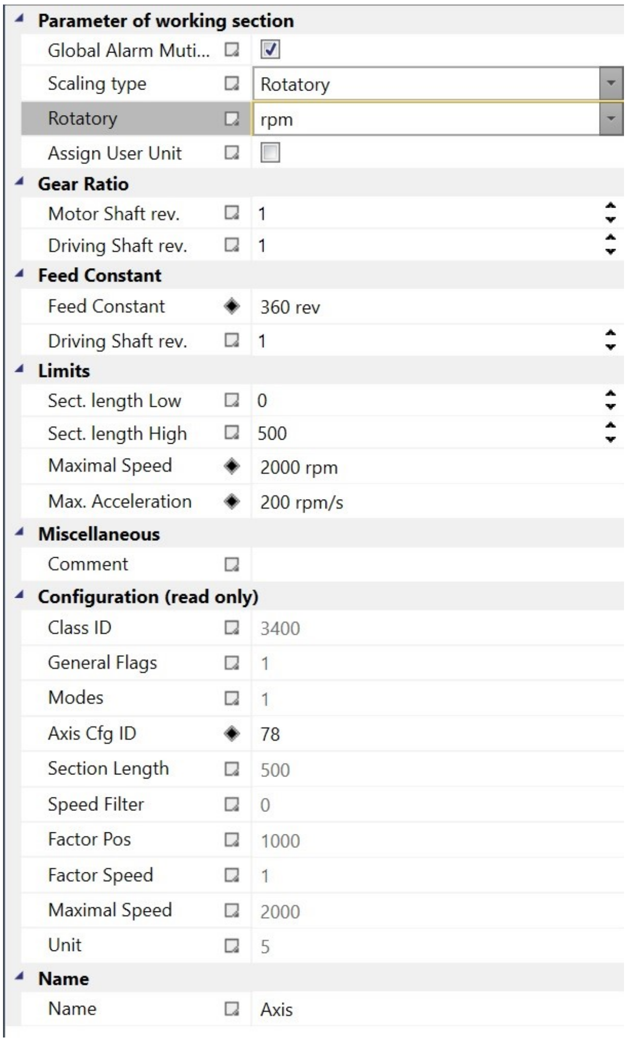
4.12.4.2. Configuration Third-party devices

The configuration is carried out in the slave module.

Observe the following chapter when using the axis values.

4.12.4.2.1. Settings od axles

To use the encoder for monitoring functions, the axis must be configured. The settings depends on the encoder type used:



Parameter	Description	Value
Global Alarm Muting	Suppression of alarm messages when exceeding/falling below the permitted limit values position/speed	
Scaling type	Linear or rotatory setting	Linear Rotatory
Rotatory	Unit	Linear-> mm/s Linear-> m/s Rotatory -> degrees/s Rotatory-> rev/s
Direction of Rotation	Direction of rotation	Clockwise Counter Clockwise
Motor Shaft rev.	Transmission ratio gearbox numerator	
Driving Shaft rev.	Transmission ratio gearbox denominator	

Feed constant	Ratio distance / encoder revolution - numerator	
Driving Shaft rev.	Ration distance / encoder revolution denominator	
Negative Position Limit	Minimum measuring length	
Positive Positions Limit	Maximum measuring length	
Maximum speed	Max. speed range	
Max. acceleration	Max. acceleration application	

4.12.5. Function blocks

4.12.5.1. Logic functions

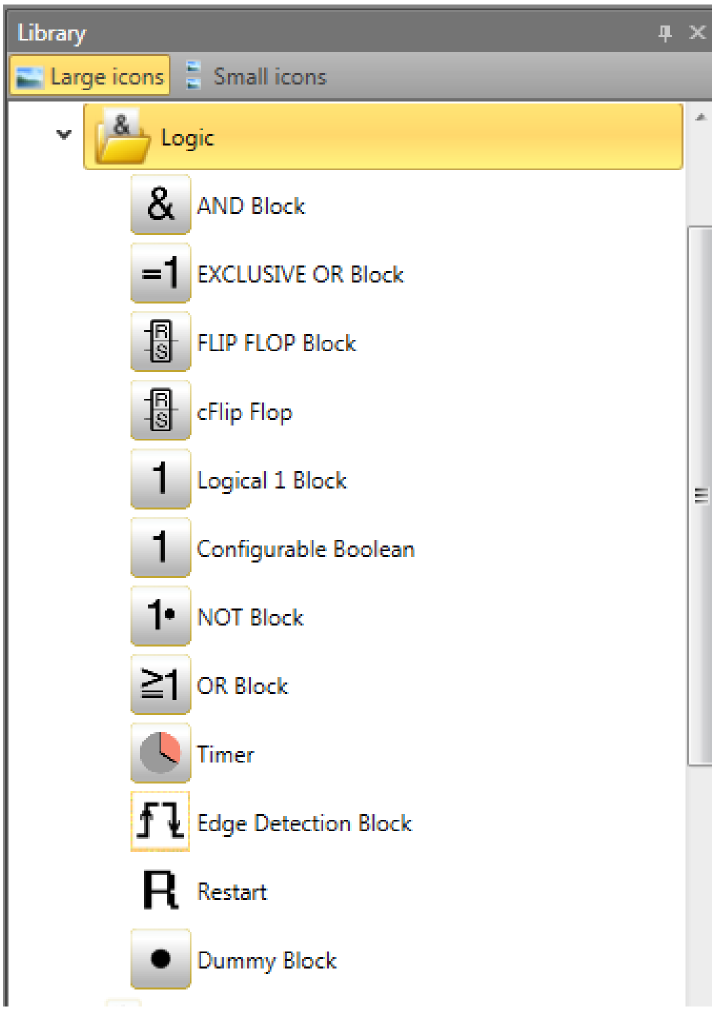


Figure 187 Listing of the logic functions in the library

These blocks are the basis for the creation of a program for the safety application. They allow the logic connection of the inputs with monitoring functions with the outputs. The insert of function blocks is only possible in the "functional scheme" view. Otherwise, the corresponding menu options are deactivated. The menu options are deactivated if the resources for a module are already exhausted, e. g., after all timer modules have been inserted.

AND block

"AND" links of not more than 5 output signals of other function blocks. The AND-link provides the signal state "1"for all input signals "1"as logic result. Otherwise, it is "0".

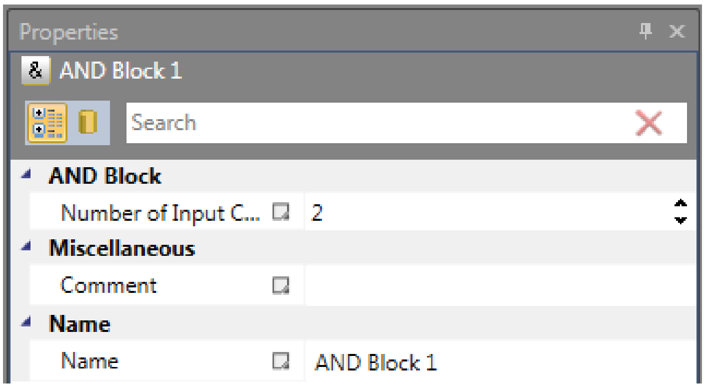


Figure 188 properties window "AND" block

NOTICE: The number of input connections can only be reduced in case of free connections. If all connectors are connected, these connections must be deleted before.

EXCLUSIVE OR block

"EXCLUSIVE OR" connections of 2 output signals from other function blocks. The "EXCLUSIV OR" module provides the logical result "1"if the input has the input signal "1", and if the input has the input signal "0". Otherwise, it is "0".

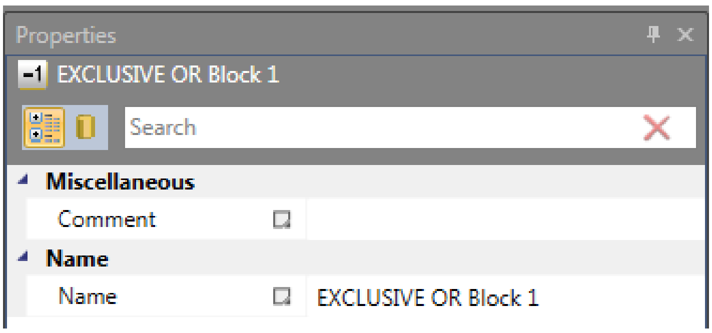


Figure 189 properties window "EXCLUSIVE OR" block

FLIP FLOP-Block

Setting / reset of a contact element. The switch element has the following properties:

- The logical result during the initialization of element is “0”.
- The logical result changes to “1” if an edge change from “0” to “1” occurs in the “Setting” input. The output stays “0”, even if the state in the “Setting” switches to “0” again.
- The logical result switiches to “0” if an edge change from “0” to “1” occurs in the “Setting” input.
- If both inputs are set to “1”, the result is “0”!

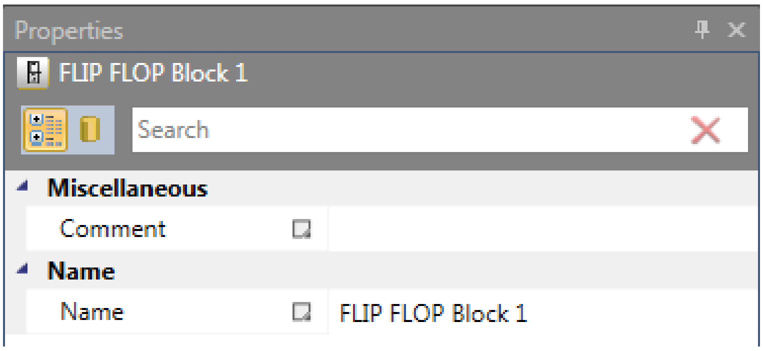


Figure 190 properties window "FLIP FLOP" block

NOTICE: The desired switching state in this element is chieved only according to the name (reset / setting).

cFLIP FLOP block/permanent FLIP FLOP block

Set / reset switching element.
The functionality is the same as for the standard FLIP FLOP block. If the check mark is set at "Save Permanent" then the state is saved voltage-proof. After a POR, the last saved state is active again.
Functioning like RS FlipFlop. If both inputs are set to "1", the result is "0"!

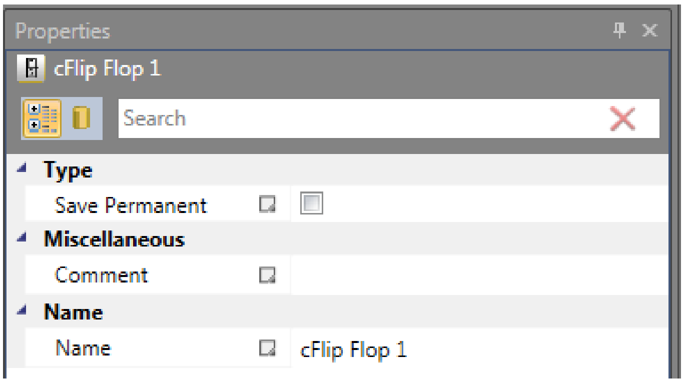


Figure 191 properties window "cFLIP FLOP" block

NOTICE: After configuration data have been loaded on the safety device, the result is reset.

Logical 1-block

This module always provides the value "1". This function can be used for programming static states in the functional scheme.

<u>Example</u>: assignation of an input that is not used to a direction-dependent SDI

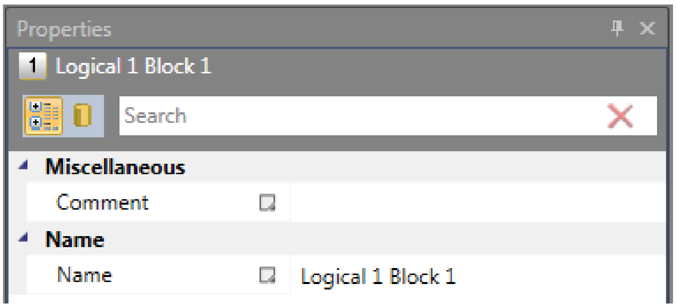


Figure 192 properties window of logical 1- blocks

Configurable Boolean

This module always provides the parameterized Boolean value “TRUE/FALSE”. This function can be used for programming static states in the functional scheme.

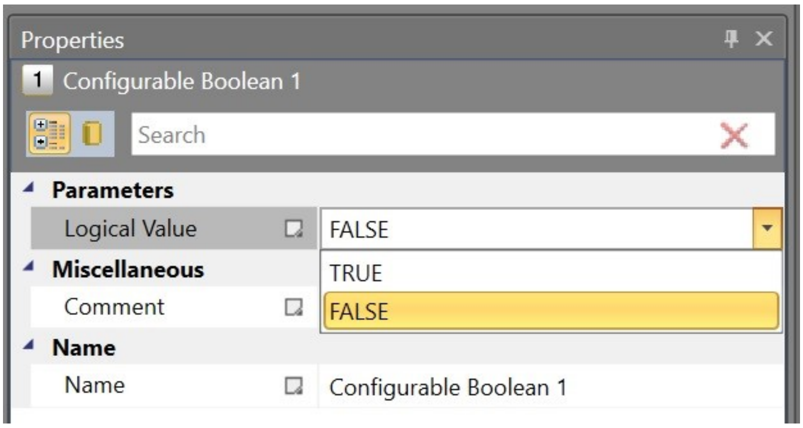


Figure 193 properties window of configurable boolean

NOT block

The logical result of this function block is the negation of the input signal. Here, the term “negation” signifies that the logical result is reversed (negated).

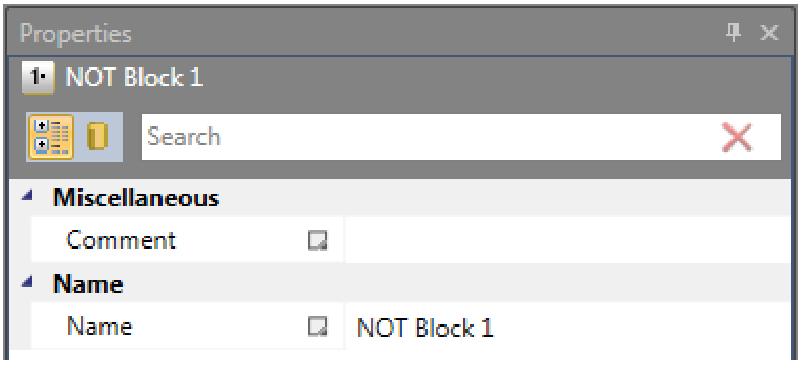


Figure 194 properties window of NOT function block

OR-Block

"OR" connections of not more than 5 output signals from other function blocks. The OR connection provides the signal state "1" for at least one input with the signal state "1". Otherwise, the signal state is "0".

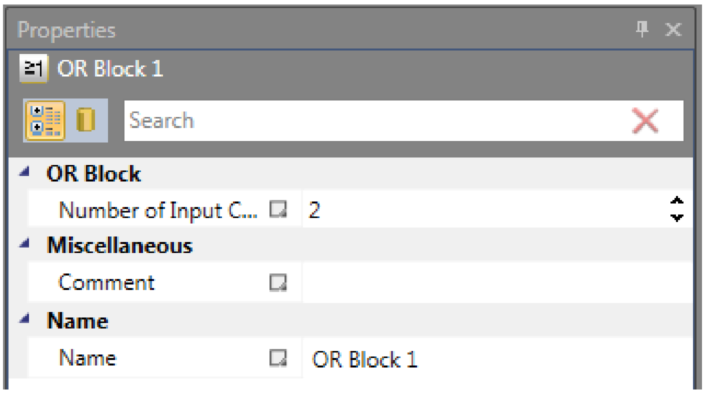


Figure 195 properties window of OR function block

Timer

Function block that starts a meter in case of an edge change. After the fixed time lag, the result is “1” or “0”.

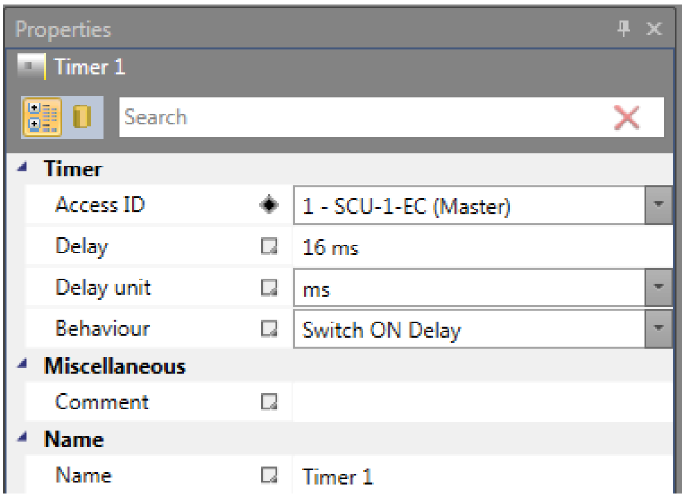


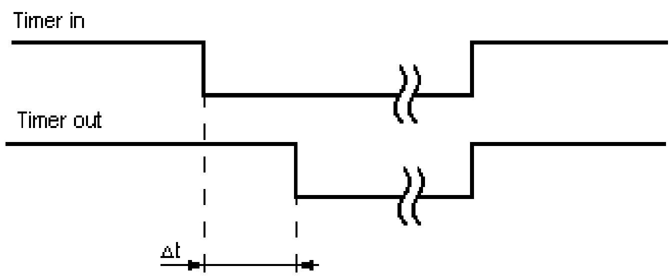
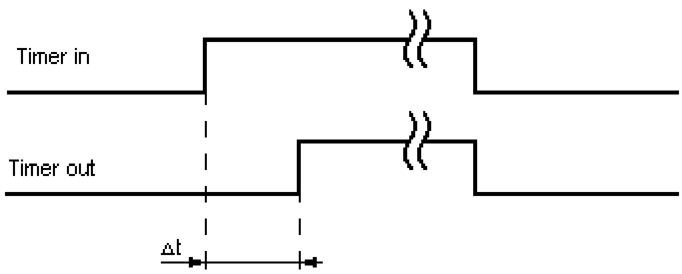
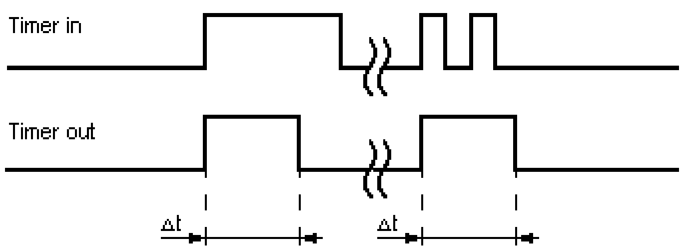
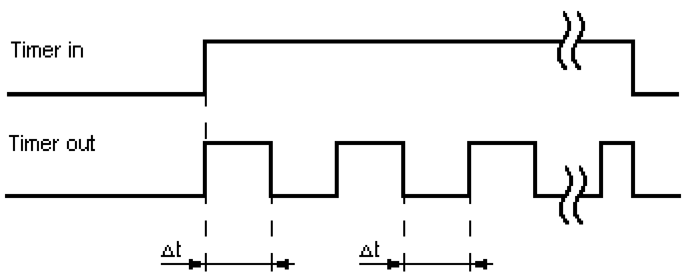
Figure 196 properties window „Timer“

Block-ID: Timer number. The timer number can be set when the timer is plugged in. If all timers are used, the timer command in the menu is disabled.
Delay: desired period, during which the timer shall run.

T min	=	16 ms
T max	=	2592.000.000ms
		43.200min
		720h

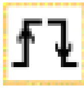
NOTICE:
The programmable values equals always an integr multiple of the cycle time and will be adjusted to the valid smaller value in case of wrong input! This change is also signaled by a warning window!

Behaviour

Function	Activation of timer	Time diagram
Fall-delayed	Falling edge	
Delayed closing	Rising edge	
Impulse	Rising edge	
Periodical	Rising edge	
Save permanent	According to function	Voltage-proof saving of timer value. After unit POR, the timer value saved last is loaded again. After loading of configuration data on the SCP unit, the saved timer value reset again (“0”). (available from FW version 2.0.0.1)

Note: Δt = PLC timer value

Edge detection

 Block for edge detection rising, falling or both.

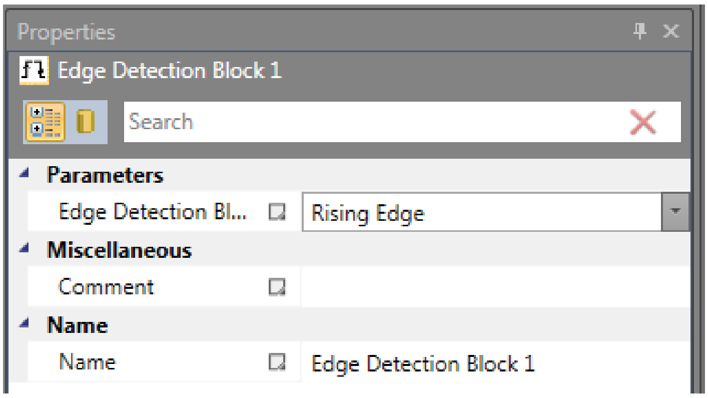


Figure 197 properties window of edge detection

Edge detection:

- Positive:
  Evaluation of rising edge. Signal at block output changes to "high", if rising edge is detected at block input.
  Output signal remains set for one cycle.

- Negative:
  Evaluation of falling edge. Signal at block output changes to "high", if falling edge is detected at block input.
  Output signal remains set for one cycle.

- Positive and Negative
  Evaluation of rising / falling edge. Signal at block output switches to "high" if rising / falling edge is detected at the block input.
  Output signal remains set for one cycle.

Restart-block

Block for confirmation signal for stopping and for the following request to restart the application.

The basic requests for the block are:

- Commissioning of the device (cold start after the Power-Cycle)
- Restart of the device (warm start after programme download, software reset etc.)

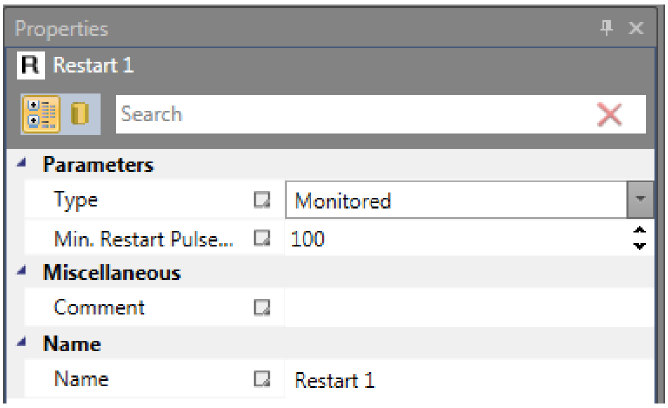
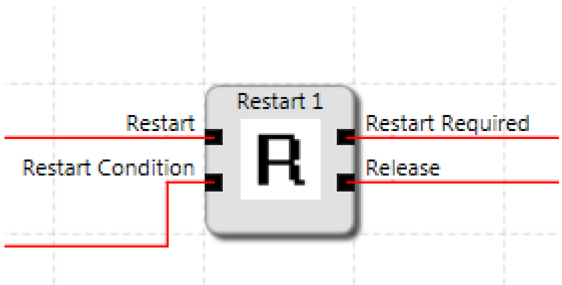


Figure 198 properties window of restart block

Type of reset start "Monitored":
"Release" condition reacts to the falling Edge Restart signal.
The falling edge must occur > "Min. Restart Pulse" and < 15s after activation.

Type of restart "Manually":
"Release" condition reacts to the rising Edge Restart signal .

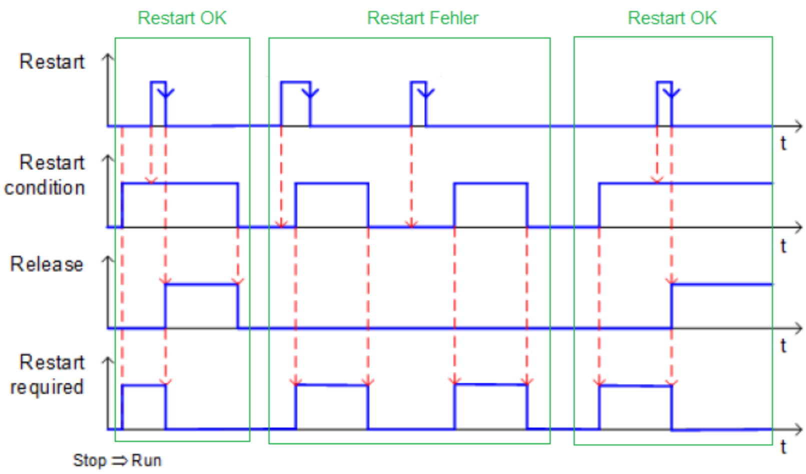


Restart:
Evaluation of Restart signal after selection of type.

Restart condition:
Restart condition. As condition, a "high" signal is expected.

Restart required:
The message "Restart required" that the function block expects a valid restart impulse at the "Restart" input that the "Release" output can switch to "high".
As a rule, this output is used to control a signal lamp.

<u>Release:</u>
The output switches to "high", if the following conditions are fulfilled:



Dummy-block

This block has no effect on the functionality of the device and the running program and should normally only be used temporarily for troubleshooting.

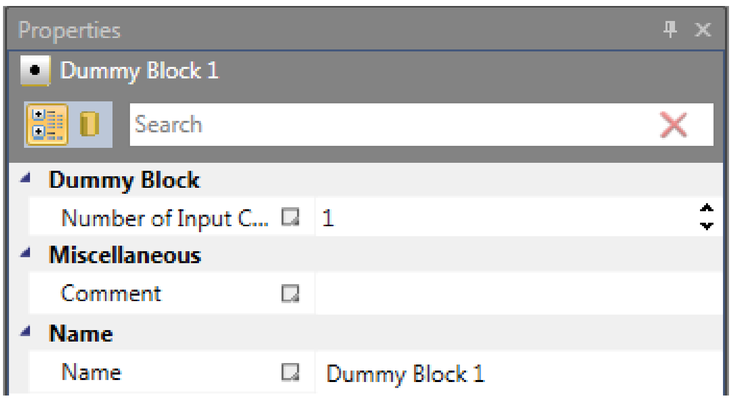


Figure 199 properties window of dummy block

4.12.5.2. Safe Arithmetic („SARC")

The **S**afe **Ar**ithmetic **C**alculation[1)] function library offers the possibility of a "floating point" calculation of analog values. For this purpose, the library consists of input elements for the transfer of position, speed and analog values, the calculation/processing elements, constant/fixed parameter elements and the output elements for the return of the calculated values in the integer range. Complex tasks with dependencies of several axes or angular positions etc. can thus be realized and safely monitored. All safety functions such as SOS, SLS, SLP, SCA and also SWM (Safe Workspace Monitoring) are available for monitoring.

With the use of the SARC functions, for example, a safe kinematic calculation for multi-axis applications such as safe tool-center-point position and speed calculation including the upstream joints can be executed in the SCU safety device.
For this purpose, input values (safe position / speed, configurable constants, etc.) are converted into a "single precision float" value and normalized in order to be able to use them further for the SARC. By means of the library of mathematical functions, in the form of building blocks that can be individually linked together, the specific calculation rule is implemented. These links ultimately produce an instruction list of the kinematic calculation, which is loaded onto the safety controller and executed. The result of the SARC calculation(s) can be converted into a virtual position or speed and linked in turn with the existing safety functions. In this way, a wide variety of application requirements can be realized. In particular for Safe Robotics applications, matrix blocks are available as complex mathematical functions. DH matrix calculations can thus be implemented easily and directly.

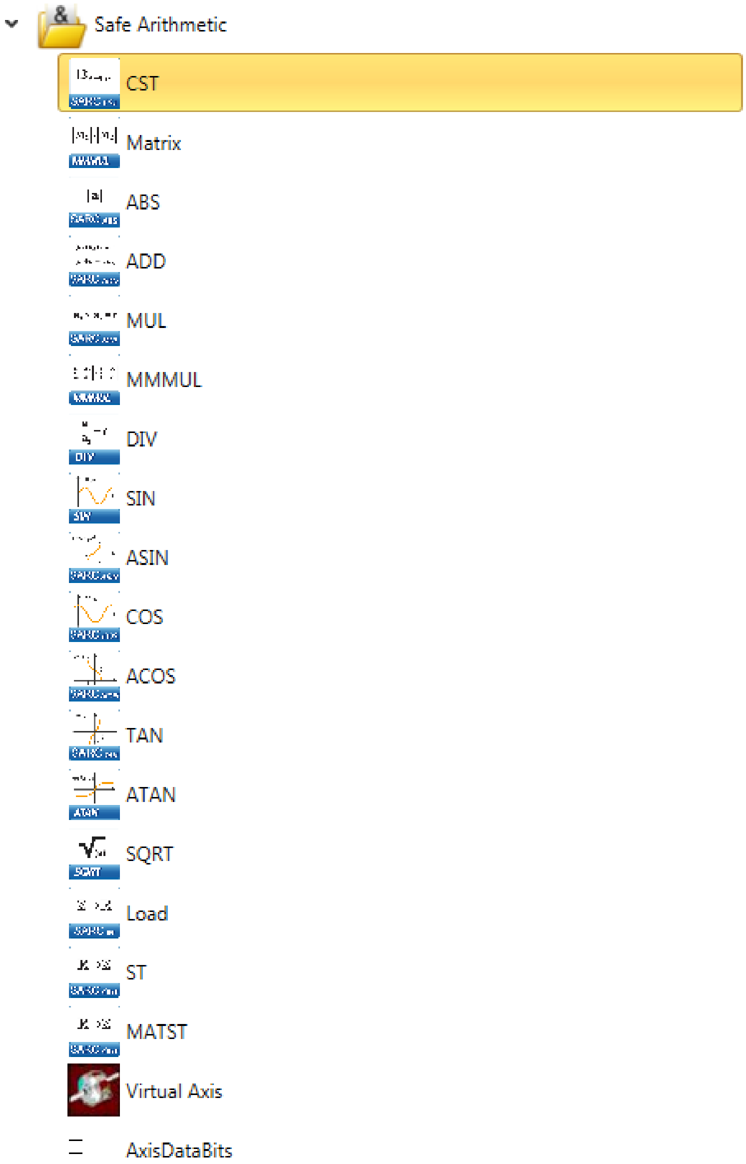
A maximum of 250 SARC instructions can be used.

The linking of the mathematical functions or conversion of the calculation rule is freely selectable, the correct calculation sequence is guaranteed by the conversion in the SCU device.

In the SARC environment, diagnostics are available in the same way as for logic and analog values. Both intermediate and result values can be displayed in the online diagnostics and thus the correct implementation of the specific calculation rule can be checked.

1) The use of the function library in the SCU requires the entry of a chargeable release key.

4.12.5.2.1. Overview of SARC functions



4.12.5.2.2. Description of SARC functions

Function for transfer and constants

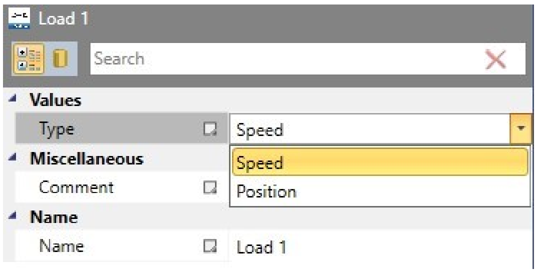
Load

Loading a speed or position of an axis

Input: Encoder axis, Selection position or speed

Output: Speed or position value in float format for SARC calculation

Parameter: Type = Position or speed



All available encoders can be used, e.g.. EnDat

- Positions (m, mm, rev, deg)
- Speed (m/s, mm/s, rev/s, deg/s)
- Conversion to float values via "Load" block

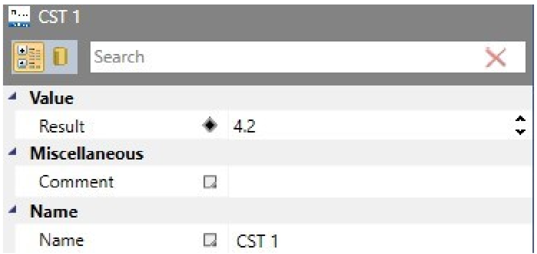
Constant (CST)

Definition of a constant

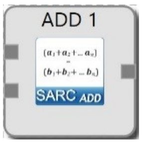
Input: -

Output: Constant in float format for SARC

Parameter: Value of the constant

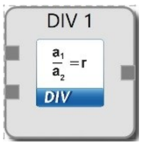


Basic arithmetic operations



Addition
Signed addition of two operands
Input: Operand 1 and 2
Output: Sum of the two operands in float format for SARC
Parameter: -

Multiplication
Signed multiplication of two operands
Input: Operand 1 and 2
Output: Product of the two operands in float format for SARC
Parameter: -



Division
Signed division of two operands
Input: Operand 1 and 2
Output: Quotient of the two operands in float format for SARC
Parameter: -

Note: Division by 0 leads to alarm

Trigonometric functions

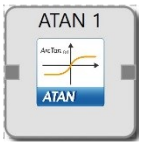
Cosine
Calculation of the cosine value of the input operand
Input: Operand in radians
Output: Cosine value of the operand in float format for SARC
Parameter: -

Arcus cosine
Calculation of the arc cosine value of the input operand
Input: Operand
Output: Arc cosine value of the operand in radians for SARC
Parameter: -

Sine
Calculation of the sine value of the input operand
Input: Operand in radians
Output: Sine value of the operand in float format for SARC
Parameter: -

Arcus sine
Calculation of the arc sine value of the input operand
Input: Operand
Output: Arc sine value of the operand in radians for SARC
Parameter: -

Tangent
Calculation of the tangent value of the input operand
Input: Operand in radians
Output: Tangent value of the operand in float format for SARC
Parameter: -



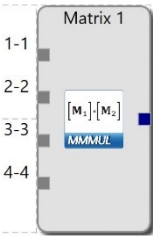
Arcus tangent
Calculation of the arc tangent value of the input operand
Input: Operand
Output: Arc tangent value of the operand in radians for SARC
Parameter: -

Advanced calculation functions

Absolute value
Determines the absolute value of the operand
Input: Operand
Output: Absolute value of the operand in float format for SARC
Parameter: -

Square root
Calculation of the square root of the absolute value of the operand
Input: Operand
Output: Value of the square root of the operand in float format for SARC
Parameter: -

Matrix functions



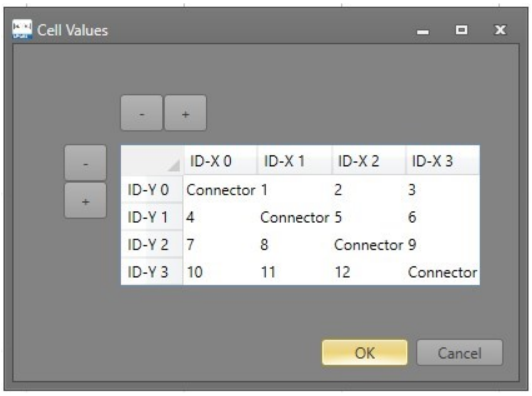
Matrix

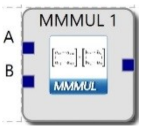
Definition of an X/Y matrix with max. 6 rows/columns to calculate the matrix value

Input: Max. 36 Operands, depending on the dimension of the matrix and assignment of the contents as operand or constant value

Output: SARC Matrix

Parameter: Dimension (Rows and columns, max. 6x6), Matrix entries (constant or input=input connector)





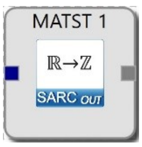
Matrix matrix multiplication

Multiplication of 2 matrices

Input: SARC Matrix output value

Output: Product of 2 matrices

Parameter: -



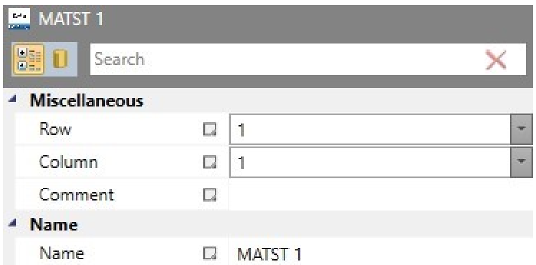
Matrix store

Output of the result value of an entry of a matrix

Input: SARC Matrix

Output: Result value of an entry of a matrix

Parameter: Position of the entry in the matrix (row and column)



Output functions

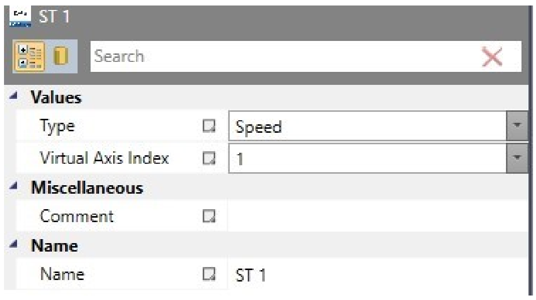
Store

Saving a SARC variable in a virtual axis

Input: SARC value

Output: -

Parameter: Type of the value to be stored (speed or position), index of the virtual axis.



Virtual axis

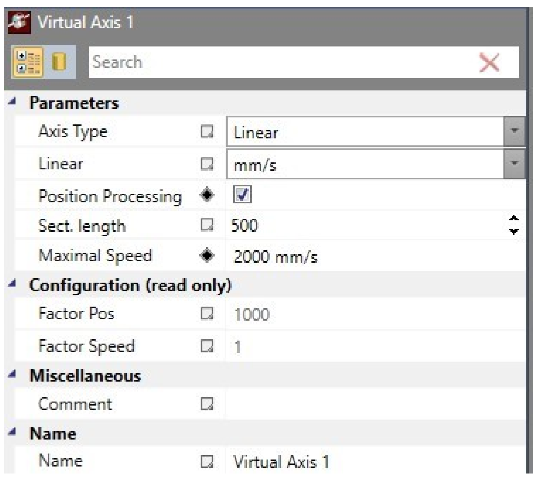
Virtual axis with output values position / speed for further processing in integer range, input value for standard monitoring functions such as SOS, SLS, SLP, SCA, SWM

Input: -

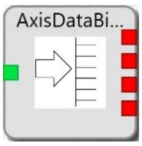
Output: Speed and/or position

Parameter: Analog to the transfer of encoder values:

- Axis Type: linear or rotatory
- Linear/rotatory: unit of virtual axis
- Position processing: position active/inactive
- Sector length: Maximum position value
- Maximal speed: Maximum speed



- Store the calculated results in "virtual axes".
  - Type: Speed or position Emulate standard axes with similar properties
  - Virtual axis index: 1…16
  - The values are stored directly after "Type". No further conversion/calculation takes place (calculation of the speed must take place in the SARC).
- Virtual axes can be used as standard axes,
  - Connection to all available monitoring functions for speed/position
  - Configuration of virtual axis, see “Virtual Axis”



AxisDataBits

Output of the value of an analog axis in bit format. By means of downstream logic processing, range functions are thus possible.

Input: Virtual axis (speed or position)

Output: Value in bit format

Parameter:

- Type: Position or speed
- Number of bits (1 .. 32)
- Sign bit: Output of the sign (for speed)

NOTICE:

Precision of the calculations

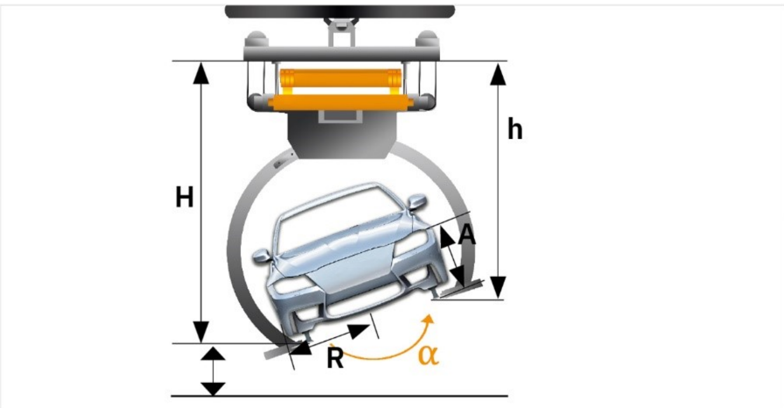
The data types of the "Single precision data type for IEEE 754 arithmetic" are used for the calculation. The following tolerances are allowed

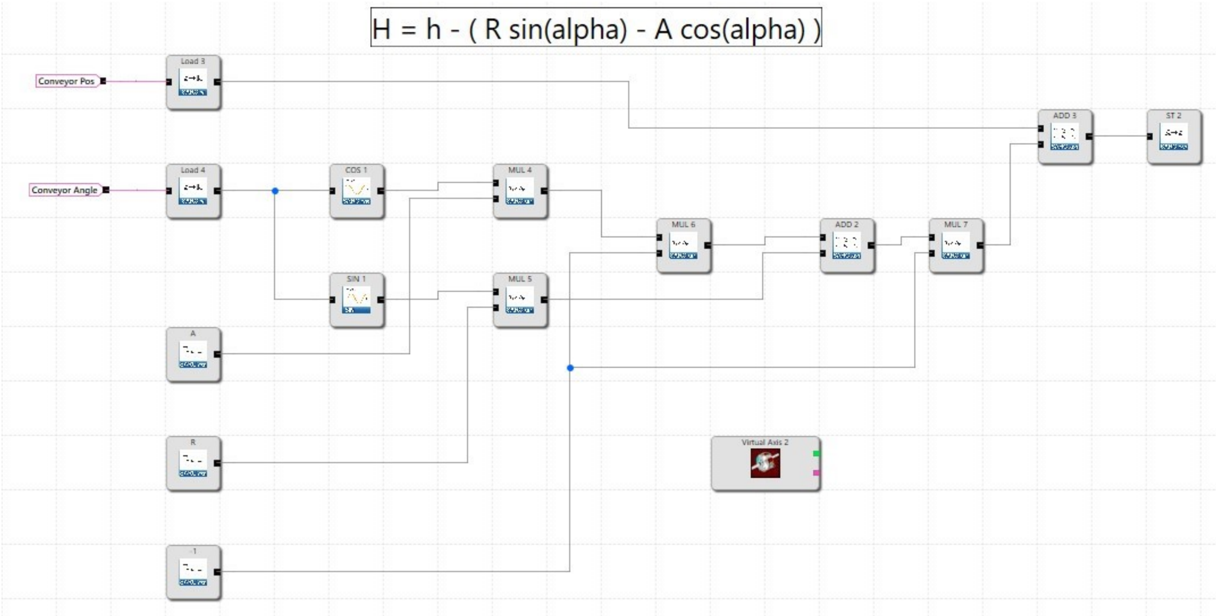
SARC function	Precision
Addition	0x00000001 (Difference within the mantissa)
Division	0x00000001 (Difference within the mantissa)
Square root	0x00000002 (Difference within the mantissa)
Cosine	0.0002 (Absolute difference from 0.0002)
Sine	0.0002 (Absolute difference from 0.0002)
Arcus cosine	0.002 (Absolute difference from 0.002°)
	0.04 (Absolute difference from 0.04° in the range |0.99| to |1.0|)
Arcus sine	0.002 (Absolute difference from 0.002°
	0.04 (Absolute difference from 0.04° in the range |0.99| to |1.0|)
Tangent	0.004 (Absolute difference from 0.004)
Arcus tangent	0.0002 (Absolute difference from 0.0002°)
Matrix multiplication*	Default: 0x00000005 difference within the mantissa, checked for each matrix entry).

- The user must ensure that the calculation is performed correctly and with sufficient accuracy according to his requirements.

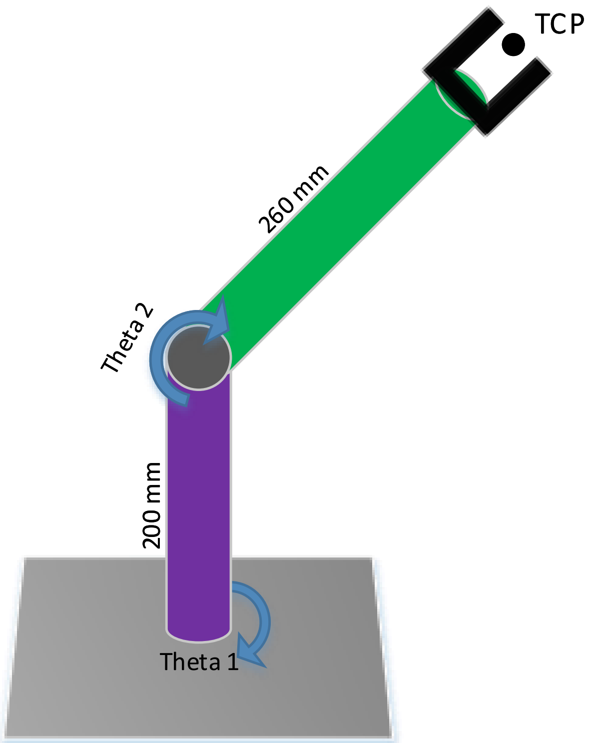
4.12.5.2.3. Examples of SARC functions

Example 1 : Safe speed/position monitoring of a 3-axis EMS





Example 2 : Matrix calculation for two-axis robots

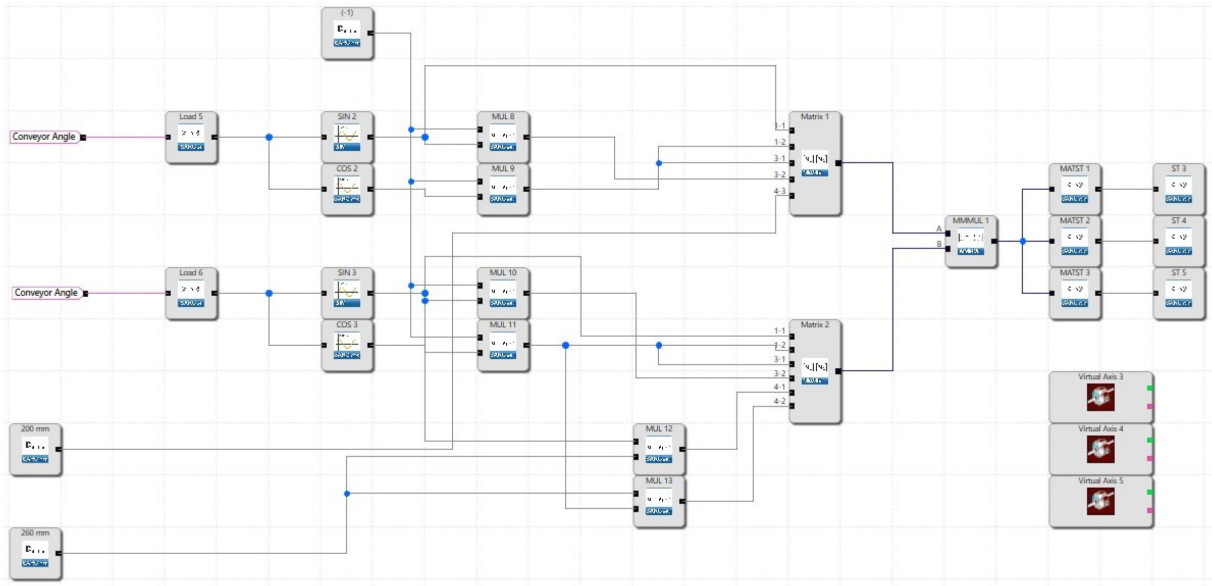


	θ [deg]	a [mm]	d [mm]	α [deg]
A_1	$\theta - \pi/2$	0	200	$\pi/2$
A_2	$\theta - \pi/2$	260	0	$\pi/2$

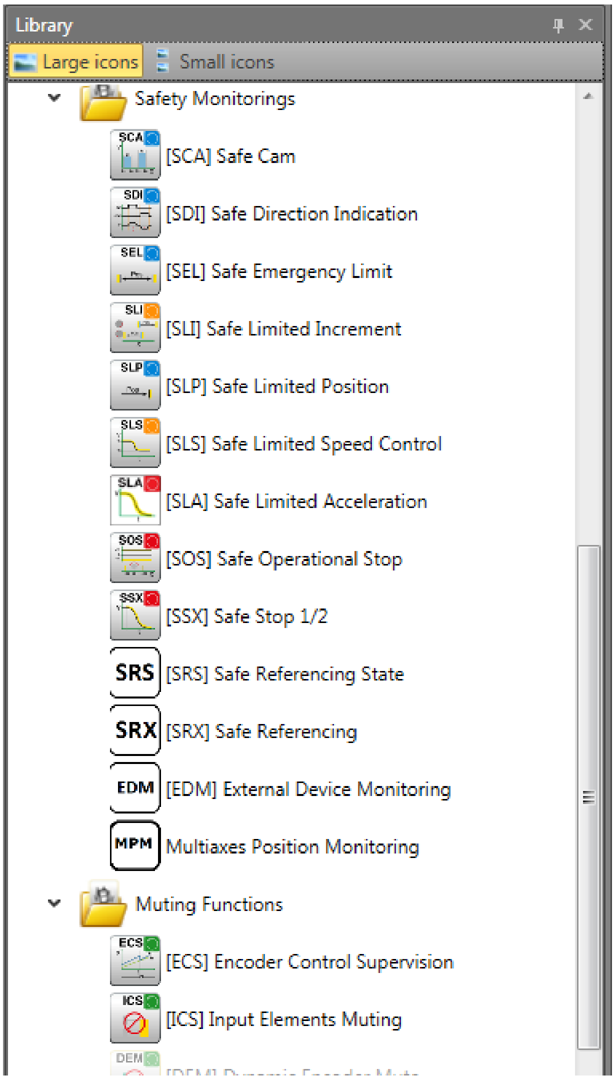
$$A_1 = \begin{pmatrix} \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ -\cos\theta_1 & 0 & -\sin\theta_1 & 0 \\ 0 & 1 & 0 & 200 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} \sin\theta_2 & 0 & -\cos\theta_2 & 260\cos\theta_2 \\ -\cos\theta_2 & 0 & -\sin\theta_2 & -260\cos\theta_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_0^{TCP} = A_1 \cdot A_2$$
$$\rightarrow x = 260 \sin\theta_1 \sin\theta_2$$
$$\rightarrow y = -260 \sin\theta_2 \cos\theta_1$$
$$\rightarrow z = 200 - 260 \cos\theta_2$$



4.12.5.3. Safety functions



The safety functions are an important function of the SCU system. The safety functions are preset functions and available for:

- Speed monitoring
- Position detection
- Monitoring of limits and target positions
- Functional emergency monitoring
- Standstill monitoring
- Direction monitoring
- Function monitoring of external shut-off devices
- Reset functions
- Muting

The function to monitor position, speed and cut-off is enabled only after the successful configuration of an encoder. After the encoder has been successfully configured, the respective functions can be inserted, if the SCU-module has resources available for this purpose. After all resources have been used, the menu option for the corresponding function block is deactivated.

4.12.5.3.1. Overview of safety modules

SCU +SIO +SSB devices

Module	Amount	Use	Single axle	cascading	Reset	SCU - devices	SIO-devices	SSB/ GBox
FDB	8	RUN, RUN internal	No	No	No	X		
EDM	16	RUN, RUN internal	No	No	Yes	X		X
SBR	2	Procession on GBox	--	--	--	--		X
SRS	12	RUN, RUN internal	Yes	No	No	X		X
SRX	48	RUN	Yes	No	No	X		X
SEL	16	RUN	Yes	Yes	Yes	X		
SLP	12	RUN	Yes	No	Yes	X		
SCA	48	RUN	Yes	Yes	No	X		
SSX	18	RUN	Yes	No	Yes	X		
SLI	12	RUN	Yes	No	Yes	X		
SDI	12	RUN	Yes	No	Yes	X		
SLS	48	RUN	Yes	Yes	Yes	X		
SOS	16	RUN	Yes	Yes	Yes	X		
ECS	12	RUN	Yes	No	Yes	X		X
MPM	2	RUN	Yes	No	Yes	X		
SWM	16	RUN	Only virtual axis	Yes	No	X		

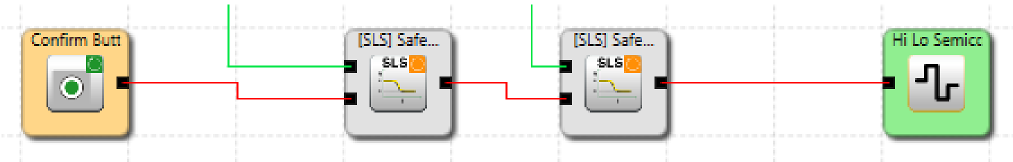
SDU devices

Module	Amount	Use	Single axle	cascading	Reset
EMU	2	RUN, RUN internal	No	No	Yes
SEL	2	RUN	Yes	No	Yes
SLP	2	RUN	Yes	No	Yes
SCA	16	RUN	Yes	No	No
SSX	4	RUN 2 per axis	Yes	No	Yes
SLI	2	RUN	Yes	No	Yes
SDI	2	RUN	Yes	No	Yes
SLS	8	RUN	Yes	No	Yes
SAC	8	RUN	Analog input	No	No
SOS	2	RUN	Yes	No	Yes
ECS	2	RUN	Yes	No	Yes
EOS	4	RUN	Yes	No	Yes

If the cut-off shall be displayed externally by a monitoring function, e. g. a control unit, an auxiliary output can be used for this purpose. If "1" has been signaled to the outputs, while the monitoring functions are in the "OK" status, the result must be negated for feedback according to the following example.

Cascading

In case of safety modules with cascading, subsequently, another safety modul can be triggered, without losing a system cycle.



With safety modules without cascading, 1 cycle must be added for response time.

4.12.5.3.2. List of safety functions

EDM (External Device Monitoring)

monitoring of external switch contacts

Number: c. f. "Overview of safety modules"

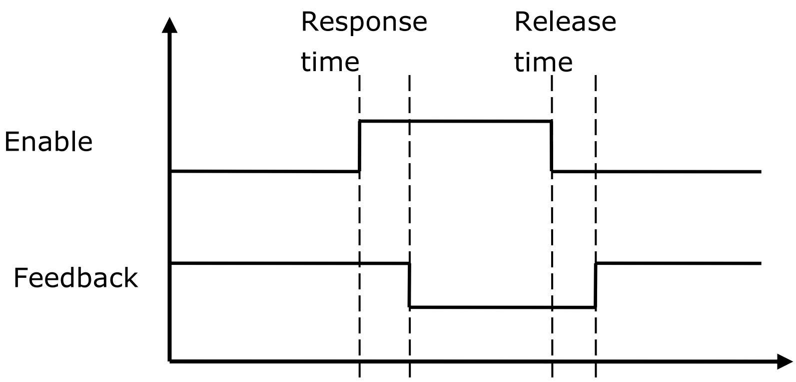
Function: Normally, additional switch devices are necessary for contact multiplication and for power multiplication. These switching devices are triggered via the outputs of the safety control (SCU, SIO). The EDM monitoring device implements the "Safety relay" function by processing an external feedback circle. For applications according to category 4 of EN 954-1, for these switch devices, among others, functional monitoring is required. For functional monitoring, the switching devices must be equipped with positively driven auxiliary contacts.

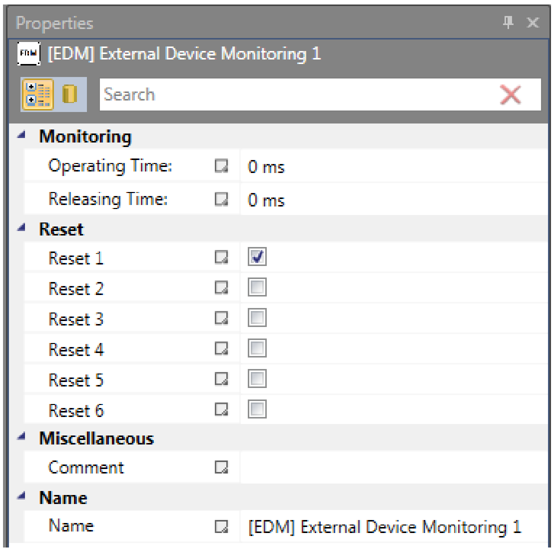
Input: Activation signal external switch contact
Reset of external switch contact

RESET function: Up to 6 resets are configurable.

Description of the function:

- Test of external Feedback signal for state of external switch contact.
- State of Feedback signal must change within response time / release time.
- A faulty error signal or a status change larger than the response time / the release time is signaled at the "Error" output as "high".





Response time

Variables time window (turn-on delay) for the test of the external switch contact

Min{T_{EDM}} = 16 msec

Max{T_{EDM}} = 3000 msec

Release time

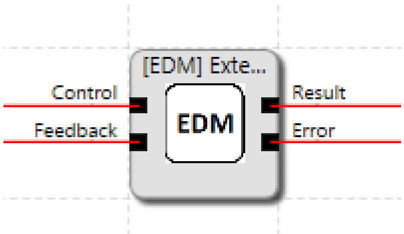
Variable time window (turn-off delay) for test of the switch contact

Min{T_{EDM}} = 16 msec

Max{T_{EDM}} = 3000 msec

Reset

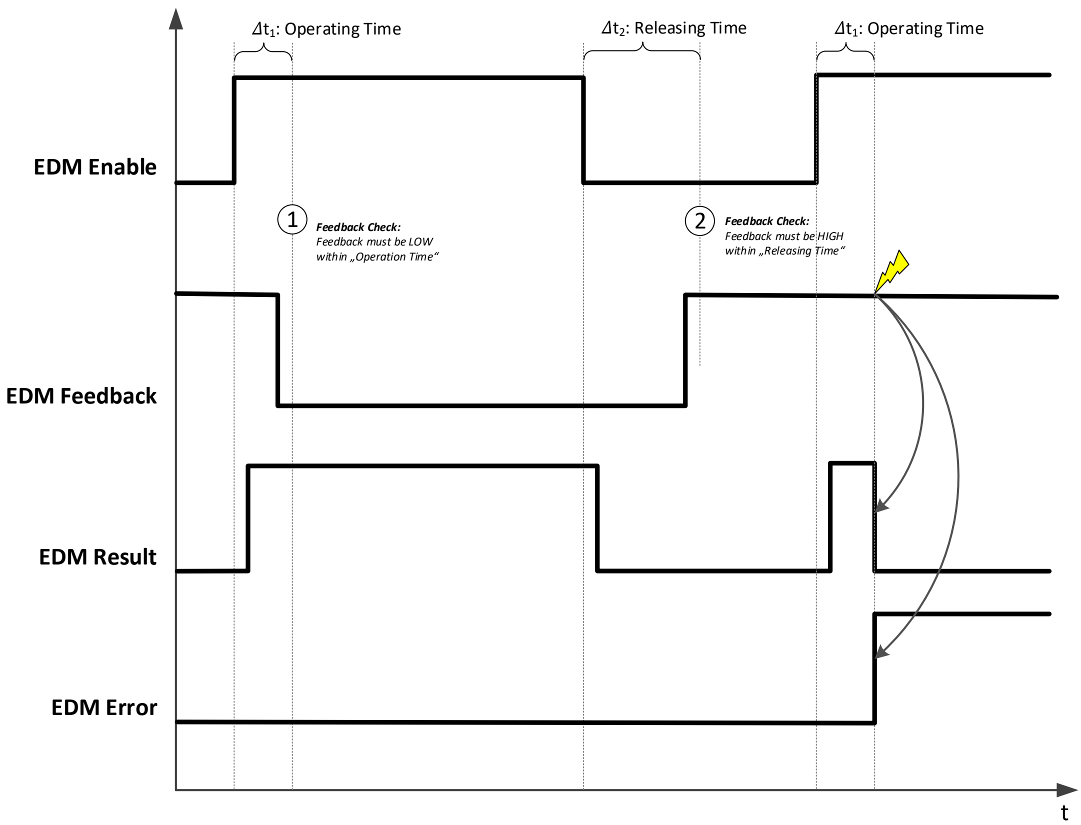
Selection of Reset



Signal PLC block	Description
Enable	Activationsignal for EDM monitoring
Feedback	Read-back signal for external switch contact monitoring
Result	Activation signal at output 0: EDM output is inactive 1: EDM output is active
Error	Error diagnosis EDM 0: EDM OK 1: EDM has diagnosed an error

Logic table

Control	Feedback	Result	Error
0	0	0	1
0	1	0	0
1	0	1	0
1	1	0	1



***NOTICE*:**

- The EDM error is stored and can only be reset via Reset.
- Function can only be reset if feedback signal is "1". (Inactive state)
- As soon as an error is present, Result is always inactive, regardless of the status of the feedback signal.
- Adjacent feedback circuits of the safety monitoring module should have different signatures (Pulse1/Pulse2) in order to be able to detect a cross-circuit.

SRX (Safe Referencing on X-Axis)

SRX Safe referencing on a X-axis

Number: c. f. "Overview of safety modules"

Function: Mit der SRX-Funktion kann die Position einer Achse auf einen definierten Wert gesetzt und damit auf eine physikalische Position referenziert werden.

RESET-Function: The axis referencing can be reset with the RESET function within the SRS function.

Description of function:

If you want to realize a position monitoring with the safety device, it is necessary in many applications to adapt the internal position of the safety device with an offset to the physical position in the plant. This is the case, for example, during initial commissioning or when replacing an encoder. With the SRX function, you can adapt the internal position of the safety device to the physical position of the plant without subsequently changing the safety program.

For this purpose, the reference position must be approached by moving to the physical mark and then safe referencing must be performed. When the SRX is activated for safe referencing, the safety device calculates the offset from the current encoder value and adjusts its internal position to the physical position of the system. In this way, the safety device can be referenced after disassembly or after an encoder exchange without having to adapt the safety program. Further validation is therefore not required.

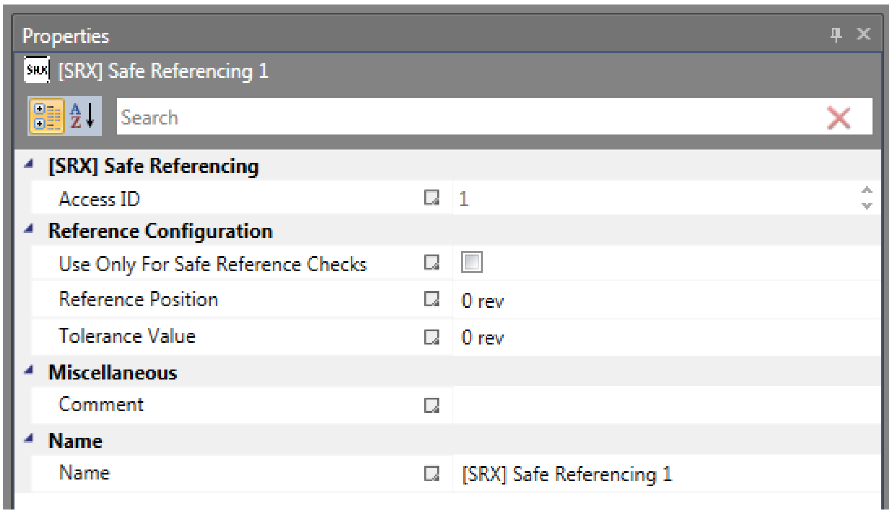
fter successful referencing, the configured measuring length (measuring distance) can be traversed independently of the encoder system used and its current position. The current position of the system is permanently stored in the non-volatile memory and used together with the calculated offset to check the referencing status. Therefore, absolute multiturn encoders do not require re-referencing after power down or power failure of the device, but a position check must be performed. For this purpose, the reference position must be approached again and a Safe Turn Counter Restore must be evaluated.

If a sine/cosine or resolver encoder is used, the SRX function can also be used for position processing. In this case, however, referencing must be restored by the Turn Counter Restore after switching on and at regular intervals of 12-48 hours.

⚠ WARNING

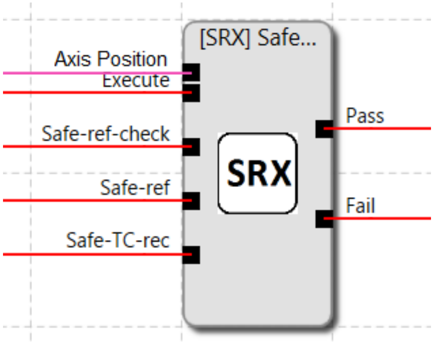
- Before performing safe referencing, make sure that the axis to be adjusted is in the correct position (reference mark).
- For successful adjustment and to keep the position tolerances low when performing referencing, we recommend that you only perform referencing when the associated axis is at a standstill
- You can only perform safe referencing when the safety control is in the "RUN" state.
- When using the SRX function, no further offset must be configured for the respective encoder system.
- After referencing, the activating processing for SRX must be blocked by the logic or by selecting the corresponding input elements.

Parameter:



Parameter	Description	Unit/Range
Access ID	The access identifier is used to identify the function element in the application program.	-
Use Only for Safe Reference Checks	If this option is checked, the SRX function can only be used to check the safe referencing status.	Checked off Not checked
Reference Position	The physical position (e.g. physical mark) to which the position of the axis to be referenced is to be set.	[Position]
Tolerance Value	Tolerance value for the referencing check.	[Position]

Comment	Input option for an application-related comment
Name	Input option for an application-related name



Axis Position	Input signal for assigning the axis to be referenced.
Execute	Edge triggered signal for execution the selected function.
Safe-ref-check (safe reference check)	Function selector for checking the referencing.
Safe-ref (safe referencing)	Function selector for executing the referencing.
Safe-TC-rec [Safe TC restore]	Function selector for restoring the referencing.
Pass [Passed]	Output signal to indicate successful execution of the selected functionality.
Fail	Output signal to indicate an incorrect execution of the selected function.

Logic table

	Signal					
Function	Execute	Safe-ref-check	Safe-ref	Safe-TC-rec	Pass	Fail
Inactive	0	-	-	-	0	0
Check the referencing	1	1	0	0	X	!X
Set the referencing	1	0	1	0	X	!X
Restore the referencing	1	0	0	1	X	!X
Invalid	1	1	1	-	0	1
Invalid	1	1	-	1	0	1
Invalid	1	-	1	1	0	1

INFORMATION

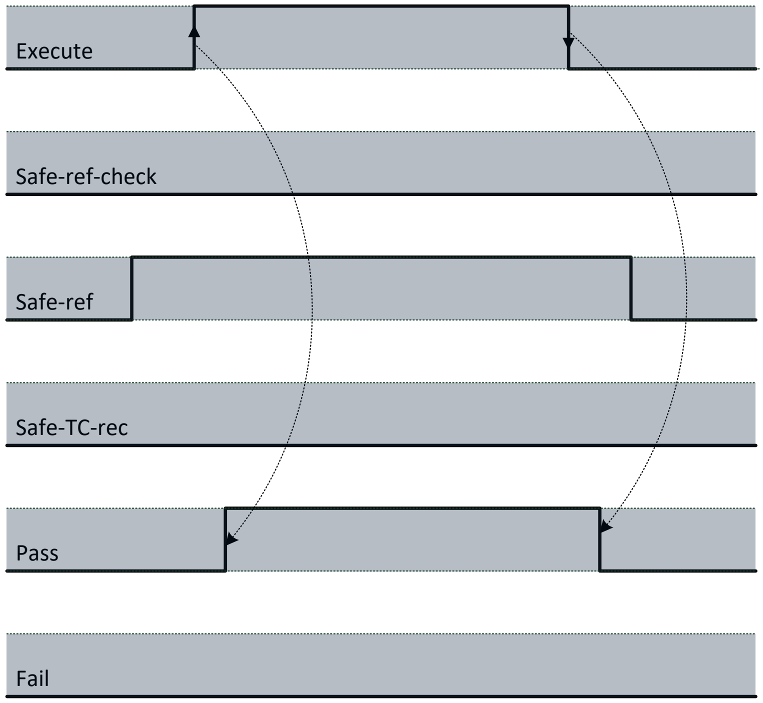
The Safe-ref-check, Safe-ref and Safe-TC-rec inputs are basically used as function selectors - a high level marks the functionality to be executed.

With a rising edge of the Execute input, the selected functionality is executed.

Only one function selector signal may be active at the time of execution.

The output state is set after activation of the functionality and remains in the corresponding state until the Execute signal changes to a low level again.

Execute a successful referencing:



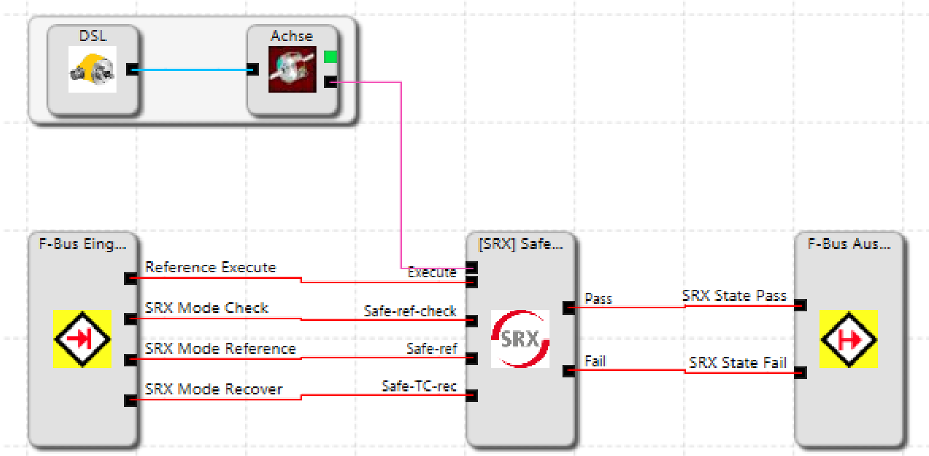
Input Example 1

A safe multiturn absolute encoder provides position information for a rotating shaft. After mounting the encoder, the system is to be referenced to a physical mark that is defined by the user as the reference position. For this purpose, the system can be moved to the physical mark at reduced speed and safe referencing can be performed.

Input Example 2

At a production station, a carriage uses a rotary absolute encoder in combination with an incremental encoder as a system for position processing.

Since the absolute encoder must be removed for maintenance, it must be possible for the internal position of the safety device to be referenced to the physical mark after maintenance without having to change the safety program. For this purpose, the reference position must be approached by moving the system to the physical mark and then a safe referencing must be carried out.



SRS (Safe Referencing State)

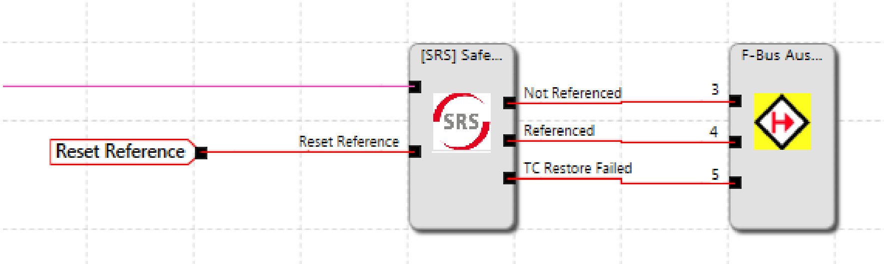
Safe Referencing state

Number: c. f. "Overview of safety modules"

Function:

Input:

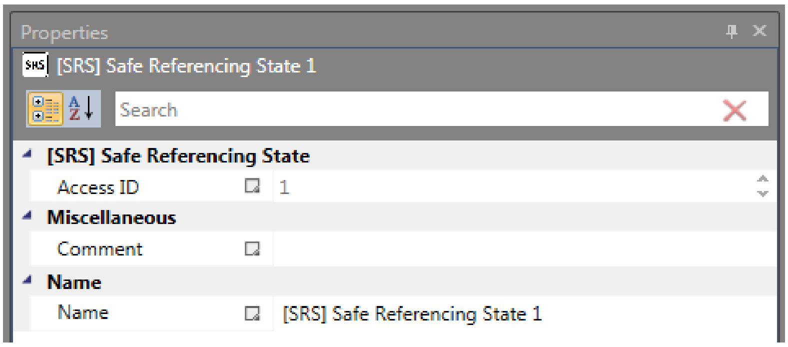
RESET function: The axis referencing can be reset by a rising edge at the "Reset Reference" input.



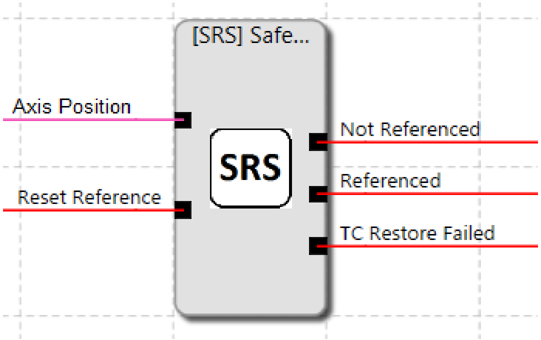
Description of function:

The SRS function is to be regarded as a status block of the SRX. It signals the status of the referencing via the corresponding logic output. In addition, the SRS can be used to reset the referencing of the axis by a rising edge at the "Reset Reference" input.

Parameter:



Parameter	Description
Access ID	The access identifier is used to identify the function element in the application program.
Comment	Input option for an application-specific comment.
Name	put option for an application-specific name



Axis Position	Input signal to assign the corresponding axis.
Reset Reference	Edge triggered signal to reset the referencing of the axis.
Not Referenced	The axis is not referenced: Encoder position was plausible after switch-on (system did not move), but no referencing has been performed yet.
Referenced	The axis is referenced.
TC Restore Failed	The axis is no longer referenced: After switching on, a deviation of the encoder position was detected and thus referencing was deactivated. Referencing can be restored via safe-TC-rec.

SEL (Safe Emergency Location)

Monitoring of the maximum movement range

Amount: c. f. "Overview of safety modules"

Function: Monitoring of the permissible speed in relation to the relative distance to the maximum limit of the movement or to the maximum limit of the adjustment range. This function replaces the conventional emergency limit switch!

Input: Standardised position signal X from the encoder interface.

RESET function: The violation of the permitted monitoring range is saved and requires a RESET acknowledgement. This is done alternatively via:

- RESET function in the "Input elements" group
- Function button at the front part of a basic unit
- Input of F-Bus Reset

NOTICE: Reset can only be carried out, if the current position is within the configured travel range.

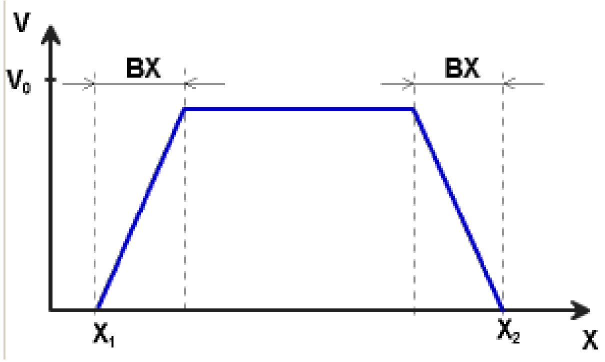
Description of the function:

- Calculation of the current velocity V from the position signal X
- Determination of the stop distance related to the current status of acceleration and speed

  => Cyclical determination of $stop\_distance_{act.}$ = f (V, a) with a = acceleration
- Comparison: $Pos_{act.}$ + $stop\_distance_{act.}$ < $target\_pos_{max}$
- Comparison: $Pos_{act.}$ - $stop\_distance_{act.}$ > $target\_pos_{min}$

The calculation is based on a velocity profile that can be either trapezoidal or sigmoid. For a trapezoidal speed profile, the limit curve results from parameterised acceleration, whereas for a sigmoid speed profile additionally acceleration change is included in the calculation.

Trapezoidal velocity profile:



X1 = Min. Position

X2 = Max. Position

V0 = maximum sped for (X1 + BX) < X < (X2 – BX)

F = type of the speed profile (trapezoidal or sigmoid)

-Trapezoidal

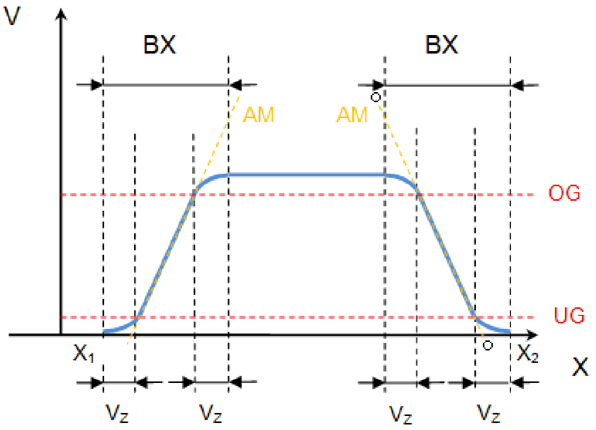
BX = brakes/proximity range

-Sigmoidal

AM = maximum acceleration

DA = type of acceleration

Sigmoid speed profile



BX = brakes/proximity range

X1 = min. position

X2 = max. position

VZ = scattering time S

AM = max. acceleration

UG/OG = range of max. acceleration

Output function:

Range		HI	LO
X < X X > X2	OR		X
X >= X1 X <= (X1 + BX) V < limit curve	AND AND	X	
X >= (X2 – BX) X <= X2 V < limit curve	AND AND	X	
X >= X1 X <= (X1 + BX) V >= limit curve	AND AND		X
X >= (X2 – BX) X <= X2 V >= limit curve	AND AND		X

Limit curve = speed profile deduced from current parametrization

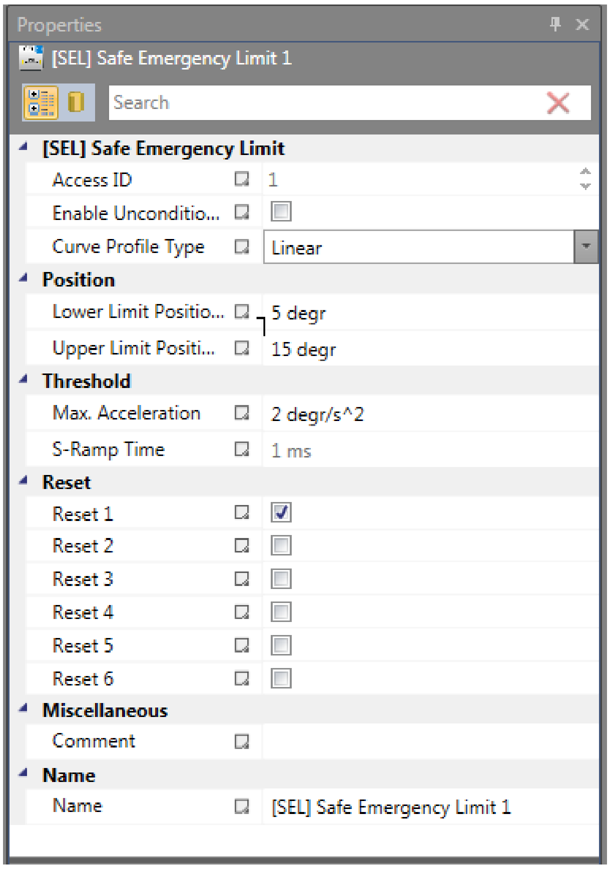
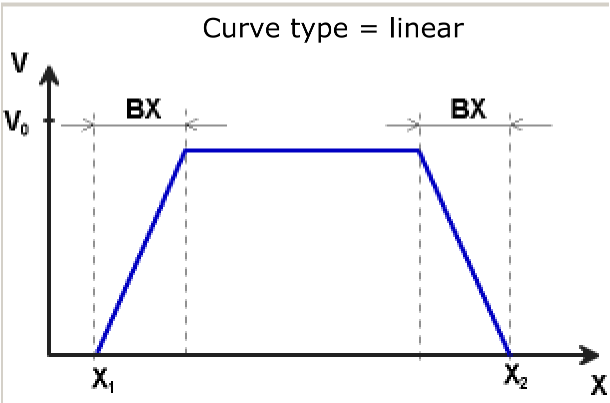


Figure 200 properties window of SEL monitoring

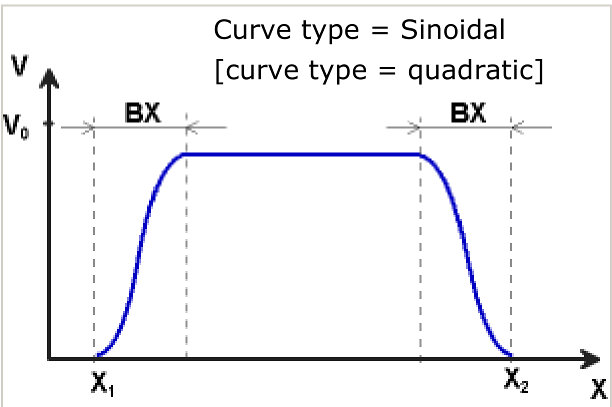
Parameters:

Activate permanently
If this option is set, the monitoring function has no input connection. This function is active from the start of the device.

Curve type
- Linear
Linear calculation method for the stop distance in relation to limit position



- Sinoidal
Quadratic calculation method for the stop distance in relation to the limit position



Lower limit position X1
Lower limit position

Upper limit position X2
Upper limit position

Max. acceleration
Max. acceleration value within BX

Time of S-ramp
Edge time of acceleration => time from acceleration = 0 to max. acceleration

Reset
Select Reset.

Input example 1:
At a production machine, access to the work zone shall be guaranteed at certain positions of the main input axis for manual input or for settings. In this position, the drive remains active, and is only monitored for a standstill. The limits of the work zone are variable, and shall be monitored electronically in a safety-relevant mode instead of via mechanical safety limit switch. The movement that shall be actively monitored is a linear movement. An absolute encoder is connected with the main drive axis of the linear length measuring system. The drive functions with an electric motor with an integrated motor feedback system and an intermediate gear.

1. Limit position
The referential zero position of the main drive axle is situated in the top dead center. The subordinate mechanical follow-up is X1 = -5 mm.
The lower limit position is at 600 mm + a 5 mm safety limit.
=> X2 = 605 mm

2. Type of the speed selection
The drive regulator / the position regulator uses a ramp control (shock limiter) for acceleration with resulting S-slip of velocity to minimise deviations and process markers
=> selection of the S-shape

3. Selection of the limit values
The other limit values are taken from the plant setting.
Maximum acceleration = 1000 mm/s²
Maximum change of acceleration = 3000 mm/s³

SLP (Safely Limited Position)

Target position monitoring

Amount: c. f. "Overview of safety modules"

Input: standardised position signal X from encoder interface

RESET function: The violation of the permitted monitoring range and requires a RESET confirmation. The RESET confirmation is alternatively carried out via:

- RESET function in the group Input elements.
- Function button at the front side of a basic unit
- F-Bus Reset Input

Description of the function:

- Calculation of the current velocity V from the position signal X
- Determination of the stop distance in relation to the current statuses of acceleration and velocity

  => Cyclical determination of the $\text{Stop\_Distanz}_{\text{Akt.}}$ = f (V, a) with a = acceleration
- Comparison: Pos_{act} +$\text{Stop\_Distance}_{\text{act.}}$ < Target_Pos
- Comparison: $\text{Pos}_{\text{act.}}$ - $\text{Stop\_Distance}_{\text{act.}}$ > Target_Pos
- Direction-dependent enabling of the function CW = count-up CCW = count-down

NOTICE: If the function is enabled, it must be programmatically and automatically ensured that CW and CCW never are "1" at the same time because otherwise an alarm is triggered.

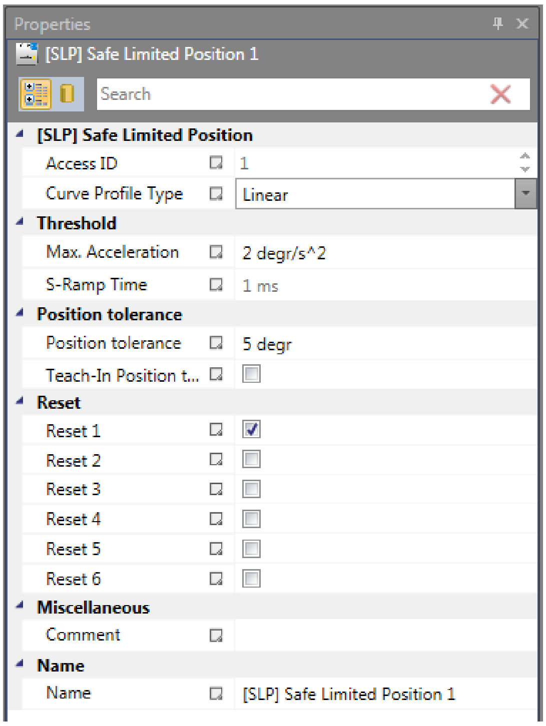


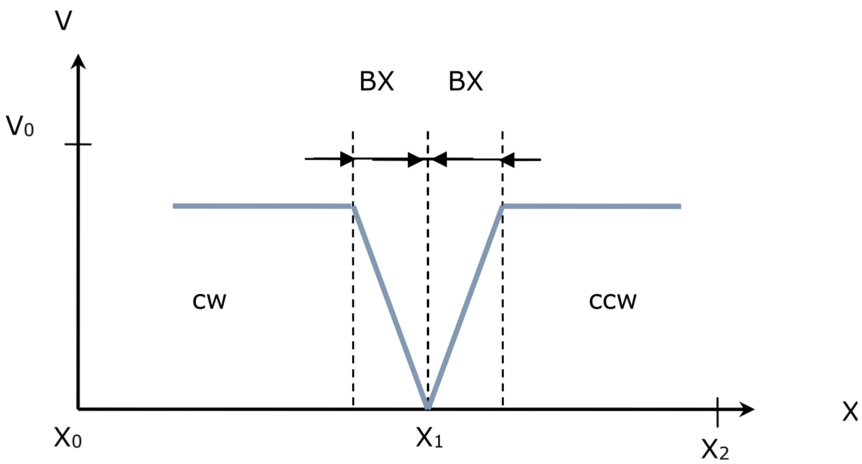
Figure 201 properties window of SLP monitoring

<u>Parameters:</u>

Curve type

-Linear

Linear calculating method for the stop distance in relation to the target position



BX = brake ration / proximity range

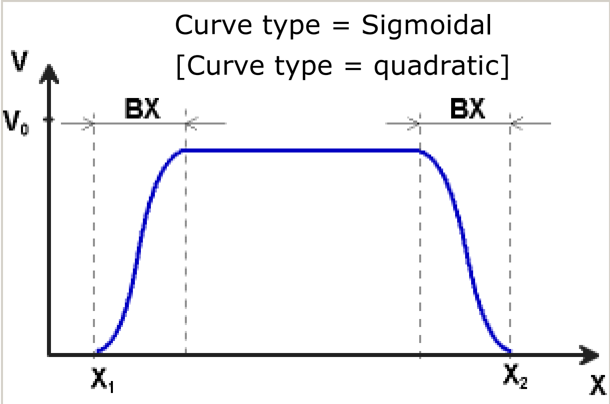
X2-X0 = measured length, cf. encoder configuration

X1 = target position

cw = cw input enabled ($pos_{act.}$ + $stop\_distance_{act.}$ < target_pos)

ccw = ccw input enabled ($Pos_{Akt.}$ - $stop\_distance_{act.}$ > target_pos)

-Sigmoidal
Quadratic calculation method for the stop distance in relation to the target position



Max. acceleration
Max. acceleration value within BX

Max. acceleration change
Value of maximum permissible acceleration change in BX when using the quadratic calculation method.

Position tolerance
Max. permitted position deviations to target position

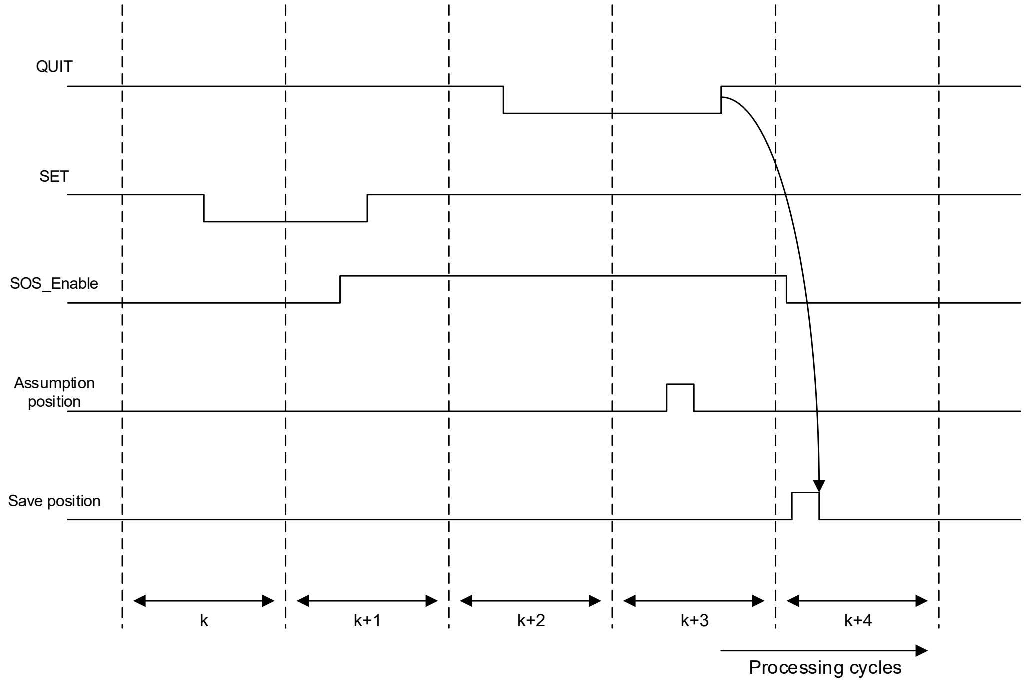
Reset
Selection of Reset.

Saving of the target position via the Teach-in
The Teach-in option can be used to save the target position in the **SDU device** without a subsequent manual setting. To do this, the following steps are necessary:

- By enabling the "Teach-In" button, the input field "target position" changes to "position range". At the same time, the input dialogue is enlarged by the SOS function.
- The recording of a position with the Teach-in option can only be carried out during a standstill with the SOS function being enabled and the SLP function being disabled.
- For the recording of a position, the two signals "Set" and "Quit" are necessary. These signals appear as input connections of the function module, if the TeachİIn option is activated.
- The Teach-In mode automatically activates the SOS function and checks the result of this function. The missing triggering of the SOS function is a condition for an active Teach-In cycle.
- The position is only recorded if the current position is situated within the determined position range.

- The successfully recorded Teach-in position appears in the process input image.
- The Teach-in position is safely stored – also in case of a current interruption.
- The Teach-in position is reset after every configuration upload.

Time characteristics of the "SET/QUIT" process:



The sequence is time-monitored and triggers an ALARM if the expected values are exceeded.

ATTENTION: The maximum time frame is 3 seconds!

Position tolerance

Tolerance value for the TeachIn position.

cw (enabled) = $Pos_{Act.}$ + Stop_distance $_{act.}$ < target_pos + position tolerance

ccw (enabled) = $Pos_{Act.}$ - Stop_distance $_{act.}$ < target_pos – position tolerance

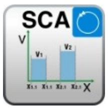
NOTICE:

If the TeachIn function is used, the monitoring limit is enlarged by the position tolerance value. Without the TeachIn function, the position tolerance value is zero.

For the input "Setting" either a push-button switch must be used, or two position switches linked with AND must be assigned to the input. When the position tolerence is determined, the maximum permissible position must be considered. =>maximum position tolerance value = maximum position in the direction of travel – Teach-in position.

Parameters of the SOS dialog: cf. SOS function

SCA (Safe Cam)



Monitoring of the position range by speed monitoring / velocity monitoring

Amount: c. f. "Overview of safety modules"

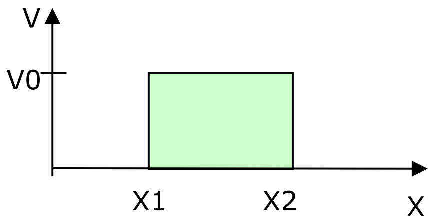
Function: monitoring of a parametrisable position range with assigned minimum limit and maximum limit. In the permitted range additionally monitoring of the speed.

Input: standardised position and velocity signal X and V by encoder interface

RESET function: The violation of the permitted monitoring range is not saved. No RESET confirmation is necessary.

Description of the function:

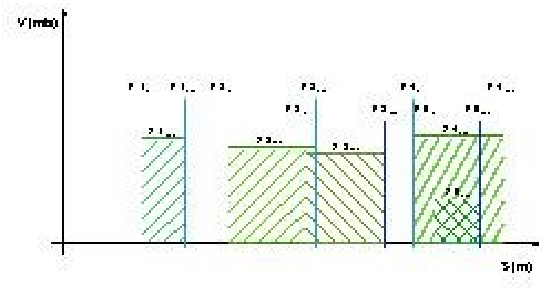
- Comparison of the current position to parametrised range limits
- Comparison of the current velocity to the parametrized limit
- Permanent activation of the block



Output function

Range		HI	LO
X < X1 X > X2	OR		X
X >= X1 X <= X2 V < V0	AND AND	X	
X >= X1 X <= X2 V >= V0	AND AND		X

Ranges can be defined as overlapping and as nested.



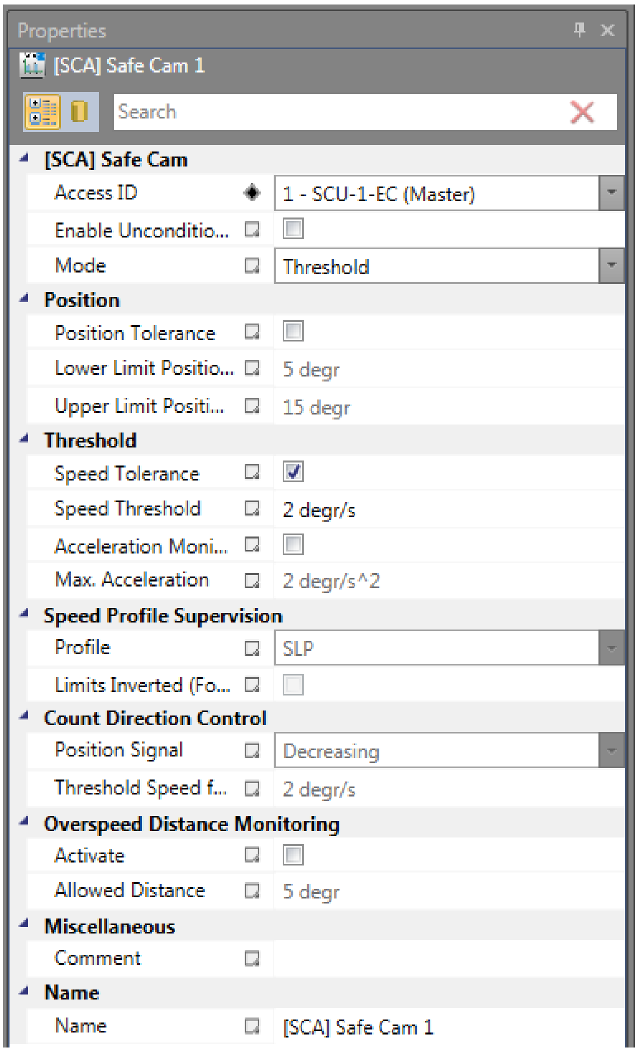


Figure 202 properties window of SCA monitoring

<u>Parameters:</u>

Permanent activation
If this option is set, the monitoring function has an input connection. The function is active from the start of the device.

Lower limit position X1
Lower limit position

Upper limit position X2
Upper limit position

Speed threshold
Maximum permissible speed in set position range

Max. acceleration
Maximum permitted acceleration in set position range

Direction-independent release
Currently not supported

Negative encoder counting direction:
Currently not supported

Positive encoder counting direction:
Currently not supported

Activation and release of the speed direction
Currently not supported

Travel monitoring
Currently not supported

<u>Example of input:</u>

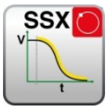
At a production machine the access to the work zone for manual insert or rather, manual setting shall be released for certain positions of the main input axis. The drive remains in this position and is only monitored for standstill. The limits of the expansion stroke are variable and shall be monitored in an electronically safety-relevant way. The active movement to be monitored is a linear movement. An absolute encoder is directly interlocked with the main feed axis as a linear path measurement system. The main axis is the referential axis for the SCU unit.

1. Range selection

With the position monitor the position of the main axis in the upper zero point shall be monitored. The upper zero is also the referential zero point in the length measuring device of the feed axis. If the range is recognised, one protective device is released for opening.

Limit range X1	= upper position = 0 mm
Limit range X2	= lower tolerance limit for position = 2 mm
Speed	= tolerated speed for conservation of position
Acceleration	= tolerated acceleration for conservation of position = 5 mm/s

SSX (safe stop 1/2)



Function monitoring for emergency stop

Amount: c. f. "Overview of safety modules"

Function: monitoring of an EMERGENCY STOP function

Input: standardised speed signal X from encoder interface

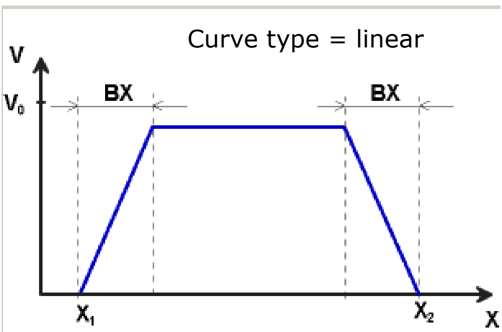
RESET function: Violation of the permitted monitoring range is saved and requires a RESET confirmation. The RESET confirmation is alternatively carried out via:

- RESET function in the group of the Input elements
- Function button at the front side of a basic device
- F-Bus (fieldbus) Reset Input

NOTICE: If this function triggers an alarm, after a RESET, the device unit is reset to the normal state either on the device or via the configured input of the "Alarm Reset" block.
Before the reset, in the monitored periphery an operating state must have been achieved, in which the "Enable" input of the function is set to zero.

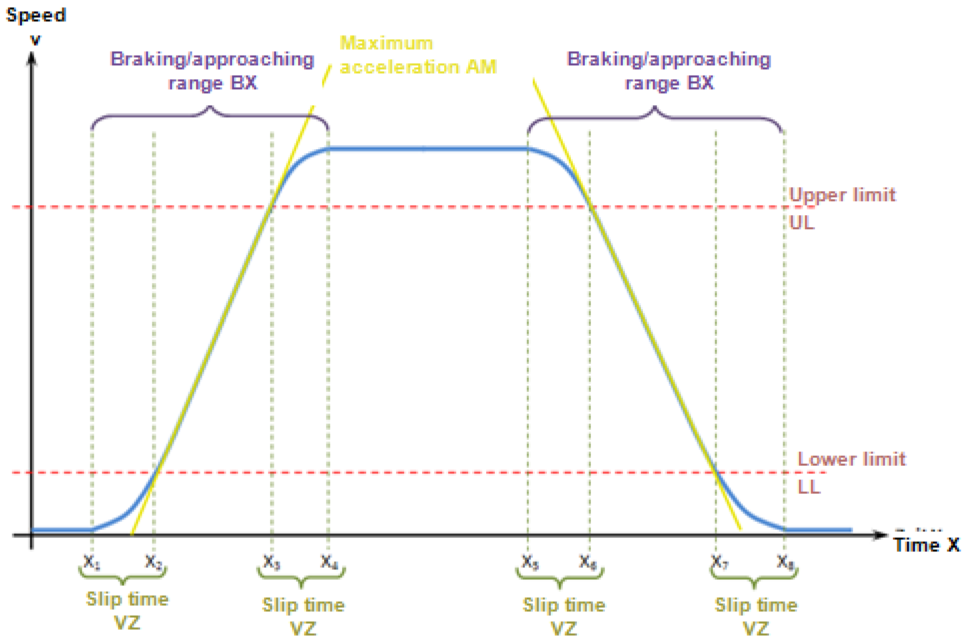
Description of the function
Monitoring of the process of a controlled EMERGENCY STOP by comparison of the speed drop over time to a parametrised monitoring limit curve. The monitoring limit curve results from latency time, the maximum speed distance zu the limit curve and their characterization, computed from acceleration and acceleration change. After the activation of monitoring, the course of the limit curve is calculatedon the basis of current speed.

Curve type
-Linear curve type



BX = braking range / proximity range
X1 / X2 = time for ramp function sequence
V0 = Start velocity of the ramp function

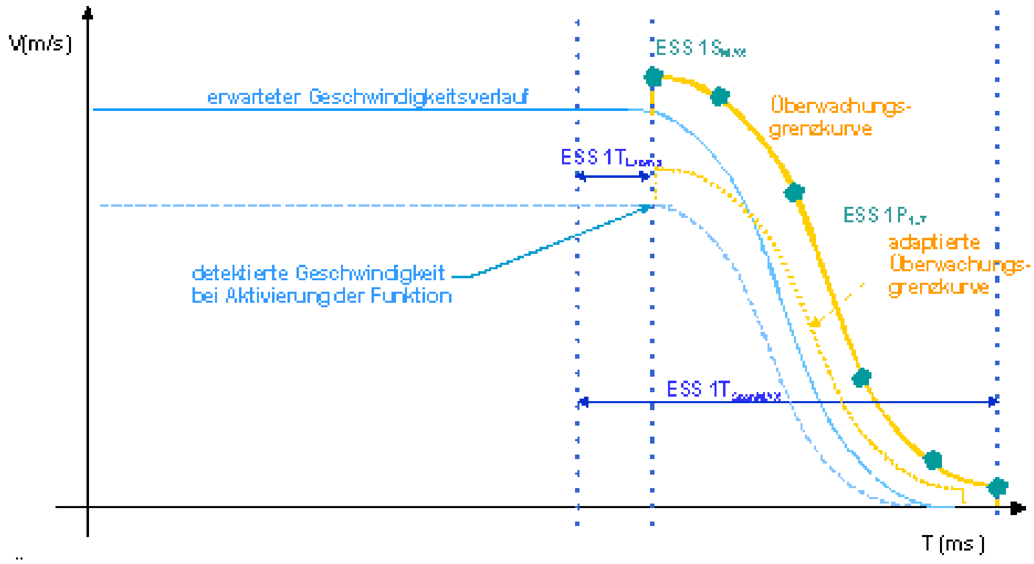
- Sigmoidal speed profile
(S-shape)



The Sigmoidal speed profile shows the change of speed over the time, or rather, the course of speed over the time.

Max. acceleration AM
Value of maximum acceleration within BX

S-smoothing time VZ
The smoothing time VZ means the period in which speed does not change linearily, or rather, the period for acceleration change from a=0 or a=amax or vice versa.



Monitored limit curves at a sigmoidal course of speed

Output function

Range	HI	LO
$T < T_{Latency}$	X	
$T > T_{Latency}$ AND $V < V_{Limit\ curve}$	X	
$T > T_{Latency}$ AND $V > V_{Limit\ curve}$		X

Every function block can be parametrised to stop-category 1 or stop-category 2. In STOP-category 2, after the expected standstill the SOS function is activated automatically.

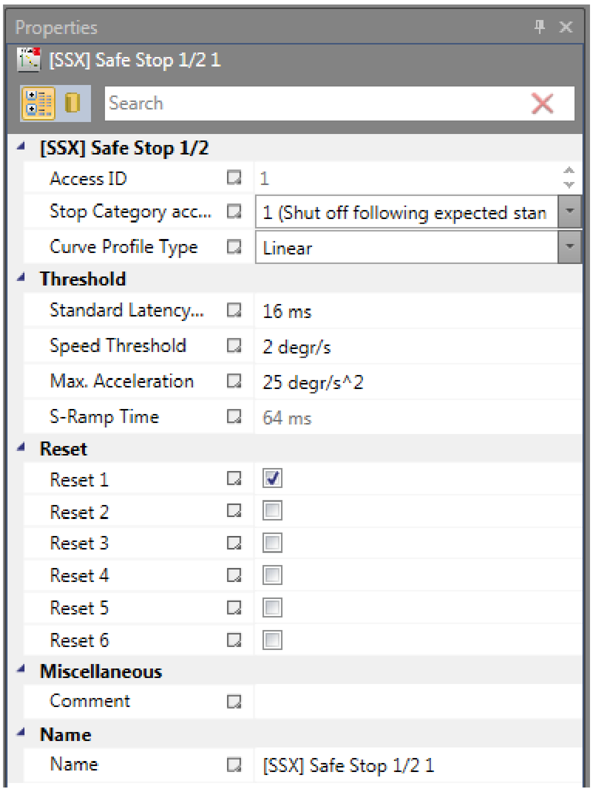


Figure 203 properties window of SSX Monitoring

Parameter:

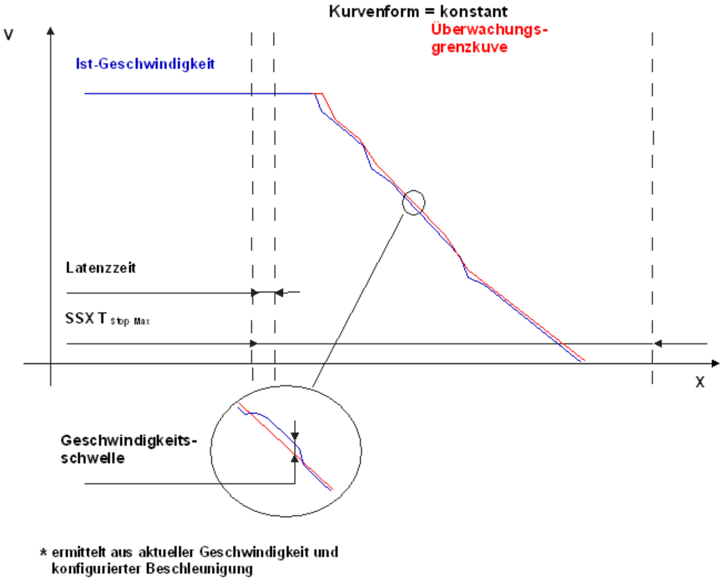
Stop-category 1
This option allows monitoring the controlled EMERGENCY STOP according to EN 60604. According to the definition in the standard, the current supply must be disconnected after the standstill of the drive. The separation is supported by the shift of the initial value of the SSX function from "1" to "0".

Stop-category 2 (SOS after expected standstill)
Currently not supported.

Curve type

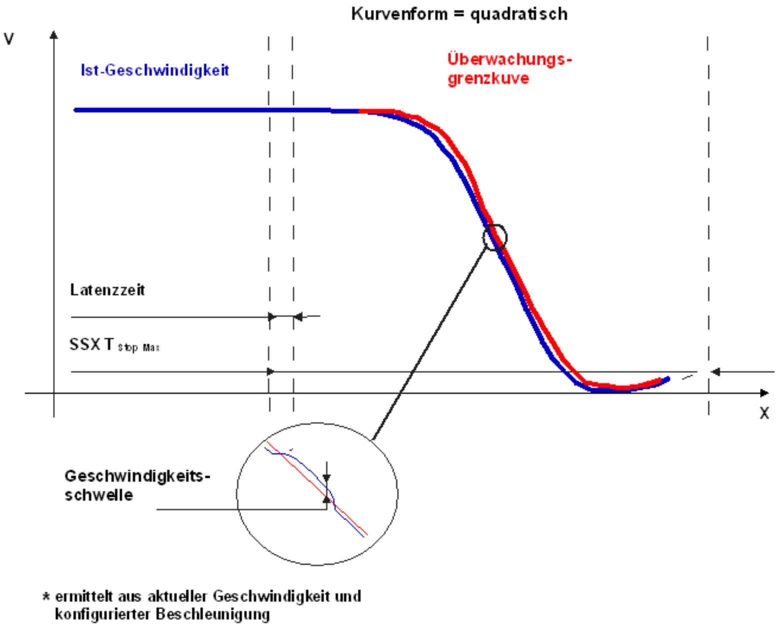
- linear curve typ

Linear speed and constant acceleration curve for the stop sequence



-Sigmoidal curve type

Sigmoidal speed an linear acceleration curve for the stop sequence



Latency time standard

Latency time until the active delay occurs.

Max. speed (speed threshold)
Tolerance value of current speed.
If the current speed is greater than the current limit curve + tolerance value, energy supply is disconnected.

Max. acceleration
Determines the increase of the stop curve.

S- slip time
The slip time indicates the period, in which the curve's form is sinoidal. The period before and after the acceleration a_{max}.

Reset
Select Reset.

Input (example):
At a production machine, at certain position of the main input axis access to the work zone shall be guaranteed for manual input or for settings. In this position, the drive remains active, and is monitored only for a standstill. The limits of the work zone are variable and shall be monitored electronically in a safety-relevant mode instead of being monitored by a mechanical safety limit switch. The movement, that shall be monitored electronically, is a linear movement. An absolute encoder is connected to this main drive axis of the linear length measuring system. The drive functions with an electric motor with an integrated motor feedback system and an intermediate transmission.

1. Selection of the stop category
To keep machine downtime and restarts as short as possible, the stop category must be used according to DIN 60604-1 (Controlled stop with subsequent actively controlled drive to V = 0) => selection of stop category 2

2. Type of speed selection
The drive regulator / the position regulator uses a ramp limiter (shock limiter) for acceleration with the resulting S-slip to minimise deviations and process. => selection of S-slip.

3. Selection of the limit value
For monitoring, the most unfavourable latency from the occurence of the emergency-stop to the start of the braking process must be entered. The braking process must be carried out with the standard control. The sequency time of the programme of the standard control causes latency = cycle duration * 2 = 50 ms

The other limit values are taken from the setting of the plant.

Maximum infeed speed	= 300 mm/s²
Maximum acceleration	= 1000 mm/s²
Maximum acceleration change	= 3000 mm/s³

SLI (Safely Limited Increment)

Monitoring of the maximum increment

Amount: c. f. "Overview of safety modules"

Function: monitoring of the maximum increment

Input: standardised position signal / speed signal V and X from encoder interface. Direction LEFT / RIGHT.

RESET function Violation of the permitted monitoring range is saved and requires a RESET confirmation. RESET confirmation is alternatively carried out via:

- RESET function in the group Input elements.
- Function button at the front side of a basic unit
- Fieldbus inputs via reset block

NOTICE: It must be programme-technically ensured that in case of a triggering event. Otherwise, the SLI-block could not be reset again.
When activating the function, it must be programme-technically ensured that CW and CCW never become 1 at the same time. Otherwise, an alarm is emitted.

Description of the function:

- Monitoring of the maximum permitted increment = relative travel ranges for continuous travel in the jog mode.
- Calculation of the current rotational direction RX from the position signal / the speed signal X
- Determination of the relative travel distance after the start of the movement
- Monitoring for observation of the given direction and of the max. given travel distance

NOTICE:
In case of a reset, connections in the block must be reset to "0". Otherwise, the function cannot be reset.
If the function is activated, it is not permitted that "cw" and "ccw" are activated simultaneously for the input signal. If both options are activated, an alarm is triggered.

Range	HI	LO
V < 0 AND DIRECTION FLAG = LEFT AND Relative travel distance < max. increment	X	
V >= 0 AND DIRECTION FLAG = RIGHT AND Relative travel distance < max. increment	X	
V < 0 AND (DIRECTION FLAG = RIGHTS OR Relative travel distance > max. increment		X
V > 0 AND (DIRECTION FLAG = LEFT OR AND Relative travel distance > max. increment)		X

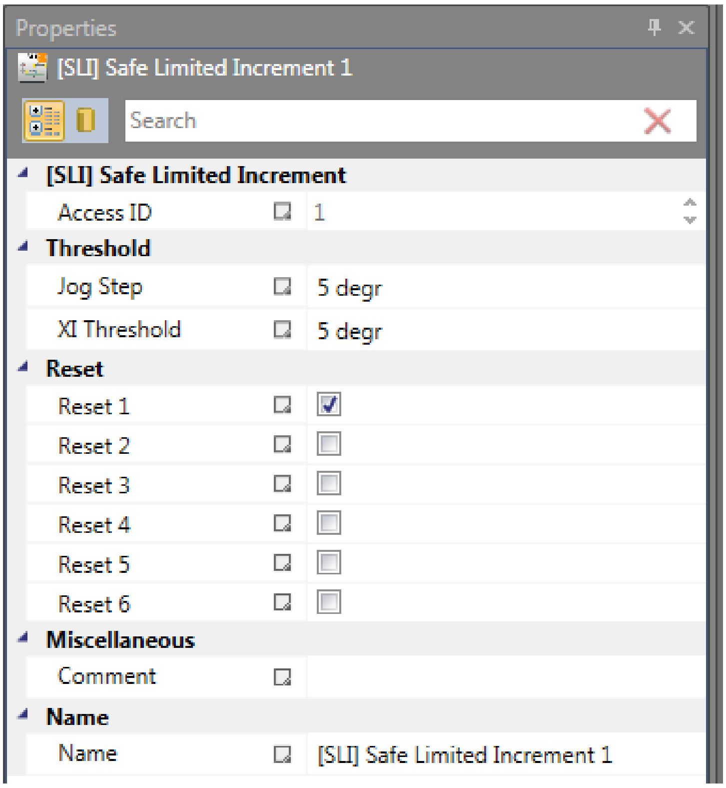


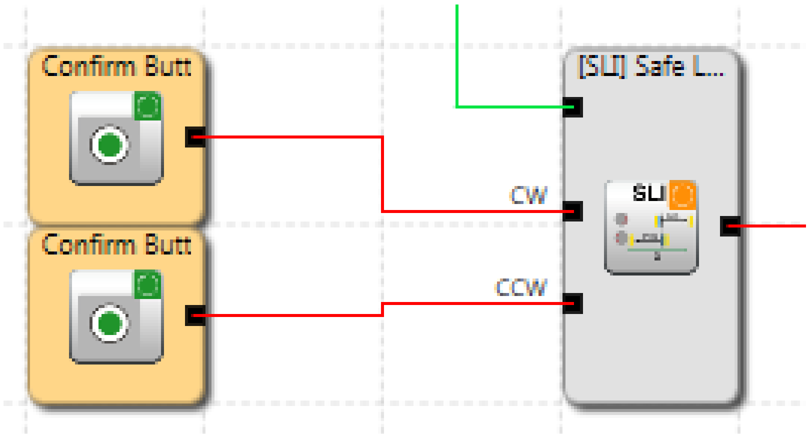
Figure 204 properties window of SLI function

Parameters:

Jog Step
Maximum relative travel distance after activation of the monitoring function

XI threshold
Tolerance threshold for the monitoring of the travel distance in the opposite direction

Example of an activation: [Achse]: 1



Example of an input:
The max. travel distance of the material feeding system of a production plant shall be safely monitored in jog operation. According to the risk analysis this travel distance is no more than 50 mm. A wrong travel distance in the opposite direction shall be monitored.

1. Jog step
The relative travel distance (only available in the incremental encoder) is monitored => input of max. travel distance according to risk analysis with a tolerance = 55 mm

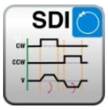
2. Monitoring of the travel distance
Permitted travel distance in opposite direction (=creep movement of the drive) = 1 mm/s

3. Monitoring input
The monitoring has two inputs to indicate the direction. An active direction signal activates the monitoring function.

After the activation of the monitoring device, the direction must be indicated by an unambiguous signal => Switches for the direction are direction are directly connected to the standard inputs of the SCU => selection I 01, I 02

NOTICE:
If both input signals are “1”, this is recognised as an impermissible status, and an alarm message is enabled.

SDI (Safe Direction of Increment)



Recognition of the direction

Amount: c. f. "Overview of safety modules"

Function: monitoring of the given rotational direction / motion direction

Input: standardised position signal / speed signal X from encoder inferface. Direction flag LEFT / RIGHT.

Reset-Function: Violation of the permitted monitoring range is saved and requires a RESET confirmation. RESET confirmation is alternatively carried out via:

- RESET function in the group Input elements.
- Function button at the front side of a basis unit
- Fieldbus input via Reset block.

NOTICE: It must be programme-technically ensured that the inputs of the SDI block do not remain on 1 in the event of a trigger. Otherwise, the SDI block could not be reset any more.
If the function is activated, it must be programme-technically ensured that at no time CW and CCW become 1 at the same time because otherwise an alarm is emitted.

Example**:**
Cf. SLI function.

Functional description

- Monitoring of the given rotational direction / the given motion direction
- Calculation the current rotational direction RX from the position signal / speed signall X

Output function:

Range	**HI**	**LO**
V < 0 AND DIRECTION FLAG = LEFT	X	
V >= 0 AND DIRECTION FLAG = RIGHT	X	
V < 0 AND DIRECTION FLAG = RIGHT		X
V > 0 AND DIRECTION FLAG = LEFT		X

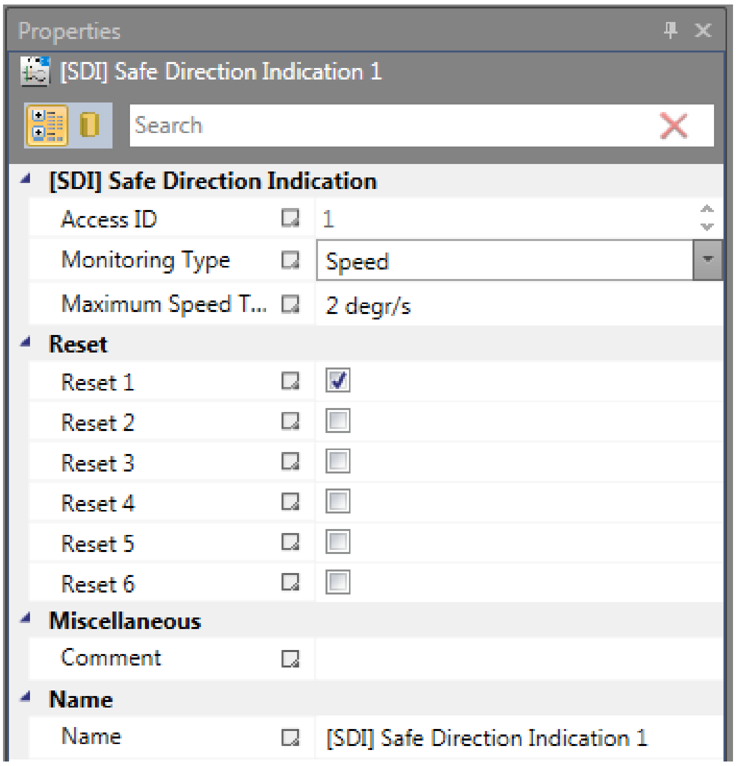


Figure 205 properties window of SDI function

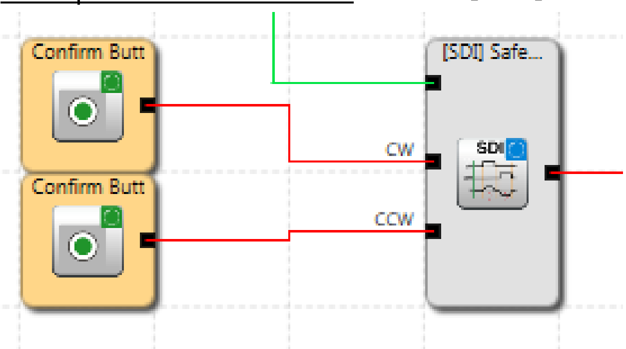
<u>Parameters:</u>

Speed monitoring mode
Maximum tolerance threshold for speed in opposite direction

Position monitoring mode
Maximum tolerance threshold for position in opposite direction

Reset
Select Reset.

<u>Example of an activation:</u> [axis]: 1



Example of input:

At a production device the speed of certain manual processes must be monitored for a safey-reduced value, and for the standstill direction and the movement direction must be monitored. The movement that shall be actively monitored is a rotating movement. The drive functions with an electric motor with and integrated motor feedback system and an intermediate transmission.

1. Input for the monitoring function
Speed monitoring (only incremental encoder available)
=>Speed

2. Speed monitoring

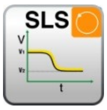
Permitted speed in opposite direction (= creep movement of drive) from the plant parameter = 1 mm/s

3. Monitoring input

The monitoring module has two inputs to indicate the direction. An active direction signal activates the monitoring function.

NOTICE:
If both input signals are “1”, this is recognised as an invalid status, and an alarm message is enabled.

SLS (Safely Limited Speed Control)



Monitoring of a minimum speed

Amount: c. f. "Overview of safety modules"

Function: monitoring of a minimum speed

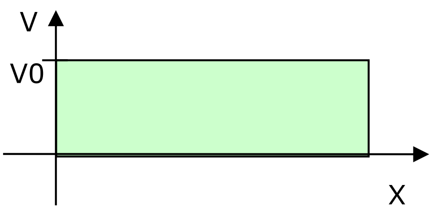
Input: standardised position signal X from the encoder interface

RESET-function: Violation of the permitted monitoring range is saved and requires a RESET confirmation. The RESET confirmation is alternatively carried out via:

- RESET function in the group of the input elements.
- Function button at the front side of a basis unit
- Fieldbus inputs via Reset block

Description of the function:

- Monitoring of the maximum speed or the maximum rotational speed of a drive.
- Calculation of the current speed V from the position signal, or rather, from the digital speed signal X
- Comparison of the actual speed to the parametrised speed threshold
- Monitoring of a change of speed from fast to slow by means of the selected SSX -ramp
- Error distance monitoring



Output function

Range	HI	LO
V < V0	X	
V >= V0		X

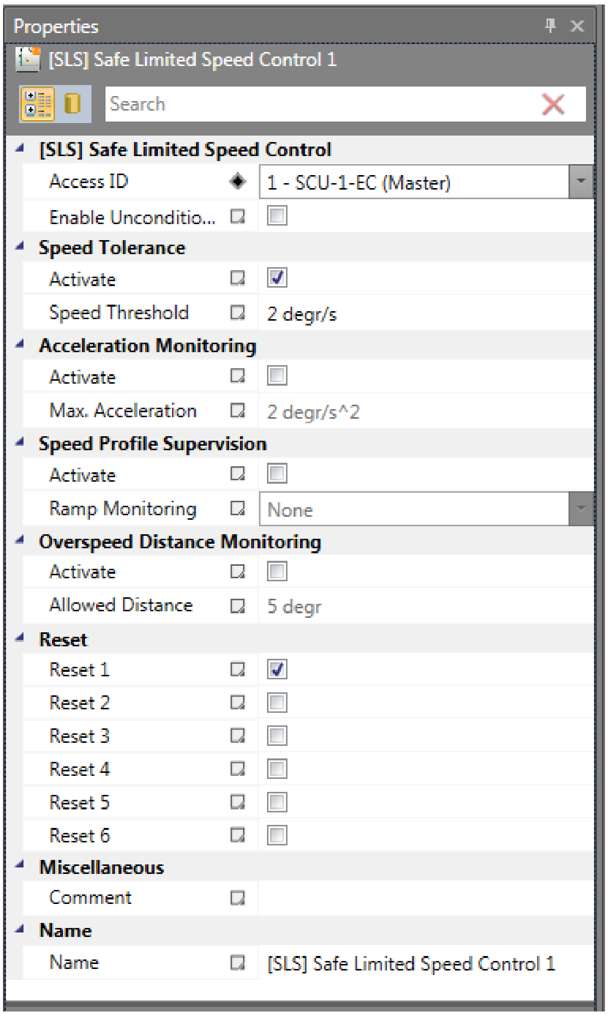


Figure 206 properties window of SLS function

<u>Activate permanently:</u>

Activate permanently
The monitoring function is always active and does not have an input connector.

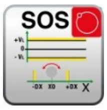
Speed tolerance
To activate speed monitoring

Travel monitoring
Currently not supported.

Error distance monitoring
Currently not supported.

Reset
Selection of Reset.

SOS (safe standstill)



Standstill monitoring

Amount: c. f. "Overview of safety modules"

Function: standstill monitoring

Input: standardized position signal / standardized speed signal

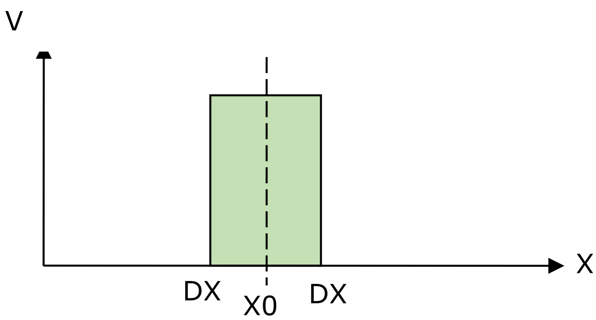
RESET-Funktion: Violation of the permitted monitoring range is saved and requires a RESET confirmation. RESET confirmation is alternatively carried out via:

- RESET function in the group of the input elements.
- Function button at the front side of a basis group
- Fieldbus input via Reset block.

NOTICE: Iit must be program-technically ensured that the input of the SOS block does not remain 1 in case of a trigger event. Otherwise, the input cannot be reset.

Description of the function:

- Standstill monitoring of the drive at current position with released drive and, if necessary, with enabled position controller.
- Calculation of the current speed V from current position signal or from current speed signal X
- Comparison of the current speed with the parametrised monitoring window



Range	HI	LO
X > (X0 – DX) AND X < (X0 + DX)	X	
X <= (X0 - DX)		X
X >= (X0 + DX)		X

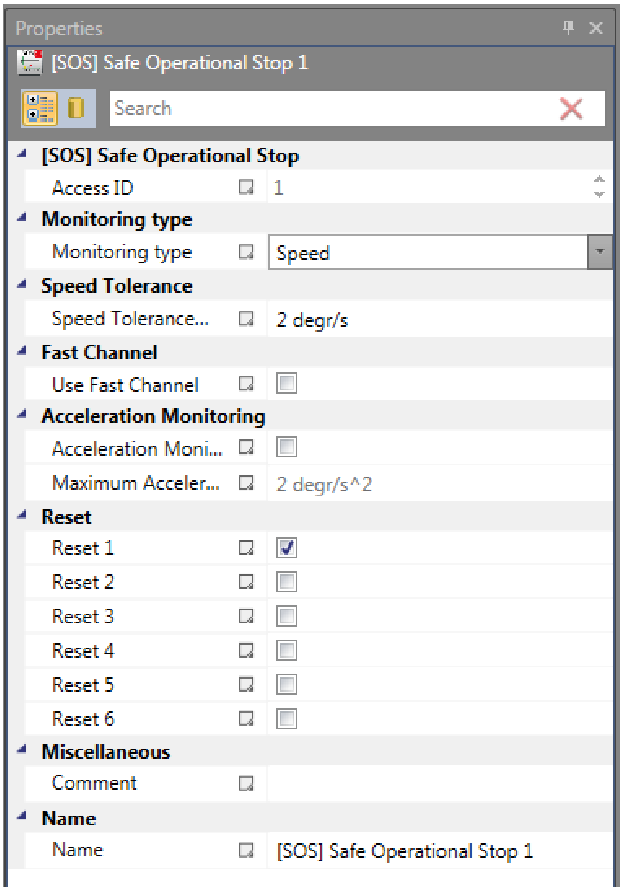


Figure 207 properties window of SOS function

Monitoring mode

Determination of the monitoring mode for standstill up to a minimum speed threshold or up to a position range

Maximum speed tolerance / maximum position tolerance

Minimum speed or a permitted relative deviation from the current position at the time when the SOS function was enabled.

Fast Channel

Currently not supported

Acceleration monitoring

Optional maximum value for speed monitoring during an active SOS function.

Reset

Selection of Reset.

Input example 1:

In a production device, at certain manual processes speed shall be monitored for a safeIy reduced value. Additionally, standstill and travel direction shall be monitored. The active movement tob e monitored is a rotating movement. The drive is an electronic motor with integrated motor feedback systeme and an intermediate gear.

1. Selection of type
Only speed is monitored (e. g. via incremental encoder) => speed monitoring

2. Speed monitoring
Specification of the tolerated value of the speed monitoring.

Input example 2:

At a production machine, the access to the work zone shall be released for manual insert or for manual configuration shall be released at certain positions of the main drivee axis. In this position, the drive remains active and is only monitored for standstill. The limits of the expansion stroke are variable, and shall be monitored in an electronically safety-relevant way instead of being monitored by a mechanical safety end-switch. The active movement to be monitored is a linear movement. An absolute encoder is a directly interlocked connection to the main drive axis. The main drive axis is driven by an electronic motor with an integrated motor feedback system and an intermediate gear.

1. Selection of type
The position is monitored (absolute encoder at hand) => position monitoring

2. Position monitoring
Specification of the tolerable value for position monitoring

MPM (safe synchronisation monitoring)

synchronisation monitoring

Amount: c. f. "Overview of safety modules"

Access-ID: identification of the function element

Axis assignment: 1 Master axis
2-12 Slave axes

Function: synchronisation monitoring

Input: Standardises position siganals form the encoder interface

RSET- function: Violation of the permitted monitoring range is stored, and a confirmation is necessary. The confirmation can be made in the following way:

- RESET function in the group of the input elements
- Function button on the front side of a base modules
- Element for fieldbus reset

Functional description:
Monitoring of the synchronous movement of a configurable Master axis with an axis that is available in the system.
The standardization, or rather, the unit of the comparable axes does not have to be identical. The standardization of the axes towards each other takes place in the monitoring function.

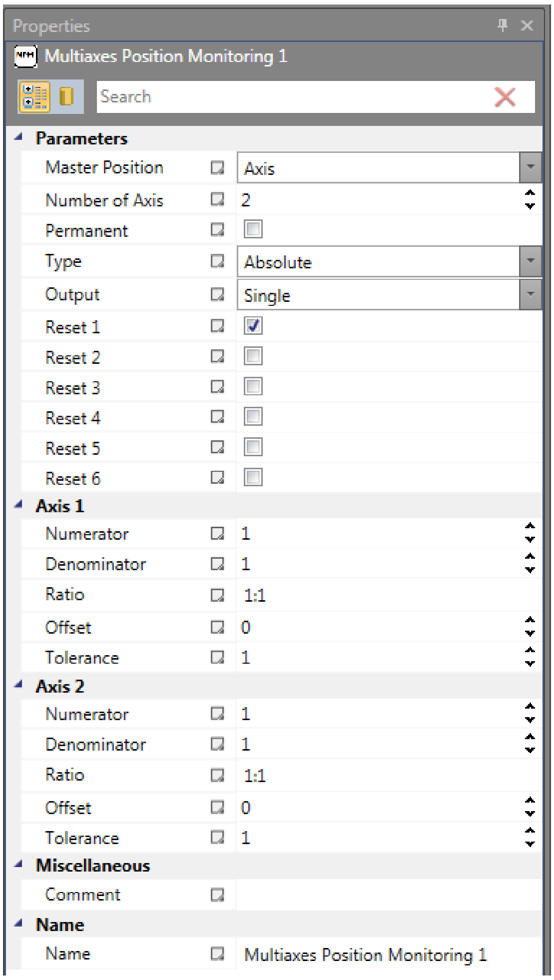


Figure 208 properties window of Multiaxes position monitoring

Position of the Master
Selection (mode) between Axis and Functional. Currently onlyone axis is supported. The supported axis is the referential axis for the comparison with the other axes.

Number of axes
Number of comparative axes 1…12

Permanent
If the function is enabled, it is permanently enabled.

Type
Selection between:

- Absolute: configuration of the offset of the referential axis to the comparative axis
- Incremental: calculation of the offset of the referential axis to the comparative axis with rising activation of the function.

NOTICE: If "incremental" type is used, the activation of "permanent" cannot be used.

Output

Selection between:

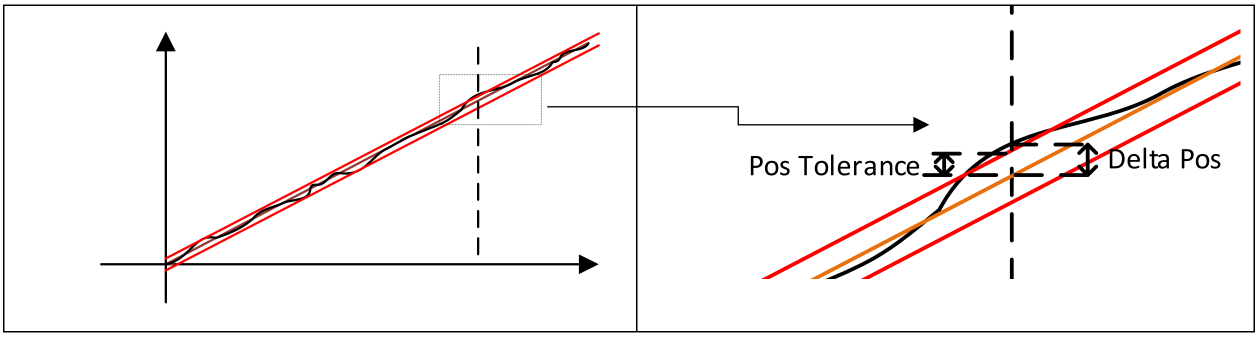
- Single: result of comparison of every axis as group feedback
- Per Axis: result of comparison of every individual axis

Reset

Selection of Reset

Axis x:

- Numerator
  Numerator for the nomination of the position to the Master axis
- Denominator
  Denominator for standardization position to Master axis
- Ratio
  Display of nominator / denominator
- Offset
  Only with Type-> Absolute. The offset is added on comparison axis
- Tolerance
  Tolerance for comparison of Master with comparison axis



SWM (Safe workspace Monitoring)

SWM

Safe Workspace Monitoring

Amount: c. f. "Overview of safety modules"

Function: Monitoring workspace (3D) using defined objects

Input: 3x virtual position (3D)

RESET-function: The violation of the permissible monitoring range is not saved and resets itself again

Description of function:

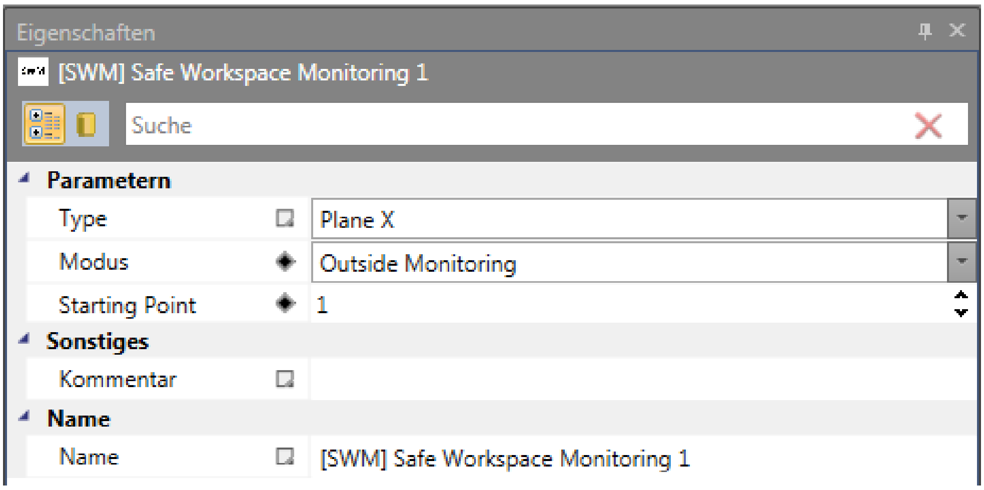
- Monitoring working area of a virtual axis using predefined configurable objects
- Configuration whether valid area is inside or outside the object.
- The result of the function is not saved and resets automatically when the invalid range is left again.

Output function:

Range	Inside	Outside	HI	LO
X,Y,Z Inside object	X		X	
X,Y,Z Outside object	X			X
X,Y,Z Inside object		X		X
X,Y,Z Outside object		X	X	

Parameter „Type“:

- Plane
    - Plane X
    - Plane Y
    - Plane Z



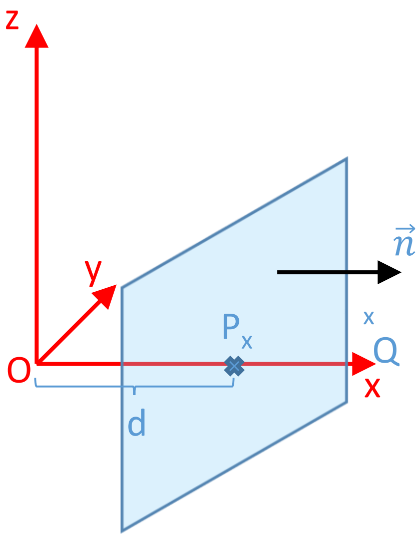
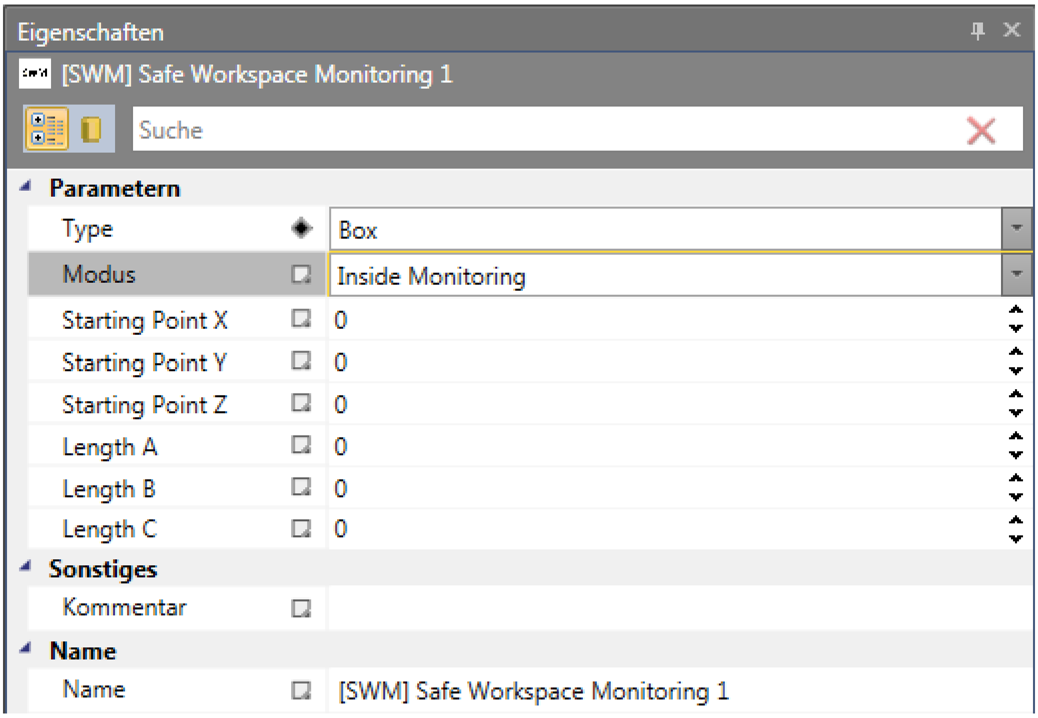
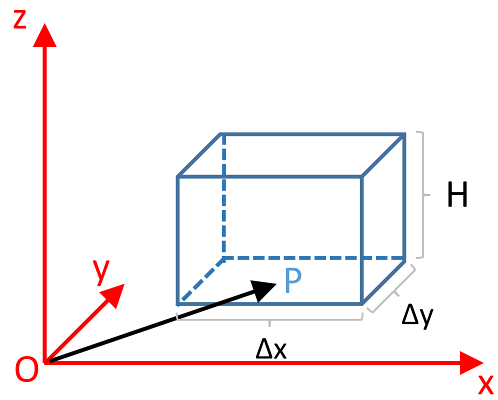
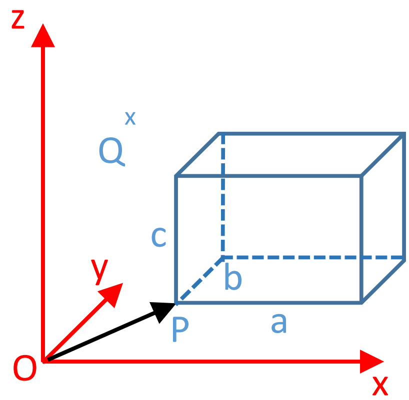


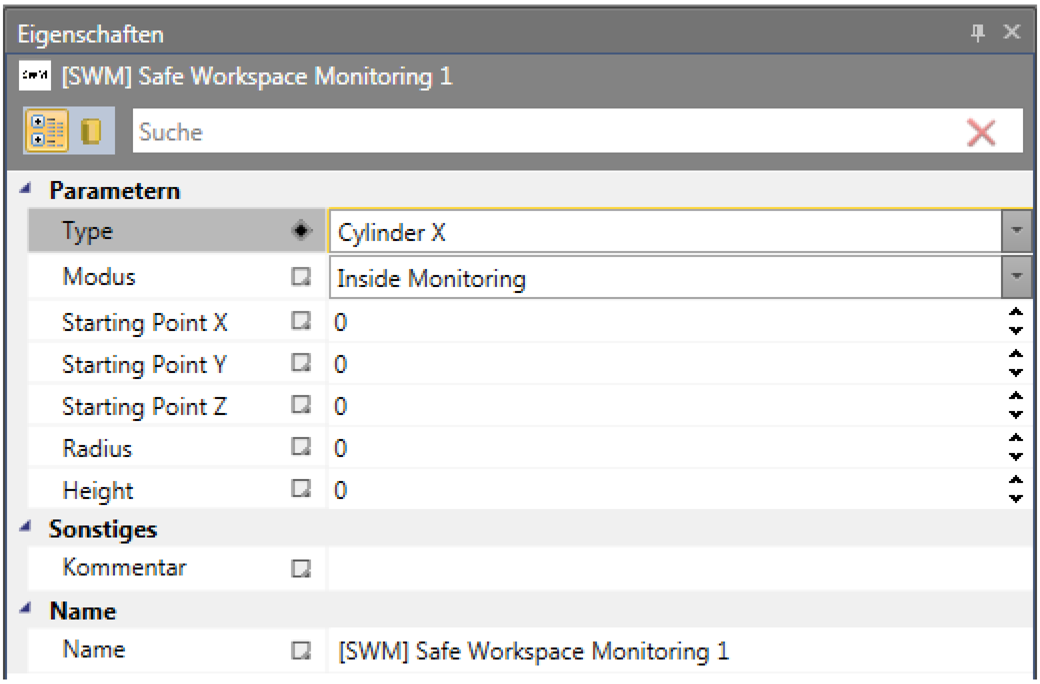
Figure 209 Example plane X

- Box
  - Box (Corner Modus)
  - Box XY (Center Modus at XY plane)
  - Box XZ (Center Modus at XZ plane)
  - Box YZ (Center Modus at YZ plane)



	
Center Modus (hier XY) • Base plane: xy, xz, yz • The side length along the x-axis: A • The side length along the y-axis: B • The side length along the z-axis: C	Corner Modus • The side length along the x-axis: A • The side length along the y-axis: B • The side length along the z-axis: C

- Cylinder
  - Cylinder X
  - Cylinder Y
  - Cylinder Z



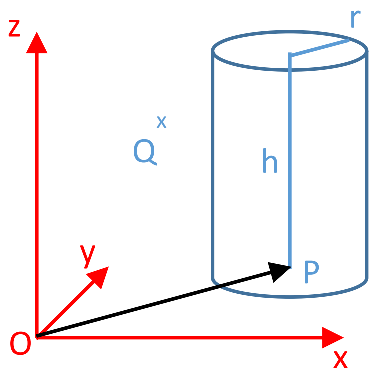
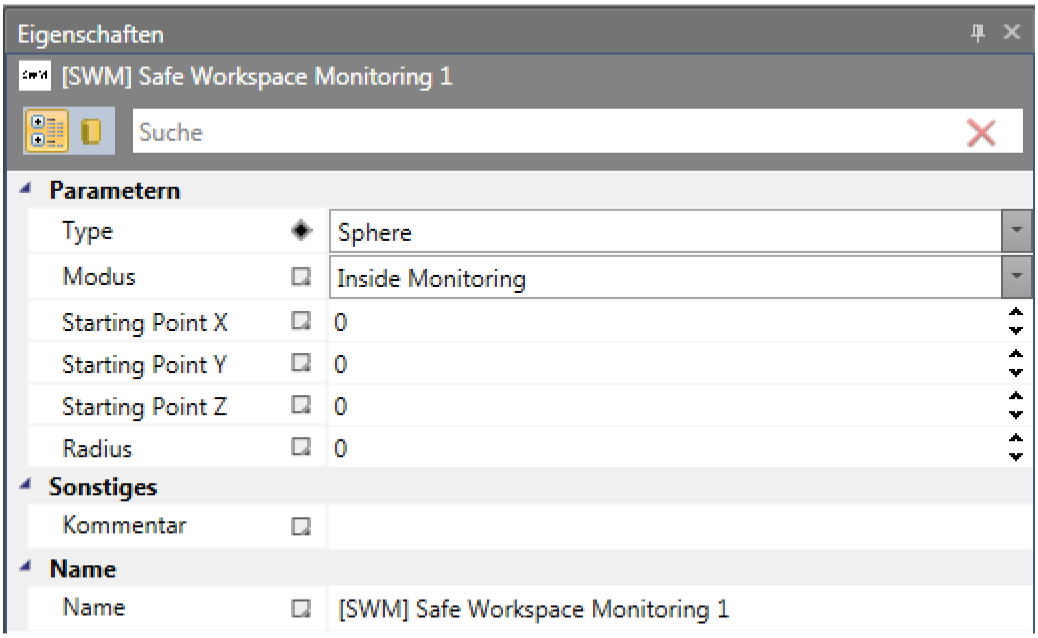


Figure 210 Example definition Zylinder Y

- Sphere
    - Sphere



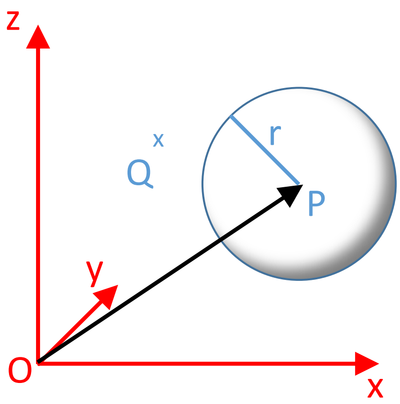
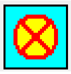


Figure 211 Defintion of a asphere as a safe zone

Configuration „Modus"
- Inside Monitoring
- Outside Monitoring

4.12.5.4. Muting functions

FDB (FSoE Disconnect Block)

 Safe disconnection of an FSoE connection

Number: c. f. "Overview of safety modules"

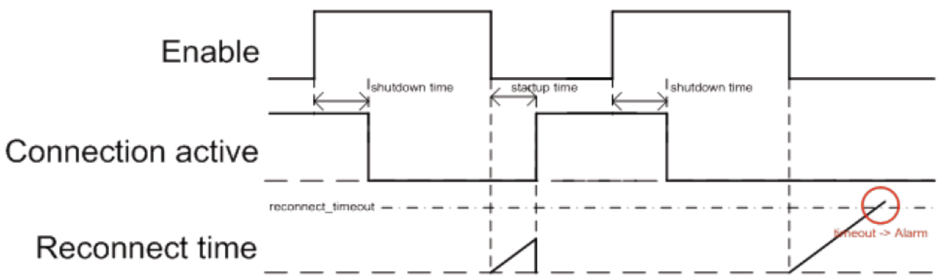
Function: "Disconnect Block" is used to temporarily disable the FSoE connection- The detection, which device must be disabled, is set via the FSoE address. By the function, an error status in the Master in case of defined shut-off / defined disconnection of a Slave unit is avoided.

Input: FSoE connection active / inactive

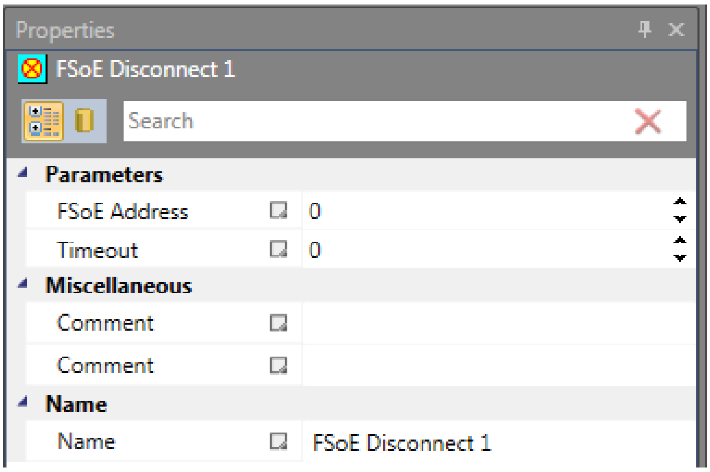
RESET function: no reset necessary

Description of the function:

- At the activation of Disconnect Block, the FSoE connection of the participant must be disables, and the process data are disabled ("0").
- At the deactivation of the Disconnect Block, the Master block tries to restore the FSoE connection to the Slave. If the connection has been successfully established, the FSoE Slave continues to work normally, and the input information / the output information work with a cyclical update of the input data / the output data. If the FSoE connection is not established within a determined, configurable time, an alarm is triggered.

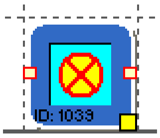


NOTICE: When using SSB, automatically both FSoE connections are disconnected.



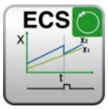
FSoE address
FSoE Slave adress to shut down the FSoE connection of the Slave block on which the function shall act.

Timeout
Timeout in ms, until FSoE connection is actively expected after deactivation of the Disconnect block.



Signal	Description
Enable	0: FSoE connection is not interrupted 1: FSoE connection shall be shut down
Output	0: FSoE connection is inactive 1: FSoE connection is active

ECS (Encoder Control Supervisior)



User-defined evaluation of the encoder status.

Amount: c. f. "Overview of safety modules"

Function: Muting of encoder error

RESET function: Violation of the permitted monitoring range is stored and requires a RESET confirmation. The confirmation can alternatively be made via:

- RESET function in the group of the input elements
- Function button on the front side of a basic unit
- Input of fieldbus reset

NOTICE: This function can considerably influence the safety of an application. It must be ensured that no safety-critical situations occur due to the use of the ESC function!

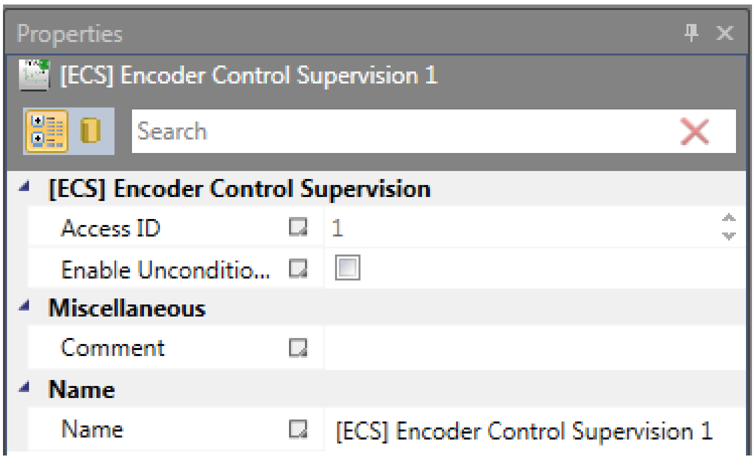
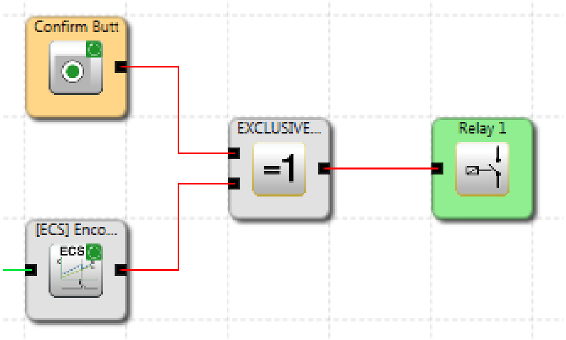
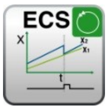


Figure 212 properties window of ECS function

Example of the use of the ECS function:



ECS (Encoder Control Supervisior)



User-defined evaluation of the encoder status.

Amount: c. f. "Overview of safety modules"

Function: Muting of encoder error

RESET Function: The violation of the permissible monitoring is stored and requires a RESET acknowledgement. This is done alternatively via:

- RESET function in the Input elements group
- Function button on the front side of a basic module
- F-Bus Reset input

NOTICE: This function can influence the safety of an application in a considerable way. It must be ensured that no safety-critical situations arise through the use of the ECS function!

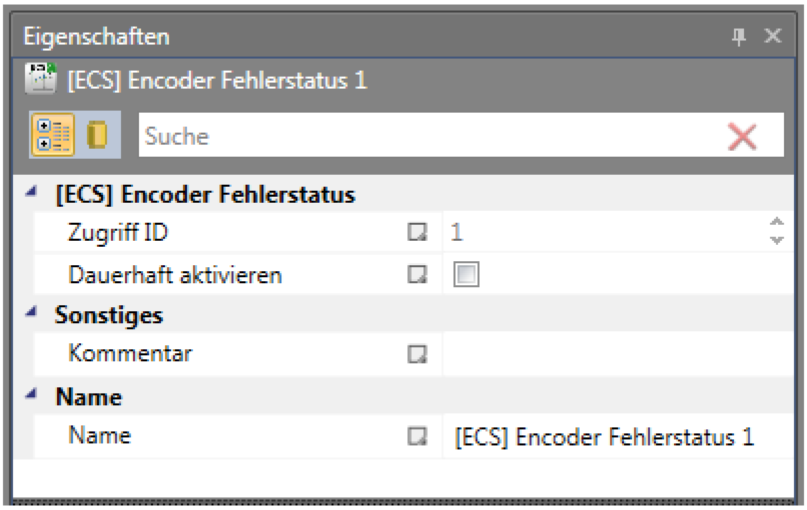
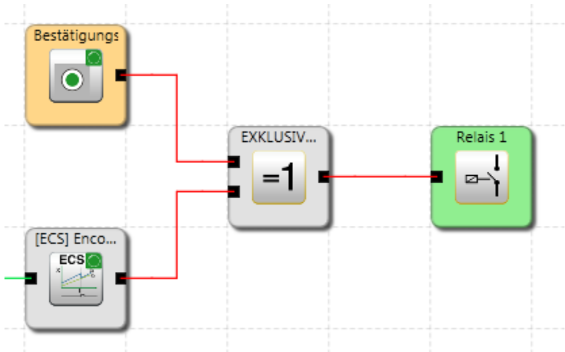


Figure 213 Properties of the ECS function

Example of the use of the ECS function:



EOS (External Offset Setup)

Setting of the encoder positions to the configured position value

Amount: c. f. "Overview of safety modules"

Access-ID: identification of the function element

Function: Calculation of a shift value for position encoders for a position encoder on the basis of a settable setting postion on the basis of the current encoder position. By the activation of the EOS function, the current position value is adjusted to a preset paranetrizable value by the revaluation and the setting of the shift value.

Input: position signal X by the encoder interface.
Selection module / axis and encoder

RESET function: no reset necessary

Operation: The activation of this function starts with a rising edge at the input of this function.

The EOS function can be used only afterthe activation position processing and after setting of the absolute encoder (e. g. SSI encoder) in the selected encoder channel.

Parameters: Then, this module can be set in the function block. For this purpose, the sensor channel is selected via the axis and the encoder number. The specification of the preset value happens in the physical unit that has been selected for the measuring distance.

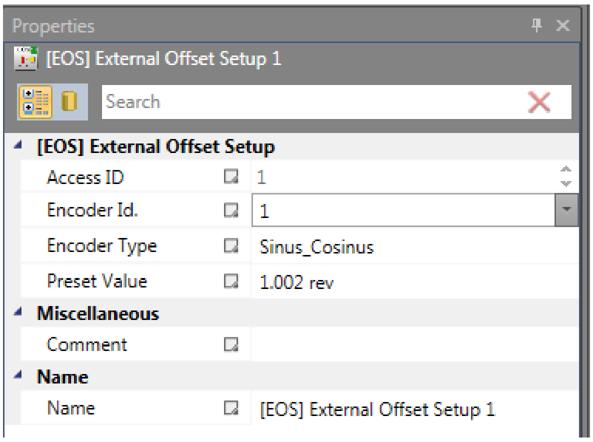


Figure 214 properties window of EOS function

Encoder – ID

selection of the sensor that is connected to encoder A (= 1) or to encoder B (= 2).

Encoder type

selection of encoder type

Preset value

preset value (set position) for the selected encoder.

NOTICE:

At the most one EOS function can be used for the absolut encoder value. An operational activation of the EOS function must be excluded. The function serves for maintenance and servicing. This must be ensured by the selection of appropriate equipment. Appropriate equipment is e. g. key-operated switches that are accessible only for qualified maintenance and service personnel.

Suitable organisational measures must be taken to grant the continuation of the physical position on the axis according to the set position.
The calculated shift value must be saved in a voltage-protected manner.

For correct functioning, the ECS function must be activated while the EOS function is used.

4.12.5.5. Elements of the global network

The global network elements include an SMMC output block and the corresponding input elements.

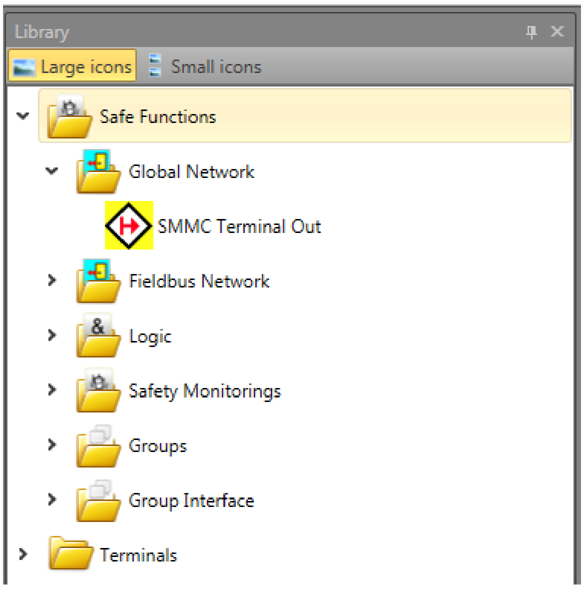


Figure 215 Library window "Global network"

SMMC “Terminal Out“[Connection point output]

This block is an output of the SMMC. Every device can write 16 bits as output on SMMC. By the connection to SMMC, these bits are defined as “Terminal out”.

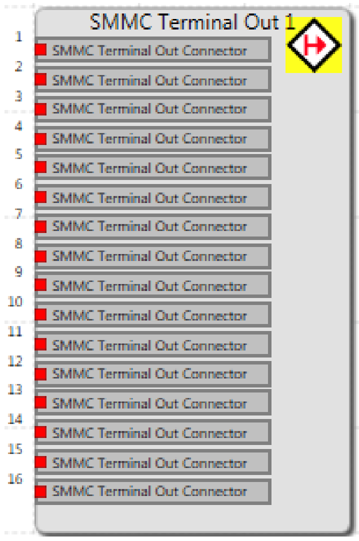


Figure 216 Terminal Out block of SMMC

The user can change the name of every output connection used.

SMMC "Terminal In" [input connection point]

This block maps the input of the SMMC. It is freely available after the user has configured the corresponding "Terminal out" [connection off] in a random functional scheme.

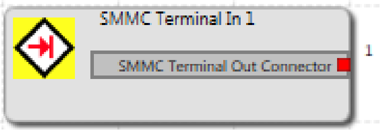


Figure 217 Terminal In block of SMMC

Number of bits: number of available bits for the input. The number must be greater or equal to 1 and smaller then or equal to 16.
Device: selection of SMMC device.
Bits: The range of the available bits depends on the selected number of bits.
Name: The user can choose a name for the connection.

4.12.5.6. Elements of the fieldbus network

In the library, the elements of the fieldbus network are shown in the folder of the fieldbus network. The elements of the fieldbus network are shown in the library in the fieldbus folder, if the "Funktionsplan" [functional scheme] tab has been selected. These elements are shown in the screenshot below. The displayed functions depend on the selected device and from the connection. You find a description of the elements in chapter 4.11.2.3.

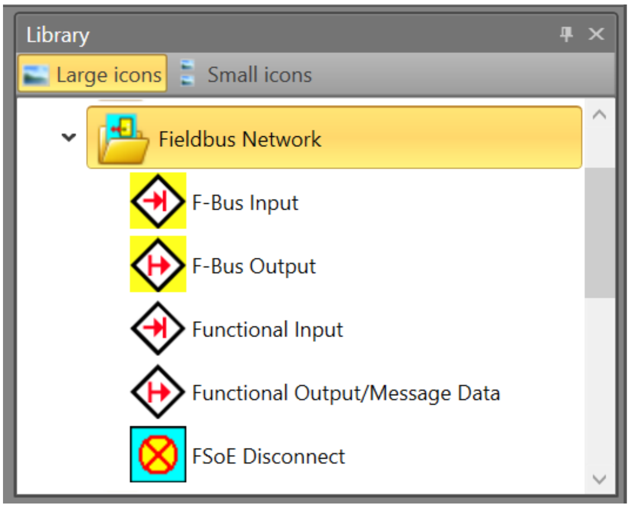


Figure 218 Library window "Fieldbus network"

4.12.5.7. Connections

These elements are intended to prove a clear representation of the function charts. These elements provide “Input connection points / Output connection points”. Afterwards, the connections between the elements are drawn.

Terminal In [input connection point]

These elements provide connections for outputs. The reference points of the connection points are created automatically. If a block for the input connection point is chosen, also the corresponding output connection points are chosen if a new output is chosen. When the input with the corresponding number has been chosen, the output with the corresponding number can be added. For identical multiport connection points, drag the connections out of the browser window.

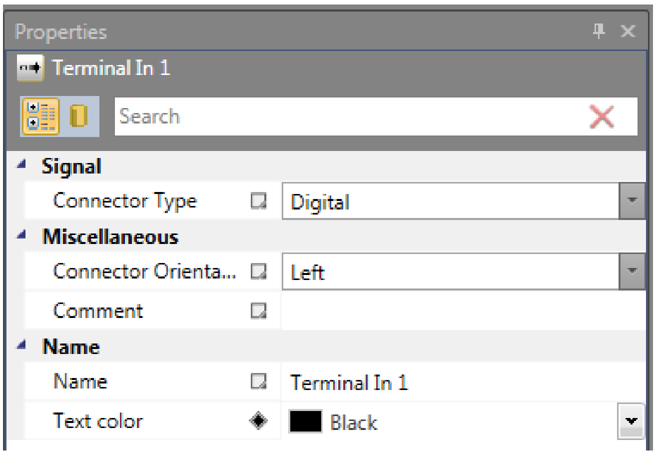


Figure 219 Properties window of the connections

Connection number: identification number of the terminal point.

NOTICE:
When deleting connection point elements that are referenced by output blocks, the user receives a warning. If this warning is confirmed, the dependent function block is deleted. If no corresponding output block has been indicated for marking, this caused a compiler error. “Nichtreferenzierter “”Eingestellter Anschlusspunkt “Baustein [“Not referenced “Set connection point “block]“.

Tip: Use the comment area. The input commentary facilitates the assignment of elements.

Terminal Out [output connection point]

This element facilitates the continuation of a signal that leads to an "Terminal In" [Set connection point] block. Thus, these elements can only be inserted after a set connection point has been defined.

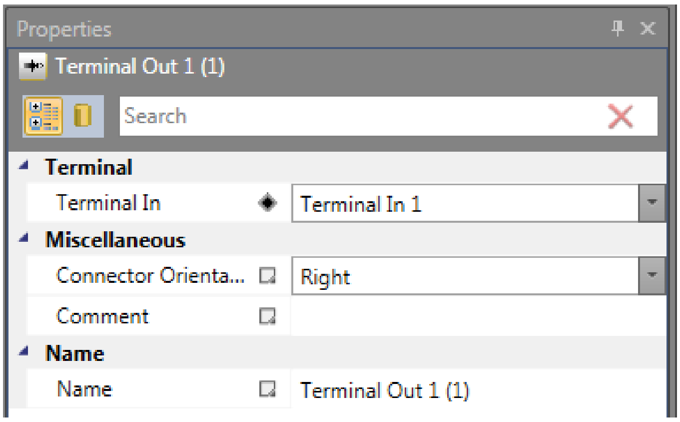


Figure 220 Properties window of Terminal out

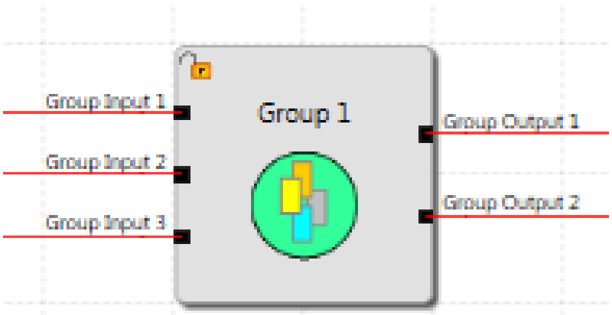
Terminal In: identification code of the set connection number.

Connecting direction: selection of the direction of the connection point on work surface (canvas).

NOTICE:
As this element refers to the set connection point, the comment for this element is shown.

4.12.5.8. Groups

Function groups connect several function blocks to a superior logic structure. This matching block group is created within the function group and is connected via this function block.



By this grouping, the function block diagram gets a clearer structure, and permits the creation of an individual function library via the export function / the import function.

4.12.5.8.1. Creating a group

1. <u>Creating a group module</u>

Creating a unique function group
The library window contains an element for a new group. To add a new group, draw the symbol "New group" from the library, and insert it in the work surface of the function chart. The created group does not have an input-output interface.

2. <u>Creating a function group from the selection</u>
   The size of the group is determined with the cursor:
   - Use the left mouse button to position the cursor in the upper left corner of the group frame and press and hold the button.
   - Afterwards, with the left mouse button pressed, drag the mouse pointer in the dsired lower edge of the group section.
   - Click on the selection with the right mouse button, and create a new group. This group is inserted in the group frame, and whose tab "Group" and be opened for editing.
   - The block types that cannot be included in a group are filtered. An info display shows the filtered blocks.

3. <u>Adding function blocks to the group</u>

The group's work surface can either be selected with a double click in the group frame or via the group sheet of a browser tree. In this section, function blocks can be inserted, shifted or deleted. If the group has not been deactivated, the blocks are automatically accepted in the group. If the function blocks are accepted, they also show the function block number. As long as the group block is activated, function modules from the section of the group frame can be added or deleted.

Please note:

- Function blocks cannot be added by shifting the group block. The modules must be shifted into the group sheet.
- In the group, only logic modules and monitoring modules can be accepted.
- Input and output modules and preset elements, e. g., signal lists, analogous modules or encoder modules are not permitted.
- In case of modules with consisting connections, it may happen that a connection protrudes over the group frame. This is permitted under no circumstances, and the connection is deleted automatically.

If modules that have already been connected shall be added to a group by shifting, you should proceed as follows:

1. Shift the group block over the function block. The affected connections must all be situated in the group file.

2. Choose the modules, and shift them in the group module by one grid position

   The following block types cannot be part of a group. They are filtered, when the modules are shifted to the frame section:
   - Input modules
   - Output modules
   - All function blocks preceeding in the functional scheme (e. g. encoders, analog modules, I/O)
   - Signal channel modules

   The maximum is determined by the size of the sheet.
   The export to the library function is shown by a click on the grop with the right mouse button.

3. Adding an interface to an input / output

   You can insert a block for a group interface by dragging a group output / a group in the library of the group interface and inserting it into the corresponding group block (or in the group in the functional scheme). After you have added a block to a group, the group interface is added.
   For further information, cf. the chapter "Group interface"

4. Creating conncetions

   -cf. the chapter "Circuit".

5. Connecting the group interface

   Via the above-mentioned interface, the function blocks within a group can only be connected to the function elements outside the group frame. In the interface, the connection type can be selected arbitrarily. However, the same connection constellation is needed for the import of the group into a different function block. The interface blocks enable a description of both, input and output of rhe function group. The description should be documented in the comment.

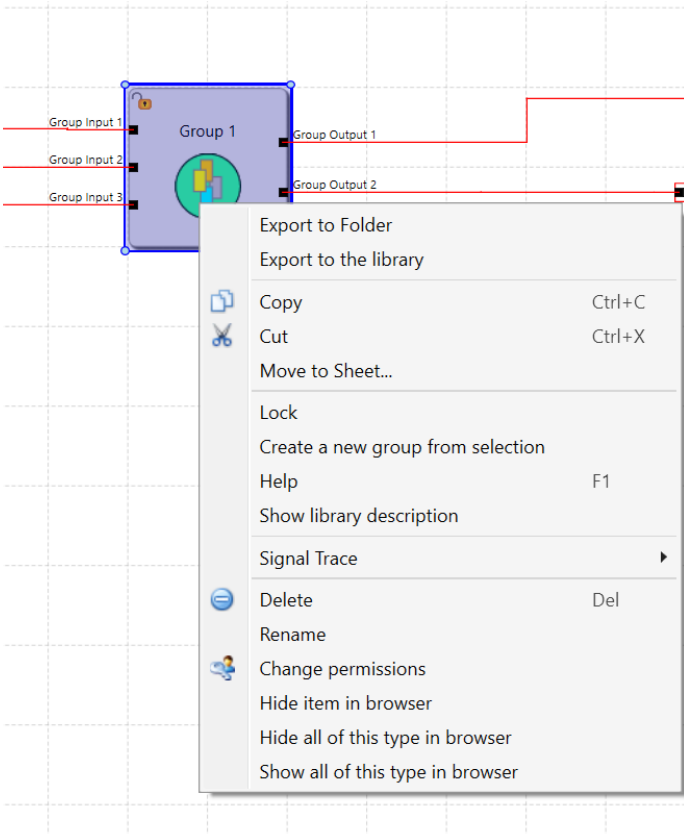
Tips:

- If possible, during which the group(s) remain in the active status should be as short as possible.
- Activate as few groups as possible in the functional scheme.
- Do not shift any groups in the functional scheme.
- If possible, only edit one group in the functional scheme.
- Deactivate the groups before saving .
- Create connections as late as possible.

4.12.5.8.2. Setting the group administration

When you click on the group with the right mouse button, the context menu for lock management appears. With this function, the block administration of the frame is deactivated, and the blocks are connected to the group:

- Module cannot be deleted from the group any more, but parameter configuration is still permitted.
- When a group frame is deleted, also all group blocks are deleted.
- No new blocks can be added to the group.



The group status “disabled” is indicated by the lock symbol in the upper left hand corner.

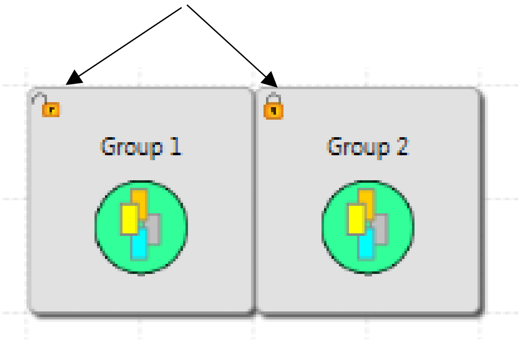


Figure 221 Group blocked / unblocked

When a new group is added, in the beginning the lock function is not set.

4.12.5.8.3. Export / Import of a function group

A right mouse click on the group shows the export to the library. The modules of can be exported to the library. An exported group can be imported into a different group sheet. This export allows you to create a library. This makes it possible to create a library with preset function groups that can be imported into new projects. In the library window, the group cannot be renamed. The user can change the image of the exported groups.

Only by means of an already inserted group frame can a function group be imported via the library.
Import includes the verification of the sensor configuration and of the already existing resources in the function chart. The group can only be imported, if resources for all modules are available. The necessary sensor settings must be checked, especially in case of position-dependent monitoring modules. If a resource is not available any more, an error message appears.

In case of resource errors, you must make sure that the sensor settings correspond to the requirements of the group. This applies especially, if position-dependent modules have been used in the function groups (SEL, SLP, SCA).

A created function group can be imported into new projects. The group to be imported must first be saved as a zip file in a folder. There are two possibilities:

1. via library – right mouse click on the group and select "Export to Folder“

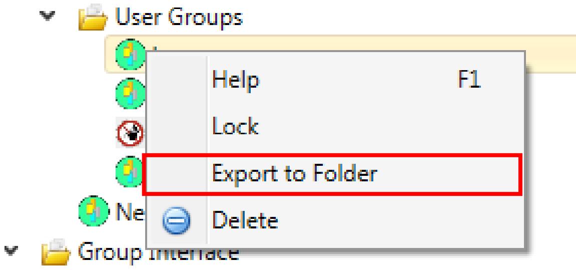


Figure 222 Exporting a group to folders via the library

2. Directly in canvas (Work surface) – right mouse click on the group to display the "Export to Folder" context menu.

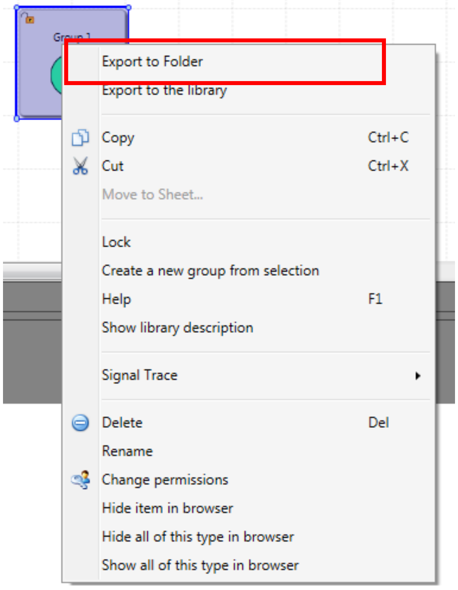


Figure 223 Export of a group via context menu of the group

3. After successful saving, select the respective "Group" file in the new project via the Start menu > "Import Group" button and add it or existing ones.

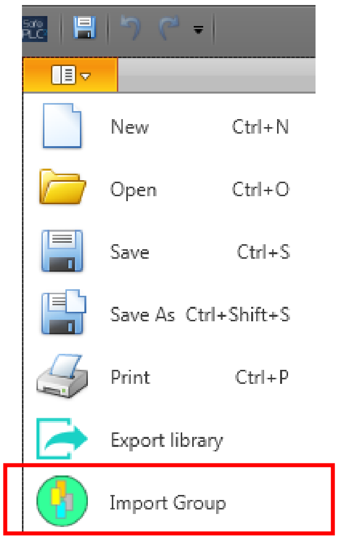


Figure 224 Mainmenu „Import Group" in new project

4.12.5.8.4. Group interface

The blocks of the group interface are the interface from the function group to the elements outside the group. Connections to function blocks outside the group are only possible via the group interface. You can insert a block for a group interface by dragging a group input / a group output into the the library of the group interfaces and inserting it in the respective group block (or in the group on the functional scheme). After you have added a block to a group, the group interface is added. In the interface modules, the connection type can be selected arbitrarily. However, the same connection constellation is necessary to import the group into a different functional scheme.

<u>Connection type</u>: This option can be used to set elements of the group inputs and group outputs and to avoid improper assignments.

<u>Example</u>: The axis of the connection type is connected to the block of the group interface. In the user mode, the group blocks must always be connected to the same connection type.

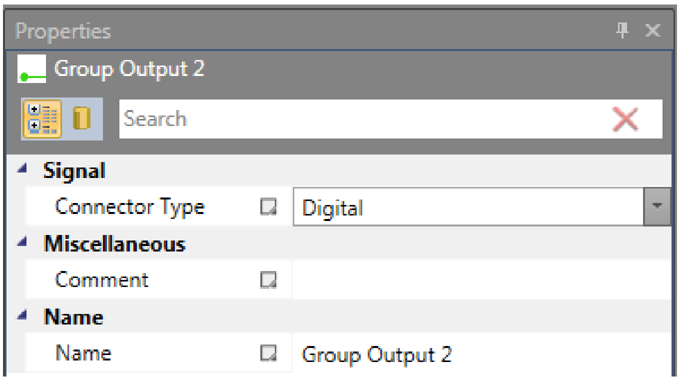


Figure 225 Properties "group output "

With this setting, the connection properties of the block can be determined as group input.

- Group input

 This element shows the connection of the function blocks outside the group to external group elements. If possible, the block shall be situated on the left side of the group area. The output connection must be forwarded within the group.

- Group output

 With this symbol a result is transmitted from the group to external elements of the function block diagram.
Input blocks / output blocks can only be deleted in the group sheet.

5. Commissioning

5.1. Start-up sequence

In case of a correct run, after every reset the following phases are completed. These phases are displayed on the front-mounted seven-segment display

7 segment display	Mode	Description
„1“	STARTUP	Synchronization between the two processor systems, and check of the configuration data / the firmware data
„2“	SENDCONFIG	Distribution of the configuration data / the firmware data, and second check of these data. Afterwards range checkof the configuration data.
„3“	STARTUP BUS	Initialization of a bus system (if available).
„4“	RUN	Standard mode of the system All outputs are connected according to the current logic state.
„5“	STOP	In the stop mode, parameter data and programmedata can be loaded externally.
„6“	Fehler [Error]	Error mode of the assemblyt. All outputs disconnected. Error can only be reset via ON / OFF oft he unit.
„7“	Alarm	Alarm mode of the assembly. All outputs disconnected. Alarm cn be reset via digital output or via front-side acknowledgement button.
„8“	Lokalbetrieb [Local mode]	Local mode of the assembly. Normal mode without network connection.

5.2. RESET behaviour

The RESET function differentiates into a start-up function after restoration of voltage = general reset and a status reset / an alarm reset = internal reset. The latter is triggered via the front-side button or an appropriately configured input = RESETelement with enabled “alarm reset” function. The following chart provides an overview about the RESET functions and their effect. The timing behavior can be taken from the SCU installation manual.

5.3. LED display

The SCU assembly has 2 LEDs: EC ST and RUN

LED	Colour	Display	Meaning
EC ST		Off	Init
	Green	Blinking	PRE-OPERATIONAL
	Green	Single Flash	SAFE-OPERATIONAL
	Green	Off	OPERATIONAL
	Red	Off	No errorr
	Red	Blinking	Invalid configuration
	Red	Single Flash	Ocal error
	Red	Double Flash	Watchdog Timeout
RUN	-	Off	No power supply voltage
	Orange	Blinking	Control is in startup phase or STOP or firmware update is carried out.
	Orange	Pernanently lit	Local mode of the assembly (without network connection)
	Green	Blinking	Control function correct; Application runs but is not (yet) validated)
	Green	Permanently lit	Control function correct. Application runs and is validated.
	Red	Blinking	Alarm (application error) – the assembly is in the in a safe operating state. Error status can be reset.
	Red	Permanently lit.	Fatal error –the assembly is in the safe operation mode.

NOTICE:
for all operating states except RUN the outputs are passivated, i. e. Switched off safely. In “RUN” state, the state of the outputs is dependent on the implemented plc program.

5.4. Parametrization

Parametrisation is carried out via the program SafePLC^2.
To be able to send the data to the assembly, a program adapter is needed. Before use, the driver of the program adapter must be installed for the first time.
The description of parametrisation must be taken from the program manual.

5.5. Regular functional test

To guarantee the safety of the assemblies, once per year a functional test of the safety functons must be carried out. For the functional test, once a year the blocks used in the parametrisation (inputs, outputs, monitoring fnctions) and logic modules must be tested with respect to their functioning, or rather, with respect to their shutoffg getestet werden.

5.6. Validation

To guarantee the functioning of the implemented safety function, the user must test the parameters and the links. After commissioning and parametrisation have been completed, the user must test these functions. This test must be documented. Documenting is supported by the validation assistant (c. f. the chapter "Safety-related verification").

5.6.1. Generating a validation report

In SafePLC², a validation report can be generated with 

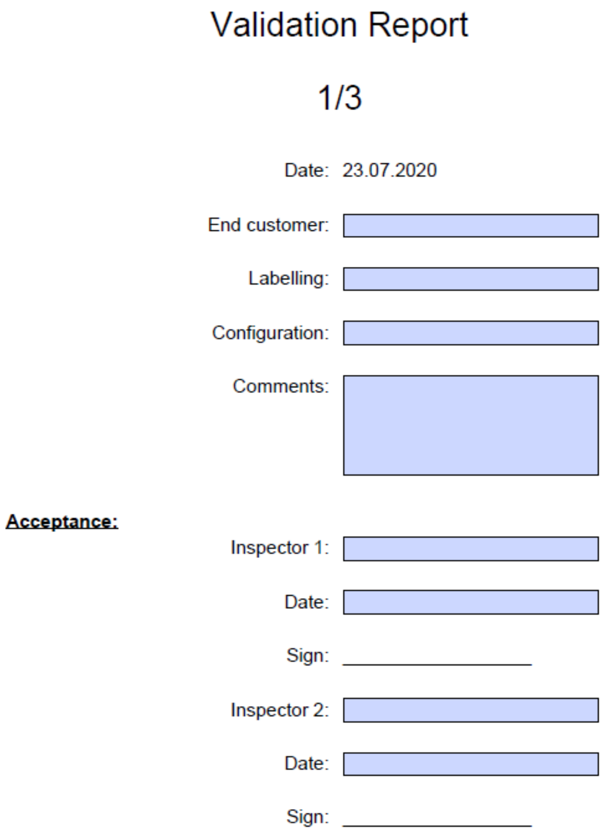
.
A *.pdf file is created.

To guarantee the functioning of the implemented safety function, the user must test the parameters and the links. After commissioning and parametrisation have been completed, the user must test these functions. This functional testing must be documented. Documenting is supported by the parametrizing software SafePLC².

On the last two pages, general details can be furnished with respect to the installation. The last page of the validation report contains the itemization of the safety-relevant inspection.

It is mandatory to carry out the safety inspections contained in the validatíon report
By his signature, the responsible tester of the safety module confirms confirms that the CRC displayed in the programming surface is has identity with the CRC defined in the SCU assembly.

On the first page, general information must be filled in.



Validation Report

1/3

Date: 23.07.2020

End customer:

Labelling:

Configuration:

Comments:

Acceptance:

Inspector 1:

Date:

Sign: ______________________

Inspector 2:

Date:

Sign: ______________________

Figure 226 Example: page 1 configuration report

Afterwards, the configured values and the user programme must be validated.

5.6.2. Validating the configuration

All set parameters must be checked and confirmed.

Example: SCA configuration data

Safe Cam (SCA)

Index	Parameter	Value	Unit	validated
SCA - 0				
	Axis	EncBox(0)Enc(0)		☐
	Speed Tolerance			
	Speed	10000	deg/s	☐
	Position monitoring			
	Lower Limit Position X1:	-10000	deg	☐
	Upper Limit Position X2:	-9000	deg	☐
SCA - 1				
	Axis	EncBox(0)Enc(0)		☐
	Speed Tolerance			
	Speed	10000	deg/s	☐
	Position monitoring			
	Lower Limit Position X1:	22000	deg	☐
	Upper Limit Position X2:	44500	deg	☐
SCA - 2				
	Axis	EncBox(0)Enc(1)		☐
	Speed Tolerance			
	Speed	9	deg/s	☐
	Position monitoring			
	Lower Limit Position X1:	-10	deg	☐
	Upper Limit Position X2:	95	deg	☐
SCA - 3				
	Axis	Robot Axis(1)		☐
	Robot	Yes		☐
	Speed Tolerance			
	Speed	40	mm/s	☐

5.6.3. Validation of the PLC program

5.6.3.1. General information

The IL programme is structured in the following way:

Index	Operator	Operand
1	LD	CBool.0
2	ST	MX.1

Index: sequential n^{of} = line in AWL program

Operator: for all permitted commands, c. f. the list "PLC commands"

Operand: for the processing code cf. the list "PLC operand". There are three operand types:

- Type1 -> Operand.x
  x: bit number
- Type2 -> Operand.x.y
  x: Index assembly / instance
  y: bit number
- Type3 -> Operand x.y.z
  x: index assembly
  y: Instance operand
  z: bit number

5.6.4. Input elements

The input elements are marked in the IL [AWL] code.

SQH	86
LD	MX.46
AND	FE0.20
ST	MX.75
SQC	55539.1

- SQH marks the start of an input element and SQC marks the end of an input element.
- SQH marks the number of the input element
- SQC marks the backup value of the input element logic.

Input elements marked with SQH/SQC must only be checked with respect to their instance and with respect to their bit location. Their logic is checked in the assembly.

5.6.5. Validation SARC

In the validation of SARC functions, the input values must be compared with the expected output values (results) in an appropriate manner.

5.6.5.1. General

SARC Code is structured as follows:

Opcode Type Dest Op1 Op2 Op3 Index

Block	Description	SARC code					
		Opcode	Type	Dest	OP1	OP2	OP3
Load	Convert integer input value to float value	LD SARC	Pos Vel CST VPos SVel	Index*) SARCMx	Index*) Type	---	---
ST	Convert float input value to integer value	ST SARC	Pos Vel VPos SVel	Index*) Type	Index*) SARCMx	---	---
ADD	Addition R = a1 + a2	ADD SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	a2 Index*) SARCMx	---

MUL	Multiplication R = a1 * a2	MUL SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	a2 Index*) SARCMx	---
DIV	Division R = a1 / a2	DIV SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	a2 Index*) SARCMx	---
SQRT	Square root R = SQRT(a1)	SQRT SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
ABS	Absolute value R = ABS(a1)	ABS SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
SIN	Sine value R = SIN(a1)	SIN SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
ASIN	ArcusSine value R = ARC SIN(a1)	ASIN SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
COS	Cosine value R = COS(a1)	COS SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
ACOS	ArcusCosine value R = ARC COS(a1)	ACOS SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
TAN	Tangent value R = TAN(a1)	TAN SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
ATAN	ArcusTangent value R = ARC TAN(a1)	ATAN SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
NEG	Change of sign R = -1*a1	NEG SARC	---	R Index*) SARCMx	a1 Index*) SARCMx	---	---
MATST	Convert float value to SARC Merker Array	STMAT SARC	---	Result Index*) SARCMx	Index*) row Matrix	Index*) column Matrix	---
MMMUL	Multiplication of two matrices (Contains also matrix-vector multiplication)	MMUL SARC	---	Index*) result MatrixMx	Index*) Matrix 1 in MatrixMx	Index*) Matrix 2 in MatrixMx	

MATRIX	Creates an empty matrix in the flags array	CRMAT SARC	---	Index*) MatrixMx	Index*) row MatrixMx	Index*) column MatrixMx	
	Changes a specific entry of a matrix to a flag value. (auxiliary function, cannot be called by the user)	CHMAT SARC	---	---	New value Index*) SARCMx	Index*) MatrixMx	Index*) value in Matix
	Changes a specific entry of a matrix to an operand value. (auxiliary function, cannot be called by the user)	CHMAT2 SARC	---		New value (Config)	Index*) MatrixMx	Index*) value in Matix

Note *) Index is 0-based

HINWEIS:

- Matrix has its own flag area (MatrixMx) and is not stored in the SARC flag (SARCMx). I.e. diagnosis only possible via process image.
- Individual matrix values can be transferred to the SARC SARCMx via code "MATST".
- user must ensure that the calculation is performed correctly and with sufficient accuracy according to his requirements.

5.6.5.2. Online Diagnose

All calculated values (SARC flags) can be checked online in the functional scheme.

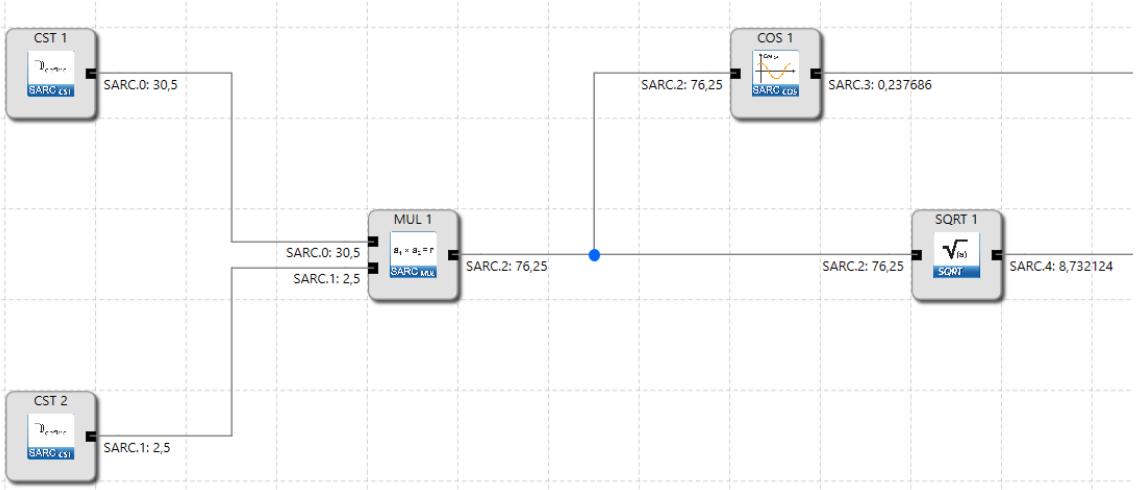


Figure 227 View calculated SARC flags in functional scheme

5.6.6. Monitoring functions

For the monitoring functions, there are two way of processing that generate additional IL code.

5.6.6.1. Cascading

The following functions are cascadable:

- SEL, SLS, SOS:

  The following code is generated:
  Example SOS:

```
LD      MX.x
ST      DriveSOS_EN0.0.0
NOT
AND     MX.y
OR      DriveSOS0.0.0
ST      MX.y
ST      DriveSOS0.0.0
```

- SCA, SWM
  Special case

```
LD      MX.x
ST      DriveSCA_EN0.0.0
NOT
OR      DriveSCA0.0.0
ST      DriveSCA0.0.0
```

If "enable unconditioned" is used, the following code is generated:
Example SLS:

```
S1      DriveSLS_EN0.0.0
```

With all other monitoring functions, simple LD commands and simple ST commands are used.

5.6.6.2. Axis groups

Monitoring functions can be combined using axis groups. Basically, one monitoring block is one monitoring block is provided for every axis in an axis group. In the IL code, the individual results are summarized (AND).

- SLS

  Example: SLS (axis group with 3 axes)

```
LD MX.63
ST DriveSLS_EN0.5.0
ST DriveSLS_EN0.6.0
ST DriveSLS_EN0.7.0
NOT
AND MX.64
ST MX.64
LD DriveSLS0.7.0
AND DriveSLS0.6.0
AND DriveSLS0.5.0
OR MX.64
ST MX.64
ST DriveSLS0.5.0
```

Attention:

- Special function SCA:

  Example: SCA mit 2 axes:

```
LD MX.58
ST DriveSCA_EN0.4.0
ST DriveSCA_EN0.5.0
NOT
ST MX.71
LD DriveSCA0.5.0
AND DriveSCA0.4.0
OR MX.71
ST DriveSCA0.4.0
```

- Special function SRS:

Result: axis group is already edited in the function. Thus, the result of the smallest entity is used.

Example: SRS axis group with 6 axes

```
LD MX.35
ST DriveSRS_EN0.0.0
ST DriveSRS_EN0.1.0
ST DriveSRS_EN0.2.0
ST DriveSRS_EN0.3.0
ST DriveSRS_EN0.4.0
ST DriveSRS_EN0.5.0
```

```
//---------------
LD DriveSRS0.0.2
ST FA0.19
//---------------
LD DriveSRS0.0.1
ST FA0.20
//---------------
LD DriveSRS0.0.0
ST FA0.21
//---------------
LD DriveSRS0.0.3
ST FA0.22
```

Figure 228 Special function SRS

NOTICE:
special function SRX: Result: axis group is differently linked.

<u>Example:</u> SRX axis group with 6 axes

```
LD MX.44
ST DriveSRX_EN0.0.0
ST DriveSRX_EN0.1.0
ST DriveSRX_EN0.2.0
ST DriveSRX_EN0.3.0
ST DriveSRX_EN0.4.0
ST DriveSRX_EN0.5.0
LD MX.49
ST DriveSRX_EN0.0.1
ST DriveSRX_EN0.1.1
ST DriveSRX_EN0.2.1
ST DriveSRX_EN0.3.1
ST DriveSRX_EN0.4.1
ST DriveSRX_EN0.5.1
LD MX.50
ST DriveSRX_EN0.0.2
ST DriveSRX_EN0.1.2
ST DriveSRX_EN0.2.2
ST DriveSRX_EN0.3.2
ST DriveSRX_EN0.4.2
ST DriveSRX_EN0.5.2
LD MX.51
ST DriveSRX_EN0.0.3
ST DriveSRX_EN0.1.3
ST DriveSRX_EN0.2.3
ST DriveSRX_EN0.3.3
ST DriveSRX_EN0.4.3
ST DriveSRX_EN0.5.3
LD DriveSRX0.5.0
AND DriveSRX0.4.0
AND DriveSRX0.3.0
AND DriveSRX0.2.0
AND DriveSRX0.1.0
AND DriveSRX0.0.0
ST DriveSRX0.0.0
LD DriveSRX0.5.1
OR DriveSRX0.4.1
OR DriveSRX0.3.1
OR DriveSRX0.2.1
OR DriveSRX0.1.1
OR DriveSRX0.0.1
ST DriveSRX0.0.1
```

Figure 229 SRX function with 6 axes

5.7. Safety-related verification

To ensure the implemented safety functions, the user must check and document the parameters and links after commissioning and parameterization. This is supported by the SafePLC2 parameterization software. (see additionally programming manual "HB-37500-820-10-xxF-EN Programming manual SafePLC2").

5.7.1. The validation process

General information about the plant can be given on the first two pages. On the following pages of the validation report, all the functions used are printed with their parameters as an itemization of the safety-related test.

- Check index for consecutive numeration
- Check command available in list "PLC commands"
- Check operand available in list "PLC operand" and type correct
    - entity: Slave number
    - entity cf. "Process image"
    - Bit number cf. "Process image"
- Check additional IL code monitoring functions
- Functional test
- Lock configuration data on assembly after successful test

After the transfer of the configuration data and the program data to the SCU assembly, the status LED flashes in yellow. This indicates that the configuration data have not yet been validated.
By confirmation of "LOCK CONFIGURATION" at the end of the validation dialog, the data are marked "Validiert" [validated], and the LED flashes GREEN.

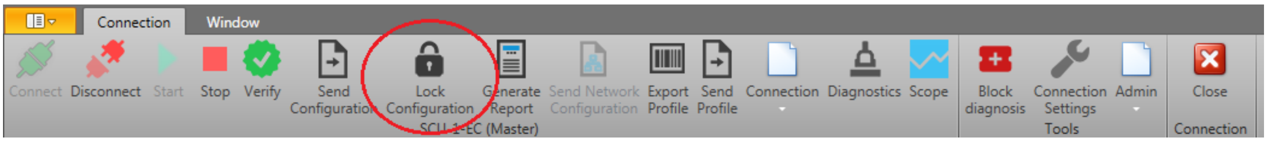


Figure 230 Tab "connection" before data are marked as validated

6. Failure and troubleshooting

If the assembly should not work properly, it autonomously passes into the safe status and indicates the fault status via LED (cf. installation manual SCU, chapter LED display).
Please, check the displayed error code (7-segment display) by means of the SCU error code (error codes and measures).
If the error cannot be successfully emitted, please contact the LED manufacturer (cf. "Manufacturer")

7. List of abbreviations

Abbreviation	Meaning	Comment
AC	Alternating Current	Alternating current
IL	Instruction List	List of commands within the assembly
BBH	Manufacturer of assemblies	
CRC	Cyclic Redundancy Check	Cyclic checksum calculation
DC	Diagnostic Couverage	Diagnostic Coverage
BG	Berufsgenossenschaft	
Cat.	Category according to EN 13849-1	Architectural category
CE	Communauté Européenne	Symbol of conformity with relevant EC directions
CLK	Clock	Clock
CPU	Central Processing Unit	Central Processing Unit
DC	Direct Current	
DIN	Deutsches Institut für Normung	German Institute for Standardization
[EMU]	Emergency Monitoring Unit	Safety function
[EDM]	External Device Monitoring	Safety function (for SCU-/SIO devices)
[ELC]	Emergency Limit Control	Safety function
EMV [EMC]	Elektromagnetische Verträglichkeit	Electromagnetic compatibility
EN	Europanorm	
EtherCAT	EtherCAT (Name)	Data portocol
FSoE	Fail Safe over EtherCAT	Safe data transmission via EtherCAT protocol
GND	Ground	Ground potential 0 VDC
H / HISIDE	High Side	Output switching to plus 24 VDC
HW	Hardware	
I.	Input.	
IO	Input Output	Digital Input/Output
IP	International Protection	Protection class according to standard
ISO	Internationale Organisation für Normung	
LED	Light Emitting Diode	Light emitting diode
LOSIDE	Low Side	Output switching to GND 0 VDC
O.	Output.	

Abbreviation	Meaning	Comment
[OLC]	Operational Limit Control	Safety function
PAA	Prozessabbild der Ausgänge	Process image of the outputs
PAE	Prozessabbild der Eingänge	Process image of the inputs
PELV	Protective Extra Low Voltage	Protection low voltage
PLC	Programmemable Logic Controller	
POR	Power On Reset	Reset process
[PSC]	Position Supervision Control	Safety function
SafePLC²	Program for the safe programming of PLC	Programming interface by BBH for BBH devices
SCU	Safe Control Unit	FSoE master module for processing encoder and input/output data
SDDC	Safe Device to Device Communication	Safe communication from component to component
SDU	Safe Drive Unit	FSoE-Slave unit to record encoder values
SELV	Safe Extra Low Voltage	Safe (fused) low vollage
SIO	Safe IO	FSoE-Slave unit to record digital inputs /digital outputs
SMMC	Safe Master to Master Communication	Safe communication from Master to Master
SSB	Safe Sensor Box	Assembly Baugruppe FSoE slave tp record encoder data from 6 encoders
SSI	Synchronous Serial Interface	Synchrone serielle Schnittstelle
SW	Software	
SWM	Safe workarea monitoring	Safety function “safe workspace”
T.	Pulse output.	Pulsed signal
VDE	Verband der Elektrotechnik	Registered Association of the Electrical, Electronic and Information Technology

8. Appendix

8.1. CoE object list

Obj.ID	Parameter	Amount	Byte	Type	Unit	Access	Process config	Description
0x1	**Type label EtherCAT**	1						
0x1000	Device type	1	4	UNSIGNED32		R		Device type
0x1008	Manufacturer device name	1	24	STRING		R		Device type
0x1009	Manufacturer hardware version	1	32	STRING		R		HW version
0x100A	Manufacturer software version	1	32	STRING		R		FW version
0x1018	Identity object	1				R		
0	Number of subindizes	1	1	UNSIGNED8		R		Number of subindizes
1	Vendor ID	1	4	UNSIGNED32		R		EtherCAT Vendor ID
2	Product code	1	4	UNSIGNED32		R		Product code of SCU
3	Revision number	1	4	UNSIGNED32		R		Revision number of SCU
4	Serial number	1	4	UNSIGNED32		R		Serial number of SCU
0x6	**Type label Device**	1						
0x60B0	**Functional Outputs**	32	1	UNSIGNED8		R		Functional SCU outputs
0x7	**Type label Device**	1						
0x70B0	**Functional Inputs**	18	1	UNSIGNED8		R		Functional SCU inputs
0x9	**FSoE Connection Communication Parameter**	1						
0x9nn1								
1	**Version**	2	1	STRING		R		Supported FSoE Version = 01
2	**FSoE Slave Adress**	1	2	UNSIGNED16		R		FSoE Slave Adress
3	**Connection ID**	1	2	UNSIGNED16		R		FSoE Connection ID
4	**Watchdog Time**	1	2	UNSIGNED16		R		FSoE Watchdog Time
5	**Unique device ID**	6	1	STRING		R		6 Byte unique device ID (0 if not used)
6	**Connection Type**	1	1	ENUM		R		1: Slave connection 0: Master Connection
7	**ComParameter Length**	1	2	UNSIGNED16		R		Number of bytes of the communication parameter
8	**ApplParameter Length**	1	2	UNSIGNED16		R		Number of bytes of the application parameter

9	**SRA CRC**	1	4	UNSIGNED32		R		CRC checksum of the safety related application parameter set
0xA	**Connection diagnosis**	1						
0xAnn0								
1	Connection state	1	2	ENUM[16]		R		100 = Reset 101 = Session 102 = Connection 103 = Parameter 104 = Data 105 = Failsafe
2	Connection diagnosis	1	2	UNSIGNED16		R		Diagnosis bits of FSoE Connection Bit0..3: 0 = no error 1 = wrong command 2 = unknown command 3 = wrong connection ID 4 = wrong CRC 5 = watchdog expired 6 = wrong FSoE address 7 = wrong data 8 = wrong communication parameter length 9 = wrong communication parameter 10 = wrong application parameter length 11 = wrong application parameter 12 = unexpected session command 13 = if failsafe data are communication error: failsafe data received Bit 4: 0 = error in FSoE Master 1 = error in FSoE Slave Bit 5: 1 = FSoE slave sends no process data Bit 6: 1 = connection in startup Bit 7: 1 = FSoE Master sends no process data (FSoE Master in failsafe)

8.2. Functional Input

Structure of functional Input

8.2.1. SCU-x-EC/NM

Byte	Bit	
Byte 0	0..7	Functional Input (0..7)
Byte 1	0..7	Functional Input (8..15)
Byte 2	0..7	Functional Input (16..23)
Byte 3	0..7	Functional Input (24..31)
Byte 4	0..7	Functional Input (32..39)
Byte 5	0..7	Functional Input (40..47)
Byte 6	0..7	Functional Input (48..55)
Byte 7	0..7	Functional Input (56..63)
Byte 8	0..7	Functional Input (64..71)
Byte 9	0..7	Functional Input (72..79)
Byte 10	0..7	Functional Input (80..87)
Byte 11	0..7	Functional Input (88..95)
Byte 12	0..7	Functional Input (96..103)
Byte 13	0..7	Functional Input (104..111)
Byte 14	0..7	Functional Input (112..119)
Byte 15	0..7	Functional Input (120..127)
Byte 16	0..7	Functional Input (128..135)
Byte 17	0..7	Functional Input (136..143)

8.2.2. SDU-x

Byte	Assignment
Byte 0	Logic data (Bit ID: 1..8)
Byte 1	Logic data (Bit ID: 9..16)
Byte 2	Logic data (Bit ID: 17..24)
Byte 3	Logic data (Bit ID: 25..32)
Byte 4	SD-Gateway - Command
Byte 5	SD-Gateway - Address
Byte 6	SD-Slave 1 – Request/call
Byte 7	SD-Slave 1 - Reserved
Byte 8	SD-Slave 2 - Request/call
Byte 9	SD-Slave 2 - Reserved
…	…

Byte 66	SD-Slave 31 - Request/call
Byte 67	SD-Slave 31 - Reserved

8.3. Functional Output

Structure of functional outputs

8.3.1. SCU-x-EC/NM

Byte	**Bit**	**„Run“ mode (2, 3, 4, 8)**	**Error case (A, F)**
DEBUG 0	0..3	SCU mode 1, 2, 3, 4, 5, 6 = FatalError, 7 = Alarm, 8	
	4	0x1 (define)	
	5..7	Alive counter (3 Bit)	
DEBUG 1	0..7	0	
DEBUG 2	0..7	0	
DEBUG 3	0..7	0 = no errorr Error code low byte	
DEBUG 4	0..7	0 = no error Error code high byte	
ByteOut 0	0..7	Functional Output (0..7)	
ByteOut 1	0..7	Functional Output (8..15)	
ByteOut 2	0..7	Functional Output (16..23)	
ByteOut 3	0..7	Functional Output (24..31)	
ByteOut 4	0..7	Functional Output (32...39)	
ByteOut 5	0..7	Functional Output (40...47)	
ByteOut 6	0..7	Functional Output (48...55)	
ByteOut 7	0..7	Functional Output (56...63)	
ByteOut 8	0..7	Functional Output (64...71)	
ByteOut 9	0..7	Functional Output (72...79)	
ByteOut 10	0..7	Functional Output (80...87)	
ByteOut 11	0..7	Functional Output (88...95)	
ByteOut 12	0..7	Functional Output (96...103)	
ByteOut 13	0..7	Functional Output (104...111)	
ByteOut 14	0..7	Functional Output (112...119)	
ByteOut 15	0..7	Functional Output (120...127)	
ByteOut 16	0..7	Functional Output (128...135)	

The bits of the device status show the status of the control. The statuses 1-5 are output analogously on the 7-segment display. Status 6 indicates an error, status 7 an alarm.

NOTICE:

The meaning of the error codes in decimal representation can be taken from the HB-37500-813-02-xxF EN Error list SCU:

8.3.2. SDU-x

Structure of the total frame:

Total size diagnostic data: always 128 bytes, 16 bytes of which can be used for diagnostics.

Byte	Bit	„Run" mode (2, 3, 4)	Error case (A, F)
Byte 0	0...3	Device status 1, 2, 3, 4, 5, 6 = Fatal error, 7 = Alarm	
	4	0x1 (always 1)	
	5..7	Alive counter (3 Bit)	
Byte 1	0...7	Logic data (Bit ID: 49..56)	
Byte 2	0...7	Logic data (Bit ID: 41..48)	
Byte 3	0...7	Logic data (Bit ID: 33..40)	
Byte 4	0...7	Logic data (Bit ID: 9..16)	
Byte 5	0...7	Logic data (Bit ID: 1...8)	
Byte 6	0..6	Logic data (Bit ID: 25.. 31)	Error-Code: high Byte
	7	„0"	„1"
Byte 7	0..7	Logic data (Bit ID: 17..24)	Error-Code: low Byte

Logic data of SDU-x

The bits of the device status show the status of the control. The statuses 1-5 are output analogously on the 7-segment display. Status 6 indicates an error, status 7 an alarm.

NOTICE:

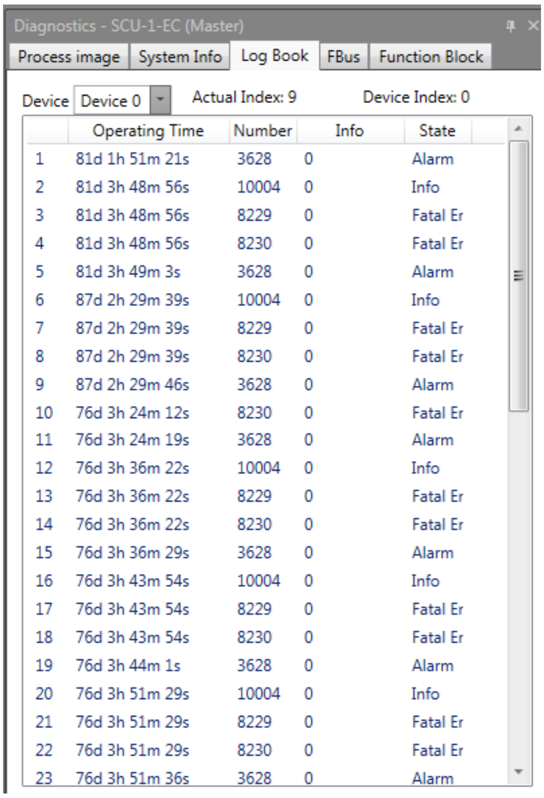
The meaning of the error codes in decimal representation can be taken from the TS-37350-130-xx-xxF error list SMX or HB-37500-813-02-xxF error list SCU-SDU modules.

The process data follow with a byte offset of 7; byte 0 of the process data is byte 8 of the total frame/input assignment.

Byte	Assignment
Byte 0	Status
Byte 1	Logic data (Bit ID: 49..56)
Byte 2	Logic data (Bit ID: 41..48)
Byte 3	Logic data (Bit ID: 33..40)
Byte 4	Logic data (Bit ID: 9..16)
Byte 5	Logic data (Bit ID: 1..8)
Byte 6	Logic data (Bit ID: 25..31) / error code
Byte 7	Logic data (Bit ID: 17..24) / error code
Byte 8	Process data (Bit: 57..64)
Byte 9	Process data (Bit: 49..56)
Byte 10	Process data (Bit: 41..48)
Byte 11	Process data (Bit: 33..40)
Byte 12	Process data (Bit: 25..32)
Byte 13	Process data (Bit 17..24)
Byte 14	Process data (Bit: 9..16)
Byte 15	Process data (Bit: 1..8)
Byte 16	Not used
…	…
Byte 127	Not used
Byte 128	SD-Gateway - Diagnosis
Byte 129	SD-Gateway - Data
Byte 130	SD-Slave 1 - Answer
Byte 131	SD-Slave 1 - Diagnosis
Byte 132	SD-Slave 2 - Answer
Byte 133	SD-Slave 2 - Diagnosis
…	…
Byte 190	SD-Slave 31 - Answer
Byte 191	SD-Slave 31 - Diagnosis

Logic and process data of SDU-x

8.4. Diagnostics Logbook



	Operating Time	Number	Info	State
1	81d 1h 51m 21s	3628	0	Alarm
2	81d 3h 48m 56s	10004	0	Info
3	81d 3h 48m 56s	8229	0	Fatal Er
4	81d 3h 48m 56s	8230	0	Fatal Er
5	81d 3h 49m 3s	3628	0	Alarm
6	87d 2h 29m 39s	10004	0	Info
7	87d 2h 29m 39s	8229	0	Fatal Er
8	87d 2h 29m 39s	8230	0	Fatal Er
9	87d 2h 29m 46s	3628	0	Alarm
10	76d 3h 24m 12s	8230	0	Fatal Er
11	76d 3h 24m 19s	3628	0	Alarm
12	76d 3h 36m 22s	10004	0	Info
13	76d 3h 36m 22s	8229	0	Fatal Er
14	76d 3h 36m 22s	8230	0	Fatal Er
15	76d 3h 36m 29s	3628	0	Alarm
16	76d 3h 43m 54s	10004	0	Info
17	76d 3h 43m 54s	8229	0	Fatal Er
18	76d 3h 43m 54s	8230	0	Fatal Er
19	76d 3h 44m 1s	3628	0	Alarm
20	76d 3h 51m 29s	10004	0	Info
21	76d 3h 51m 29s	8229	0	Fatal Er
22	76d 3h 51m 29s	8230	0	Fatal Er
23	76d 3h 51m 36s	3628	0	Alarm

Value	Description
Operating time	Operating hours meter in s
number	Number of errors [decimal display]
Info	Info [decimal display]
status	1: Fatal Error 2: Alarm 3: Info [text display]

Information number	Description
10001	DOWNLOAD Configuration
10002	Not supportedt
10003	Alarm reset
10004	POR
10005	System switched from STOP to RUN
10006	System in STOP mode
10007	SDDC Timeout
10008	SDDC CRC
10009	AI communication
10010	AI communication KI Kommunikation incorrect receive-length
10011	AI communication NETx status error

8.5. PLC processing

8.5.1. PLC commands

Operator	Operand	Description
LD	All input operands and all output operands	Equals the current result to the operand
LD NOT	All input operands and all output operands	Equals the current result to the operand and inverts the operand
ST	Only output operands	Stores current result under the operand address
AND	All input operands and all output operands	Boolean AND
AND NOT	All input operands and all output operands	Negated Boolean ANDD
OR	All input operands and all output operands	Boolean OR
OR NOT	All input operands and all output operands	Negated Boolean OR
XOR	All input operands and all output operands	Boolean Exklusiv OR
NOT	All input operands and all output operands	Inverts the value of the accumulator
S	PLC_FLAG in the output image	Sets FlipFlop
R	PLC_FLAG in the output image	FlipFlopreset
S1	All input operands and all output operands	Sets operand to 1
R1	All input operands and all output operands	Sets operand to 0
SQH (MACRO_INFO)	Description of macro element	Operand field : 2 bytes for identification of macro
SQC (MACRO_CRC)	CRC of the preceding macro field	Operand field 1. Operand: CRC_LO (8 Bit) 2. Operand: CRC_HI (8 Bit)
INFO	Infobox	Operand field: 1. Operand: reserved free ! 2. Operand: reserves free !

8.5.2. Resource assignments

Element	In	Out	Amount MX	IN/OUT Process image	PLC-Code	IL display
AND2	2	1	1	0	LD x1.y1 AND x2.y2 ST MX.z	3
AND5	5	1	1	0	LD x1.y1 AND x2.y2 AND x3.y3 AND x4.y4 AND x5.y5 ST MX.z	6
OR2 .. OR5					Analog AND	3 … 6
XOR 2					Analog AND	3
NOT	1	1	1	0	LD x1,y1 NOT ST MX.z	3
RS-Flipflop	2	1	0	Output = 1	LD x1.y1 (Quelle S) S M.z LD x2.y2 (Quelle R) R M.z	4
Permanent Flipflop (RS)	2	1	2	Input = 2 Output = 1	LD MX.0 ST CFlipFlop_EN0.0 LD MX.1 ST CFlipFlop_EN0.1 LD CFlipFlop0.0 ST MX.2	6
Timer	1	1	0	Output = 1	Timer freigeben : LD x1.y1 ST PLCTimer_EN.z	2
Single semiconductor output	1	1	0	Output = 1	LD x1.y1 ST DO.x_y	2
Semiconductor output Redundant	1	2	0	Output = 2	LD x1.y1 ST DO.x_P ST DO.x_M	3
EDM	2	2	0	Input = 2 Output = 2	LD MX.0 ST DriveEDM_EN0.0.1 LD MX.2 ST DriveEDM_EN0.0.0 AND DriveEDM0.0.0 ST DriveEDM0.0.0	6
Restart	2	2	2	Input = 2 Output = 2	LD MX.0 ST Restart_En0.0 LD MX.1 ST Restart_EN0.1 AND Restart0.1	10

					ST Restart0.1 LD Restart0.0 ST MX.2 LD Restart0.1 ST MX.3	
Edge deduction (rising)	1	1	2	0	LD MX.0 AND NOT MX.1 ST MX.2 LD MX.0 ST MX.1	5
Edge deduction (falling)	1	1	3	0	LD NOT MX.0 AND MX.1 ST MX.2 LD MA.0 ST MX.1	5
Edge deduction (rising / falling)	1	1	3	0	LD MX.0 XOR MX.1 ST MX.2 LD MA.0 ST MX.1	5
Fieldbus In	0	1	0	Output = 1	LD FBus_In0.0	1
Fieldbus Out	1	0	0	Input = 1	ST FBus_Out0.0	1
Functional In	1	1	2	Input = 1 Output = 1	SQHx.y LD MX.0 AND FE0.0 ST MX.1 SQCx.y	5
Functional Out	1	0	1	Input = 1	LD MX.0 ST FA0.0	2

8.5.3. PLC-operand

In the following all supported operands with indication of the number places.

Operand	Numberplaces	Description
DriveSAC	3	Result monitoring function SAC
DriveSDI	3	Result monitoring function SDI
DriveSLI	3	Result monitoring function SLI
DriveSEL	3	Result monitoring function SEL
DriveSSX	3	Result monitoring functionSSX
DriveBase	3	Result monitoring function DriveBase
DriveSLP	3	Result monitoring function SLP
DriveSLS	3	Result monitoring function SLS
DriveSCA	3	Result monitoring function SCA
DriveEOS	3	Result monitoringfunction EOS
DriveSOS	3	Result monitoringfunction SOS
DriveECS	3	Result monitoring function ECS
DriveACS	3	Result monitoring function ACS
DriveICS	3	Result monitoring function CS
DriveDEM	3	Result monitoring function DEM
DriveEDM	3	Result monitoring function EDM
Drive ESA	3	Result monitoringfunction ESA
Drive SBT	3	Result monitoring function SBT
PLC Timer	1	Result PLC timer
E Timer	2	Result start element
Starttest	1	Result start-up test
Twohand	1	Result two-hand switch
Masterswitch	1	Result Master switchr
Edge	1	Result edge dectection
Restart	1	Result Restart element
SDI	2	Digital inputs Master / binary inputs Slavee
E Address	2	Binary input data Slave assembles
FE	2	Functional input
SCO_Status	2	SOC status
DriveSZMc	3	Result monitoring function SZMc
DriveSLSc	3	Result monitoring function SLSc
DriveSCUc	3	Result monitoring function SCUc
DriveSOM	3	Result monitoring functionSOMc
DriveSAC_EN	3	Enable montoring function SAC
DriveSDI_EN	3	Enable monitoring function SDI
DriveSLI_EN	3	Enable monitoring function SLI
DriveSEL_EN	3	Enable moitoring function SEL
DriveSSX_EN	3	Enable monitoring function SSX
DriveBase_EN	3	Enable monitoring function DriveBase
DriveSLP_EN	3	Enable monitoring function SLP
DriveSLS_EN	3	Enable monitoring function LS
DriveSCA_EN	3	Enable monitoring function SCA
DriveEOS_EN	3	Enable monitoring function EOS
DriveSOS_EN	3	Enable monitoring function SOS
DriveECS_EN	3	Enable monitoring function ECS
DriveACS_EN	3	Enable monitoring function ACS
DriveICS_EN	3	Enable monitoring function ICS

DriveDEM_EN	3	Enable monitoring function DEM
DriveEMU_EN	3	Enable monitoring function EDM
DriveESA_EN	3	Enable monitoring function ESA
PLC Timer_EN	1	Enable PLC timer
E Timer_EN	1	Enable input timer
Starttest_EN	1	Enable start- test
Twohand_EN	1	Enable two-hand switch
Masterswitch_EN	1	Enable Master switch
Edge_EN	1	Enable edge detection
Restart_EN	1	Enable restart element
DO	2	Digital outputs
SDO	2	Safe Digital Output
SRO	2	Safe Relay Output
DOut	2	Non-safe digital output
S_Hi	2	Safe HISIDE output
S_Lo	2	Safe LOSIDE output
A Address	2	Binary outputs
PLCMXMerker	1	Flag elements for PLC cache
SCO_cmd	2	Data internal system bus
DriveSZMc_EN	3	Enable monitoring function SZMc
DriveSLSc_EN	3	Enable monitoringfunction SLSc
DriveSCUc_EN	3	Enable monitoringfunction SCUc
DriveSOM_EN	3	Enable monitoring function SOMc
EAE	2	Safe input Sichere Eingänge Erweiterungsbaugruppe
EAA	2	Safe output extension module
FA	2	Ffunctional output
CBool	1	Configurable Boolean
CFlipFlop	2	Result configurable saveable FlipFlop
CFlipFlop_EN	2	Enable configurable saveable FliipFlop
DriveFDB	3	Result FSoE disconnect block
DriveFDB_EN	3	Enable FSoE disconnect block
DriveSRX	3	Result SafeReferencing
DriveSRX_EN	3	Enable SafeReferencing
DriveSRS	3	Result SafeReferencing State
DriveSRS_EN	3	Enable SafeReferencing State
DriveSRTc	3	Result Robotic block
DriveSRTc_EN	3	Enable Robotic block
DriveSLA	3	Result monitoring function SLA
DriveSLA_EN	3	Enable monitoring function SLA
FBus_Slave_In	2	Profisafe input data
FBus_Slave_Out	2	Profisafe Output data
SSF_Slave_In	2	SpecialFunction Input Slave device
SSF_Slave_Out	2	SpecialFunction Output Slave device
GBox_State	2	Status GBox
DriveESM	3	Result Encoder Standstill Monitoring
DriveESM_EN	3	Enable Encoder Standstill Monitoring

Agenda:

- 3 digits z.B. LD DriveSLS_ENx.y.z (corresponds to 3 digits below)

  x: number of assembly where block is carried out. (0: Master, 1: Slave with logical address 1, etc.

  y: Instance of the block

  z: Bit address of the block
- 2 digits, e. g. LD SDIx.y
  x: number of the assembly where block is carried out. (0: Master, 1: Slave with logical address 1, etc.
  y: Instance of the block
- 1. digit, e. g. CBool.x (implementation only on Master device)

  x: Instance of the block