Servo Drive Software with Safety PLC Function

Programming Manual

for MSD with integrated safety control

Servo Drive Software with Safety PLC Function - Programming Manual

ID no.: CB78095-001, Rev. 1.3

Date: 02/2018

Safety firmware V1.00-04

Technical alterations reserved

The contents of our documentation have been compiled with greatest care and in compliance with our present status of information.

Nevertheless we would like to point out that this document cannot always be updated parallel to the technical further development of our products.

Information and specifications may be changed at any time. For information on the latest version please refer to drives-support@moog.com.

NOTE: This document does not replace the Operation Manuals

MSD Servo Drive AC-AC Single-Axis System (ID no.: CA65642-001)
MSD Servo Drive DC-ACMulti-Axis System (ID no.: CA97554-001)
MSD Servo Drive Specification “Functional Safety” (ID no.: CB38398-001).

Please always follow the information given in "For your safety", "Intended use" and "Responsibility" in the above-mentioned operation manuals.

You will find information on mounting, installation and commissioning as well as the assured technical characteristics of the MSD device series in the supplementary documents (operation manual, application manual, etc.).

Contents

1 TERMINOLOGY

Servo drive (master)

The servo drive (master) is the first servo drive in the axis group that is inserted in the terminal diagram. The servo drive (master) can manage up to five servo drives (slave). The communication between the servo drives (master/slave) is via the axis cross-communication (SCC).

Of course, a servo drive (master) can also be operated as a single axis.

Servo drive (slave)

Every servo drive in the axis group that is managed via the SCC by the servo drive (master) is a servo drive (slave).

Axis group

Axis group is an overall term for a combination of several MSDFS that are connected via the SCC. An axis group comprises:

- A minimum of 1x servo drive (master) and 1x servo drive (slave)
- A maximum of 1x servo drive (master) and 5x servo drives (slave)

Attribute

Non-graphic property of a function block. An attribute comprises a designator and a value.

Instruction list (AWL)

Assembler-like programming language that can be loaded into a PLC. The instruction list is generated with the aid of Safety PLC from the function blocks defined, their attributes and operators.

Function block group

Classification of the function blocks by how they can be positioned in the function block diagram (input, output, logic).

Function block

PLC block that has either a physical or logical effect on the execution of a program in a PLC. A physical (hardware) function block is, e.g., a button or an output on the MSDFS. However a function blocks is also a logical operator (for instance AND or OR) applied to input and output signals within the PLC.

Function block diagram (function block language)

Graphically-orientated, function, block-based, descriptive "programming language" in accordance with IEC 1131 that is used to show links between the inputs and outputs on the function blocks in a PLC. In the function block diagram (FBD) the function blocks and their links are shown graphically.

Function block type

More detailed identification of the function block within a group. (E.g. "emergency stop")

InPort / OutPort

Point in a function block to which a link can be made to other function blocks. Each input connector represents an InPort and each output connector an OutPort.

Information display

The delayed display of information on a function block similar to the Windows tooltip mechanism. To display this information the mouse pointer must be moved over an object.

Connector

Connection point between the start and end of a link between an input and output on a function block (see InPort / OutPort)

Configuration

Configuration is a collective term for a monitoring program and the related parameters for the deviations allowed or the minimum and maximum values. In this respect it is important that a monitoring program always contains further data related to the program.

Message Window

Multiple line output window embedded in the Windows toolbar element. This window is used to display errors, warnings and information on the program to the user. The message window can be enabled and disabled.

MSDFS

MSD System in the version **F**unctional **S**afety. Here the first servo drive is always the servo drive (master) and all other servo drives in the group are servo drives (slaves).

OSSD

Abbreviation for "**O**utput **S**ignal **S**witching **D**evice". This is a safe semiconductor output on which the switching capability is tested with the aid of test pulses. The test pulses are manufacturer-dependent and are normally so short that they do not interrupt downstream actuators.

PLC

Programmable **L**ogic **C**ontroller. Only the term PLC is used in the MSD System.

PLC input signal list

Signal lines entering the PLC, shown as a table. The labels for the PLC inputs can be defined by the user in Safety PLC. They have a unique number and must be assigned to the inputs of a function block.

PLC output signal list

Signal lines leaving the PLC, shown as a table. The labels for the PLC outputs can be defined by the user in Safety PLC and have a unique identification number, like the inputs.

Routing

Horizontal and vertical arrangement of the links on a function block diagram such that there are no overlaps with function blocks and the links with the same connector are combined at an early stage (referred to the distance to the destination function block).

SCC

Abbreviation for **S**afe **C**ross **C**ommunication. This is safe axis cross-communication for the servo drive. The servo drive (master) communicates with the servo drive (slaves) in the servo group using this connection.

Safety PLC

Safety PLC is the user interface for configuring, setting parameters and programming the SMC in the servo drive MSDFS.

SRP/CS

The English term from which this abbreviation is derived is "safety-related parts of control systems". This term refers to the safety-related part of a control system that reacts to safety-related input signals and generates safety-related outputs signals (cf EN ISO 13849-1).

Signal list

Signal lines that enter and leave the PLC, shown in a table.

Signal cell

Area within the signal list that can be selected and in which a comment can be entered.

SMC

SMC is the abbreviation for"**S**mart **M**onitoring **C**ontrol". This is a modular monitoring tool from Moog for reliable servo monitoring that is integrated into the MSDFS.

Link

A named connection between:

- A function block output (OutPort) and a function block input (InPort)
- A PLC input and a function block input (InPort)
- A function block output (OutPort) and a PLC output

Validation

Validation is a check to evaluate if the desired safety functionality is achieved (see EN ISO 13849-2).

2 MOUSE AND KEYBOARD COMMANDS

2.1 Mouse-dependent actions

TIP: If for "Set flag" the CTRL key is pressed while making the selection, the related "Flag output" blocks are also selected.

Action	Explanation
Left mouse button on a function block	Selected display (highlight); previous selections are cleared.
Shift + left mouse button on function block	Multiple selection (add to an existing selection).
Ctrl + left mouse button on selected function block	Clear the selection of the function block (removal from the selection).
Delete key	Delete the elements in an existing selection incl. the connections!
Double-click on function block	Edit the settings.
Right mouse button on function block	Display the context menu for function block.
Right mouse button in the drawing area	Display the context menu for drawing area.
Left mouse button on connector	Highlight the existing link(s).
Ctrl + move the mouse pointer over an object	Display information even if the display of the information is disabled via the menu.
Turn the scroll wheel in the mouse	Dynamically zoom the function block diagram.
Drag the mouse with scroll wheel pressed	Move the function block diagram.

2.2 Keyboard commands

Action	Explanation
Ctrl + Q	Start zoom-in command
Ctrl + W	Start zoom-out command
Ctrl + A	Zoom all command
Ctrl + I	Enable, disable automatic display of information
Ctrl + O	Open file
Ctrl + S	Save file
Ctrl + M	Enable, disable message window
Ctrl + N	New file
Esc	Clear the selection of elements marked
Del	Delete the selected objects
Ctrl+Left arrow	Line scroll function block diagram left
Ctrl+Right arrow	Line scroll function block diagram right
Ctrl+Up arrow	Line scroll function block diagram up
Ctrl+Down arrow	Line scroll function block diagram down

3 SHORT DESCRIPTION OF THE PROCEDURE

The program Safety PLC from Moog is a graphically-orientated software application for preparing a PLC-based monitoring program for the MSDFS. This version permits the safe monitoring of servo motors.

The structure of the programming task in Safety PLC is the result of many years of experience at Moog with safety-related control tasks. The following procedure has been proven to be effective on programming the MSDFS, but it is not mandatory to follow this procedure. Screenshots of the related toolbars or commands are included for illustration.

NOTE: The program requires the user to have write and read permissions on the computer used for programming. Insufficient permissions can have side-effects during function block diagram debugging, or cause problems on saving function block diagrams in directories with restricted permissions.

3.1 "Push & pop" instead of "drag & drop"

Is should be noted that the development team for Safety PLC has decided against "drag & drop" as favoured in Windows. Instead, simply click an icon on a toolbar or a menu command ("push") and in this way change to pop mode. This mode is apparent from the different mouse pointer. You do not need to keep the left mouse button pressed. To "pop" the selected function block click the point where it is to be pasted. The "Esc" key cancels this mode. The steps suggested for the procedure match the considerations that should be undertaken on planning the safety-related monitoring of a drive axis.

3.2 Setting the device types to be programmed

After the Safety PLC program has been started or if a new function block diagram is to be added, the following view appears:

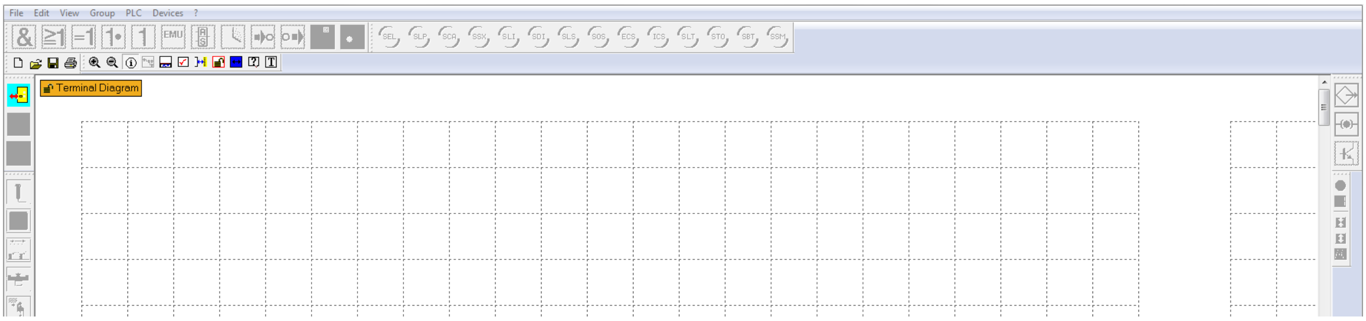


Figure 1: Start view

Use the  icon to open the dialog box to add the first axis, the servo drive (master), to the terminal diagram.

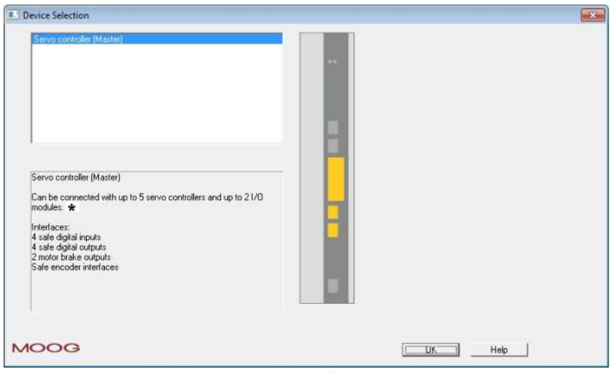


Figure 2: Device Selection *I/O modules not available

After adding the servo drive (master) another dialog box opens, Device Configuration.

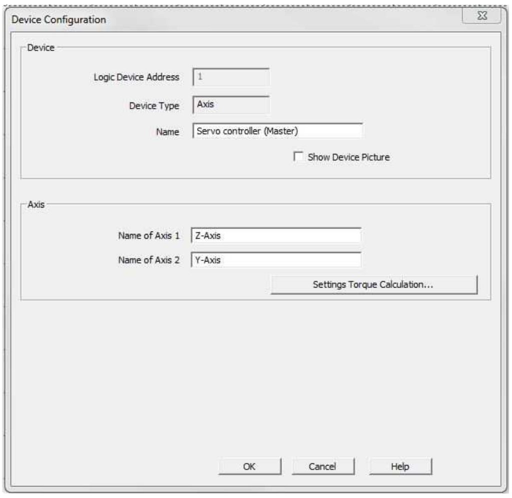


Figure 3: Device Configuration

Here you can assign the name for the servo drive (master) added and the axis to be monitored.

The servo drive (master) is then available in the terminal diagram:

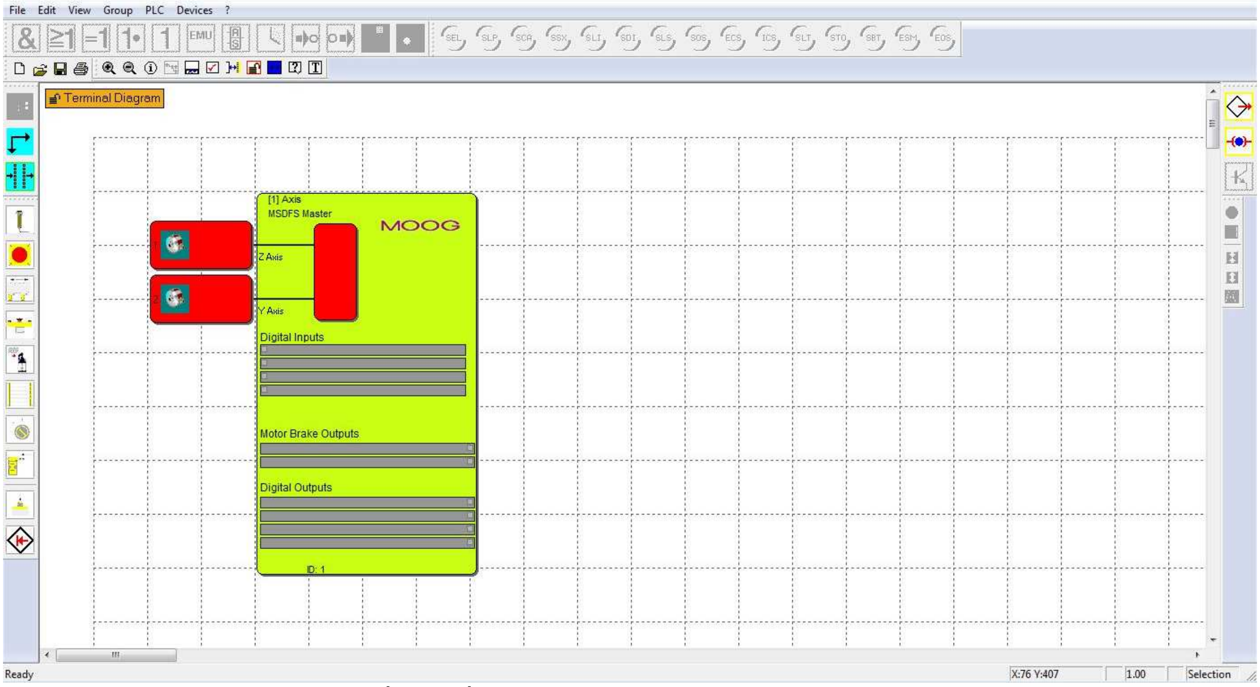


Figure 4: Display of the servo drive (master) in the terminal diagram

To monitor further axes, use the  icon to add servo drive (slaves).

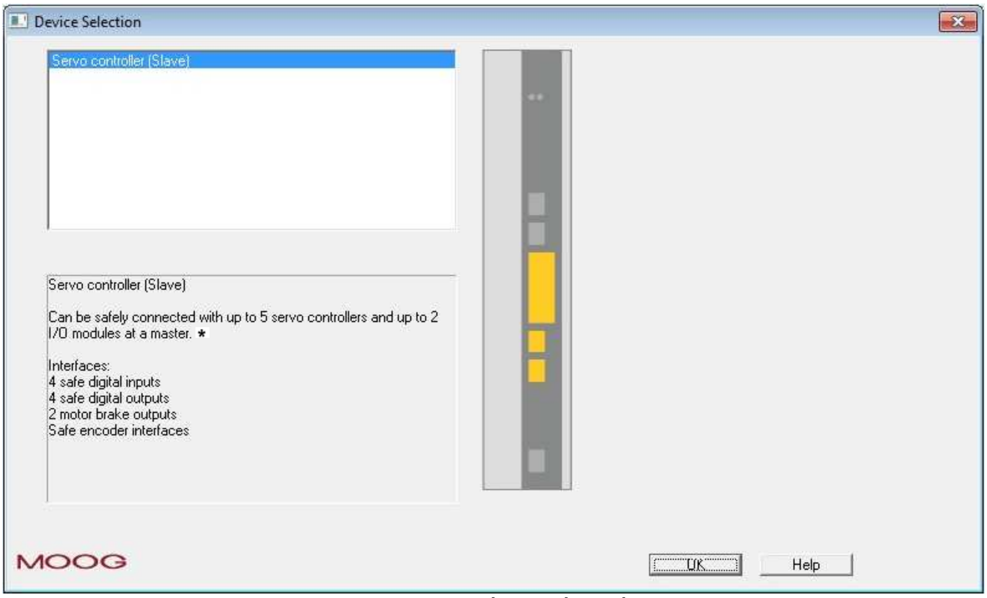


Figure 5: Selection of additional servo drives (slaves) *I/O modules not available

Here you can assign the name for the servo drive (slave) added and the axis to be monitored.

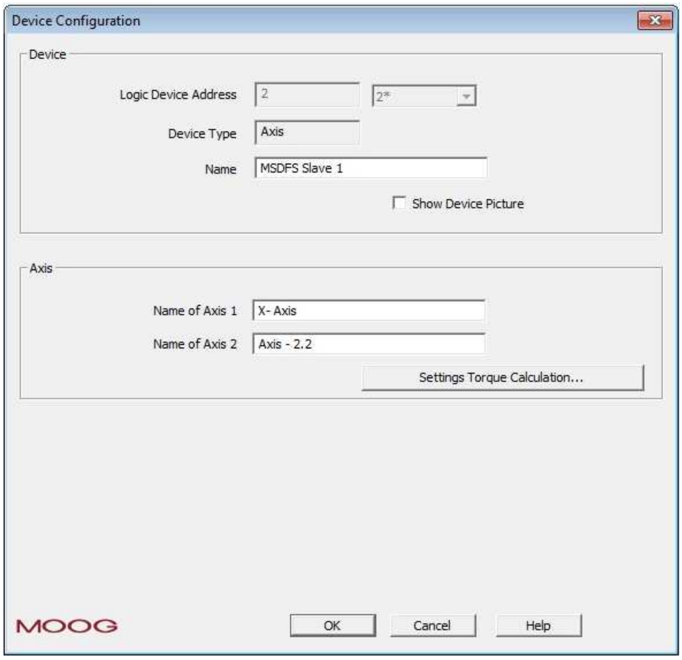
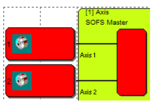


Figure 6: Device configuration for additional axes

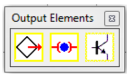
NOTE: A maximum of six servo drives can be operated in an axis group and therefore managed in the Safety PLC programming application.

3.3 Defining the peripherals in the terminal diagram

The terminal diagram depicts the view from the exterior on the axes to be monitored. Here you should undertake the following work steps.

-  If necessary define the encoders used and their parameters by double-clicking the encoder icon for the related axis.

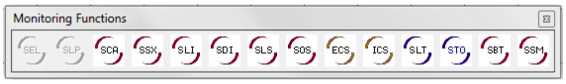
-  Define the input peripheral blocks used (emergency stop, guards, sensors etc.), their configuration and define the monitoring input on the available axes to which these peripherals are to be connected.

- 

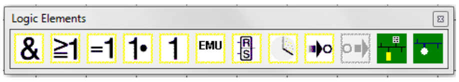
Define the output blocks (semiconductor / brake output) and the channels on which feedback is to be provided in case of malfunction.

3.4 Defining the monitoring functions and logic blocks in the function block diagram

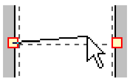
The function block diagram shows the logic blocks and their link to a program schematic inside the servo drive (master). In the function block diagram you can:

- 

Define the type of monitoring and its parameters (this action is only possible if the encoder parameters have been defined).

- 

Define the operators and logic elements such as timers, flip flops and terminal blocks.

-  Wire the peripheral devices, monitoring and logic blocks "internally". For this purpose place the mouse pointer over the start connector, press the left mouse button and with the button pressed drag to the destination connector and release.

- Use the diagnostic and analysis tools. These include the information display, signal trace, the display of function block attributes in the Message Window, and quickly jumping to blocks in the function block diagram by double-clicking the BlockID marked in colour in the Message Window.

3.5 Compiling the monitoring program

After the required blocks have been defined and connected, the program for the servo drive (master) can be prepared. The servo drive (master) can manage up to five servo drives (slave); communication is undertaken via the safe cross-communication (SCC). Safety PLC undertakes the following for you:

- Checks for open connectors in the function block diagram

- Checks the boundary conditions for the monitoring functions
- Checks for the correct distribution of cross-circuit pulse numbers
- Prepares OP programming code that can be transferred to the servo drive (master)

3.6 Program transfer to the servo drive (master)

Once the monitoring program for the safe controller has been compiled without errors, it is transferred to the servo drive (master). For this purpose undertake the following steps:

- Using "Verbinden" (Connect) establish the connection to the servo drive (master)
- Transfer the "CONFIG" data
- Transfer the "PROG" data
- Test the monitoring program on the safe controller
- Lock the function block diagram once the program has been approved
- Prepare a configuration report and validate the configuration

NOTE: The first axis that is added to the terminal diagram is always the servo drive (master), all other axes are the servo drives (slaves).

You can configure communication settings using the application "Options". This application is saved in the installation folder for Safety PLC.

C:\...\Moog\Safety PLC Functions\SafetyComInterface

The USB connection on the servo drive (master) must be used for the communication with the servo drive (master) and the PC. In this way a "Peer to Peer" connection is ensured.

4 Function block diagram

Function block diagrams are Windows documents with the file extension "plc" prepared using Safety PLC. These files contain all the information for the automatic preparation of a program for the servo drive (master) from Moog.

The function block diagram is divided into fields that can accommodate the function blocks. The function blocks are added and moved in this grid. It is not possible for function blocks to overlap.

Within the function block diagram there are two views available to the user: "Terminal Diagram" and "CFC" (function block diagram).

There are two ways you can switch between the views:

- **Menu:** View → Layout umschalten (Switch layout)
- **Keyboard:** Crtl + Tab
-  button on the "Zeichenhilfen"(Drawing aids) toolbar

The servo drive (master/slave) peripherals to be monitored are defined on the terminal diagram.

The links between the inputs and outputs to be monitored and the logic and monitoring functions blocks are made on the function block diagram.

4.1 Continuous status indication

There is a status indication in the top left corner of the function block diagram. This indication provides information on the following states:

Active function block diagram view: This information is displayed as text and changes between "Terminal Diagram" and "CFC" (function block diagram)

Actual diagram access: This information is displayed as an open or locked padlock.

Compiler status: The background colour of the status indication defines the current compilation status of the function block diagram.

- **Orange:** It is still necessary to compile the function block diagram
- **Green:** The current function block diagram has been compiled, the program can be transferred to the MSDFS.

NOTE: For safety-related reasons the Windows "Cut" and "Paste" commands cannot be used on function blocks.

TIP: Use the context menu on the function block diagram.

4.2 Terminal diagram

The terminal diagram shows the links between the switches and sensors to be monitored and the MSDFS. Now the components dependent on the servo drive (master) and the servo drives (slaves) must be defined in the terminal diagram. On the addition of a new function block or if you double-click a function block already added, the related attribute editor is opened and the parameters can be modified.

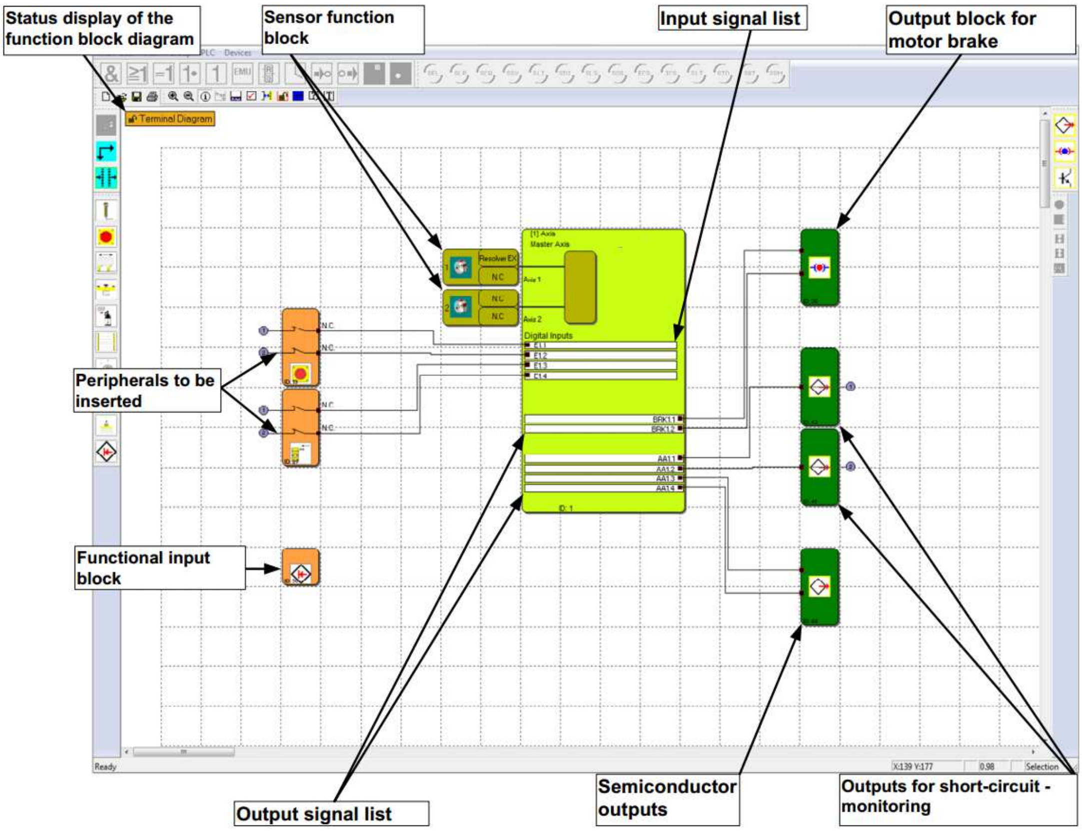


Figure 7: Terminal diagram

If you add function blocks to the terminal diagram, the elements are then wired automatically. In some cases the connections may not be shown in the best possible manner. This **does not affect the functionality in any way**. You can force the connection to be redrawn by moving the related block. You can also delete existing connections and then add them by hand.

NOTE: If it is not allowed to define any logic elements in this view, the related commands are not available.

TIP: Start at the left edge of the function block diagram and work down.

4.3 Function block diagram

In the function block diagram links are made in the program between input elements, monitoring elements and logic blocks as well as the outputs on the MSDFS. The outputs on the input elements correspond to the MSDFS inputs in this view. Conversely, the inputs for the output elements correspond to the outputs on the MSDFS.

To be able to design a clear function block diagram, you can define so-called terminal blocks. These terminal blocks form a named connection between input and output connectors on function blocks. One or more flag output blocks (output terminals) can be defined for a Set flag block (input terminal).

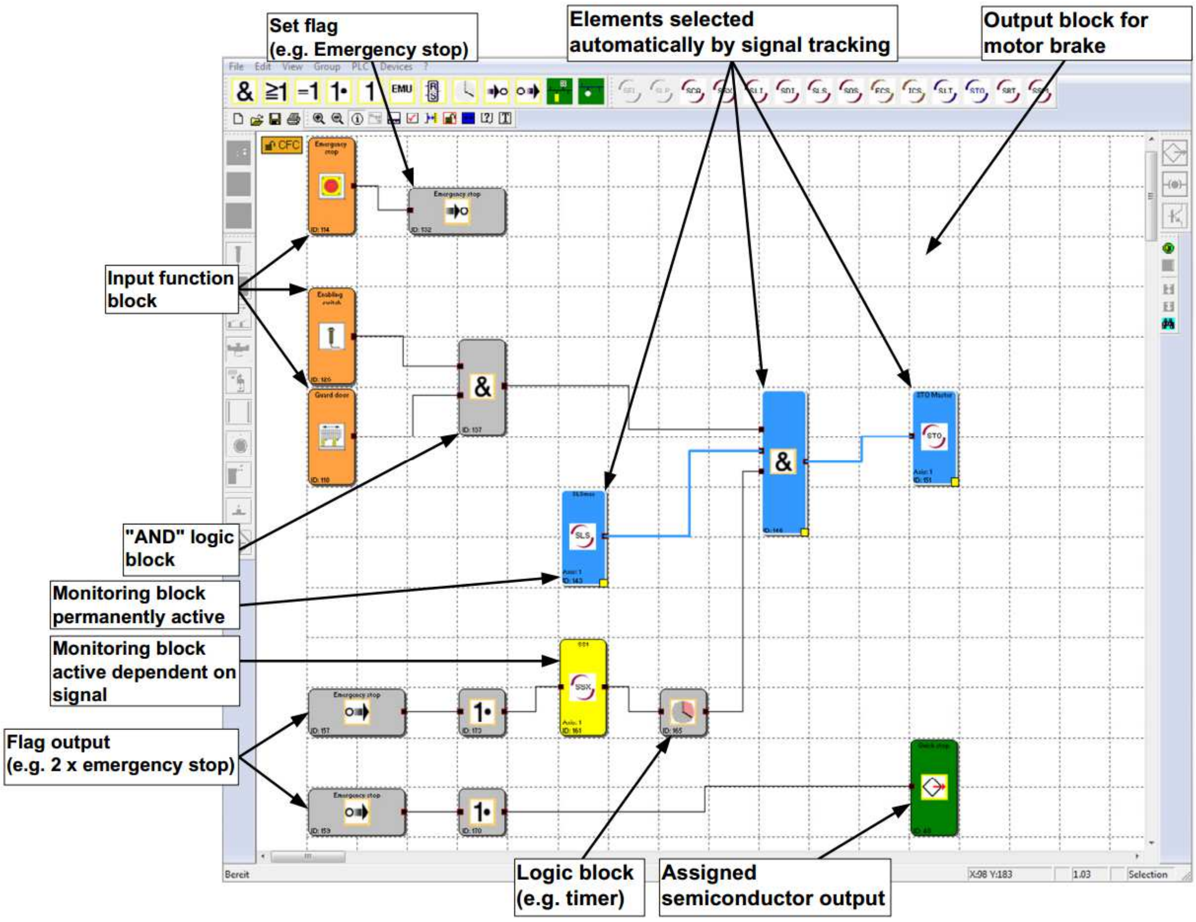


Figure 8: Function block diagram

NOTE: It is not possible to modify the parameters for the input elements in this view.

TIP: Use the comment line on the Set flag blocks. The comment appears if the block is selected and as a label on the flag output block. This information improves clarity!

5 PREPARING WIRING

The function block diagram is built-up by connecting the input and output connectors on function blocks. An output on a block can, if necessary, be connected to several inputs on other blocks, while an input can only be used once. Also, for technical reasons, certain block groups cannot be connected together. If a connection is not valid a message is displayed by the program.

Preparing a connection:

1) Select a start connector using the left mouse button
2) Position mouse pointer with left mouse button pressed
3) Release mouse button over the destination connector

NOTE: You can select connections by clicking them using the mouse or by selecting a connector; it is not possible to select a network.

TIP: If all connections on a block are to be deleted, the best method is to delete the function block. The related connections are deleted automatically.

The program draws the connection automatically by adding additional intermediate points (corner points) based on a halving algorithm. The graphic display can be varied by moving the function blocks. In case of complex diagrams, a connection line may cross a function block. This behaviour has no effect on the preparation of the program.

There is an additional command for drawing self-defined connection lines. These user-defined connection lines are retained until it is necessary to re-calculate the intermediate points due to the movement of a related function block.

A user-defined connection is made as follows:

1) Either select the connection that is to be edited and open the command: "Benutzerdefinierte Verbindungspunkte" (User-defined connection points) on the "Edit" menu.
2) Or open the context menu (right mouse button) with the mouse pointer over the related connection and select the command "Benutzerdefinierte Verbindungspunkte" (User-defined connection points).
3) Enter the intermediate points (corner points). The program starts at the output connector. It is only possible to generate orthogonal intermediate points, i.e. the connection lines are always horizontal or vertical. The program collects the points entered until the draw command is terminated.
4) Terminate the command using the Enter key (Return). The connection is then drawn.

NOTE: The program adjusts the first and last intermediate point to the related function block connector. The output and input connector are not intermediate points and therefore also do not need to be entered.

TIP: Visual corrections on the function block diagram should only be made just before locking the function block diagram. Then the layout is complete and it is no longer necessary to move the blocks.

6 MESSAGE WINDOW

Along with the output of status and error messages and the display of results from the function block diagram check, the Message Window is a powerful tool for checking the function blocks within their context.

Quick jump

If you double-click the block IDs marked in colour in the Message Window you can display the related block in the middle of the function block diagram window. In this way you can quickly find the function blocks related to a task.

Figure 9: Message Window

<u>Context menu in the Message Window</u>

Message Window

Hides the Message Window

Fenster Delete Content

Clears the contents of the Message Window. It is recommended to clear the contents at regular interval, as for instance on the extensive usage of the "Attributes in the Message Window" function a very large amount of data can accumulate in the Message Window, which could slightly reduce the processing speed for the output of messages.

Select and Copy Content

Copies the entire content of the Message Window to the clipboard such that the text is available in other Windows programs using the "Paste" command.

Text search

Makes it possible to find text in the Message Window.

Message Window Help

Opens this help page

Dock Window

Toggle button to dock the Message Window to the frame for the main program or to position the window anywhere on the screen.

NOTE: It is not possible to configure the "docking" behaviour for the Message Window via the settings.

MOOG

7 GENERATING THE PROGRAM

Once all connectors on the blocks added have been connected, a monitoring program can be generated.

The program is compiled after opening the compiler using the steps described below. The results are displayed in the Message Window, which appears automatically when the compiler is started.

1) Check for open connectors

Safety PLC ensures all connections between the function blocks can be resolved. Connectors that are not connected are displayed as errors.

2) Check for unreferenced "flag" blocks

Safety PLC ensures that all "Set flag" blocks added to the function block diagram are used. Otherwise unreferenced addresses would be used in the program.

3) Check the value ranges for the monitoring functions

Prior to generating the instruction list Safety PLC checks whether the parameters for the monitoring functions are within the value ranges for the current encoder configuration. Otherwise, if the encoder settings are modified with the monitoring functionality already defined, a value may exceed a range unnoticed. This check does ***not*** replace the context-related evaluation of the data by the user after a change!

4) Generate the instruction list

The instruction list code generated from the function blocks is output in the Message Window and can be verified there, or copied to the clipboard for documentation purposes (see: context menu for the Message Window). The code blocks related to the function blocks are segmented by the related block ID, which is output as a comment.

5) Generate the OP code

This step generates the machine code that is then transferred with the configuration data to the servo drive (master).

NOTE: After a compiler run, the actual program CRC and the compile date are displayed in the Layout Management dialog box. Please lock the function block diagram after successful compilation and transfer so that unintentional modifications by compiling are no longer possible. To retrieve the CRC see section **11 Retrieving the CRCs**

TIP: Use the "Quick Jump" to jump directly to a block in the diagram by double-clicking the related *block ID* in the Message Window. In this way you can easily find the related function block in case of error messages.

8 TRANSFERRING THE PROGRAM TO THE SERVO DRIVE (MASTER)

This section describes the transfer of the data and program to the servo drive (master) via a network. The window shown below appears when the interface is started.

Figure 10: Connection

NOTE: The program prepared must only be transferred to the servo drive (master). This drive manages the servo drives (slaves) present using safe cross-communication (SCC).

More detailed actual transfer states or any errors that occur are written to the Message Window. However, this window is consciously not opened automatically for every message for reasons of space so that as much as possible of the function block diagram can be displayed for the diagnostics.

Connection

To be able to establish a connection to the servo drive (master), it must be connected to the PC via the USB connection.

Connect

Starts the connection to the servo drive (master) connected.

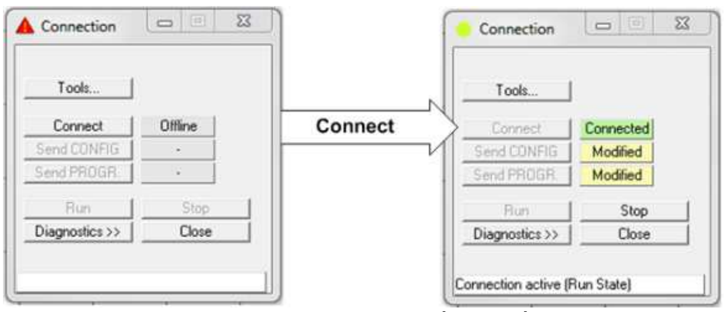


Figure 11: Connection to the servo drive (master)

Stop

Stops the program running on the servo drive (master). The dimmed buttons "Send CONFIG" and "Send PROGR" can now be used.

Send CONFIG

Sends the configuration data for the function block diagram to the servo drive (master).

Send PROGR

Sends the program data for the function block diagram to the servo drive (master).

Start

Starts the program transferred.

Close

Closes the communication.

9 DIAGNOSTICS

If the diagnostics are enabled in the Connection window, the dialog box is expanded with further elements that are used for diagnostics.

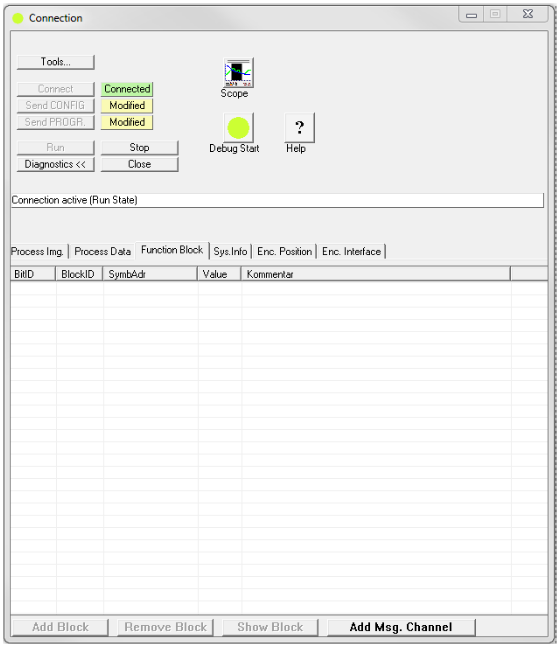


Figure 12: Diagnostics

The dialog box element "Debug Start" is a toggle button for starting and stopping the diagnostics. The related mode is displayed in the dialog box label such that feedback on the status is provided even with the dialog box minimised.

Off

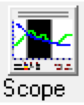
On

Alarm or error status

Once the diagnostics have been started successfully, the label on the button changes to "Debug Stop".

NOTE: Before you can start diagnostics, it is ensured that the same program is referenced in the current function block diagram and in the servo drive (master). For this reason there is a configuration check at the start of the diagnostics. This check results in the indication of the status in the message bar in the Connection dialog box. If the two programs do not match, an error message is displayed and the diagnostics cancelled.

9.1 The scope monitor



The "Scope" button opens the Scope dialog box. Here you can monitor changes in velocity, acceleration and position over time.

Setting the parameters for the drive monitoring requires detailed knowledge of the process data. It is particularly important to know how the velocity, acceleration and position change over time. Only in this way is it possible to set the correct thresholds and limit parameters.

All available graphic functions read the necessary process data ONLINE via the communication interface on the servo drive (master) and display this data in real time. Current values are displayed on the right edge of the scope monitor and move to the left as the recording progresses until they finally disappear at the left edge. Even though the data have disappeared from the visible window, these data are still retained in a cache and can be moved back into the visible area by moving the slider under the graphic window.

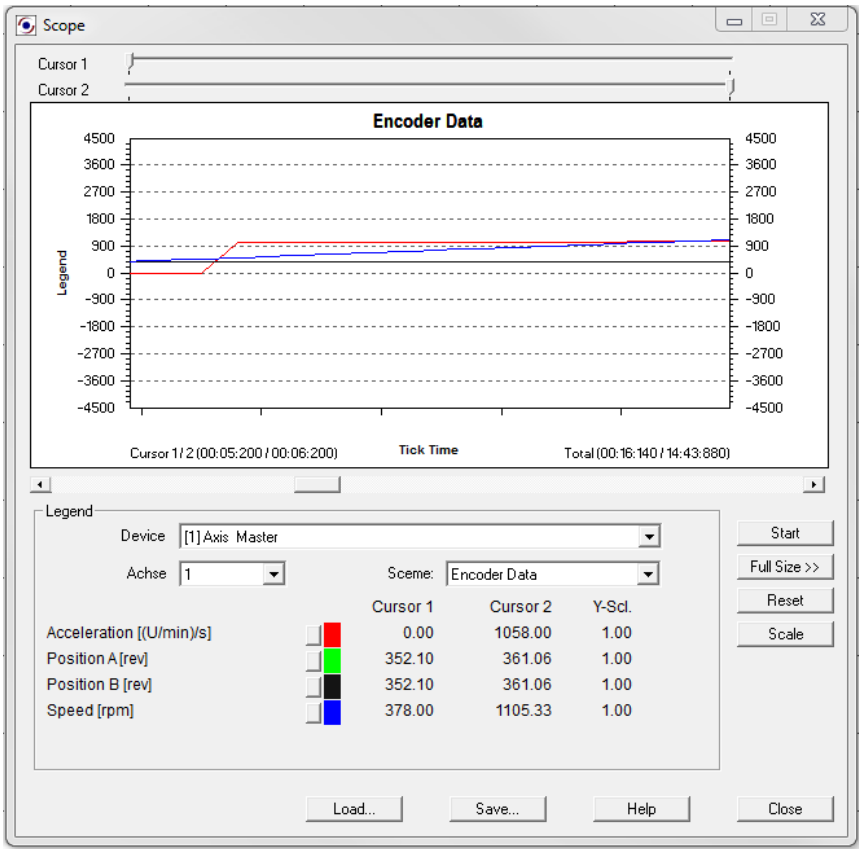


Figure 13: Scope monitor

NOTE: While a scope monitor is active the process image or function block diagram debugging is hidden and the diagnostics tab in the Connection dialog box disabled. These data cannot be provided for performance reasons.

Cursor 1, Cursor 2

Using these sliders you can set two cursor positions to display specific values on the diagram. The line indicated on the graphic moves as you move the slider. During this process the values for the related cursor positions are displayed in the Legend group box. The times for related to the cursor positions are output on the graphic.

Scale

Opens a dialog box for scaling the graphic functions displayed. In this way the Y values on the individual graphs can be scaled if they do not fit in the value range displayed due to the configuration.

Start / Stop

Start / stop recording.

Full Size >>

Increases the size of the scope monitor to the full screen area available. If you have switched to full size, you can reduce the size of the dialog box back to normal using the "Normal <<" button.

Scheme

You can select the context for the required data display using the scheme. The context of the graphs changes depending on the scheme selection in the list box. These are assigned via the colours given in the Legend group box. The following contexts are available:

- Encoder data
- Gebergeschwindigkeit (Encoder speed)
- SSX Daten (SSX data)
- SEL (Position-related)
- SLS Filter (SLS filter)
- SCA Filter (SCA filter)

The advancing time tick is displayed on the X axis, while the Y values relate to the scheme selected.

It is not possible to change the scheme while a measurement is in progress.

Load...

Using this button you can load a measurement saved in a scope XML file into the scope monitor. The Scope dialog box then changes to the viewing mode. Due to the possibly different encoder configuration for the measurement viewed compared to the actual program and the resulting differences in the scaling of the position or velocity values, the "Start" button and the Scheme list box are not available if data have been loaded for viewing. It is no longer possible to undertake a measurement until the scope monitor is restarted.

Save...

If the scope is stopped, you can save a current recording in a file. The scope data are written to a file as ASCII values. XML tags are applied to the individual values so that the recording can be used for documentation or for the analysis of the encoder configuration. The data can be viewed using the current version of Microsoft Explorer or in another XML viewer.

NOTE: All Internet or LAN-based applications (e.g. mail program) that may be running in the background must be closed prior to the measurement!

9.1.1 Measuring using the scope monitor

If the scope monitor is started from the Connection dialog box, it is in the stop mode. To be able to undertake measurement with as little trouble as possible, you should use the procedure given below.

1) Prepare measurement

Select the required measurement scheme.

During a speed-orientated measurement, the elapsed tick time for the servo drive (master) is displayed on the X axis. The axis is to be considered a sequentially incrementing counter of the system ticks from the servo drive (master). The measured data for the graphs are continuously updated and retained in the cache. The cache can hold approx. 15 minutes of data from the recording.

If the cache is full, the measurement is automatically restarted. The previous measurement is automatically saved under the name "ScopeTempData.ScpXml".

During a position-orientated measurement the measuring range configured for encoder 1 is displayed on the X axis. The trailing pointer (cursor 1) is at the current position. Using cursor 2 you can display in the Legend group box measured values as a function of the position to which the cursor is moved.

NOTE: If you change the scheme, data already recorded from the previous measurement are deleted! The cache for the ongoing measurement must also be deleted if the size of the dialog box is to be changed. A message appears if you switch to full size. The current measurement is lost.

2) "Start" measurement

If you click this button, the elapsed tick time in the graphic for the scope moves from right to left, **if there is an existing connection**. The measured values are now entered in the scope's cache. This cache saves the data as the measurement progresses. After starting the scope monitor the label on the button changes to "Stop".

NOTE: If the measurement exceeds the maximum measurement duration of approx. 15 minutes, the measurement will be restarted.

3) "Stop" measurement and view data

Once the measurement has been made you can move the recording along the X axis to view the data using the slider under the graphic. To display specific measured values you can position a line over the graphic in the X direction using the Cursor 1 / Cursor 2 sliders. The related Y values can then be read in the "Cursor 1" or "Cursor 2" column in the Legend group box.

9.1.2 Measuring schemes

The measuring schemes available are listed in the following with their function and application.

Measuring scheme:	Encoder data
Functionality:	Recording the scaled position values from system A and system B over time. Recording the process values for speed and acceleration over time.
Comment:	The process value for the position is formed internally from position value for system A. Recording the process values for speed and acceleration over time.
Application:	Scaling encoder systems A and B in the case of position monitoring. Given a correctly scaled encoder system there should be any significant difference between the position A and B. Analysis and change in the encoder signal over time for diagnostics (e.g. troubleshooting etc.) Acceleration and speed behaviour of the drive. Finding thresholds.

Measuring scheme	Gebergeschwindigkeit (Encoder speed)
Functionality:	Recording the actual speed from system A and B over time. Recording the difference between the speed signal from system A and B over time.
Comment:	The process value for the speed is formed internally from speed value for system A. Recording the process values for speed and acceleration over time.
Application:	Scaling the encoder systems A and B in the case of speed monitoring. Given a correctly scaled encoder system there should be any significant difference between speed A and B. Analysis and change in the encoder signal over time for diagnostics (e.g. troubleshooting etc.).

Measuring scheme:	Daten SSX Baustein (SSX block data)
Functionality:	Recording the process data speed and acceleration over time. Recording the speed limit for the monitoring function over time.
Application:	The graphic shows the dynamic behaviour of the drive by displaying the speed and acceleration. The speed limit remains zero if the SSX function is not activated. On the activation of the SSX function the speed limit is applied from the current speed and calculated downwards. If the current speed of the drive remains below the speed limit, there is no shut down.

Measuring scheme:	SEL (Positionsbezogen) (SEL position related)
Functionality:	Recording the process data speed and acceleration against the position. Display of the current position in the form of a moving cursor. Display of the current stopping distance as a trailing pointer.
Application:	The graphic shows the dynamic value for the stopping distance as a minimum value for the braking distance. Checking the parameter values set in the SEL function, in particular the reserve retained for the shutdown.

Measuring scheme:	SLS Filter (SLS filter)
Functionality:	Monitoring the maximum speed of the monitoring function.
Application:	The graphic shows the current speed referred to the limit speed set. Checking the shutdown if the limit speed is exceeded. Display of the integrated speed.

Measuring scheme:	SCA Filter (SCA filter)
Functionality:	Monitoring the maximum speed of the monitoring function.
Application:	The graphic shows the current speed referred to the limit speed set.

9.2 Process Img., Function Block, Sys. Info, Enc. Position, Enc. Interface

The tabs available are listed in the following with their function.

Process Img.: Display of the states of all addresses in the input and output image in the servo drive (master).

Function Block: Enables you to monitor selectively the memory states of selected function blocks.

(See "9.2.1 Procedure during function block diagram diagnostics")

Sys. Info.: System information on the servo drive (master). The CRC of the active configuration as well as the state of an internal transfer counter are displayed. This counter is incremented on each transfer to the servo drive (master) and can be used as a reference for the documentation. The values are displayed after you have triggered "Debug Start" and the data have been compared.

Enc. Position: Shows the actual position values transferred for encoder A and encoder B. The normalised positions are displayed in the scope monitor.

Enc. Interface: Shows the voltage differences on the driver blocks and the state of the input jumpers on the encoder interface.

If one of the values for the voltage states is 0, the encoder is faulty or not connected.

The value for the input jumpers for incremental encoders is:

0 := Jumper OK

1 := Error

9.2.1 Procedure during function block diagram diagnostics

During function block diagram diagnostics the actual input and output states of the function blocks (0 or 1) are displayed in the function block diagram.

Selection of the blocks

If you have opened the Function Block tab, the program initially expects you to select function blocks on which the states are to be monitored. As soon as you have made a selection in the diagram, the "Add Block" button is no longer available. You can apply the selected function blocks from the function block diagram to the monitoring list using this "Add Block" button.

If the diagnostics have not yet been started, initially the symbol addresses related to the connectors are displayed in the diagram.

NOTE: These addresses related to the blocks are also given in the instruction list that is output in the Message Window when the program is compiled.

TIP: If all blocks are to be selected you can use the "Alles auswählen" (Select all) command from the context menu for the function block diagram. Move the mouse pointer over an empty area in the diagram and press the right mouse button.

To select mutually dependent function blocks, use the "Signal Trace" command from the context menu for a function block. For this purpose move the mouse pointer over a single selected function block and press the right mouse button.

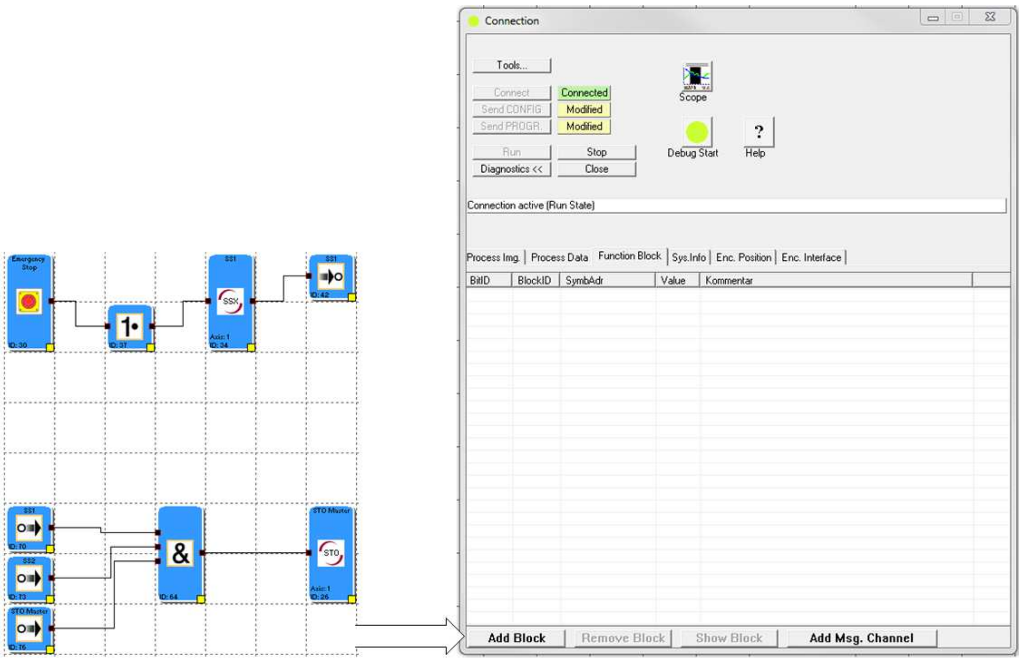


Figure 14: Selection of the function blocks

Starting the debugger

If the servo drive (master) is in the run mode, the "Debug start" button is enabled. If you click this button, first a plausibility check between the function block diagram and the servo drive (master) is undertaken (message in the Message Window). This check uses the program CRC to ensure the data are synchronised. If the check is successful, the memory states (0 or 1) for the related addresses in the servo drive (master) are displayed both in the monitoring list and in the function block diagram.

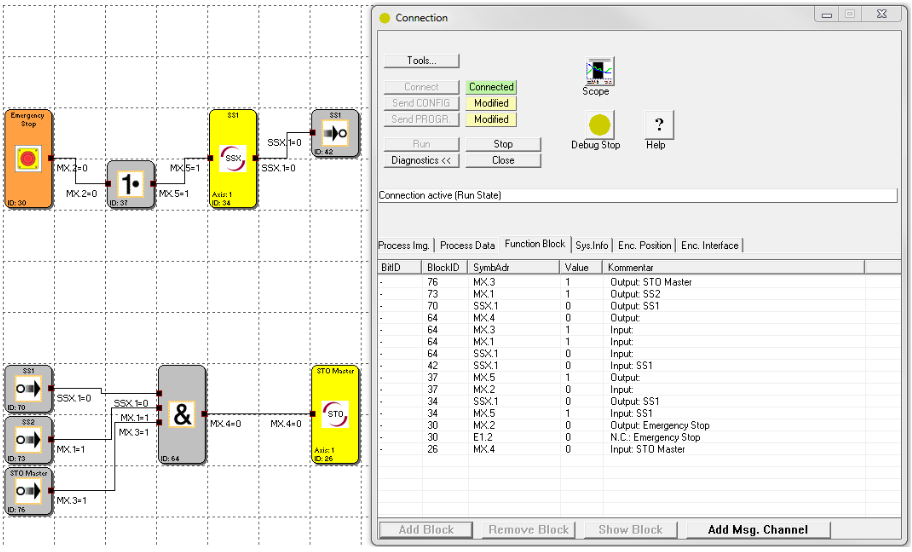


Figure 15: Function block diagram analysis

Remove Block

Click this button to delete the related display of selected entries in the monitoring list.

Show Block

Click this button to centre selected entries in the function block diagram window.

NOTE: The values that are displayed in the debugger mode are updated approx. every 50 ms.

If the servo drive (master) changes to an alarm state, the process image is no longer updated. Changing levels on the inputs no longer have any effect and are also not displayed in the diagnostics. If you change from "Function Block" to a different diagnostics mode using the tabs (e.g. "Enc. Position"), a question mark is displayed instead of the address value to indicate that the value cannot be displayed.

TIP: Double-click a list entry in the function block list to display the block in the function block diagram (quick jump).

10 TOOLS FOR VALIDATION

Always define a validation plan. The tests and analyses you have used to demonstrate the compliance of the solution with the requirements from your application are defined in the plan.

The "Tools…" dialog box helps you to obtain the necessary documentation for the integrated safety control.

In addition, you can save the validated program and configuration data in the dialog box.

Tools…

A dialog box opens…

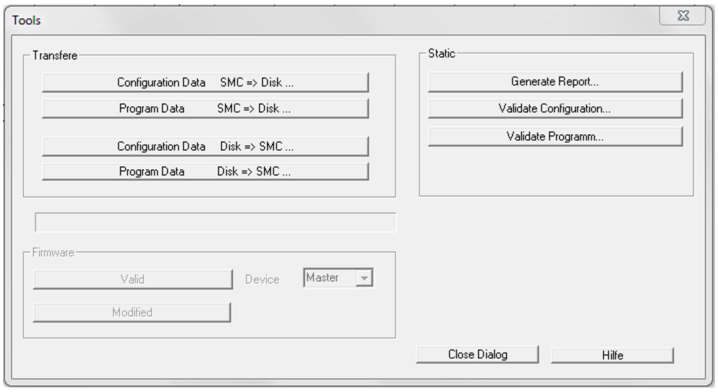


Figure 16: Tools for validation

Configuration Data SMC => Disk …

Reads the configuration data saved on the servo drive (master) and saves them on the hard disk.

Program Data SMC => Disk …

Reads the program data saved on the servo drive (master) and saves them on the hard disk.

NOTE: The program and configuration data should be read at the end of the validation so that the final state is archived and can be used for identical applications.

Configuration Data Disk => SMC…

Transfers the configuration data saved on the hard disk to the servo drive (master).

Program Data Disk => SMC...

Transfers the program data saved on the hard disk to the servo drive (master).

NOTE: Check the CRC for the program data and configuration data to make sure the correct program is loaded in the servo drive (master) (see "**Retrieving the CRCs**“).

Generate Report...

Reads the program and configuration data saved in the servo drive (master) and saves this data on the hard disk in text form as a validation report (see "**Validation report**").

Validate Configuration...

Click this button to confirm the successful validation of the configuration data saved in the servo drive (master).

Validate Program...

Click this button to confirm the successful validation of the program data saved in the servo drive (master).

10.1 Validation report

Using the validation report you can save the instruction list program generated and the monitoring parameters configured for the function blocks in an Excel file for documentation purposes.

The printed file is used as a template for the safety-related check.

NOTE: You can only generate the report after you have given a new "unnamed" function block diagram a file name and saved it. Also there must be an active connection established to the servo drive (master) on which the Safety PLC program is running and that is in the "RUN" state. The file generated (*.xls) then has the same name and is saved in the same directory as the related function block diagram.

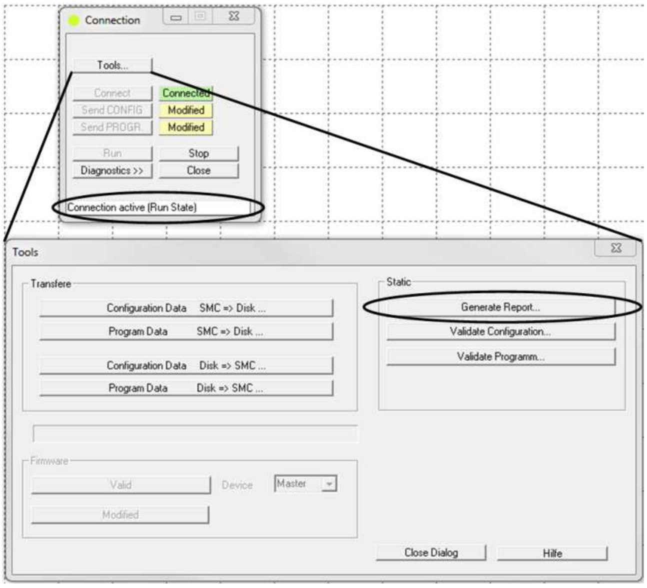


Figure 17: Generating validation report

Only after the safety-related check and completion of the validation report can you click the "Validate Configuration…" and "Validate Program…" buttons.

Only then is the MSDFS correctly validated.

10.2 Completing the validation report

The validation report is divided into various parts that must be completed during the safety-related check.

Abnahme (Acceptance)

Both inspectors are to be identified by name here. After the check has been completed successfully, the printed validation report is to be signed here.

Ansprechpartner (Contacts)

Anlage (System): Describes the system to be monitored.

General name, version, system code used internally

Kunde (Customer): Organisation operating the system

Here you can enter the customer's name, telephone number and fax number.

Lieferant (Supplier): Manufacturer of the safety control

Here you can enter the supplier's name, telephone number and fax number.

Errichter (Installer): Organisation that is placing the system in operation

Here you can enter the name of the company placing the system in operation, its telephone number and fax number.

Anlagebeschreibung (System description)

These fields must be completed in hand writing after printing out the report.

Anlagenbezeichnung (System designator): Describes the functionality or the application area of the system

Aufstellungsort (Installation location): Describes the exact location of the system

Endkunde (End customer): Organisation operating the system

Kurzbeschreibung (Brief description): Safety-critical features of the system

Funktionsbeschreibung (Function description): Safety-critical features of the system to be monitored by the safety device.

Einzelprüfung der verwendeten Systemkomponenten und Funktionen (Individual check on the system components and functions used)

Here the inspector must again validate all configured data in the printed out report, in addition to a visual inspection.

It must be checked the parameter and program CRCs and the CRCs displayed in the device match.

NOTE: The CRCs can be read on the status display (D1 and D2 on the MSDFS) (see "**Retrieving the CRCs**")

The correct program and parameter data must be loaded for the preparation of the validation report!

All limits set for the monitoring functions used must be checked.

11 RETRIEVING THE CRCs

There is a configuration data CRC and a program data CRC for the safety program saved in the integrated safety control in the servo drive (master). For both CRCs the CRCs displayed in the Safety PLC user interface must match the CRCs that can be seen in the servo drive (master).

The CRC for the program and configuration data can be viewed in three places.

1) CRCs in Safety PLC (offline)

You can view the program and configuration data CRCs for a compiled program in Safety PLC in "Layout Management".

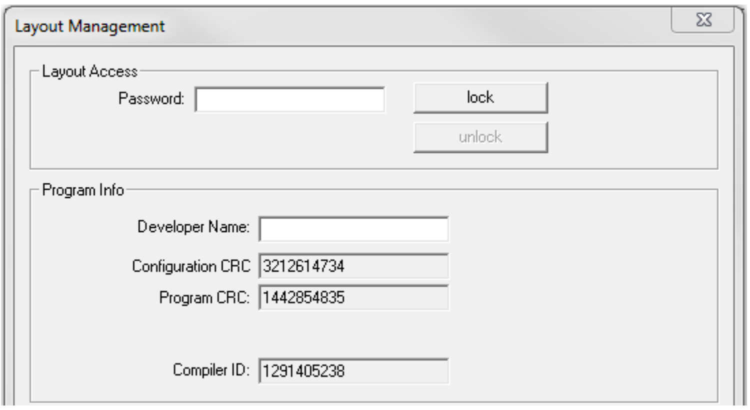


Figure 18: CRCs in Layout Management

NOTE: The CRCs displayed in Layout Management are the CRCs for the Safety PLC program currently opened and compiled.

2) CRCs in the Connection dialog box (online)

 As soon as you have established a connection to the servo drive (master) using the Connection dialog box, you can read the program and configuration data CRCs for a program in the servo drive (master) in Sys. Info.

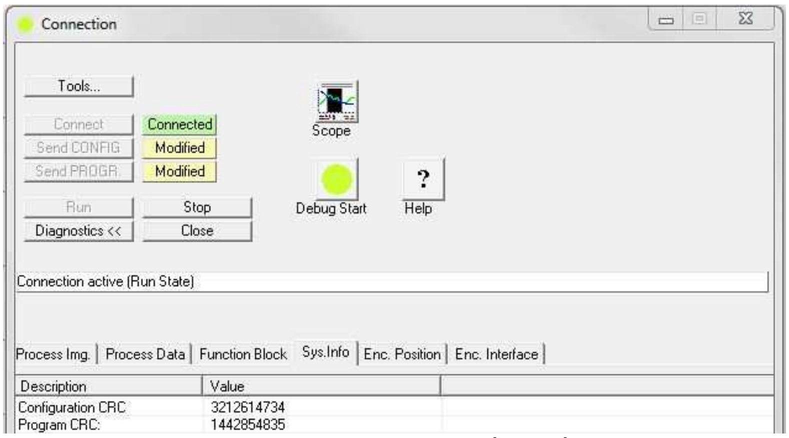


Figure 19: CRCs in the Connection dialog box (online)

NOTE: The CRCs displayed in the Connection dialog box are the CRCs for the program currently saved in the servo drive (master). These CRCs must always match the CRCs that can be read on the servo drive (master).

3) CRCs on the servo drive (master)

The program and configuration data CRCs for the program in the servo drive (master) can be read on the 7-segment display. For this purpose the servo drive (master) must be free of errors and the integrated safety control must be in the "RUN (4)" mode.

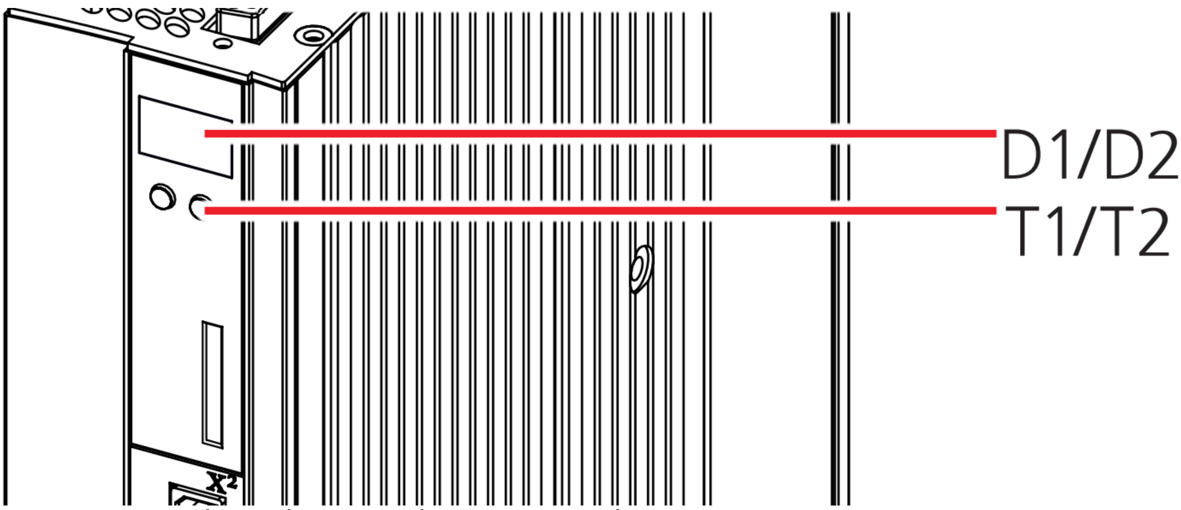


Figure 20: Servo drive (master) display D1/D2 and button T1/T2

Using the T1 and T2 buttons you can open the menu for the integrated safety control. The figure below shows the button sequence for reading the CRCs.

Button sequence for retrieving the CRCs on the servo drive (master)

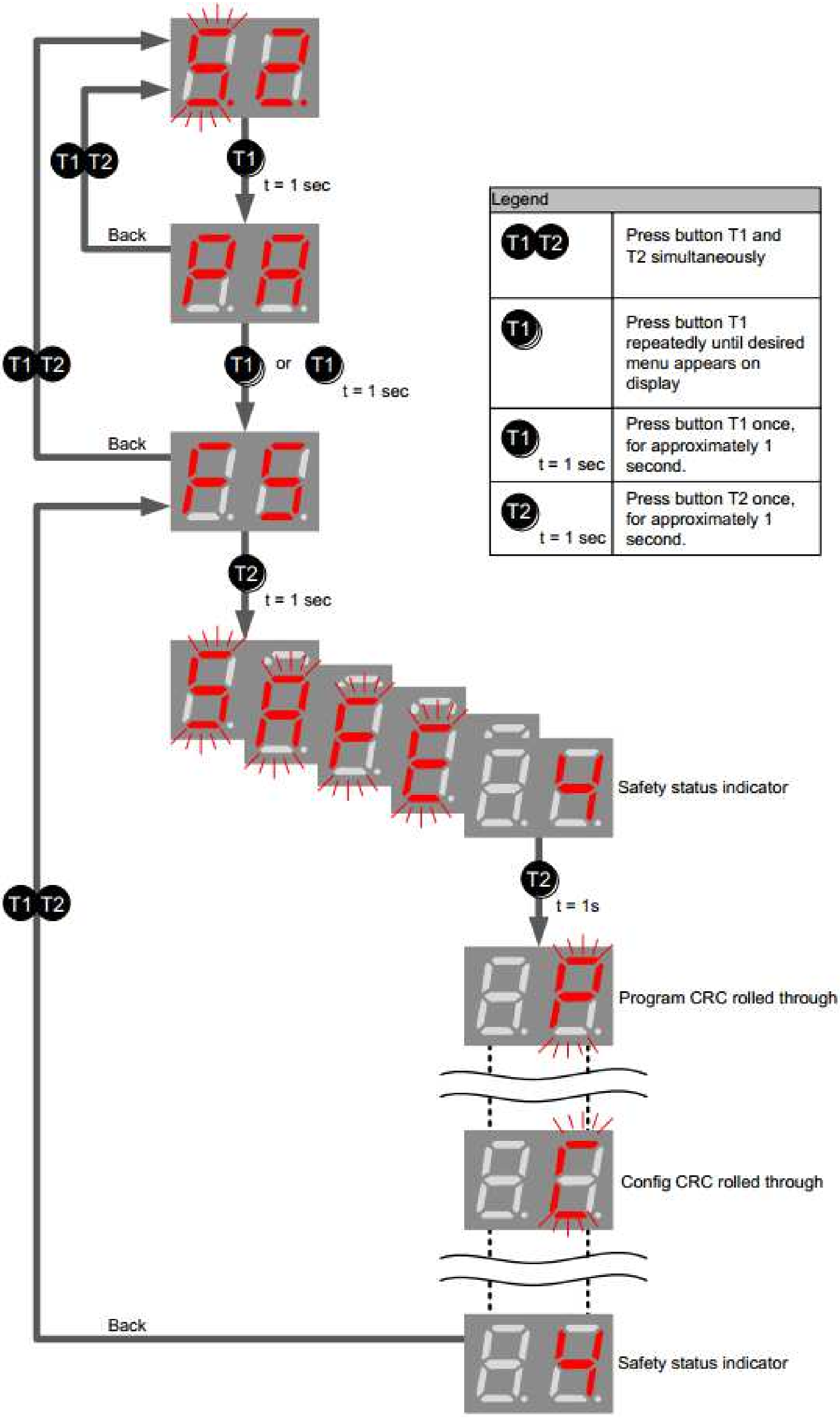


Figure 21: Retrieving CRC on the servo drive (master)

12 AIDS DURING PROGRAM DEVELOPMENT

Information display

If information display is enabled, the attributes of the element currently under the mouse pointer are displayed. The information display has a delay similar to a Windows tooltip. This delay can be set in the "Einstellungen" (Settings) dialog box. The information display can be activated by pressing "Ctrl" key if necessary. The information is displayed until this key is released again.

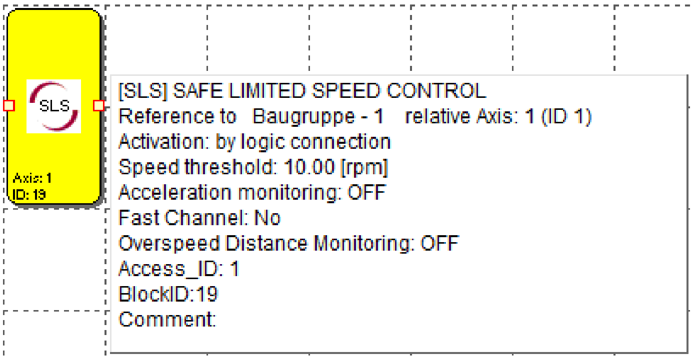


Figure 22: Information display

Signal trace

This command selects all other function blocks connected to the currently selected block. In this way the related block links can be displayed.

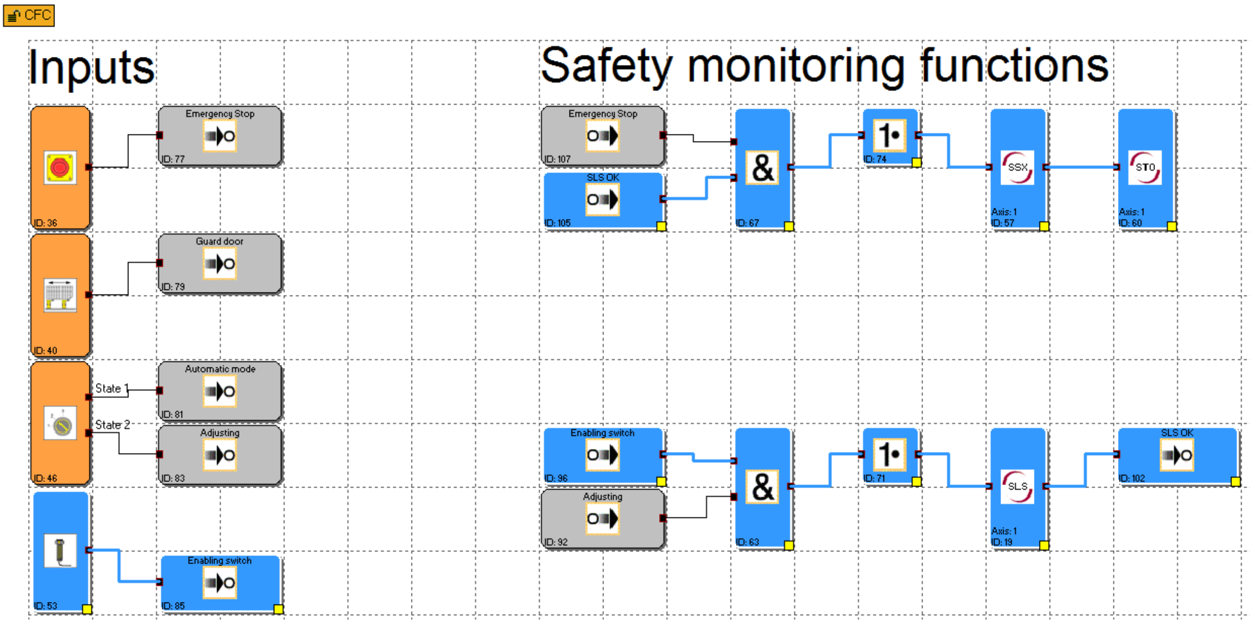


Figure 23: Signal trace

NOTE: This command is only active if just one function block has been selected.

TIP: Using this function you can find, e.g., blocks that are connected indirectly to a "Set flag" block.

All attributes related to a block selection can also be output in the Message Window. For this purpose use the "Edit -> Attributes in Message Window" menu command or the function block context menu.

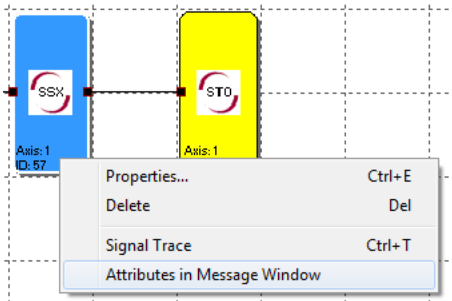


Figure 24: Copying attributes to the Message Window

NOTE: If you run the command via the context menu, it is important the mouse pointer is over a selected block, as otherwise the selection of other blocks will be cleared

TIP: If you want to display the attributes of <u>all</u> blocks in the Message Window, run the menu command without any elements selected.

Quick jump

If you <u>double-click</u> the block IDs marked in colour in the Message Window you can display the related block in the middle of the function block diagram window. In this way you can quickly find the function blocks related to a task and make any necessary changes.

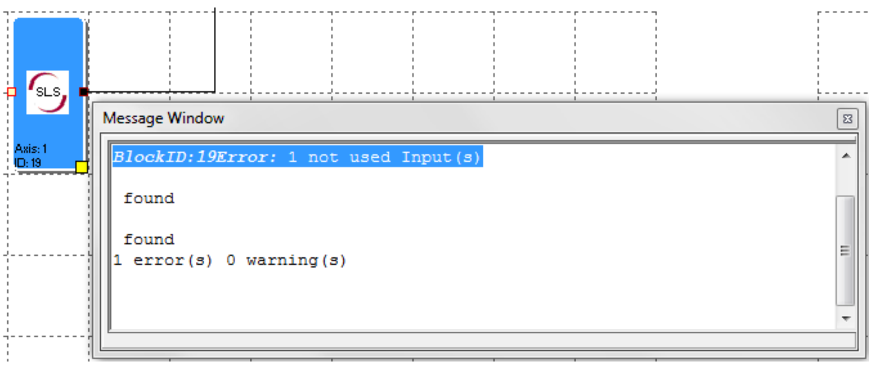


Figure 25: Quick jump

Pre-defined function blocks

These elements represent the inputs and outputs available on the MSDFS in the terminal diagram.

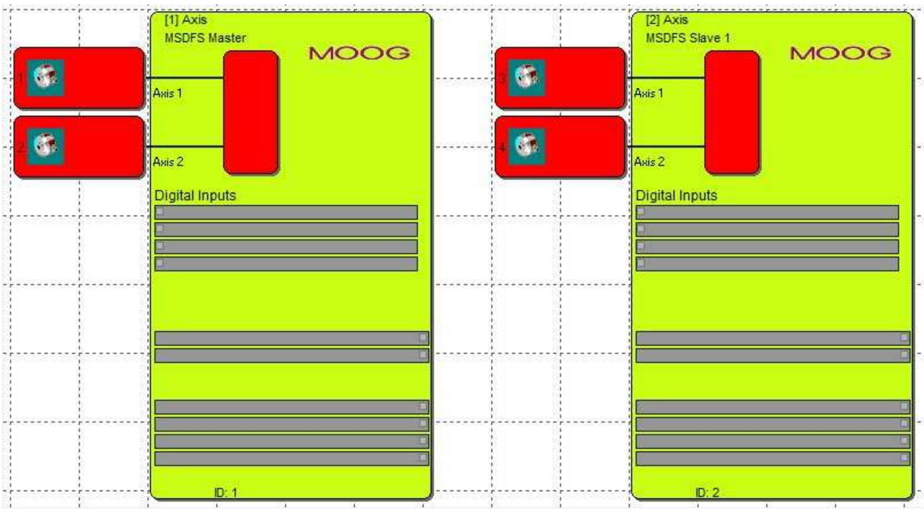


Figure 26: Pre-defined function blocks

MSDFS inputs and outputs

This block describes the speed and position sensors as well as the signal list for digital inputs and outputs. You can select the individual areas and modify them by either double-clicking, or using the "Properties..." context menu command.

Speed and position sensors

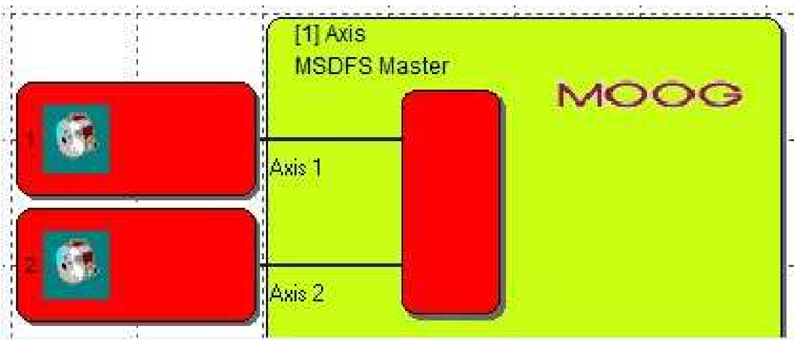


Figure 27: Speed and position sensors

Double-click an encoder element to open the dialog box for the encoder configuration. The parameters to be entered are described in detail in the MSDFS model description.

Digital inputs

In this area the input signals are linked to the peripheral elements. The link is made automatically on adding the function blocks described below. Double-click a signal list to open a comment window. Here you can enter descriptive text for the input.

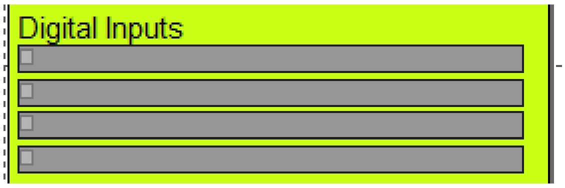


Figure 28: Digital inputs

Digital outputs

This block comprises the signal lists for the semiconductor outputs and the brake output. As for the input signals the wiring is undertaken automatically on adding the related function blocks.

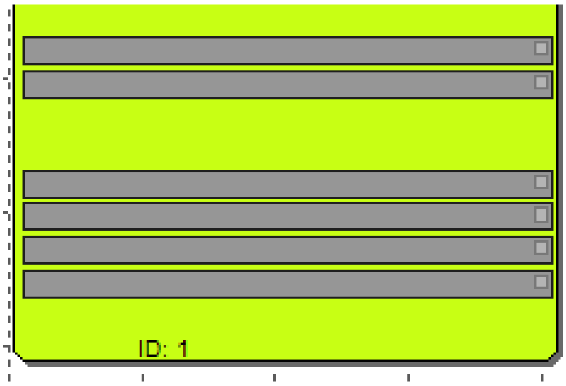


Figure 29: Digital outputs

13 LAYOUT MANAGEMENT

 Using Layout Management you can lock the function block diagrams against unintentional or unauthorised modifications. It is also possible to document the preparation of the program.

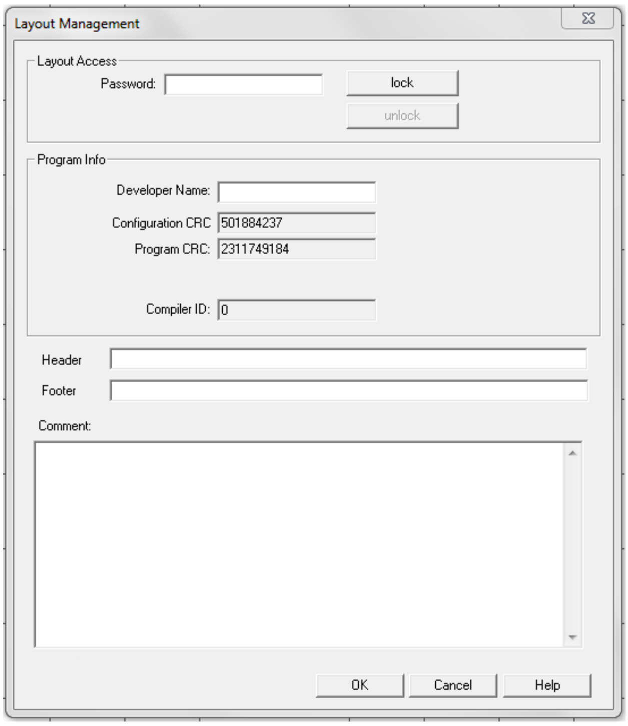


Figure 30: Layout Management

<u>Layout Access</u>

Here you lock or enable the access to function blocks in the current function block diagram. This means that all menu commands and toolbars for the addition of function blocks are dimmed (= locked) on a locked function block diagram. It is also not possible to change any parameters in function blocks already added.

Password

A password must be assigned for "unlocking". The values and the function blocks in a locked diagram can be viewed, but not modified. This functionality prevents unauthorised persons making changes to a function block diagram.

If a function block diagram is locked, on leaving Layout Management the "Datei Speichern" (Save file) dialog box appears so that any changes made are not lost.

NOTE: You can only unlock function block diagrams using the password assigned on "locking". It is no longer possible to compile a locked function block diagram! However, it is possible to access the servo drive (master).

Program Info

This information is used to document changes or other specific items of information that are to be saved with the function block diagram.

Developer Name

Name of the responsible developer.

Header

Footer

Configuration CRC

Checksum for the Safety PLC configuration data that were generated from the function block diagram elements during the last compiler run. This value is used to compare the programs during debugging. The value does not change if nothing has been changed in the configuration of the function block diagram elements used.

Program CRC

Checksum for the Safety PLC program data that were generated from linking the function block diagram elements during the last compiler run. This value is used to compare the programs during debugging. The value does not change if nothing has been changed in the existing links for the function block diagram elements.

NOTE: If the CRCs are used for the test report, it is recommended to lock the function block diagram, as in this way unintentional modification will be prevented!

Comment

You can save comments on a function block diagram in this field. It is strongly recommended to log the changes to a program using a revision scheme.

14 ADDING INPUT BLOCKS

The input elements provide the data on the operating state of the system monitored by the MSDFS. These components, outside the device from the point of view of the MSDFS, can only be added and configured in the terminal diagram.

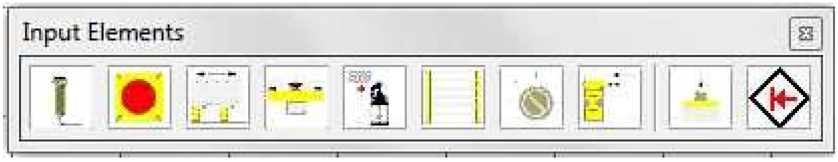


Figure 31: Input Elements

The automatic resource monitoring of the function block elements for the MSDFS ensures that only the elements available are enabled in the program. This aspect affects above all the time-monitored peripheral devices. If there are no more resources (memory) in the MSDFS for the monitoring program, it is no longer possible to add further function blocks to the program. These resources can be enabled again by deleting corresponding function blocks.

The peripheral components are structured by their usage and by input signal type such that targeted MSDFS resource monitoring is possible.

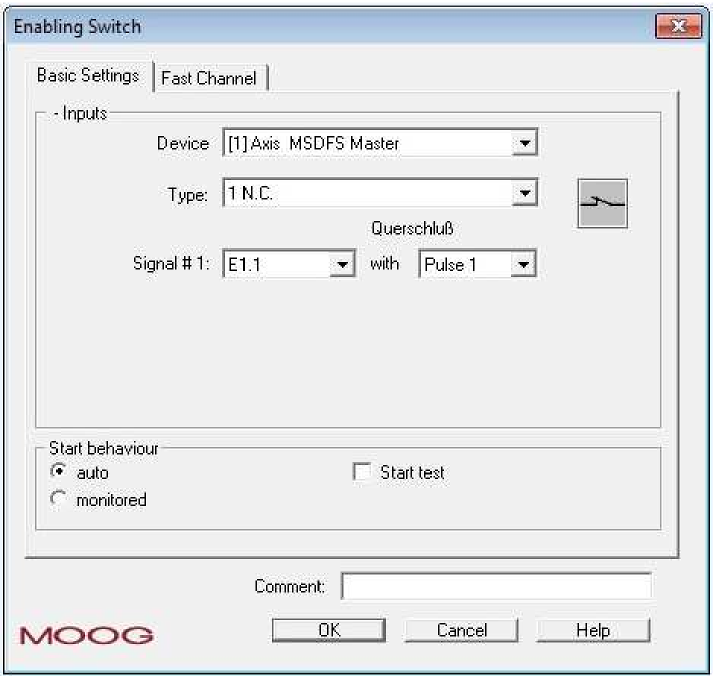


Figure 32: Basic Settings input block

Basic Settings

Device

If there are several MSDFS, you can define here the MSDFS to which the input element is to be connected.

Type

Switch type used in the component that is connected to the MSDFS. The number of related input signals and the monitoring behaviour of the MSDFS change depending on the selection.

NOTE: With *time-monitored switch elements* a further signal change must occur within $t = 2$ s after the occurrence of the first signal. If this does not happen, a malfunction is detected and a corresponding alarm message sent.

Signal #:

Number assigned to the signal on the MSDFS digital input. The input signal designators still free (e.g. "E1.1") on the MSDFS are displayed in this list box. The user assigns these numbers. It is not allowed to make duplicate assignments to input signals. If the resources of the MSDFS are almost exhausted and too many input signals would be used due to the switch type selected, the list box is empty. The input block cannot be added. In this case a switch type with fewer connections must be used.

Querschluß (Cross circuit check)

Number of the pulse that is to be present on the input signal. To ensure reliable monitoring for short circuit or wire break, you should use different pulse numbers on adjacent inputs on the MSDFS. If this is not the case, a warning will be output.

Start behaviour

Using these settings you can define how the peripheral devices are to behave on switching on the system or a device reset.

Auto

This default start type makes it possible to start the MSDFS without the need for feedback from the user.

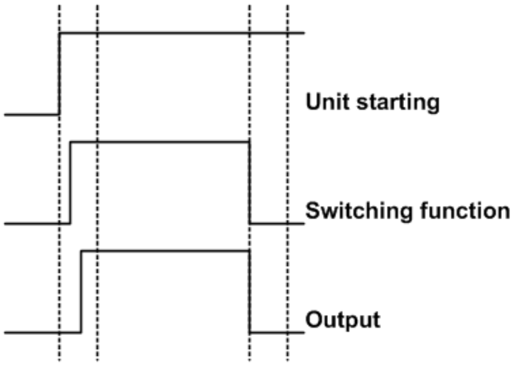
Start type	Function	Schematic
Automatic start	Automatic start after device reset. Output on the input element is 1 if safety circuit is closed/active as per definition of switch type	

Table 1: Automatic start

Monitored

Monitored input element enabled on falling edge on the stated monitoring input. This enable is required every time the monitored input element is to be switched. Example: Start a servo only once confirmation has been provided by the operator.

With the monitored start type an additional connector for the connection to a **start element** is provided. In this element you can further configure the behaviour for monitoring the component on starting the system using the MSDFS.

Start test

Manual start after device reset with test on the monitoring device connected. The monitoring device must trigger 1x in the monitoring direction and switch on again. Normal operation then follows. This one-off actuation of the input element on starting (or resetting) the monitored system ensures the input element functions at the time of the start. A start test can be undertaken for two input elements, then the dialog box element is no longer available.

If the start test is set, this situation is indicated by a *red square on the function block added*.

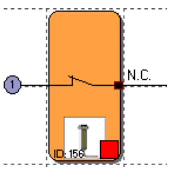
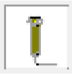


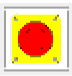
Figure 33: Start test

14.1 Enable button



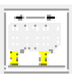
Switch type	Designation	Comment
1 (eSwitch_1o)	1 normally closed contact	Simple enable button SIL 2
2 (eSwitch_1s)	1 normally open contact	
3 (eSwitch_2o)	2 normally closed contacts	Enable button increased requirement SIL 3
4 (eSwitch_2oT)	2 normally closed contacts time monitoring	Monitored enable button SIL 3

14.2 Emergency stop



Switch type	Designation	Comment
1 (eSwitch_1o)	1 normally closed contact	Simple emergency stop SIL 2
3 (eSwitch_2o)	2 normally closed contacts	Emergency stop increased requirement SIL 3
4 (eSwitch_2oT)	2 normally closed contacts time monitoring	Monitored emergency stop SIL 3

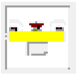
14.3 Door monitoring



Switch type	Designation	Comment
3 eSwitch_2o	2 normally closed contacts	Door monitoring increased requirement SIL 3
4 eSwitch_2oT	2 normally closed contacts time monitoring	Monitored door monitoring SIL 3
5 eSwitch_1s1o	1 normally open contact + 1 normally closed contact	Door monitoring increased requirement SIL 3

6 eSwitch_1s1oT	1 normally open contact + 1 normally closed contact time monitored	Monitored door monitoring SIL 3
7 eSwitch_2s2o	2 normally open contacts + 2 normally closed contacts	Door monitoring increased requirement SIL 3
8 eSwitch_2s2oT	2 normally open contacts + 2 normally closed contacts time monitored	Monitored door monitoring SIL 3
9 eSwitch_3o	3 normally closed contacts	Door monitoring increased requirement SIL 3
10 eSwitch_3oT	3 normally closed contacts time monitored	Monitored door monitoring SIL 3

14.4 Two-hand buttons



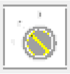
Switch type	Comment	Category categorisation	SIL categorisation
2 changeover contacts	Two-hand buttons increased requirement	Type III C category 4	SIL 3
2 normally open contacts	Monitored two-hand buttons	Type III A category 2	SIL 1

NOTE: For these input elements there is a fixed pulse assignment that cannot be changed by the user!

14.5 Light curtain

Switch type	Designation	Comment
3 eSwitch_2o	2 normally closed contacts	Light curtain increased requirement SIL 3
4 eSwitch_2oT	2 normally closed contacts time monitoring	Monitored light curtain SIL 3
5 eSwitch_1s1o	1 normally open contact + 1 normally closed contact	Light curtain increased requirement SIL 3
6 eSwitch_1s1oT	1 normally open contact + 1 normally closed contact time monitored	Monitored light curtain SIL 3

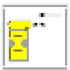
14.6 Mode selector switch



Switch type	Designation	Comment
13	Selector switch - normally closed contact/normally open contact	Monitored mode selector switch SIL 3
14	Selector switch 3 stages	Monitored mode selector switch SIL 3

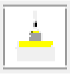
NOTE: On a change in the state of the switch it is to be ensured by the Safety PLC program to be prepared that the requirement from IEC/EN 60204-1 (section 9.2.3 Operation modes) is met.

14.7 Sensor



Switch type	Designation	Comment
1 eSwitch_1o	1 normally closed contact	Simple sensor input SIL 2
2 sSwitch_1s	1 normally open contact	Simple sensor input SIL 2
3 eSwitch_2o	2 normally closed contacts	Sensor input increased requirement SIL 3
4 eSwitch_2oT	2 normally closed contacts time monitoring	Monitored sensor input SIL 3
5 eSwitch_1s1oT	1 normally open contact + 1 normally closed contact time monitored	Monitored sensor input SIL 3

14.8 Start / reset switch



This input element offers both expanded monitoring functionality and can be used to reset an alarm that has occurred.

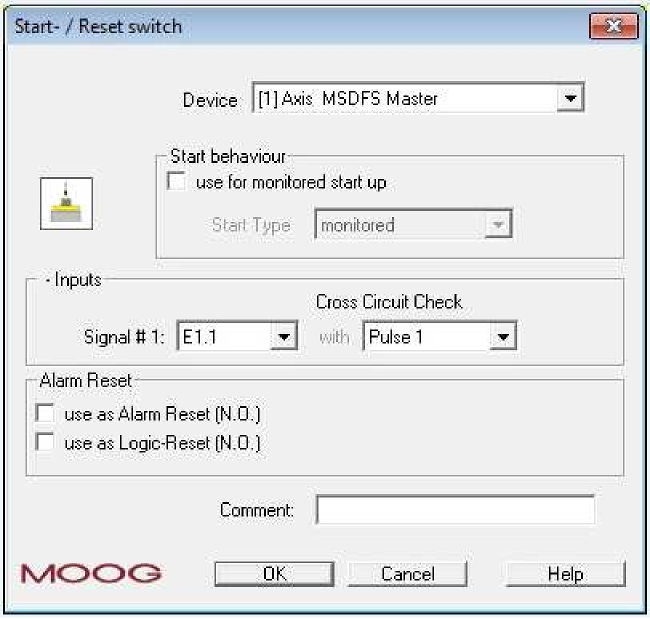


Figure 34: Start / Reset switch

Inputs

As for the other input elements, you can select the input on the MSDFS to be used using this list box. If only the start of other input elements is to be monitored (start behaviour), then the input number can be arbitrarily assigned.

14.8.1 Start behaviour

Use for monitored start up

If start monitoring is set, special instruction list code segments for monitoring an input segment when a system is started or reset are generated. This function check on a peripheral element (e.g. actuation of the emergency stop button) is intended to ensure it is functioning correctly on starting the system.

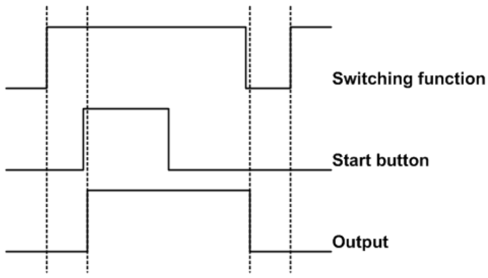
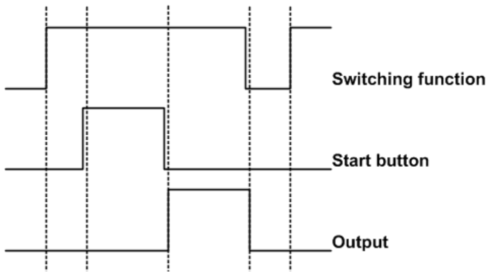
Start type	Function	Scheme
Manual start	Manual start after device reset. Output on the input element is 1 if safety circuit is closed/active as per definition of switch type and start button has been pressed 1 x. Output is 0 after safety circuit open. E1: Switching function E2: Start button M.(X1): Auxiliary flag 1	
Monitored start	Manual start after device reset with monitoring of the start circuit for static 1 signal. Output on the input element is 1 if safety circuit is closed/active as per definition of switch type and start button has been pressed 1 x and released again. Output is 0 after safety circuit open. E1: Switching function E2: Start button M.(X1): Auxiliary flag 1 M.(X2): Auxiliary flag 2	

Table 2: Manual start

The monitoring input on the start element is to be connected to the output on the input elements labelled "Start element". Several elements can be monitored e.g.:

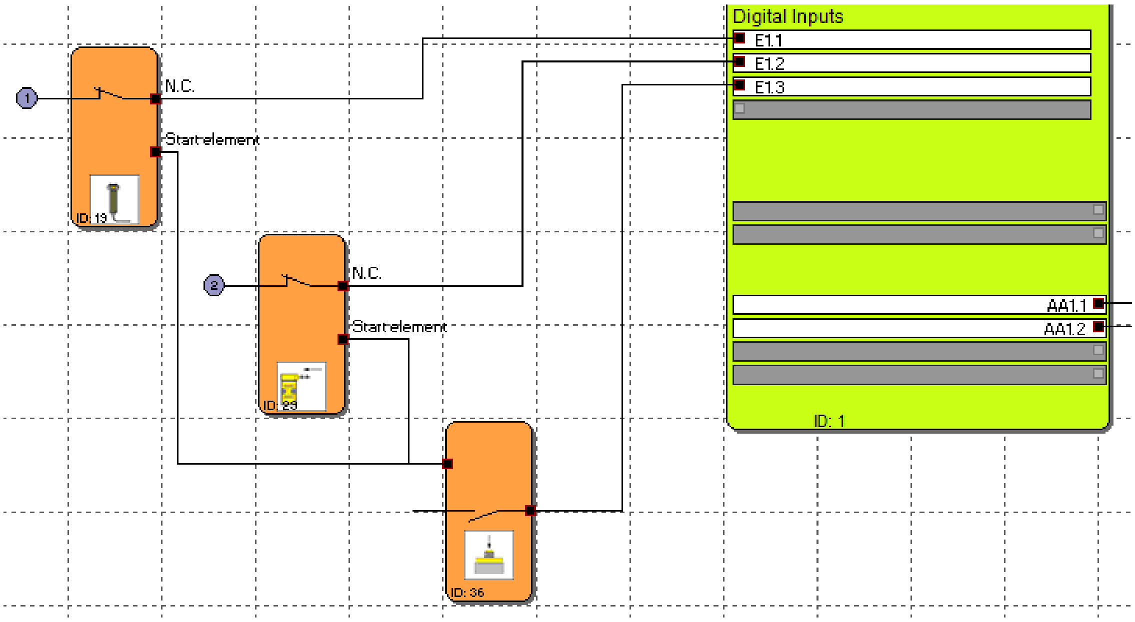


Figure 35: Connection of the start element

NOTE: On editing the related input element the connection to the start element is deleted and cannot be re-established automatically. It must be added manually.

14.8.2 Alarm Reset

Use as Alarm Reset (N.O.)

If this check box is selected, a malfunction that occurs during operation can be reset (acknowledged) using the related button. No special program code is generated, instead this input is processed directly by the MSDFS in the event of an alarm.

NOTE: If a reset element is used, it is not possible to process any short circuit monitoring for this input. In this situation the cross circuit check is set to "AUS" (OFF) when you close the dialog box.

The reset element can only be used on the servo drive (master).

The alarm reset input can be operated with a continuous voltage of 24 V (without test pulses) and is edge controlled.

A restart inhibit is not implemented in the alarm reset. Should a restart inhibit be necessary to safeguard the machine/system, you must program it into the logic.

Use as Logic-Reset (N.O.)

If this additional check box is selected, the function block output is provided in the function block diagram. Here you can use the output to link to logic functionality. This feature is provided so that an SCA error that occurs in a RS block remains set and is only reset after the actuation of the reset button on the RS block, e.g.:

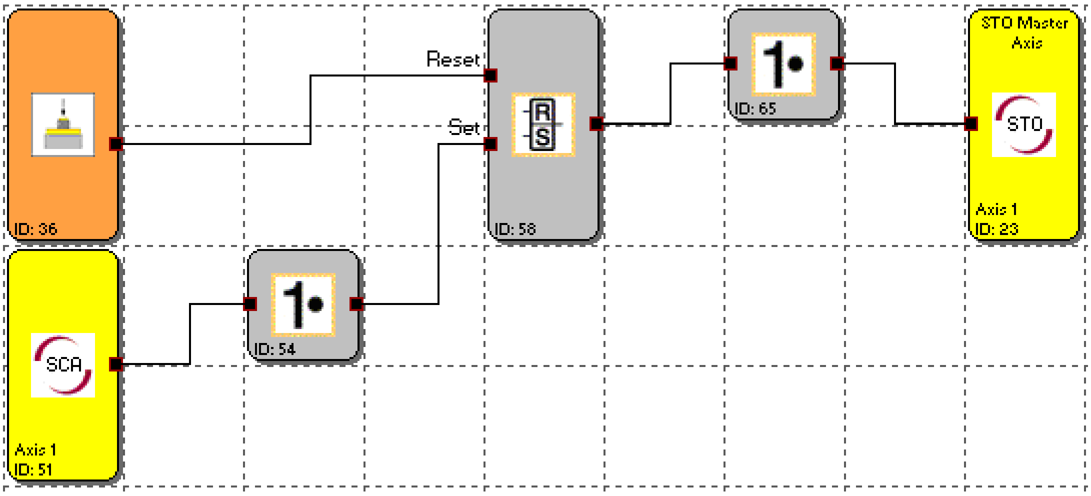
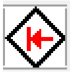


Figure 36: Logic reset

NOTE: If the logic reset is simply used with a normally open contact, PL d in category 3 as per EN ISO 13849-1, or SIL 2 as per IEC/EN 62061 can be achieved.

14.9 Functional input



A functional input has an input and an output connector; here the input connector must be connected to a signal from the function block diagram. Internally the functional input is processed with a non-safe signal from the servo drive using a logical AND operator. Which signal from the servo drive is to be processed in the function block diagram can be set in the servo drive. The output connector for the functional input is logical 1 if the signal from the function block diagram AND the non-safe signal from the servo drive are logical 1. In this way non-safe signals from the servo drive can be processed in the safe section.

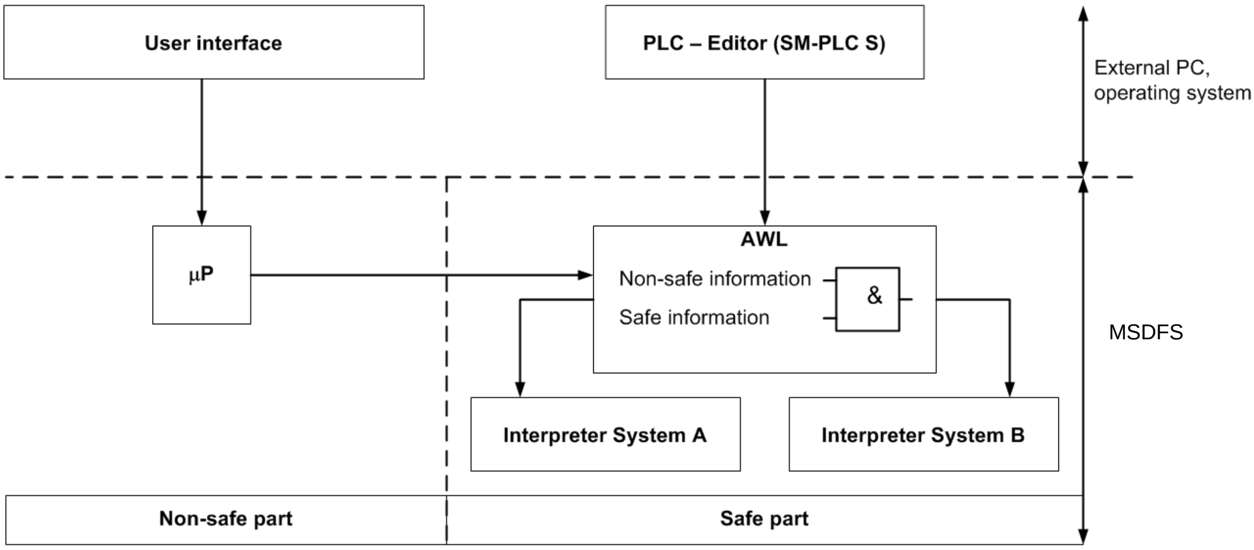


Figure 37: Information linking

Non-safe information

The following information from the servo drive can be processed as non-safe information in the safe part:

- Status information as per the function selection for functional outputs (see **Appendix 2 Functions of the functional outputs**)
- Information on field bus
- Information on PLC
- Information on parameter access

Safe information

The following information from the integrated safety control can be processed with non-safe information from the servo drive using an AND operator in the safe part:

- All function blocks that have an output connector

NOTE: So that the logical AND operation on the non-safe and safe information provides a safe result, a signal from the Safety PLC function block diagram that in safety engineering terms is also considered safe is to be used. As such processing with a **permanent logical "1"** block is possible, but will not provide a safe signal in safety engineering terms.

Every user in the axis group provides a specific number of functional inputs that can be used in the Safety PLC as required.

Number and designation of the functional inputs per axis or servo drive		
Axis	**Number of functional inputs**	**Designation of the functional inputs**
MSDFS master	32 8	FE0.1 to FE0.32 FE1.1 to FE1.8
MSDFS slave 1	8	FE2.1 to FE2.8
MSDFS slave 2	8	FE3.1 to FE3.8
MSDFS slave 3	8	FE4.1 to FE4.8
MSDFS slave 4	8	FE5.1 to FE5.8
MSDFS slave 5	8	FE6.1 to FE6.8

Non-safe information

For example to output a status message from the servo drive (non-safe part) on a digital output in the safe part, any safe signal from the servo drive can be processed with a permanent logical 1 block. On the transfer of information from the servo drive (non-safe part) it is to be ensured that safe signals are not muted by non-safe signals. Here particular care is required.

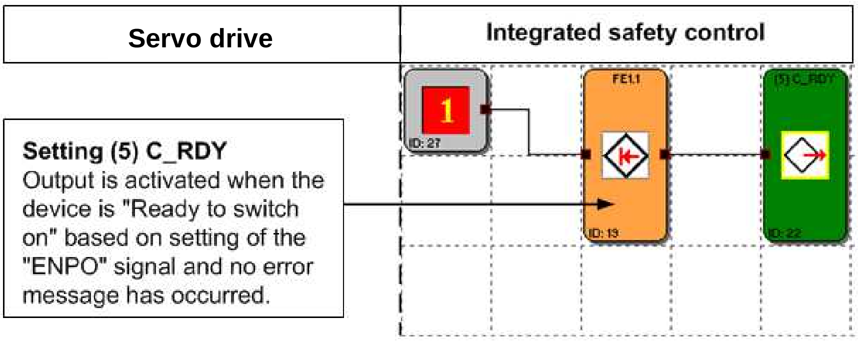


Figure 38: Non-safe status message from servo drive on digital output

14.9.1 Function selection on the functional inputs

You can define the function the related functional input is to have via a function selection in the servo drive. **Parameter P1808** (selector) is used for this purpose and can be accessed using DRIVEADMINISTRATOR 5.

Parameter 1808 MPRO_OUTPUT_FS_FKTIN [0] to [15]

<table>
<tr><td>Subject area</td><td>Functional settings</td></tr>
<tr><td>Values</td><td>
<table>
<tr><td>Index</td><td>Min value</td><td>Max value</td><td>Standard</td><td>Type</td><td>LE</td><td>SE</td></tr>
<tr><td>[0]</td><td>0</td><td>65535</td><td>0</td><td>Uint16</td><td>1</td><td>1</td></tr>
<tr><td>[1]</td><td>0</td><td>65535</td><td>0</td><td>Uint16</td><td>1</td><td>1</td></tr>
<tr><td colspan="7">To</td></tr>
<tr><td>[14]</td><td>0</td><td>65535</td><td>0</td><td>Uint16</td><td>1</td><td>1</td></tr>
<tr><td>[15]</td><td>0</td><td>65535</td><td>0</td><td>Uint16</td><td>1</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>Setting</td><td>
<table>
<tr><td>Index</td><td>[0]</td><td>[1]</td><td>[2]</td><td>[3]</td><td>[4]</td><td>[5]</td><td>[6]</td><td>[7]</td></tr>
<tr><td>Functional input Safety PLC</td><td>FE*x.1</td><td>FE*x.2</td><td>FE*x.3</td><td>FE*x.4</td><td>FE*x.5</td><td>FE*x.6</td><td>FE*x.7</td><td>FE*x.8</td></tr>
<tr><td colspan="9">Setting: See <u>Function selectors digital outputs in the servo drive</u>
*x Placeholder for related device, where the following numbering is assigned:
1 = axis 1 (master), 2 = axis 2 (slave 1), 3 = axis 3 (slave 2),
4 = axis 4 (slave 3), 5 = axis 5 (slave 4), 6 = axis 6 (slave 5)</td></tr>
</table>
<table>
<tr><td>Index</td><td>[8]</td><td>[9]</td><td>[10]</td><td>[11]</td><td>[12]</td><td>[13]</td><td>[14]</td><td>[15]</td></tr>
<tr><td>Functional input Safety PLC</td><td>FE0.1</td><td>FE0.2</td><td>FE0.3</td><td>FE0.4</td><td>FE0.5</td><td>FE0.6</td><td>FE0.7</td><td>FE0.8</td></tr>
<tr><td colspan="9">Setting: See <u>Function selectors digital outputs in the servo drive</u>
FE0.1 to FE0.8 can only be used on the master axis. Index [8] to [15] is processed with index [0] to [7] from parameters 141 using an OR operator.</td></tr>
</table>
</td></tr>
</table>

ℹ	**NOTE:** In this parameter it is defined how the related functional input in the integrated safety control is to be behave. The functional input must be linked to a safe signal in the Safety PLC. As such the functional input is processed with the safe signal using an AND operator.

14.9.2 Setting for the functional inputs in the MSDFS master

The servo drive (master) provides a total of 40 functional inputs of which 16 inputs can be assigned a function. The remaining 24 inputs are permanently assigned to the access via field bus, MSD PLC and parameters. The result is the following assignment:

FE1.1 to FE1.8 and FE0.1 to FE0.8

The functional inputs FE1.1 to FE1.8 can be assigned a function using the parameter 1808 in the servo drive.

The functional inputs FE0.1 to FE0.8 can be assigned a function using the parameter 1808 in the servo drive and at the same time logically processed with **parameter 141** using an OR operator.

The setting for the functional inputs FE1.1 to FE1.8 and FE0.1 to FE0.8 in the servo drive (master) is shown graphically in Figure 39.

FE0.9 to FE0.32

The functional inputs FE0.9 to FE0.32 are permanently assigned to the controller via **parameter 141**. They can be controlled directly via field bus, MSD PLC or parameter access.

Parameter 141 OUTPUT_CTRL

<table>
<tr><td>Subject area</td><td>Functional settings</td></tr>
<tr><td>Values</td><td>
<table>
<tr><td>Index</td><td>Min value</td><td>Max value</td><td>Standard</td><td>Type</td><td>LE</td><td>SE</td></tr>
<tr><td>[0]</td><td>00000000h</td><td>FFFFFFFFh</td><td>00000000h</td><td>Uint32</td><td>0</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>Setting</td><td>
<table>
<tr><td>Byte 0</td><td>Byte 1</td><td>Byte 2</td><td>Byte 3</td></tr>
<tr><td>Bit 0 to 8</td><td>Bit 0 to 8</td><td>Bit 0 to 8</td><td>Bit 0 to 8</td></tr>
<tr><td>*P1808 index [0] to index [7]
FE0.1 to FE0.8</td><td>FE0.9 to FE0.16</td><td>FE0.17 to FE0.24</td><td>FE0.25 to FE0.32</td></tr>
</table>
*P1808: If an index from [0] to [7] has the setting "39", it accesses parameter 141 byte 0.
</td></tr>
<tr><td></td><td>NOTE:
Parameter 141 be controlled directly via field bus, MSD PLC or parameter access.</td></tr>
</table>

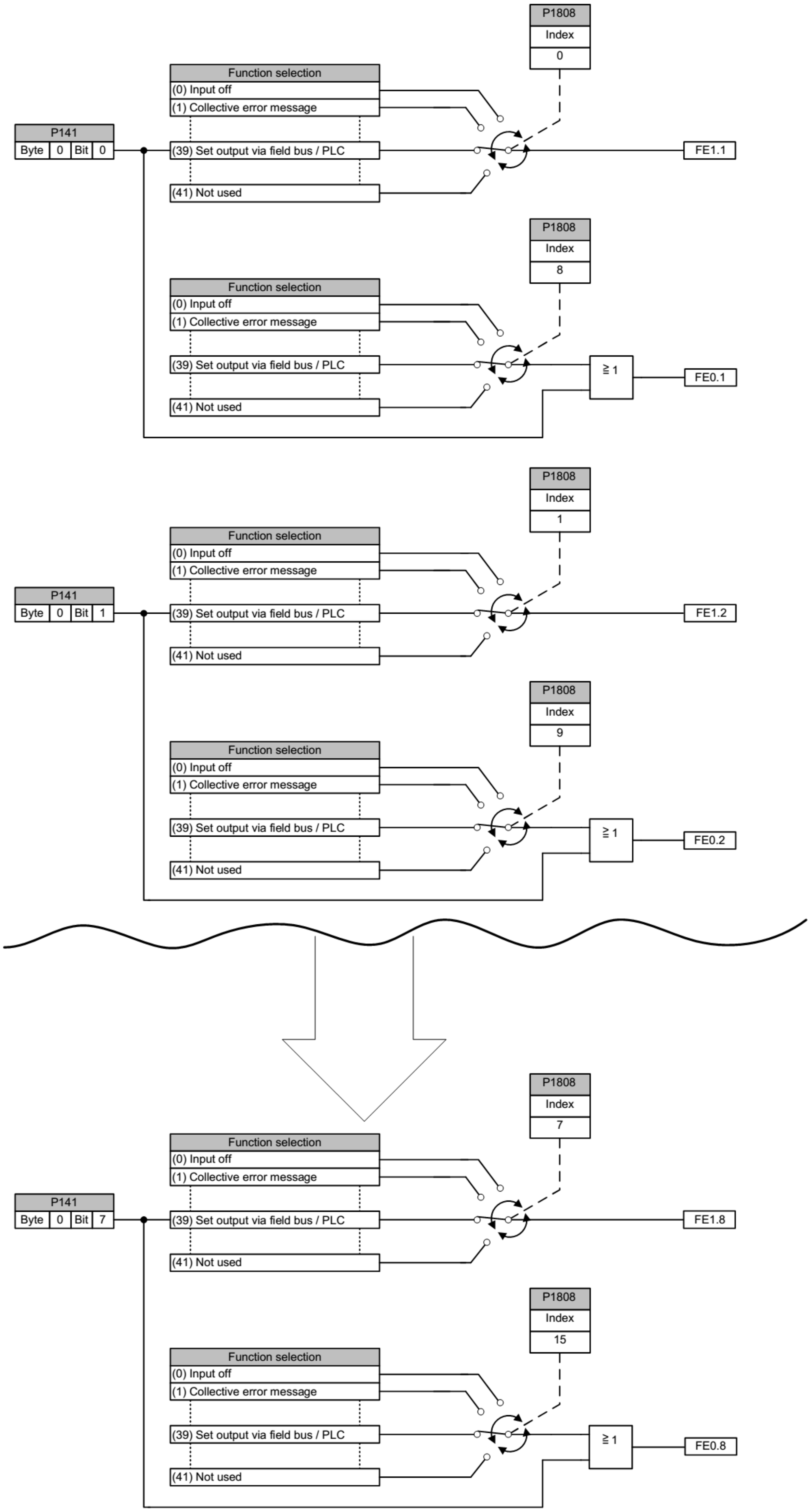


Figure 39: Setting for the functional inputs in the servo drive (master)

The function of the functional inputs FE0.9 to FE0.32 and their assignment in **parameter 141 OUTPUT_CTRL** are shown graphically in Figure 40.

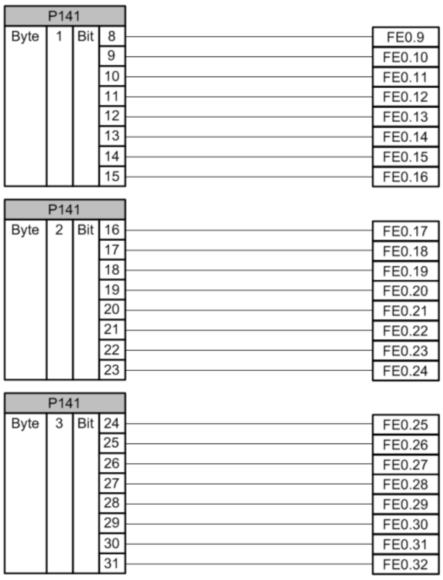


Figure 40: Assignment of the functional inputs FE0.9 to FE0.32

14.9.3 Setting for the functional inputs in the MSDFS slave

Each servo drive (slave) provides 8 functional inputs. These functional inputs are called FEx.1 to FEx.8 in the Safety PLC, where the "x" stands for the related servo drive (slave). As the functional inputs FE1.1 to FE1.8 are already assigned in the servo drive (master), the numbering for the servo drive (slave) starts at FE2.x:

- Servo drive (slave) 1 → FE2.1 to FE2.8
- Servo drive (slave) 2 → FE3.1 to FE3.8
- Servo drive (slave) 3 → FE4.1 to FE4.8
- Servo drive (slave) 4 → FE5.1 to FE5.8

FEx.1 to FEx.8

The functional inputs FEx.1 to Fx1.8 can be assigned a function in the related servo drive (slave) using the parameter 1808.

The setting for the functional inputs Fex.1 to Fex.8 in the servo drive (slave) is shown graphically in Figure 41.

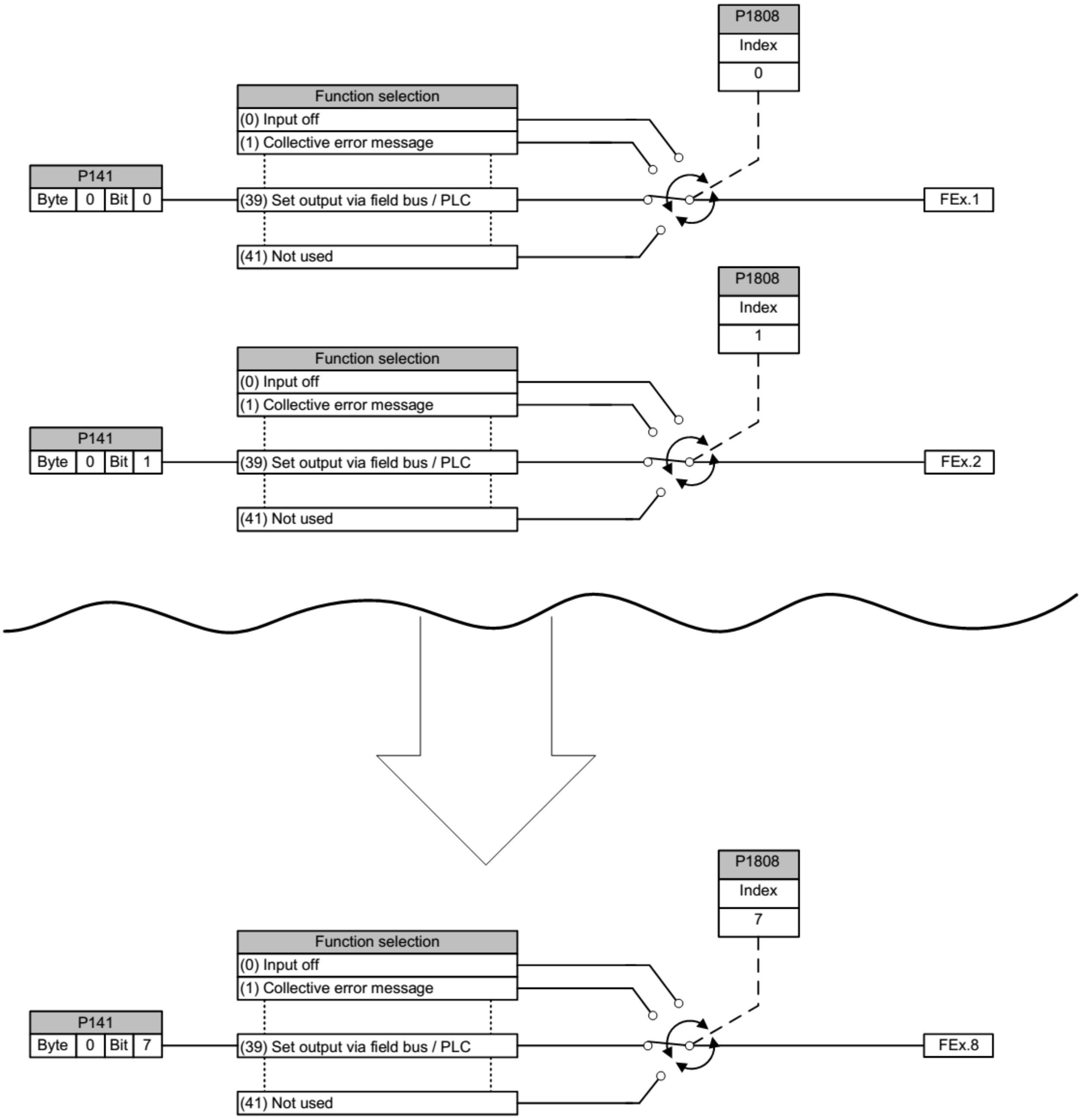


Figure 41: Setting for the functional inputs FEx.1 to FEx.8 in the servo drive (slave)

15 ADDING OUTPUT BLOCKS

These function block diagram elements are used to exercise a direct effect on the drive to be monitored. In addition, it can be defined how external switchgear is to be monitored.

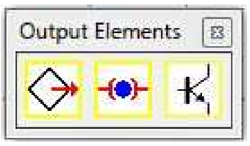
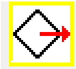


Figure 42: Output Elements

The automatic resource monitoring of the function block elements for the MSDFS ensured that only the elements available are enabled in the program. If there are no longer any resources (memory) in the MSDFS for the monitoring program, the commands for adding the related components or function blocks are not available (menu commands or toolbars dimmed). This is, e.g., the case if all digital outputs on the servo drive (master) and servo drive (slaves), as far as present, are assigned. These resources can be enabled again by deleting corresponding function blocks.

15.1 Safety-Output



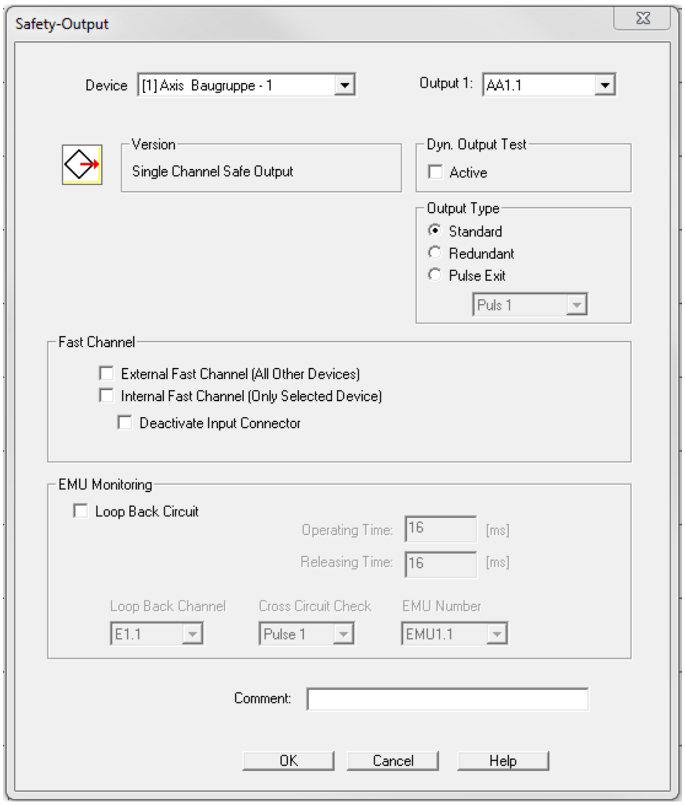


Figure 43: Safety-Output

Device

If there are several MSDFS, you can define here the MSDFS on which the output is to be used.

15.1.1 Output 1

Assignment number for the digital output used. The output signal designators still free (e.g. "AA1.1") on the MSDFS are displayed in this selection list. The user assigns these numbers. It is not allowed to make duplicate assignments to output signals. If the resources of the MSDFS are almost exhausted and too many output signals would be used due to the selection of the output type, the list box is empty. The output block cannot be added.

NOTE: If you select the output type "Redundant", output 2 appears automatically and is used in the same way as output 1.

Version

The currently selected setting for the output appears automatically in this field.

15.1.2 Output Type

Standard

Default setting, with this setting the output is used as a single-channel safety output.

Redundant

With this setting the output is used as a two-channel safety output. The dialog box for setting the safety output is expanded with Output 2.

Pulse Exit

With this setting the output is used for cross circuit monitoring on input elements. A total of four different pulse types are available that you can select in the list box that is now enabled.

NOTE: The pulse outputs can only be set on the servo drive (master); they can be read by every servo drive (slave). It is a prerequisite that the servo drive (slaves) are connected to the servo drive (master) via the safe cross communication.

15.1.3 Dyn. Output Test

Here you can enable or disable an OSSD pulse that is used to test the switching capability of the output.

NOTE: If the OSSD pulse is disabled the output is considered a non-safe output and should only be used to display or forward non-safe signals. OSSD pulsing on the output is also necessary to achieve category4.

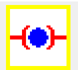
15.1.4 Fast Channel

See chapter "19.13 Fast channel"

EMU Monitoring

See chapter "15.3 EMU monitoring"

15.2 Brake output



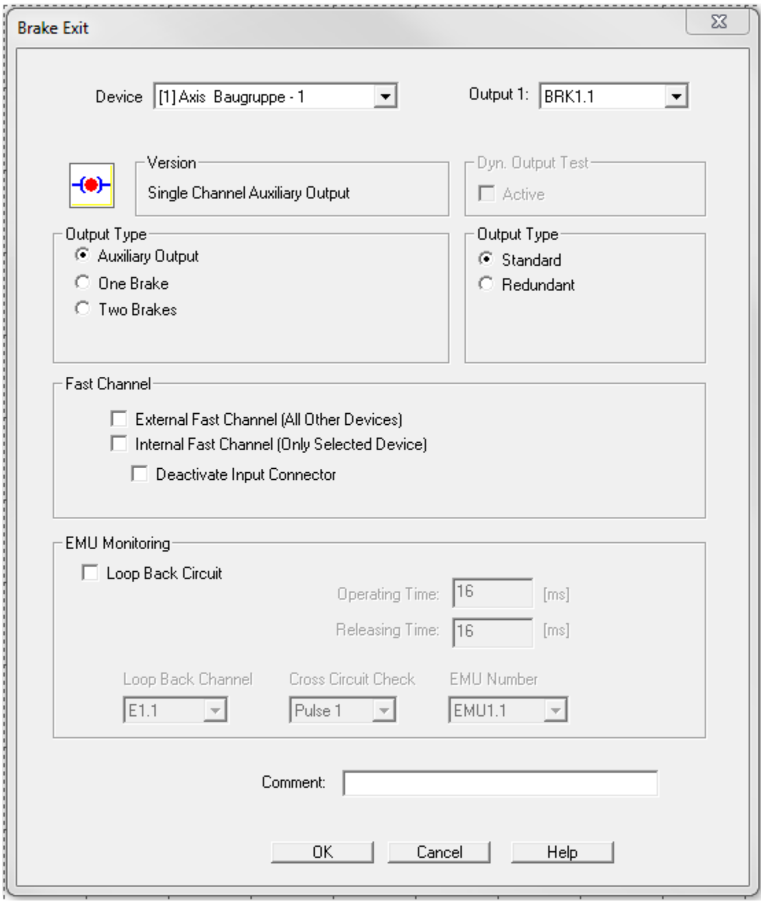


Figure 44: Brake Exit

Each servo drive (master/slave) provides a brake output that can be used in four different ways.

Device

If there are several MSDFS, you can define here the MSDFS on which the output is to be used.

Output 1

Assignment number for the brake output used. The brake outputs on the related device are displayed in this list box, provided these are not already assigned.

Single Channel Auxiliary Output

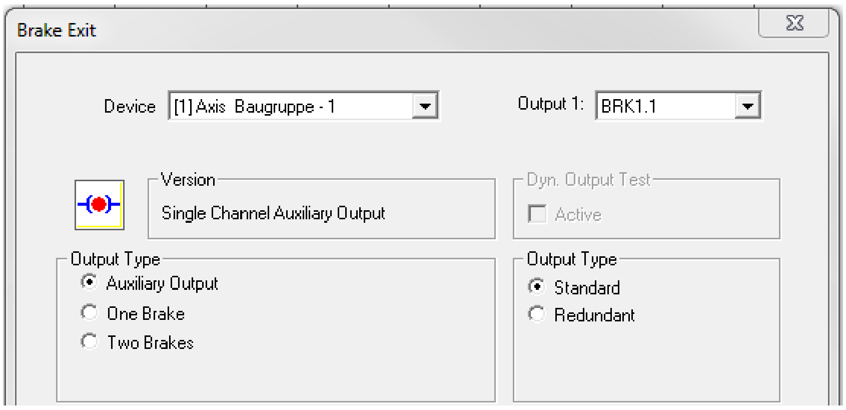


Figure 45: Brake output as single channel auxiliary output

In the "Single channel auxiliary output" version the dyn. output test is automatically deactivated. The output can be used as a non-safe digital output.

Double Channel Auxiliary Output

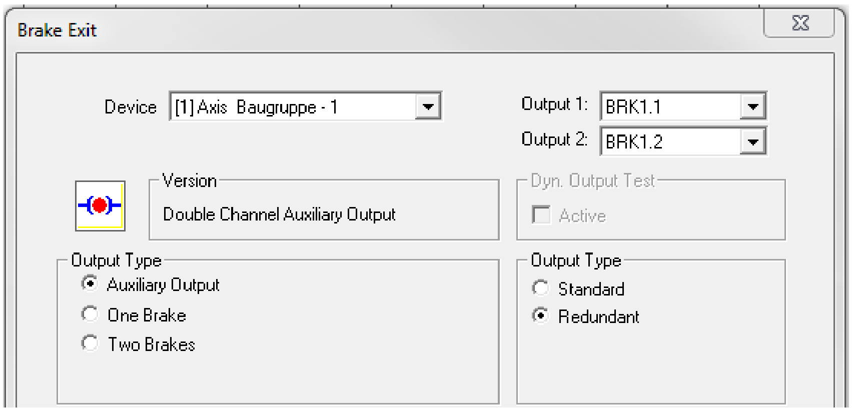


Figure 46: Brake output as double channel auxiliary output

In the "Double channel auxiliary output" version the dyn. output test is automatically deactivated and output 2 is added automatically with the corresponding designation. The redundant output can be used as a non-safe digital output.

One Brake

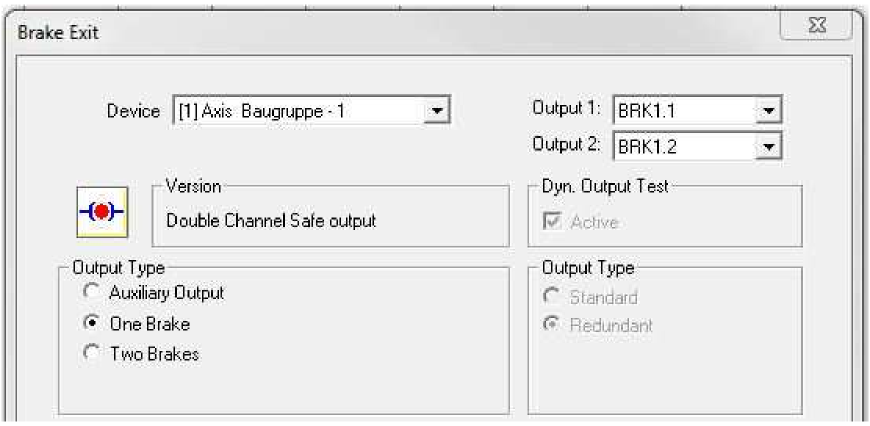


Figure 47: Brake output for one brake

In the "One Brake" version the dyn. output test is automatically activated and output 2 is added automatically with the corresponding designation. This two-channel safety output can be used for the safe operation of a corresponding brake.

Two Brakes

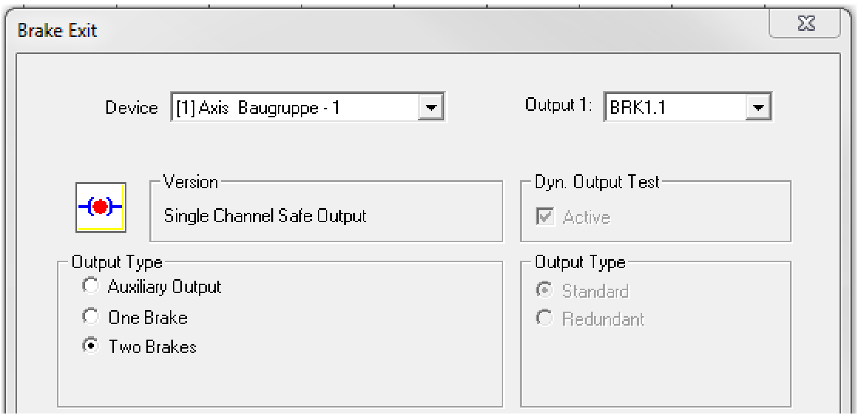


Figure 48: Brake output for two brakes

In the "Two Brakes" version the dyn. output test is automatically activated. In this setting the output can be used as a single-channel safety output and the brake operated safely using the output. For the second brake a further brake output it is now necessary to add a further brake output to the terminal diagram that is also set to "Two Brakes". With this setting the output is set automatically to "BRK1.2" and, equivalent to the first brake, the dyn. output test activated.

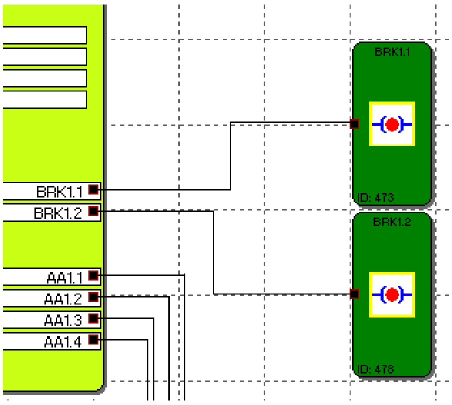


Figure 49: Brake output with two brake

15.3 EMU monitoring

To increase the number of contacts and the performance in general additional external switchgear is required that is operated via the cut-off channels on the MSDFS. In applications in accordance with category 4 as per DIN EN ISO 13849-1 functional monitoring is required for this switchgear among other aspects. For this purpose the switchgear is to be equipped with positively driven auxiliary contacts, also called mirror contacts. Such mirror contacts can be connected in series and are closed in the quiescent state. It is checked whether the mirror contacts are closed in the quiescent state and are open in the active state. The contacts can be supplied via the pulse output assigned or 24 VDC. Several mirror contacts can be connected in series.

NOTE: The EMU monitoring is functionally the same as an EDM (external device monitoring) function.

The servo drives (master/slave) each provide two EMU monitoring functions. These functions can used for "Safety Outputs" configured as safety output or for the brake output.

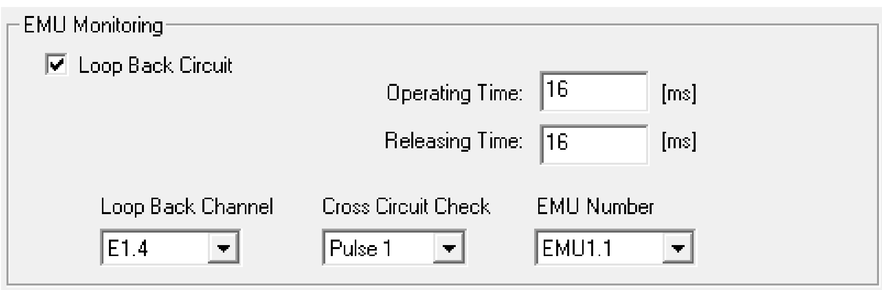


Figure 50: EMU Monitoring setting

Loop Back Circuit

Check box to activate the EMU monitoring.

Operating Time

Variable time window (switch-on delay) for the test on the safety contacts

Min_{TEMU}	=	16 msec
Max_{TEMU}	=	3000 msec

Releasing Time

Variable time window (switch-off delay) for the test on the safety contacts

Min_{TEMU}	=	16 msec
Max_{TEMU}	=	3000 msec

Loop Back Channel

Digital input on the MSDFS via which feedback is provided from the component to be monitored (e.g. contactor, relay).

Cross Circuit Check

Here you can set which pulse is connected to the safety contact to be monitored as the cross circuit test. The setting "AUS" (OFF) means that here no pulse is connected to the safety contact, instead 24 VDC static.

EMU Number

Number of the EMU loop back channel that is used. The EMU loop back channels used can be added to the function block diagram as an **EMU result block**.

NOTE: If the EMU monitoring in the master detects a fault in the plausibility monitoring, an alarm is not sent. The corresponding output is rendered passive and the related **EMU result block** in the function block diagram changes its state from logical "1" to logical "0".

If the EMU monitoring in a slave detects a fault in the plausibility monitoring, an alarm is sent and the entire axis group enters the safe state.

16 THE LOGIC ELEMENTS

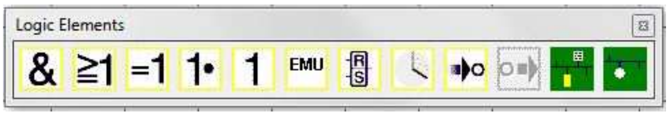


Figure 51: Logic Elements

These blocks form the basis for building up a program for the safety application. They make it possible to logically process the inputs with the monitoring functions and outputs. It is only possible to add the logic elements in the "Function block diagram" view; otherwise the related menu commands are not available. This is also the situation if the resources for a block have already been exhausted, e.g. after all timer blocks have been added.

16.1 Logical AND

"AND" operator for maximum 5 output signals from other function blocks. The AND operator provides the signal state "1" as the result if all input signals are "1", otherwise the result is "0".

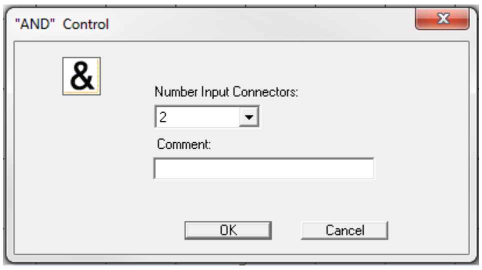


Figure 52: "AND" block

NOTE: The number of input connectors can only be reduced if there are spare connectors. If all connectors have connections, the connections must be deleted first.

16.2 Logical OR

≥1

"OR" operator for maximum 5 output signals from other function blocks. The OR operator provides the signal state "1" as the result if at least one input signal is "1", otherwise the result is "0".

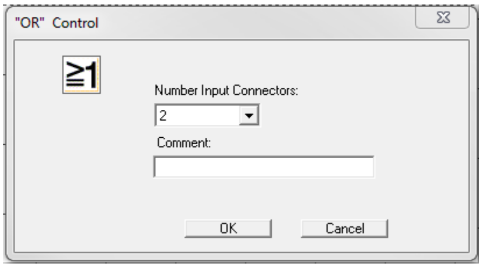


Figure 53: "OR" block

16.3 Logical EXCLUSIVE OR

=1

"EXCLUSIVE OR" operator for 2 output signals from other function blocks. The XOR block provides a "1" as the result if one input has the input signal "1" and the other input has the input signal "0", otherwise "0".

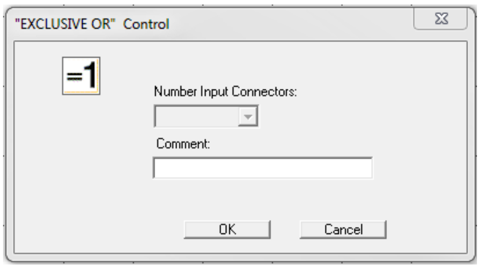


Figure 54: "EXCLUSIVE OR" block

16.4 Logical NOT

The result from this function block is the negation of the input signal. Negation is the term used if the result of an operator is inverted (negated).

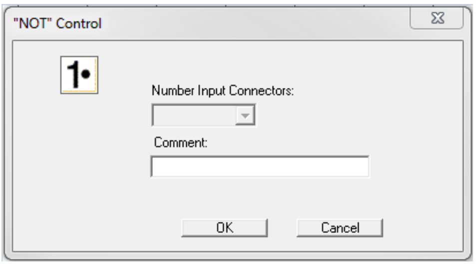


Figure 55: "NOT" block

16.5 RS flip flop

Set/reset switching element. This switching element has the following behaviour:

- The result of the operator on the initialisation of the element is "0".
- The result of the operator is "1" if there is an edge change from "0" to "1" on the "Set" input. The output remains at "1" also if the state of the Set input changes back to "0".
- The result of the operator is "0" if there is an edge change from "0" to "1" on the "Reset" input.
- If both inputs are set to "1", the result is "0"!
- If both inputs are set to "1" and there is an edge change to "0" on the "Reset" input, the result of the operator is 1 (event triggered).

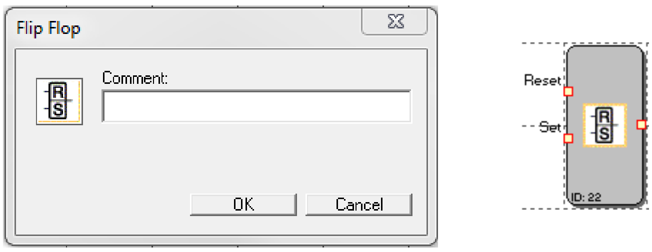


Figure 56: Flip Flop

NOTE: This element only has the required switching state on connection as per the labelling on the input connectors.

16.6 Timer

Function block starts a counter on an edge change. After the stated delay the result of the operator is "1" or "0".

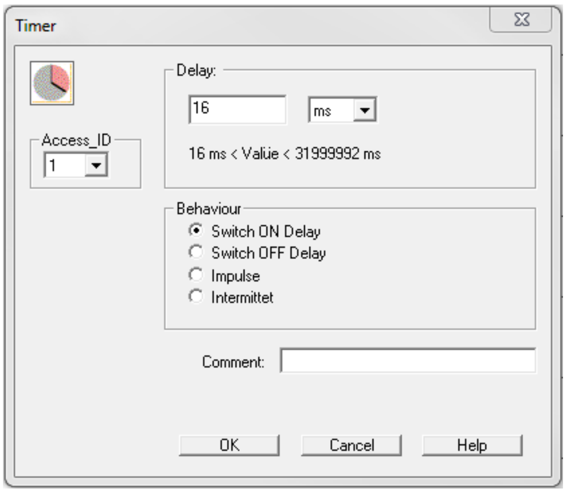


Figure 57: Timer block

Delay

Time configured (T_{min} = 16 ms // T_{max} = 533 min)

NOTE: The values that can be programmed always correspond to an integer multiple of the MSDFS cycle time of at least 16 msec!

Behaviour

Switch ON Delay	
Input	Output function
"0"	Output remains permanently at "0"
Edge "0" to "1"	After the time configured has elapsed the output state of the time block changes from "0" to "1".
State change "1" to "0"	Output changes immediately to "0"
Switch OFF Delay	
Input	Output function
"0"	Output remains permanently at "0"
Edge "0" to "1"	Output changes immediately to "1"
State change "1" to "0"	After the time configured has elapsed the output state of the time block changes from "1" to "0"

Impulse	
Input	Output function
"0"	Output remains permanently at "0"
Edge "0" to "1"	Output changes to "1" for the time configured independent of the length of the "1" signal applied.
State change "1" to "0"	A state change from "1" to "0" does not have any effect on the output of the timer block.
Intermittent	
Input	Output function
"0"	Output remains permanently at "0"
Edge "0" to "1"	Output changes its state for the time configured while the "1" signal is applied.
State change "1" to "0"	Output changes immediately to "0"

16.7 Permanent logical "1" block

This block provides the value "1" constantly. You can program static states in the logic diagram using this function.

Example: Connection to an unused input for direction dependence SDI

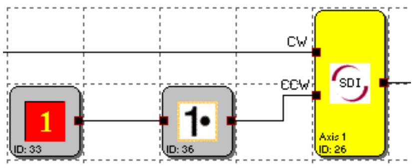


Figure 58: Usage example permanent logical "1" block

16.8 EMU Result block

This block provides the result from the **EMU monitoring** that can be configured in the related output blocks. Error-free **EMU monitoring** is fed back with the logical state "1".

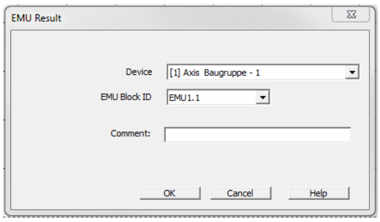


Figure 59: EMU Result

16.9 Terminal In

The "Terminal In" aids the clear depiction of function block diagrams. These blocks provide virtual connections in the logic diagram. The reference numbers for the connection points are generated automatically and cannot be changed; however, the Comment field permits corresponding assignment of the virtual connection. If you press the CTRL key and select a "Terminal In", the related "Terminal Out" blocks are also selected.

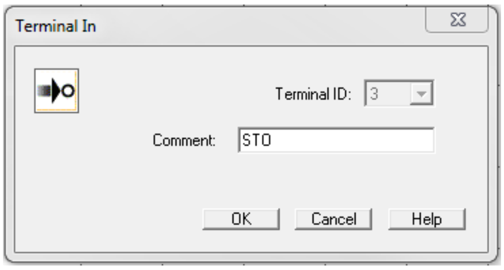


Figure 60: Terminal In

NOTE: On the deletion of "Terminal In" elements the dependent "Terminal Out" elements are automatically deleted at the same time. Prior to deletion a warning appears.

TIP: The usage of the Comment fields makes it easier to assign the elements.

16.10 Terminal Out

This element represents the equivalent to the "Terminal In". You can establish a virtual connection to a "Terminal In" function block by selecting the terminal ID.

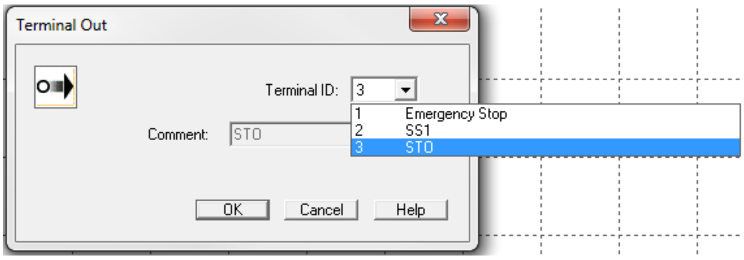
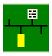


Figure 61: Terminal Out

NOTE: After assignment to a "Terminal In" element the comment saved there is applied in the "Terminal Out" element.

16.11 Status Message Data



In this block you can configure the diagnostic data to be transmitted from the process image in the status message data.

In the status message data for the logic data a 32 bit wide field of the status bits set here is transmitted cyclically; this data is then available for evaluation in the servo drive (master).

Process data can also be transmitted in a further frame. This feature makes it possible to evaluate dynamic process data.

16.11.1 Logic Data

This dialog box tab contains the list of the states that are to be transmitted from the function blocks added to the function block diagram.

The status message data list contains the references to the function block outputs added. The result bit from the block is entered at the position of the bit ID set.

The bit value for a function block result (status) is assigned to the corresponding status message data bit via the bit ID.

The monitoring blocks always have the value 1 (high) in the good state. Good state means that the monitoring block is either not activated, or is activated and there has been no limit infringement.

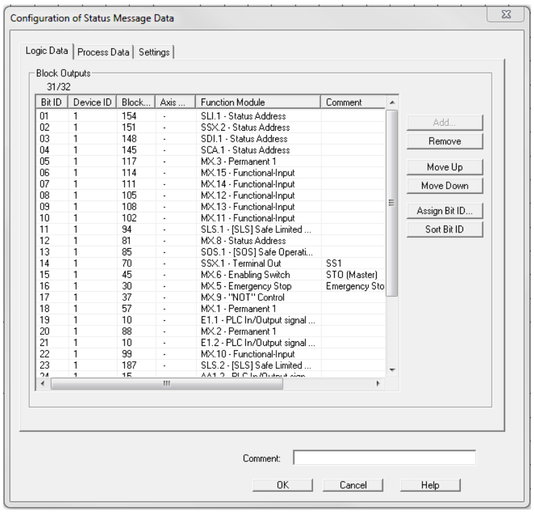


Figure 62: Configuration of the Status Message Data - Logic Data

Block Outputs

The numbers under the label "Block Outputs" show the number of status bits used / maximum possible number of status bits.

Add...

Opens the "Statusbit Hinzufügen" (Add status bit) dialog box. The block selected there is added to the end of the reference list.

Remove

Removes the currently selected row from the reference list. The bit IDs for the entries after the entry removed are retained.

Move Up

Moves the currently selected row up one line in the reference list and takes over its bit ID.

Move Down

Moves the currently selected row down one line in the reference list and takes over its bit ID.

Assign Bit ID

Here you can use any free bit ID. The assignment dialog box can only be opened under the following conditions.

- There must be at least one free bit ID available.
- A row must be selected in the status message data list. Double-clicking a row also opens the dialog box.

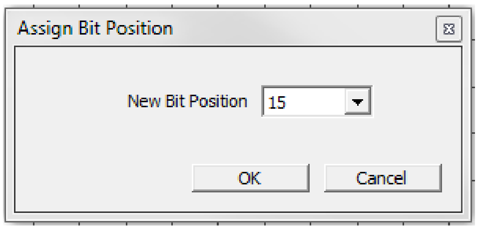


Figure 63: Assign Bit Position

Please select here the required bit position for the row selected. The count is based on 1. The result from the block referenced is entered at the position set.

Sort Bit ID

Sorts the status message data based on the bit IDs assigned.

NOTE: The status bits should only be assigned after a successful compiler run, as the addresses set by the compiler must be applied. These are displayed in the "Symboladressen" (Block Addresses) column. As long as the function block diagram cannot be fully compiled, the entries in this column are blank or not up to date.

16.11.2 Process Data

On this dialog box tab you can set which dynamic process data are to be transmitted from the safe part (integrated safety control) to the non-safe part (servo drive). A total of 96 bits are available for this purpose. The values set in the dialog box are evaluated from "top" to "bottom" and as per the resolution set. This area is to be interpreted as an integer value! It corresponds to the normalised encoder position from the actual sensor configuration.

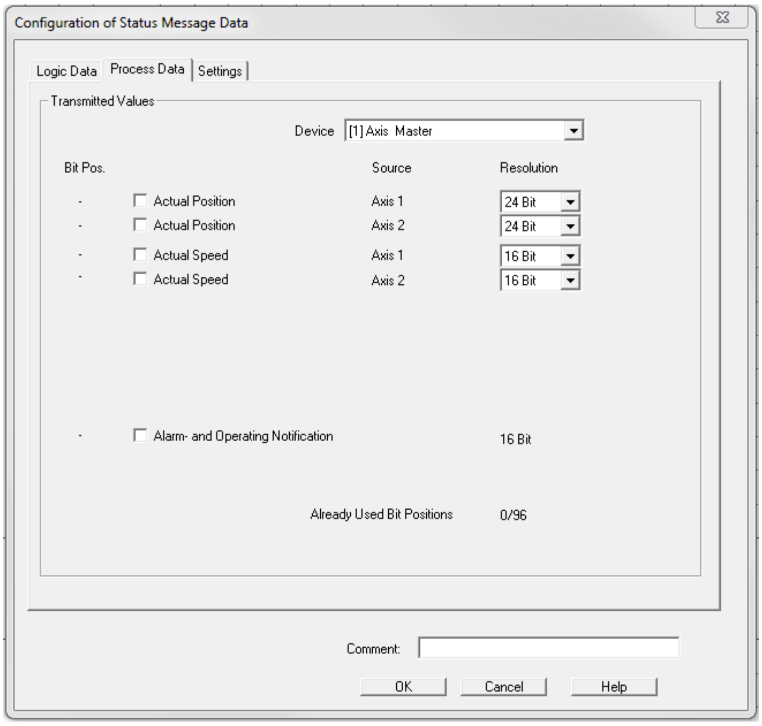


Figure 64: Configuration of the Status Message Data - Process Data

17 Functional groups

Functional groups combine several function blocks into a higher level logic structure. This related group of blocks is placed in a functional group frame and bound together via this frame.

The grouping makes the function block diagram clearer and makes it possible to build up a custom function library via the export/import functionality.

17.1 Adding a functional group frame

First use the "Add" toolbar button  to open the command "Add group frame". Alternatively you can open the menu: Group → Gruppenrahmen einfügen… (Add group frame…).

To define the size of the group frame, enter two co-ordinates using the mouse pointer.

First click the top left corner of the group frame using the left mouse button and keep the mouse button pressed.

Then drag the mouse button with the left mouse button pressed. In this way you can define the bottom right corner of the group area.

When you release the mouse button the group frame is added and the group editor opened.

17.1.1 Opening the group editor

The group editor can also be opened by double-clicking the status bar for the group frame, or using the context menu (right mouse button) for a selected block.

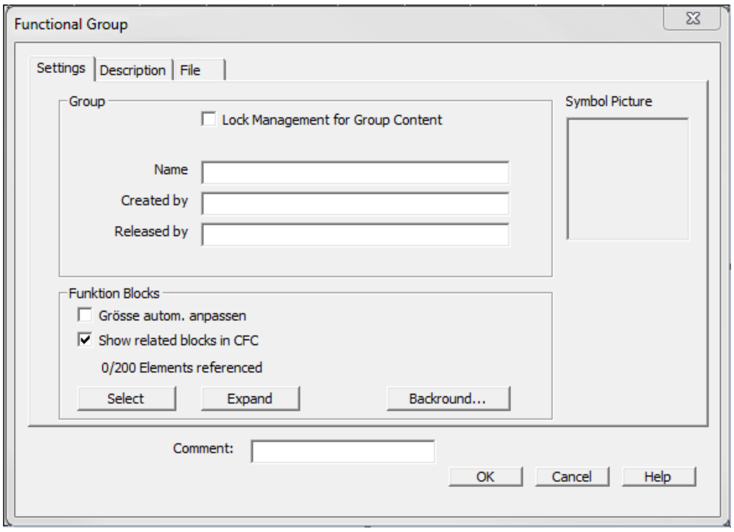


Figure 65: Group editor

The dialog box tabs "Settings" and "Description" contain the group-related settings as well as the functional description of the group. On the "File" dialog box tab you can export the group to a file, or import from a file.

Group

Lock Management for Group Content

Using the "Lock Management for Group Content" check box you can lock or unlock the group blocks.

If the check box is selected, the function block management for the frame is disabled and the blocks frozen. Blocks can no longer be removed from the group.

NOTE: It is still possible to configure the parameters. However, here caution is required on changing basic functionality (functional parameters) in the block parameters. Management has not been implemented here yet.

On the deletion of the group frame all group blocks are deleted.

It is not possible to add new blocks to the group.

The group members are displayed "dimmed".

When you lock the group it is assigned a time stamp that is displayed on opening the group editor (quasi last modification).

The text for the information fields "Name", "Created by" and "Released by" is locked.

The "locked" status of the group is indicated by the padlock symbol in the status bar for the group block at the top left.

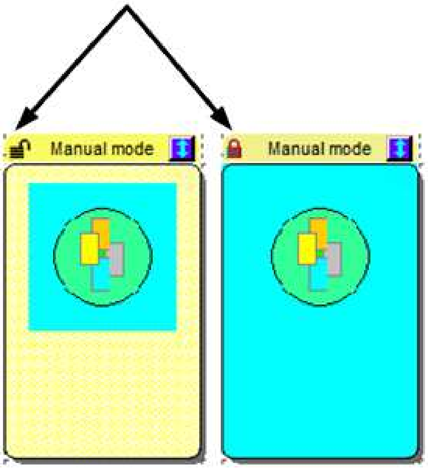


Figure 66: Indication of the “locked” status

On the addition of a new group frame the "Lock Management for Group Content" check box is initially not selected. After closing the group editor the frame drawn appears in the function block diagram and represents the active area of the group.

You can now add or drag the related function blocks to this frame. As long as the group does not have the locked status, the blocks are <u>automatically</u> added to the group. The function blocks then also indicate the group number.

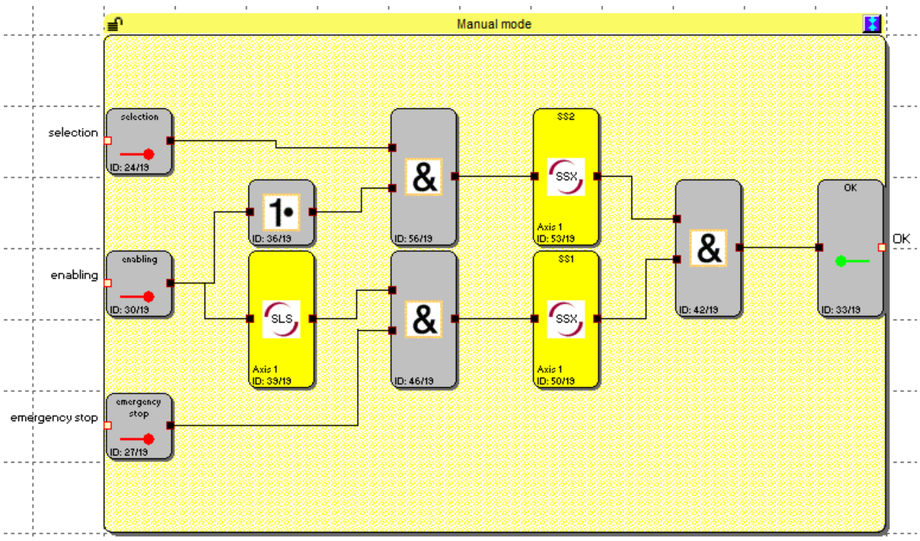


Figure 67: Group frame

NOTE: The block types listed in the following cannot be included in a group. They are filtered out on dragging the blocks into the frame.

- Input blocks
- Output blocks
- All pre-defined function blocks in the function block diagram (e.g. encoders, IOs)
- Status message data block
- Terminal blocks

A maximum of 200 function blocks can be added to the group.

17.1.2 Changing the size of a group frame

You can change the size of a selected block using its "hotspot". Select this point using the mouse pointer and drag with the left mouse button pressed. The change in size is displayed.

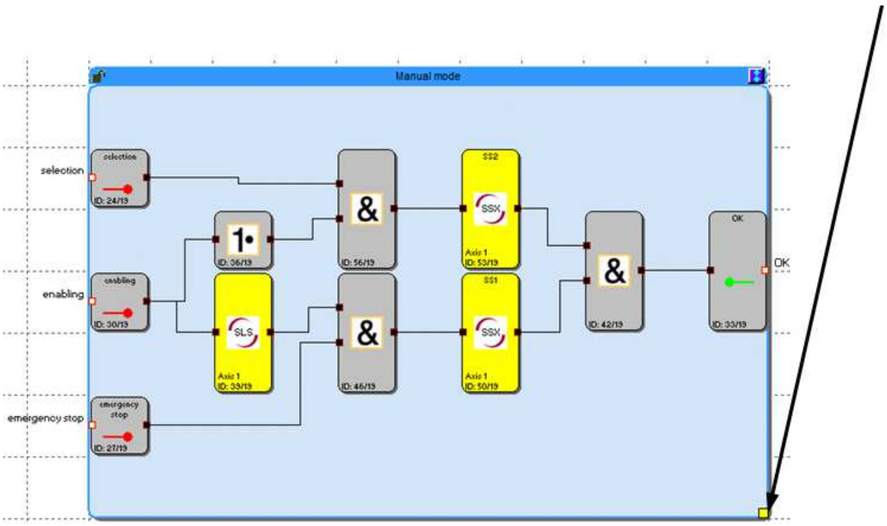


Figure 68: Changing the size of the group frame

17.1.3 Showing, hiding the function blocks

You can show or hide the blocks contained in the group by selecting the button on the status bar (single click). The group block automatically adjusts to the size of the blocks it contains on showing the function blocks.

NOTE: While you are editing the blocks, as far as possible do not show/hide, as any free space provided for additional blocks will be reset. The size of the group must then be increased again manually using the "hotspot".

TIP: The size of the group frame can be fixed via a text element positioned in the bottom right corner.

You can also define whether the related function blocks are show in the function block diagram in the Functional Group dialog box using the "Show related blocks in CFC" check box.

Blocks shown

The size of the group block is defined by the position of the function blocks it contains.

Blocks hidden

The group block is set to the size of approx. 2 x 3 fields in the function block diagram. The bit map is used for symbol display.

If several functional groups are added, you can show or hide all group blocks using the button on the group toolbar, or the Group menu.

17.2 Adding the functional group interface

The functional group interface blocks represent the interface between the functional group and the function blocks outside the group. You can only make connections to function blocks outside the group using the interface block.

Start the addition of a functional group interface using the button on the group toolbar (alternatively menu: Group->Interfacebaustein einfügen... (Add interface block...)) After you have positioned a block inside a group frame the functional group interface editor opens.

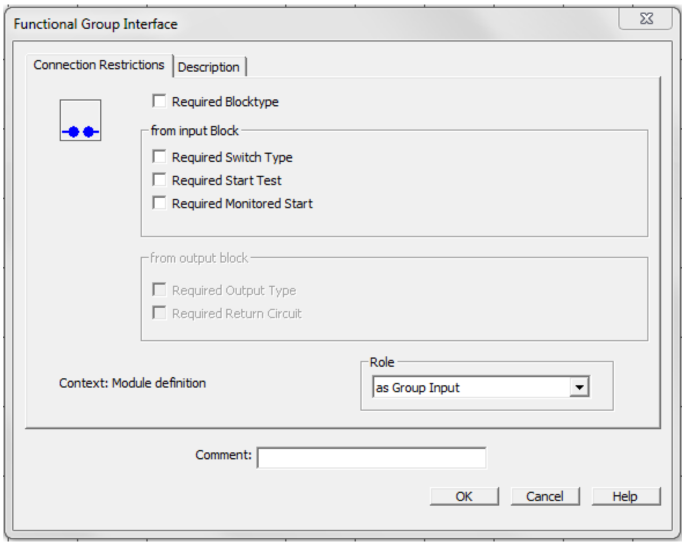


Figure 69: Functional group interface editor

17.2.1 Setting role

First set the role of the block.

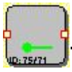
Role

Using this setting you can define how the block is used in its environment.

"as Group Input"

Connection for the function blocks outside the group as input connector. Is used to activate group functionality. The block should be positioned on the left side of the group frame. The output connector must be connected inside the group.

"als Gruppenausgang" (as Group Output)

This block provides the result of the group functionality. The block is the last element in the sequence of blocks. The output connector is connected to a function block outside the group.

17.2.2 Restrictions

The main task of the interface blocks is to check whether an imported functional group is connected to the required function elements. You can set this aspect via the connection restrictions.

Connection Restrictions

The first key issue is the state of the related group. The group can be unlocked (= definition mode) or locked (= application mode). These states reflect the context of the interface block.

NOTE: You can only set the restrictions correctly if the interface block is connected to the required function block diagram elements. This means the interface blocks must be connected to the relevant function blocks in the definition mode.

Example: A mode selector switch is connected to the group interface block. In the application mode the group block then always expects that it is connected to the "mode selector switch" function block type.

Context: defining block

The interface block reads the type and the restriction criteria from the block connected and offers it as a restriction. If the related check box is selected, the related restriction appears.

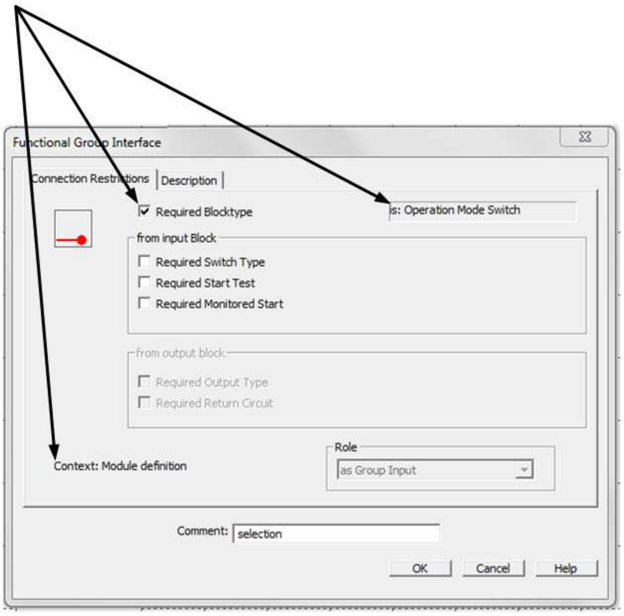


Figure 70: Context: defining block

Context: using block

If you now lock the group management, the interface block expects the restriction criteria set on connecting to an external function block. If these criteria are not met, there will be a compiler error and the program cannot be compiled.

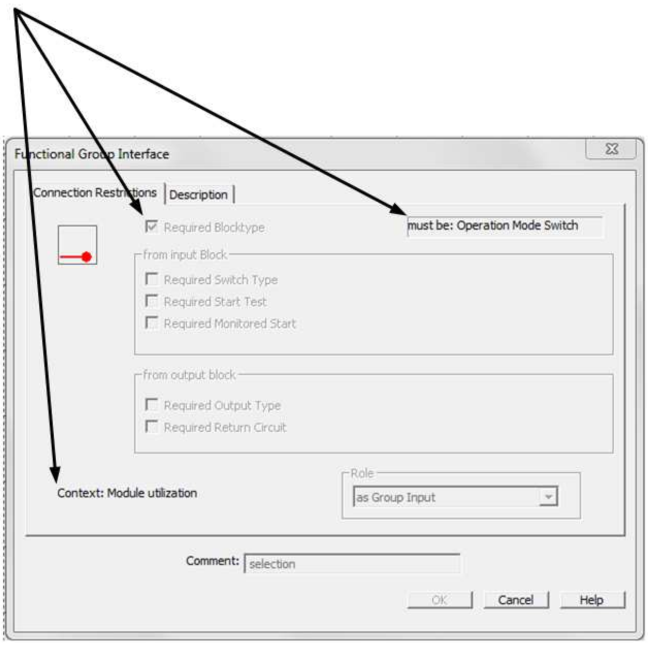


Figure 71: Context: using block

17.3 Procedure on adding a functional group

A functional group is formed via a group frame. Function blocks within the different colour area of a group frame form part of the group. As long as the group block is unlocked, you can add or remove function blocks to or from the area of the group frame. If you have added a block to a functional group, the message "Enthalten in Funktionsgruppe: Nr." (Included in functional group no.) is displayed on the information display.

TIPs:

- The group(s) should only be in the unlocked state for as little time as possible.
- Unlock as few groups as possible in the function block diagram.
- As far as possible do not move unlocked groups on the function block diagram
- Make connections as late as possible.

1) Add interface blocks

The function blocks contained in a group can only be connected to the function blocks outside the frame via the interface blocks. In the interface blocks you can set restrictions if necessary that require the same connection configuration on importing the group into a different function block diagram. The interface blocks permit a description of the input and output parameters for the functional group. If you have set restrictions, these should always be described in the block.

2) Add function blocks to the group

You can only add function blocks to the group frame in the unlocked state. This situation is indicated by the symbol on the status bar.

To add function blocks to a group, you must either add a block within the group frame or drag it into the frame.

NOTE:

- It is not possible to add function blocks by dragging the group frame! Instead the blocks must be moved into the group frame.
- Only logic blocks and monitoring blocks are added to the group, no input or output blocks and no pre-defined elements such as signal lists, analogue blocks or encoder blocks.
- On blocks with existing connections you may end up with a connection to outside the group frame if you drag the selection in stages. This situation is not allowed under any circumstances and the connection will be deleted automatically.

- If blocks with connections are to be added to the group by dragging, proceed as follows:
    - Increase size of group frame. At least two grid elements larger than necessary.
    - Then move the group frame over the function blocks. The related connections must all be within the group frame.
    - Select the blocks and move one grid position within the group block.

3) Add connections

4) Connect group interface

5) Set connection restrictions

6) Test functional group

The user is responsible for the correctness of the functionality. The parameters must be checked and modified by the user on importing the group into a different function block diagram! Add description of the usage of the functional group. The function block diagram should be in a state suitable for compiling. The inputs and outputs of the functional group are to be described.

7) Lock functional group

On locking the functional group the function blocks in the group are merged with the group block. The blocks can then no longer be deleted individually and can are only be moved via the group block.

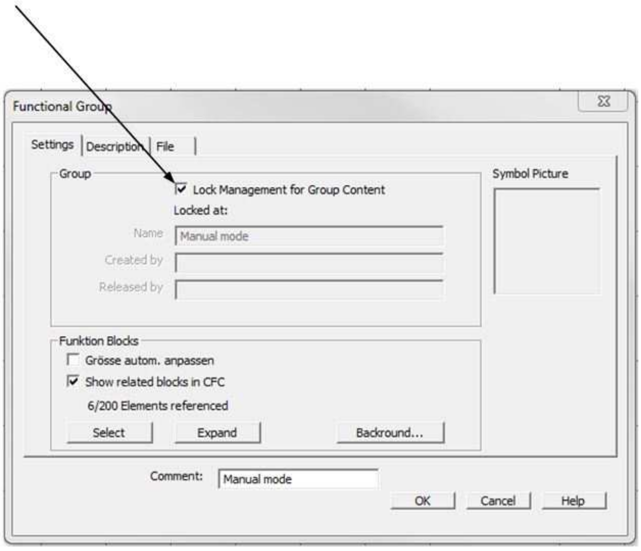


Figure 72: Locking functional group

17.4 Exporting functional groups

The blocks in the group can be exported to a *.fgr file. An exported group can be imported again into a different group frame. In this way you can build up a library with pre-defined functional groups that you can import into new projects.

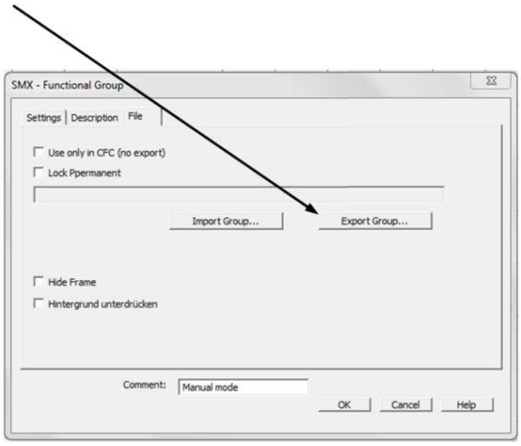


Figure 73: Exporting functional groups

Lock Permanent

If this check box is selected, it is no longer possible to unlock this group after the import.

NOTE: If this check box remains selected and the dialog box is closed using OK, the group is locked within the function block diagram and the "Lock Management for Group Content" check box is permanently hidden. It is advisable to make a backup copy with the functional group unlocked. The structural layout of the functional group can no longer be changed after the "Lock Permanent" check box has been selected!

17.5 Importing functional groups

You can only import a functional group file into an existing group frame. For this purpose open the group editor and start the "Import Group..." function.

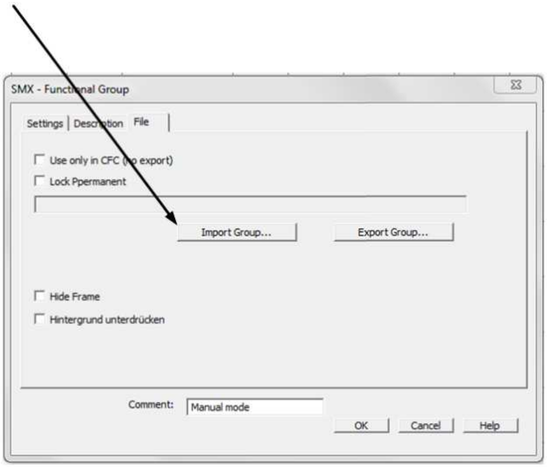


Figure 74: Importing functional groups

NOTE: Blocks that already exist in the group are deleted.

During the import the sensor configuration and the existing resources in the function block diagram are checked. The group can only be imported if the necessary resources are available for all blocks. In particular, on position-dependent monitoring blocks the required sensor settings are to be checked.

If a resource is no longer available, an error message is displayed.

In the event of resource errors, it is to be ensured the sensor settings correspond to the requirements of the group. In particular if position-dependent blocks are used in the functional groups (SEL, SLP, SCA).

18 Encoder

Using the "Encoder" window you can select the encoder type, the working section as well as the parameters for the two encoders for position and speed acquisition.

NOTE: The parameter configuration for the encoders must always relate to a common axis. If the two encoders are connected in different mechanical positions and the positions connected, e.g. using an intermediate gearbox, the working section must be defined for one of the two encoder positions and the transmission ratio in between taken into account on the other encoder.

18.1 Configuring working section parameters

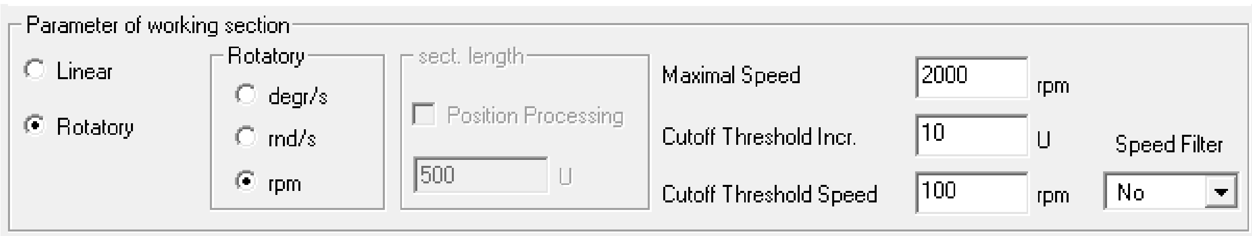


Figure 75: Parameter of working section

Parameter of working section

In this group box you can configure the parameters for normalising the working section. The following options and entries are possible:

Linear

The working section has a linear character, the movement is longitudinal. I.e. a linear movement in position and speed is acquired in the units mm and mm/s or m and m/s.

NOTE: On the usage of a resolver a rotatory working section is always to be used.

Rotatory

The working section has a rotatory character, the movement is rotational. The position is acquired in degr or U and the speed in degr/s or rnd/s or rpm.

Position Processing

This functionality is only active if you have selected an absolute encoder in the sensor area!

If you select the check box all position-related monitoring functions are enabled. If you clear the check box only speed and direction acquisition is possible.

NOTE: The section length must always be known for position acquisition.

Sect. length

Definition of the Maximum section length for the position in mm, m or degrees U. The maximum possible entry here is 1000000 in each unit.

Maximal Speed

Definition of the Maximum speed of the reference axis in the related unit of measure.

The maximum speed allowed describes the largest possible achievable speed in the actual, system-related configuration. Here you should enter the maximum value that can be reached by the monitored axis. In some circumstance this may relate only to a theoretical maximum speed, as this not reached in the control or with the parameters configured.

This value does not relate to the safety shutdown, as for example in the SLS function, but to reliability, that is the consistency of the encoders or the consistency of the mechanical situation. If this value is exceeded an alarm is triggered with shutdown and error / alarm state. This shutdown is not a planned shutdown due to a safety-related overspeed event, instead there should be doubts about the reliability of the encoders or the mechanical situation (encoder error, error in the servo drive,...), as this speed should not actually be reached by the drive.

If this situation occurs, the MSDFS switches to an alarm state and shuts down the outputs on the safe part.

Consequently, the "Maximal Speed" must always be higher than the shutdown speed for a safety function. It is used to define a fault in the measuring systems on the safe axis.

The value that is entered in this field also affects the magnitude for the encoder consistency in relation to the "Cutoff Threshold Incr." and "Cutoff Threshold Speed". A higher maximum speed permits higher cutoff thresholds between the encoders. For this reason the maximum value should also not be selected too large, as otherwise the cutoff thresholds for the reliability of the sensors in relation to each other could be selected too large. The table "Information on sensors" shows these calculated limits values for the variables V_max, V_min.

Cutoff Threshold Incr. / Cutoff Threshold Speed

The cutoff threshold defines the tolerable speed and position deviation between the two acquisition channels / encoder channels. Among other aspects, it is dependent on the arrangement of the sensors and the Maximum mech. play (e.g. due to backlash and spring strength) between the two acquisition points. The lowest possible value taking into account the dynamic processes (e.g. load/play in the gearbox) is to be selected at which the monitoring is not triggered in normal operation.

Speed Filter

Average value filter applied to the speed values acquired by the encoder to smooth peaks on the speed on systems susceptible to jerking. With the filter enabled the reaction time stated for the overall system is increased by the time set. The filter acts on all speed-dependent parameters of the monitoring blocks.

NOTE: The definition of the character of the working section as linear or rotatory has in principle an effect on all position and speed entries in the other windows related to the monitoring functions. There is therefore a change from an entry in mm, m and mm/s, m/s to degr, U and degr/s, rnd/s or rpm.

It is imperative the Maximum section length and Maximum speed are defined. If there is no entry or the entry is incorrect, the monitoring functions may trigger unintentionally.

18.2 Encoder A, and Encoder B

In these two group boxes you can configure the parameters for the encoders.

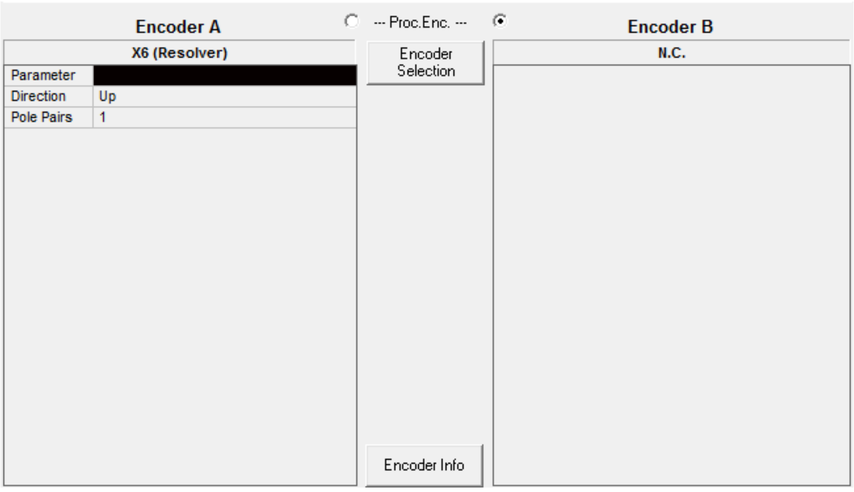


Figure 76: Encoder A and Encoder B

In general encoder A has the function of a process encoder and encoder B that of the reference encoder. If encoders with a different resolution are used, the encoder with the higher resolution should be configured as the process encoder.

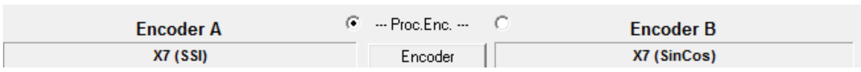


Figure 77: Process encoder selection

NOTE: The encoder with the lower resolution defines the accuracy that can be achieved in safety engineering terms.

First the encoder combination required is selected using the "Encoder Selection" button.

Figure 78: Encoder selection

The possible encoder combinations are listed in [Appendix 1 Encoder combinations](#).

18.3 Configuring resolver parameters

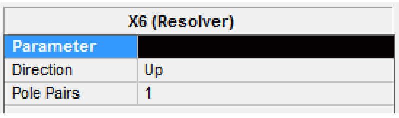


Figure 79: Resolver

Direction

Set here the direction of rotation.

Pole Pairs

Set here the number of pole pairs on the resolver used.

Parameter	Value range / setting	Explanation
Direction	Up	Positive direction of rotation
	Down	Negative direction of rotation
Pole Pairs	1 to 5	Number of pole pairs on the resolver used.

NOTE: On the usage of a resolver a rotary working section is always to be used.

18.4 Configuring Sin/Cos encoder and TTL encoder parameters

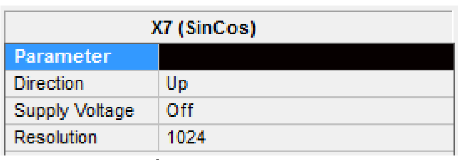


Figure 80: Sin/Cos Geber

Direction

Set here the direction of rotation.

Supply Voltage

Set here the supply voltage used to supply the Sin/Cos encoder used.

Resolution

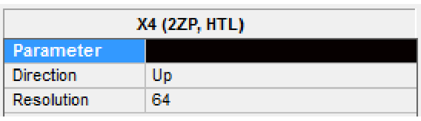
Set here the number of Sin/Cos tracks or increments.

<table>
<tr><th>Parameter</th><th>Value range / setting</th><th>Explanation</th></tr>
<tr><td rowspan="2">Direction</td><td>Up</td><td>Positive direction of rotation</td></tr>
<tr><td>Down</td><td>Negative direction of rotation</td></tr>
<tr><td rowspan="3">Supply Voltage</td><td>Off</td><td>No monitoring</td></tr>
<tr><td>4.75 V…5.25 V</td><td>Voltage range</td></tr>
<tr><td>7 V…12 V</td><td>Voltage range</td></tr>
<tr><td>Resolution</td><td>1 …*</td><td>Increments per revolution

NOTE: If the working section is set to "Linear", set here the increments per 1000 mm.

* The maximum number of increments that can be entered here depends on the maximum input frequency of the device.
<table>
<tr><th></th><th>TTL</th><th>SinCos</th></tr>
<tr><td>Maximum signal frequency that can be evaluated</td><td>400 kHz</td><td>400 kHz</td></tr>
<tr><td>Speed calculation method</td><td colspan="2">Maximum input frequency / resolution (pulses per revolution)</td></tr>
<tr><td>Signal level</td><td>Digital signals EIA422</td><td>Analog signals 1 Vpp</td></tr>
</table>
</td></tr>
</table>

18.5 Configuring 2ZP, HTL parameters



X4 (2ZP, HTL)	
Parameter	
Direction	Up
Resolution	64

Figure 81: 2ZP, HTL

Direction

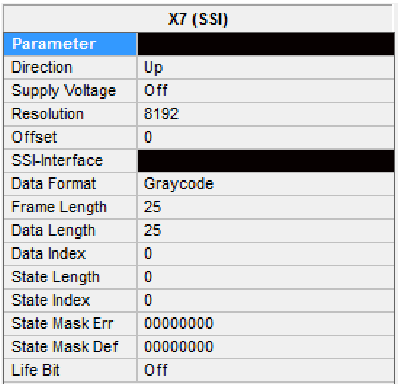
Set here the direction of rotation.

Resolution

Set here the number of HTL tracks or counting pulses.

Parameter	Value range / setting	Explanation
Direction	Up	Positive direction of rotation
	Down	Negative direction of rotation
Resolution	1…*	Incr. / counting pulses per revolution **NOTE:** If the working section is set to "Linear", set here the increments / counting pulses per 1000 mm. * The maximum number of increments that can be entered here depends on the maximum input frequency of the device.

18.6 Configuring SSI encoder parameters



X7 (SSI)	
Parameter	
Direction	Up
Supply Voltage	Off
Resolution	8192
Offset	0
SSI-Interface	
Data Format	Graycode
Frame Length	25
Data Length	25
Data Index	0
State Length	0
State Index	0
State Mask Err	00000000
State Mask Def	00000000
Life Bit	Off

Figure 82: SSI

Direction

Set here the direction of rotation.

Supply Voltage

Set here the supply voltage used to supply the SSI encoder.

Resolution

The singleturn resolution of the SSI encoder used is set here.

Offset

Here you can enter an offset that is added directly to the raw value from the encoder.

MOOG

Data Format

The coding used by the SSI encoder is entered here.

Frame Length

The total length of the SSI telegram is entered here.

Data Length

The length of the position information for the SSI encoder used is entered here. This item comprises singleturn and multiturn information.

Parameter	Value range / setting	Explanation
Direction	Up	Positive direction of rotation
	Down	Negative direction of rotation
Supply Voltage	Off	No monitoring
	4.75 V…5.25 V	Voltage range
	7 V…12 V	Voltage range
Resolution	1…*	Singleturn resolution *)
Offset	1…*	Offset in increments
Data Format	Binary	Encoder coding
	Gray code	
Frame Length	10..31 bits	Length of the complete telegram.
Data Length	10…28 bits	Length of the SSI data starting from the MSB. This area is not allowed to be interrupted, e.g. by status bits.
The Data Index,, State Length, State Mask Err, State Mask Def and Life Bit parameters are not supported. *) If the working section is set to "Linear", set here the increments per 1000 mm.		

NOTE: SSI formats (singleturn and multiturn) are stated in the specification.

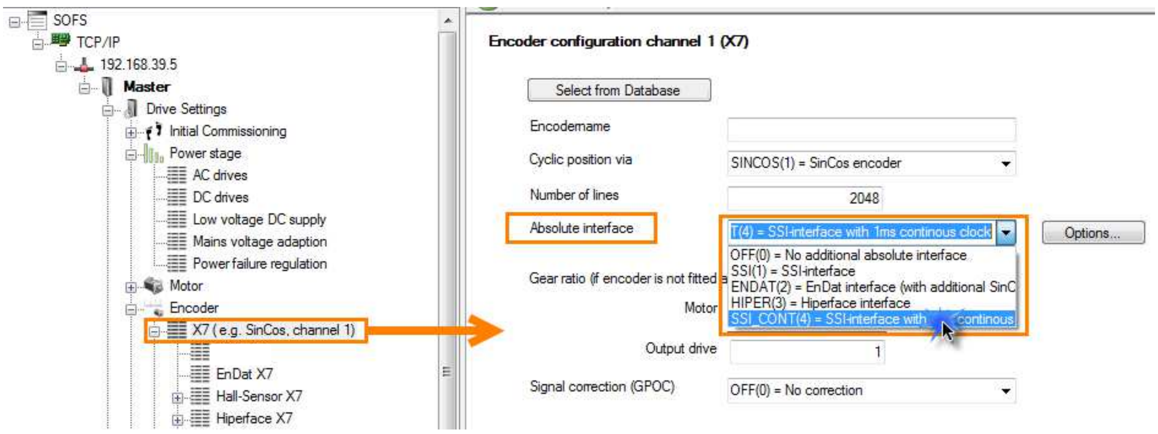
18.6.1 Configuring SSI encoder parameters in the servo drive

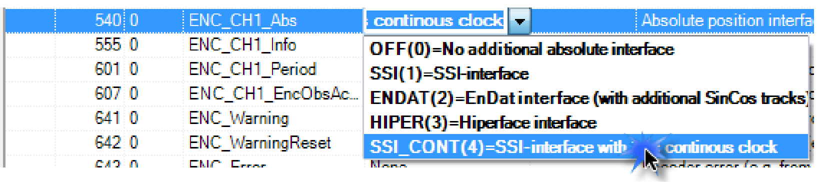
For the correct operation of the SSI encoder, settings must be made in the servo drive, along with the parameters in the safety control.

These settings can be made via the DRIVEADMINISTRATOR 5 user interface. If there is no dialog box for the parameter to be set, the setting can also be made directly in the parameter list.

Setting: continuous clock

To read the necessary data from the SSI encoder, the encoder requires a continuous clock signal on the clock lines (CLK+ / CLK-). So that the servo drive provides a continuous clock, the following parameter is to be set:

<table>
<tr><th>No.</th><th>Index</th><th>Abbreviation</th><th>Description</th><th>Setting</th></tr>
<tr><td>540</td><td>0</td><td>ENC_CH1_Abs</td><td>Absolute position interface selection</td><td>SSI_CONT(4)</td></tr>
<tr><td colspan="5">Settings dialog box in DRIVEADMINISTRATOR</td></tr>
<tr><td colspan="5">


</td></tr>
<tr><td colspan="5">Parameter list in DRIVEADMINISTRATOR</td></tr>
<tr><td colspan="5">


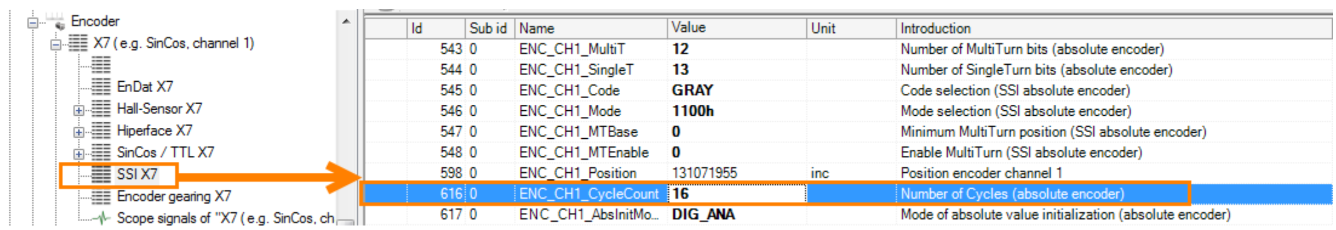
</td></tr>
</table>

Setting the repetition time for the clock signal

In addition to setting for a continuous clock, the intervals at which the clock signal is to be repeated must be set.

No.	Index	Abbreviation	Description	Setting
616	0	ENC_CH1_CycleCount	Number of Cycles (absolute encoder)	16(1)

Parameter list in DRIVEADMINISTRATOR



(1) About the setting: because the number entered here always signifies a multiple of 125 µs, 16 means that a clock signal is output every 2 ms.

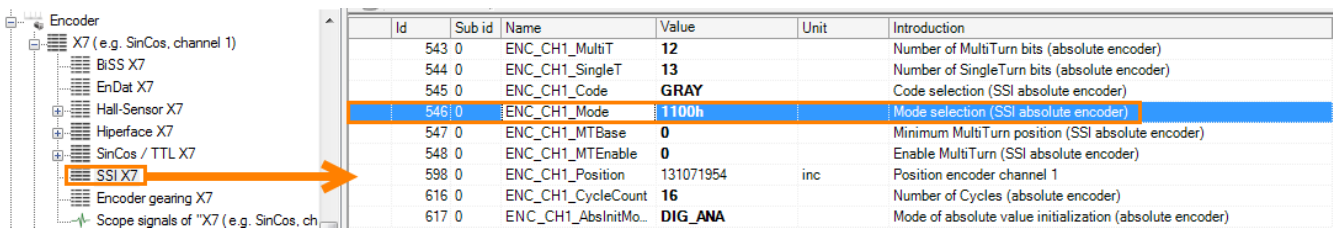
16 * 125 µs = 2000 µs ≙ 2 ms

Setting the mode

So that the mode for evaluating the SSI encoder is enabled, a parameter must also be set.

No.	Index	Abbreviation	Description	Setting
546	0	ENC_CH1_Mode	Mode selection (SSI Absolute encoder)	1100(1)

Parameter list in DRIVEADMINISTRATOR



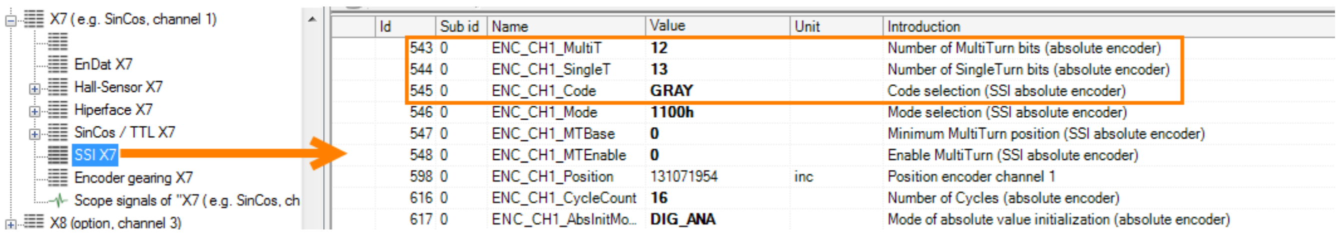
(1) About the setting: With the setting 1100, the SSI mode for the safety control is enabled.

Setting encoder-specific parameters

If not already undertaken, it is necessary to make the encoder-specific settings in the servo drive for the functional evaluation of the SSI encoder.

No.	Index	Abbreviation	Description	Setting
543	0	ENC_CH1_MultiT	Number of multiturn bits (absolute encoder)	Encoder-specific
544	0	ENC_CH1_SingleT	Number of singleturn bits (absolute encoder)	Encoder-specific
545	0	ENC_CH1_Code	Code selection (SSI absolute encoder)	Encoder-specific

Parameter list in DRIVEADMINISTRATOR

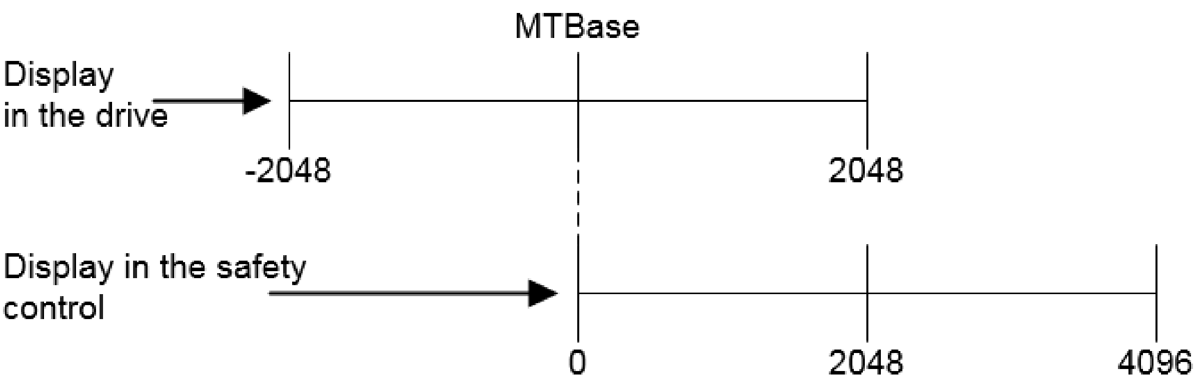


Id	Sub id	Name	Value	Unit	Introduction
543	0	ENC_CH1_MultiT	12		Number of MultiTurn bits (absolute encoder)
544	0	ENC_CH1_SingleT	13		Number of SingleTurn bits (absolute encoder)
545	0	ENC_CH1_Code	GRAY		Code selection (SSI absolute encoder)
546	0	ENC_CH1_Mode	1100h		Mode selection (SSI absolute encoder)
547	0	ENC_CH1_MTBase	0		Minimum MultiTurn position (SSI absolute encoder)
548	0	ENC_CH1_MTEnable	0		Enable MultiTurn (SSI absolute encoder)
598	0	ENC_CH1_Position	131071954	inc	Position encoder channel 1
616	0	ENC_CH1_CycleCount	16		Number of Cycles (absolute encoder)
617	0	ENC_CH1_AbsInitMo...	DIG_ANA		Mode of absolute value initialization (absolute encoder)

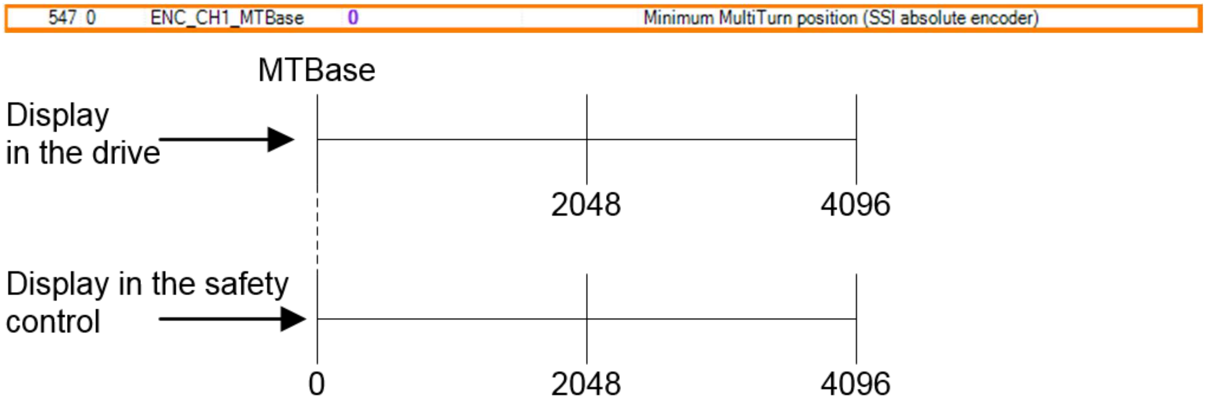
TIP: The parameter "547[0] MTBase" applies a 0-point offset to the multiturn range. With the factory setting the zero point is in the middle of the multiturn range, as specified by the MTBase setting.To see the same position in the safety control and the servo drive, it is recommended to set MTBase to 0.

1) Effect of MTBase set to the factory setting based on the example of an SSI encoder with 12 bit multiturn revolutions, normalised to revolutions:

547	0	ENC_CH1_MTBase	-2147483648	Minimum MultiTurn position (SSI absolute encoder)



2) Effect of MTBase set to 0 based on the example of an SSI encoder with 12 bit multiturn revolutions, normalised to revolutions:



18.7 Setting the parameters for Option 2 Technology

Depending on requirements, it is possible to equip the MSDFS with an Option 2 Technology. These options are installed from the factory and can be ordered together with the servo drive.

18.7.1 Second safe Sin/Cos encoder

Using this option it is possible to evaluate a second Sin/Cos or TTL encoder via the integrated safety control. The evaluation of only one Sin/Cos or TTL encoder is already included in the standard MSDFS device.

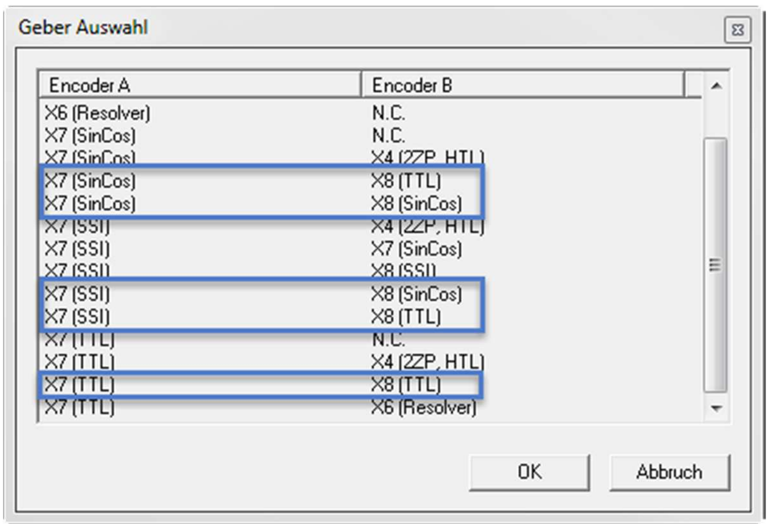


Figure 83: Additional encoder combinations due to Option 2 Technology "Second safe Sin/Cos encoder"

The parameters for the second SIN/COS or TTL encoder are configured in a similar manner to the parameters described in chapter "Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.".

18.7.2 Second safe SSI encoder

Using this option it is possible to evaluate an SSI encoder as a second channel via the integrated safety control. The evaluation of only one SSI encoder is already included in the standard MSDFS device.

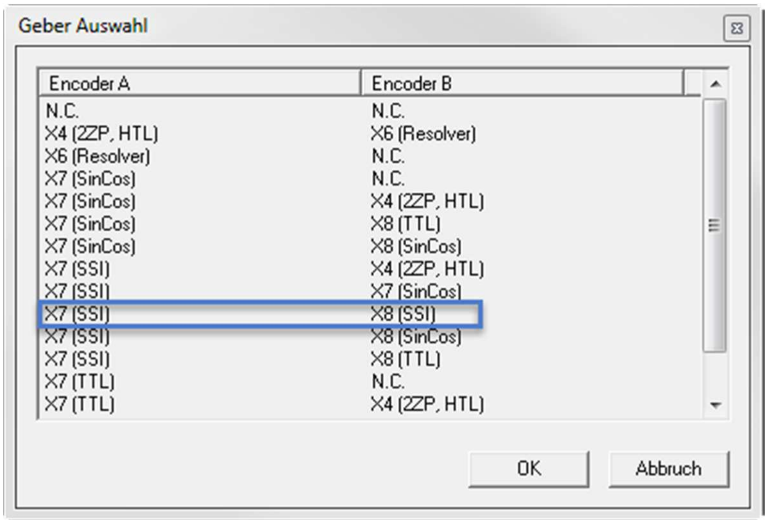


Figure 84: Additional encoder combination due to Option 2 Technology "Second safe SSI encoder"

The parameters for the second SSI encoder are configured in a similar manner to the parameters described in chapter "Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.".

18.7.3 Second safe axis monitor (Sin/Cos)

Using this option it is possible to evaluate an external drive axis via the integrated safety control in the MSDFS.

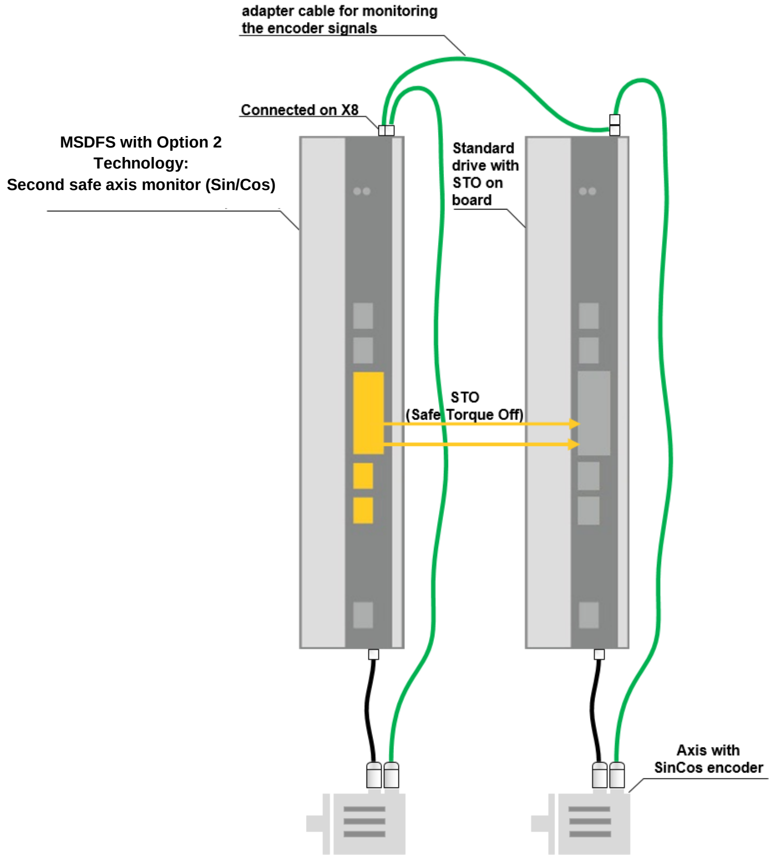


Figure 85: System example

Double-click the encoder symbol in the terminal diagram to select. The parameters for the Sin/Cos encoder are configured in a similar manner to the parameters described in chapter "Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.".

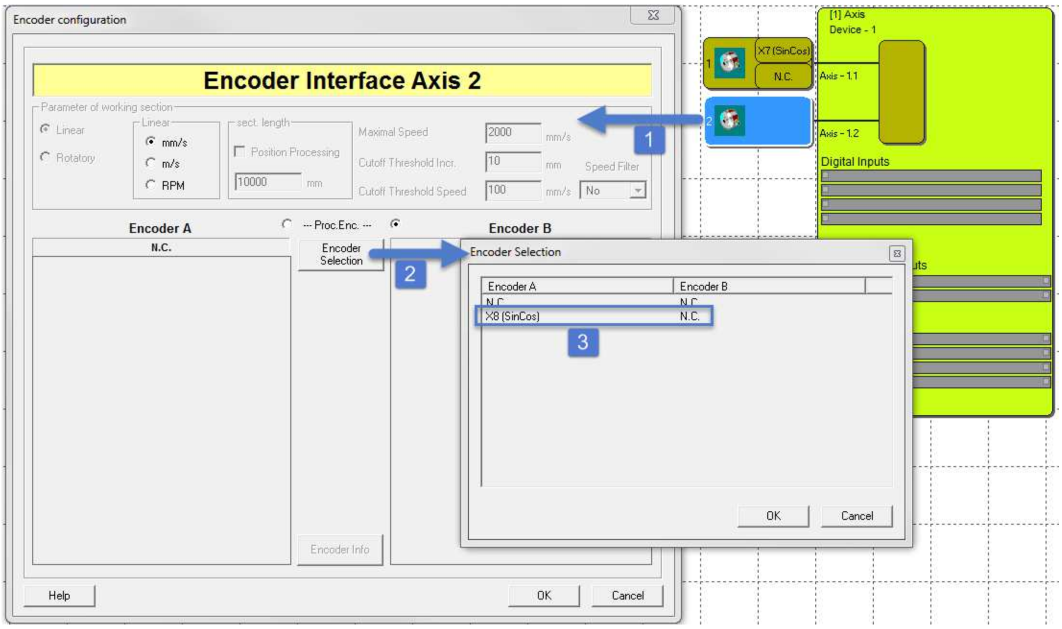


Figure 86: Additional function due to Option 2 "Second safe axis monitor (Sin/Cos)

19 Safe monitoring functions

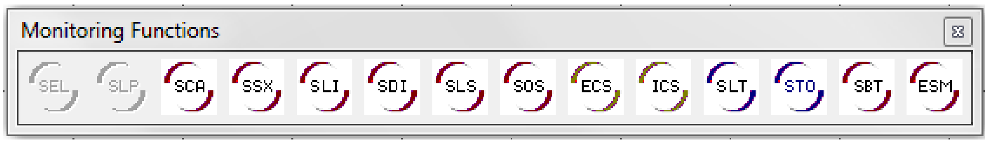


Figure 87: Monitoring Functions

The safe monitoring functions form the key functionality of the safe controller. Pre-defined functions are available for:

- SEL (Safe Emergency Limit)
- SLP (Safely Limited Position)
- SCA (Safe Cam)
- SSX (Safe Stop 1 / Safe Stop 2)
- SLI (Safely Limited Increment)
- SDI (Safe Direction)
- SLS (Safely Limited Speed)
- SOS (Safe Operating Stop)
- STO (Safe Torque Off)
- ECS (Encoder Supervisor)
- ESM (Encoder Standstill Monitoring)

The functionality for monitoring the position, speed and cutoff is only activated after successful encoder configuration on the terminal diagram. Once this action has been undertaken, the related functions can be added as long as the related resources are available in the MSDFS. Once these resources are exhausted, the menu command for the function block is no longer available.

NOTE: If position monitoring is not activated in the encoder configuration, the related controls in the dialog boxes are not available.

The monitoring functions are calculated within the cycle time of the MSDFS and provide a 1 bit result on the output. The result can be connected to an output using logical operators.

If a monitoring function is not activated, it outputs a logical "1" on the output connector. The monitoring function is activated with a rising edge on the input connector and continues to output a logical "1" in the "good" state. If a monitoring limit within the monitoring function is infringed, the state on the output connector changes to a logical "0".

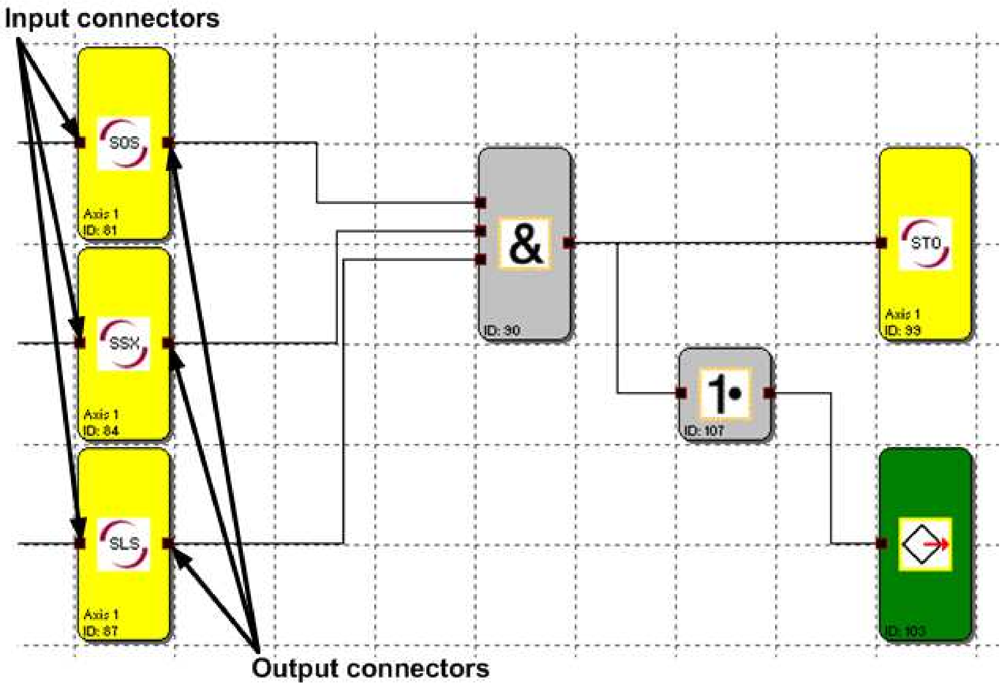


Figure 88: Example for logical processing of monitoring functions

19.1 SEL (Safely Emergency Limit)

Monitoring the maximum movement range

Quantity: 6

Access ID: Identification of the function element

Axis assignment: 1 function per axis and absolute value encoder

Function: Monitoring the permissible speed referred to the relative distance to the maximum movement range / adjustment range limit. This function replaces the usual safety limit switches (elec. emergency limit switches)!

Input: Normalised position signal X from the encoder interface

Function description:

- Calculation of the actual speed V from the position signal X
- Cyclic determination of the stopping distance referred to the status of the acceleration and speed.

  Stop_DistanceAct. = f (V, a) with a = acceleration
- Comparison: PosAct. + Stop_DistanceAct. < Target_Posmax
- Comparison: PosAct. – Stop_DistanceAct. > Target_Posmin

The calculation is based on a trapezoidal or S-shaped speed profile. For a trapezoidal speed profile the limit curve is given by the acceleration set in the parameters, while for an S-shaped speed profile the acceleration change (smoothing) also flows into the calculation.

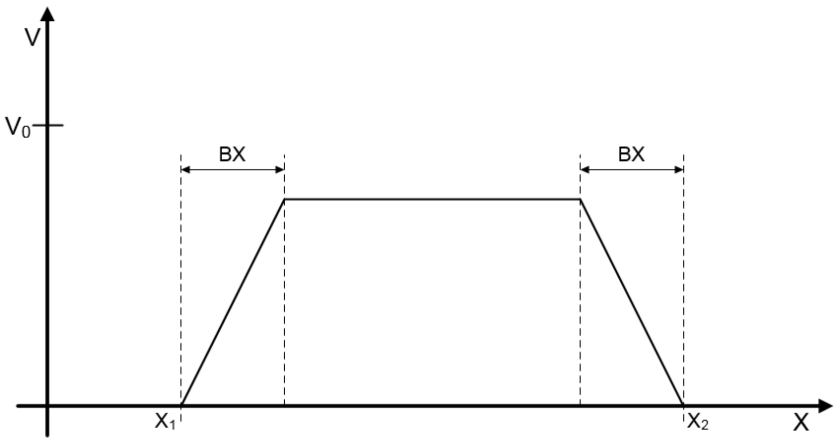


Figure 89: Trapezoid-shaped speed profile

BX	=	Braking/approach area
X1	=	Min. position
X2	=	Max. position
V0	=	Maximum speed for (X1 + BX) < X < (X2 - BX)

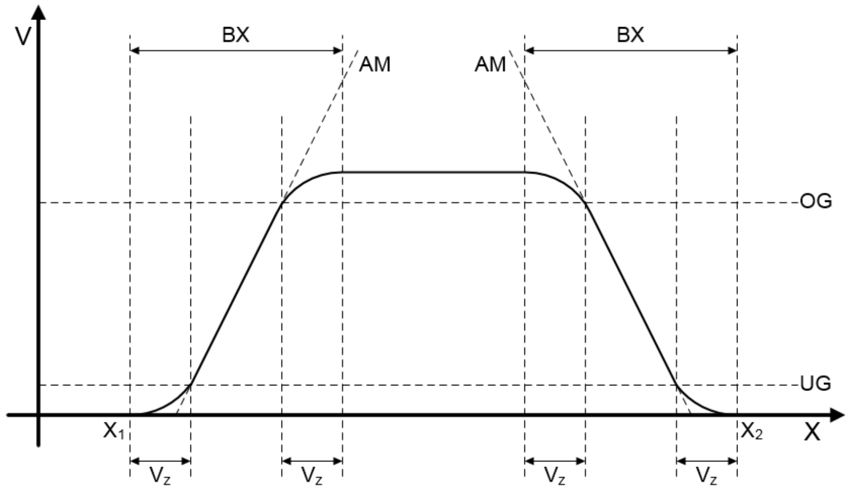


Figure 90: S-shaped speed profile

BX	=	Braking/approach area
X1	=	Min. position
X2	=	Max. position
Vz	=	Smoothing time
AM	=	Max. acceleration
UG/OG	=	Area of the max. acceleration

Output function:

Range	**HI**	**LO**
X < X1 OR X > X2		X
X >= X1 AND X <= (X1 + BX) AND V < limit curve	X	
X >= (X2 – BX) AND X <= X2 AND V < limit curve	X	
X >= X1 AND X <= (X1 + BX) AND V >= limit curve		X
X >= (X2 – BX) AND X <= X2 AND V >= limit curve		X

Limit curve = speed profile derived from the actual parameter settings

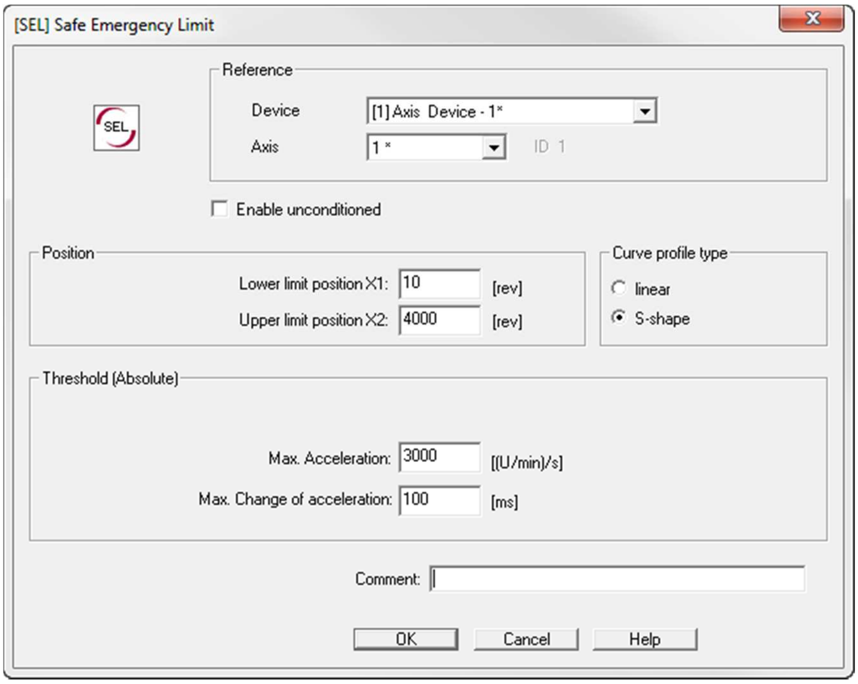


Figure 91: SEL Safe Emergency Limit

Enable unconditioned:

The monitoring function is always active and has no input connector.

Position

Lower limit position X1

Lower limit position that it is not allowed to drop below.

Upper limit position X2

Upper limit position that it is not allowed to exceed.

Curve profile type

Linear

Trapezoid-shaped speed profile.

S-shape

S-shaped speed profile (with smoothing).

Threshold (Absolute)

Max. Acceleration

Maximum acceleration.

Max. Change of acceleration

Smoothing time in ms (only active if S-shaped speed profile is selected).

19.2 SLP (Safely Limited Position)

Monitoring of movement to the target position

Quantity: 6

Access ID: Identification of the function element

Axis assignment: 1 function per axis and absolute value encoder

Function: Monitoring the permissible speed referred to the relative distance to a target position configured.

Input: Normalised position signal X from the encoder interface

Function description

- Calculation of the actual speed V from the position signal X
- Cyclic determination of the stopping distance referred to the status of the acceleration and speed.

  Stop_DistanceAct. = f (V, a) with a = acceleration
- Comparison: PosAct. + Stop_DistanceAct. < Target_Pos
- Comparison: PosAct. - Stop_DistanceAct. > Target_Pos
- Direction-dependent activation of the function, where CW = counting up and CCW = counting down.

NOTE: On the activation of the function, it must be ensured in the program that CW and CCW are not 1 at the same time. Otherwise an alarm will be output.

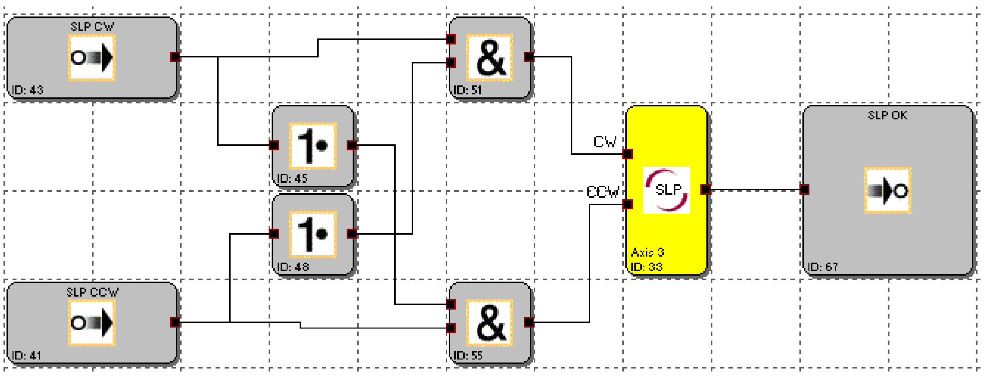


Figure 92: Example SLP request

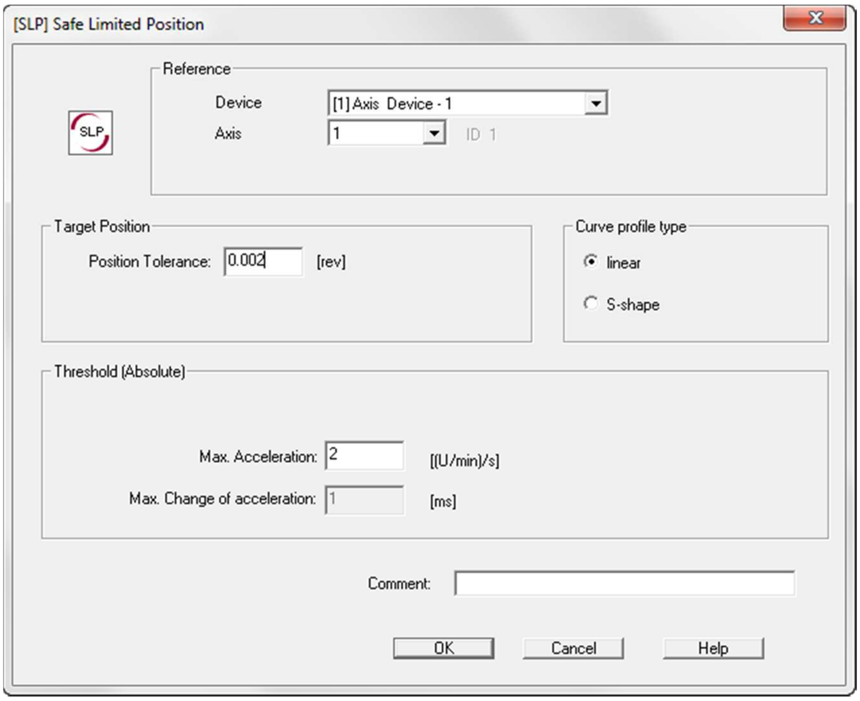


Figure 93: SLP Safe Limited Position

Target Position

Position Tolerance

Absolute position value for the target position (X1), depending on the activation of the function (CW or CCW).

Curve profile type

Linear

Trapezoid-shaped speed profile.

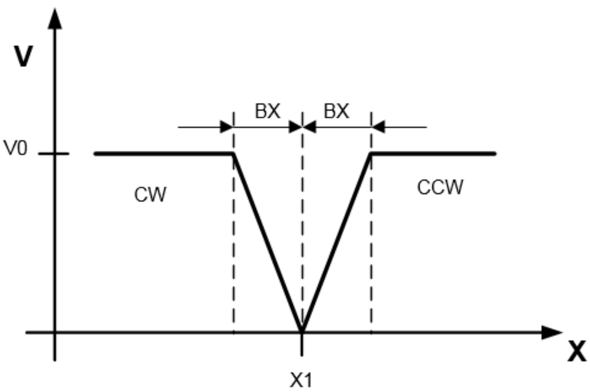


Figure 94: Trapezoid-shaped speed profile

BX = Braking/approach area

X1 = Target position

CW = CW input activated (PosAct. + Stop_DistanceAct. < Target_Pos)

CCW = CCW input activated (PosAct. - Stop_DistanceAct. > Target_Pos)

S-shape

S-shaped speed profile (with smoothing).

Threshold (Absolute)

Max. Acceleration

Maximum acceleration.

Max. Change of acceleration

Smoothing time in ms (only active if S-shaped speed profile is selected).

19.3 SCA (Safe Cam)

Position range monitoring with speed/velocity monitoring

Number:	64
Access ID:	Identification of the function element
Axis assignment:	Any
Function:	Monitoring of a configurable position range with minimum and maximum limit assigned. Additional monitoring of the maximum speed /velocity in the range allowed.
Input:	Standardised speed signal V from the encoder interface

Function description:

- Comparison of the actual position and the range limits configured
- Comparison of the actual speed with the limit configured
- Comparison of the actual acceleration with the limit configured
- Monitoring of the position limits via ramp functionality
- Detection of the direction
- Continuous activation of the module
- Overspeed distance monitoring

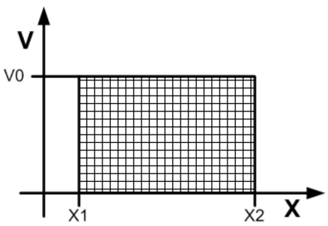


Figure 95: SCA function

Output function:

Range		HI	LO
X < X1 X > X2	OR		X
X >= X1 X <= X2 V < V0 X >= X1 X <=X2 V >= V0	AND AND AND AND	X 	 X

Ranges can be overlapping and defined so they are nested.

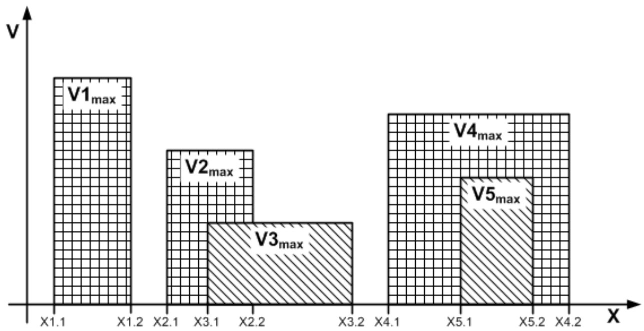


Figure 96: Nesting several SCA blocks

NOTE: You can only use position-dependent functions if position processing is activated.

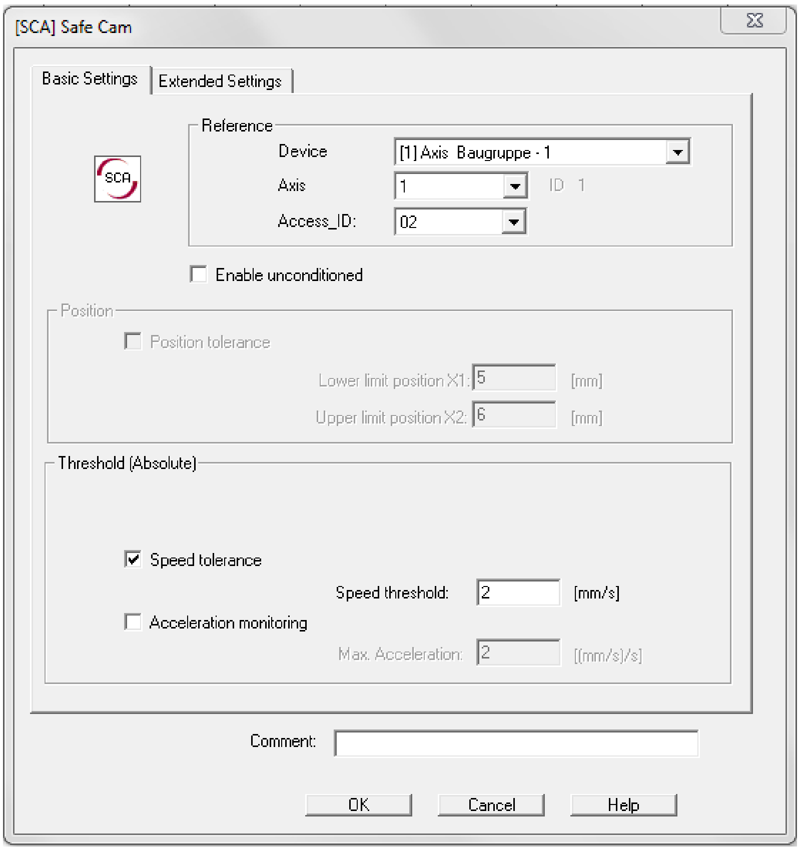


Figure 97: SCA pos. /speed monitoring basic settings

Basic Settings

Enable unconditioned:

The monitoring function is always active and has no input connector.

Position

Lower limit position X1

Lower limit position.

Upper limit position X2

Upper limit position

Threshold (Absolute)

Speed threshold
Maximum permissible speed in the position range configured

Maximum Acceleration
Maximum permissible acceleration in the position range configured

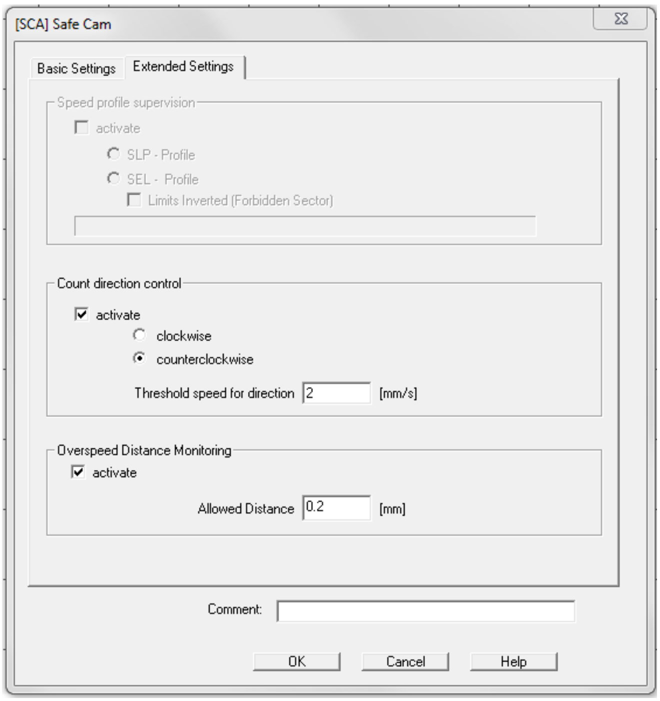


Figure 98: SCA pos./speed extended settings

Extended Settings

Count direction control (SDI Safe Direction)

Permits the activation of downstream function blocks as a function of the direction. This functionality can only be used without speed and acceleration monitoring.

Clockwise

Function block provides the output value = "1" for an increasing position signal

Counterclockwise

Function block provides the output value = "0" four a decreasing position signal

Threshold speed for direction

The direction-dependent enable is only evaluated from the limit defined. Below the speed threshold the output value = 0;

Overspeed Distance Monitoring

This additional functionality makes it possible to filter peaks in the speed during uneven operation (peaks in the speed peaks on the signal).

For further information see SLS function.

CAUTION: On the usage of this function the behaviour of the application changes. On this topic it is essential you pay attention to the information given in the installation manual!

19.4 SSX (Safe Stop 1 / Safe Stop 2)

Emergency stop function monitoring

Number: 12

Access ID: Identification of the function element

Axis assignment: Maximum 2 functions per axis

Function: Monitoring an emergency stop function

Input Standardised speed signal V from the encoder interface

Function description:

Monitoring the process during a controlled emergency stop by comparing the speed drop over time with a configurable monitoring limiting curve. The monitoring limiting curve is defined by the latency time, the maximum speed difference in relation to the limit curve, as well as its characteristic, calculated from the acceleration and the change in acceleration. After the activation of the monitoring the shape of the limit curve is calculated starting from the actual speed.

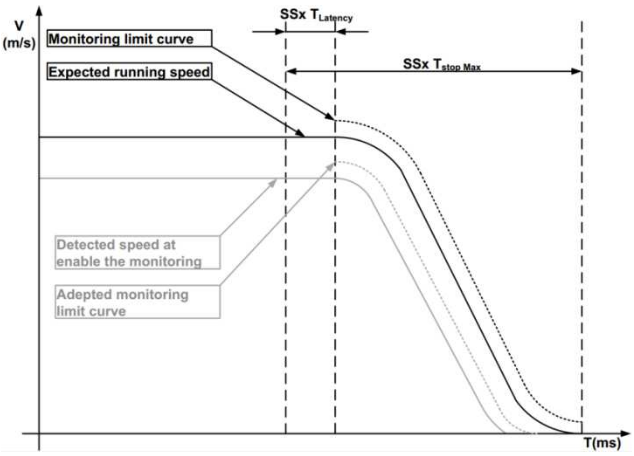


Figure 99: Monitoring curves for speed with jerk limitation

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

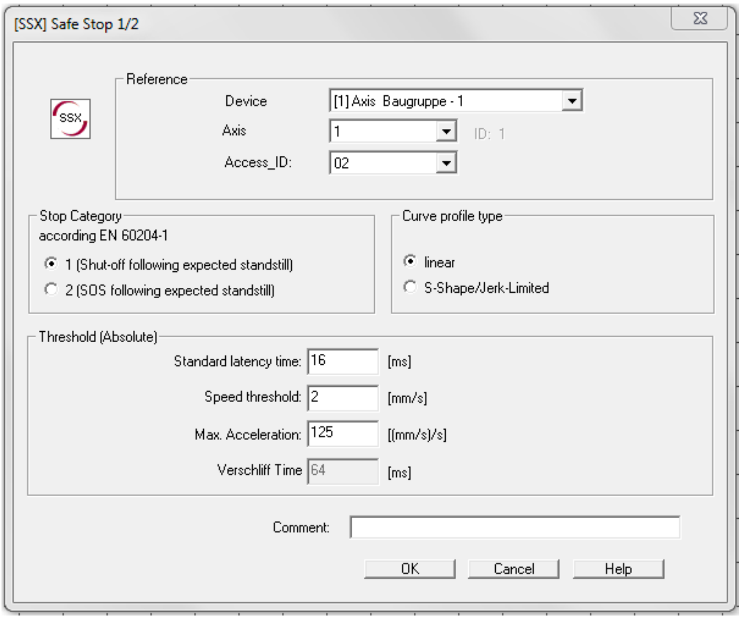


Figure 100: SSX drive emergency stop monitoring

Each function block can be configured for stop category 1 or 2.

Stop category in accordance with IEC/EN 60204-1

1 (shutdown following expected standstill)

This option realises the monitoring of the controlled emergency stop in accordance with IEC/EN 60204. As per the normative definition here the supply of power should be disconnected after the drive has reached standstill. This feature is supported by a transition on the output value from "1" to "0" on the SSX function.

2 (SOS following expected standstill)

This option realises the monitoring of the controlled emergency stop in accordance with IEC/EN 60204. After the end of the ramp monitoring, as per the normative definition, the drive is to be monitored for standstill without the disconnection of the supply of power (Safe Operational Stop). For this reason, after the end of the SSX limit curve the output value remains at the value "1".

If an SOS block is not defined in the function block diagram, it must be added. All the parameters necessary for the SOS function can be entered using this block.

NOTE: If the SSX function is used in conjunction with the SOS function, the circuit shown below is to be used. With standstill detected, the operating system automatically activates the SOS monitoring.

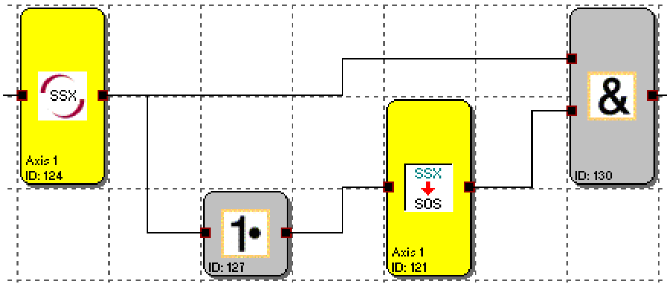


Figure 101: SS2 function

Curve profile type

Linear

The stop process is monitored using a linear monitoring limiting curve.

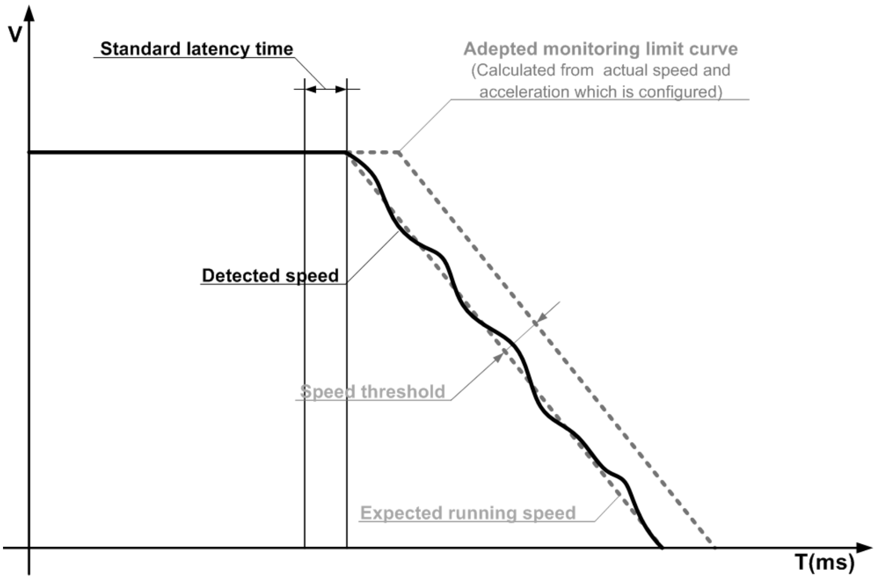


Figure 102: Linear curve profile type

S-Shape/Jerk Limited

The stop process is monitored using a monitoring limiting curve including smoothing.

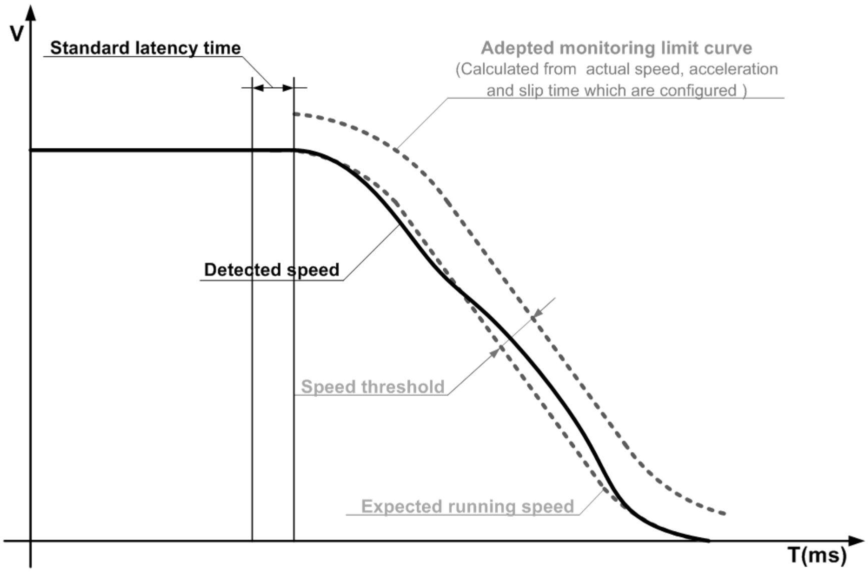


Figure 103: S-shape/jerk limited curve profile type

Standard latency time

Latency time until the active deceleration starts

Speed threshold

Relative speed along the calculated limit curve.

Maximum Acceleration

Definition of the acceleration (Maximum deceleration ramp / stop ramp) for the calculation of the limit curve.

Verschliff Time (Smoothing time)

Definition of the acceleration change (smoothing time) for the calculation of the limit curve.

Example entry

On a manufacturing machine the access to the working area for manual insertion or set-up is to be enabled at certain positions of the main feed axis. The drive remains active in this position and is monitored for standstill. The active movement to be monitored represents a linear movement. A travel measuring system is positively and directly connected to this main drive axis. Drive is provided using an electric motor with integrated encoder and an intermediate gearbox.

Selection of the stop category

To keep standstill times and restarting times low, stop category 2 in accordance with IEC/EN 60204-1 (controlled stop with subsequent actively controlled drive at V=0) must be used.

Selection: Stop category 2

Selection of curve profile type

The drive / position controller uses smoothing (jerk limitation) for the acceleration with resulting S shape for the speed.

Selection: S-Shape/Jerk Limited

Selection of standard latency time

For the monitoring the worst-case latency time from the occurrence of the emergency stop event to the start of the braking process implemented by the standard control must be entered. From the program execution time in the higher level standard control (cycle time 8 ms) the following is found:

Latency time = cycle time * 2 = 8 ms * 2 = 16 ms

Entry: 32 ms

Speed threshold

A speed deviation of 20 rpm is assumed as the tolerance or deviation from the ideal braking ramp due to the encoder accuracy.

NOTE: The actual deviation can be checked with the aid of the scope monitor, in the diagnostic functions. It is important to enter a speed deviation as small as possible.

Maximum Acceleration

3000 rpm/s are configured as the maximum deceleration in the servo drive. The value results from the design of the application and is applied directly.

Entry: 3000 rpm/s

Verschliff Time (Smoothing time)

100 ms are configured in the servo drive as the smoothing time (jerk limitation). This value is applied directly.

Entry: 100 ms

19.5 SLI (Safely Limited Increment)

SLI Monitoring the maximum increment

Number: 6

Access ID: Identification of the function element

Axis assignment: Maximum 1 function per axis

Function: Monitoring the Maximum increment allowed

Input: Normalised position / speed signal V and X from the encoder interface. Direction information LEFT/RIGHT.

Function description:

Monitoring the maximum increment allowed = relative movement range for uninterrupted movement in the jog mode. Calculation of the current direction of rotation RX from the position / speed signal X. Determination of the relative travel after the start of the movement. Monitoring for compliance with the stipulated direction and the maximum relative travel.

NOTE: On the activation of the function, it must be ensured in the program that CW and CCW are not 1 at the same time. Otherwise an alarm will be output (see SLP example).

Output function:

Range		HI	LO
V < 0 DIRECTION FLAG = LEFT relative travel < Maximum increment	AND AND	X	
V > 0 DIRECTION FLAG = RIGHT relative travel < Maximum increment	AND AND	X	
V < 0 DIRECTION FLAG = RIGHT relative travel > Maximum increment	AND OR		X
V > 0 DIRECTION FLAG = LEFT relative travel > Maximum increment	AND OR		X

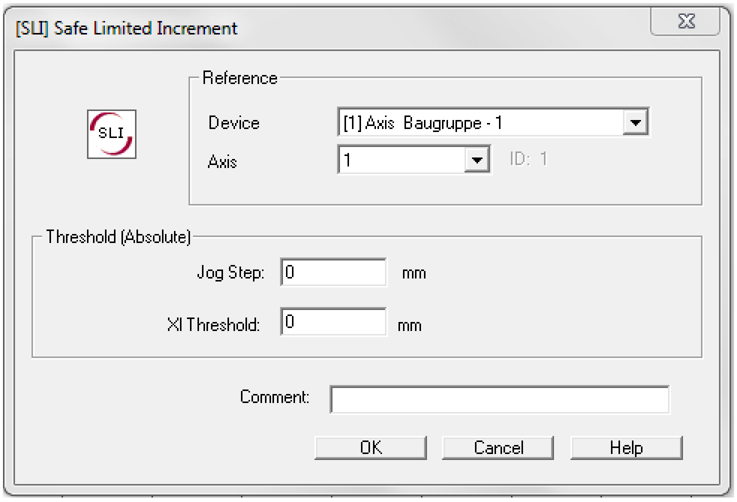


Figure 104: SLI individual step monitoring

Threshold (Absolute)

Jog Step

Maximum relative travel after activation of the monitoring function

XI Threshold

Tolerance threshold for monitoring the travel in the opposite direction

Activation example

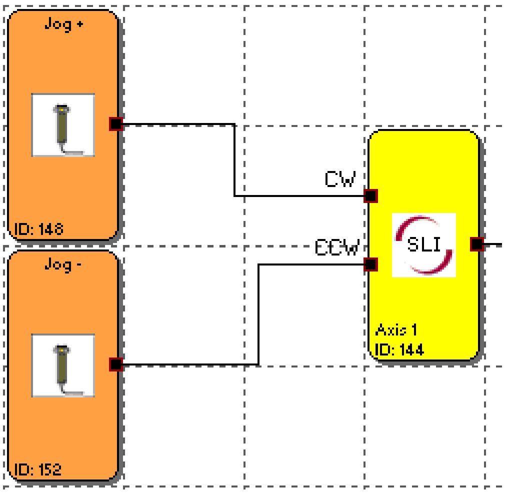


Figure 105: Activation example SLI

NOTE: Both input signals at "1" is detected as an impermissible state and will result in an alarm message.

Example entry

In a material feed on a manufacturing machine the Maximum travel in the jog mode is to be safely monitored. As per the risk analysis this Maximum travel is 50 mm. Incorrect movement in the opposite direction is to be monitored.

Jog Step

Monitors the relative travel (only incremental encoder used). As per the risk analysis a maximum travel of 55 mm incl. tolerance is allowed.

Entry: 55 mm

XI Threshold

Travel allowed in the opposite direction is 1 mm

Entry: 1 mm

Input for the monitoring

The monitoring block has two inputs for defining the direction. The monitoring function is activated with the active direction signal.

19.6 SDI (Safe Direction)

Direction detection

Number: 6

Access ID: Identification of the function element

Axis assignment: Maximum 1 function per axis

Function: Monitoring the stipulated direction of rotation / movement

Input: Normalised position / speed signal X from the encoder interface. Direction flag LEFT/RIGHT.

NOTE: On the activation of the function, it must be ensured in the program that CW and CCW are not 1 at the same time. Otherwise an alarm will be output (see SLP example).

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

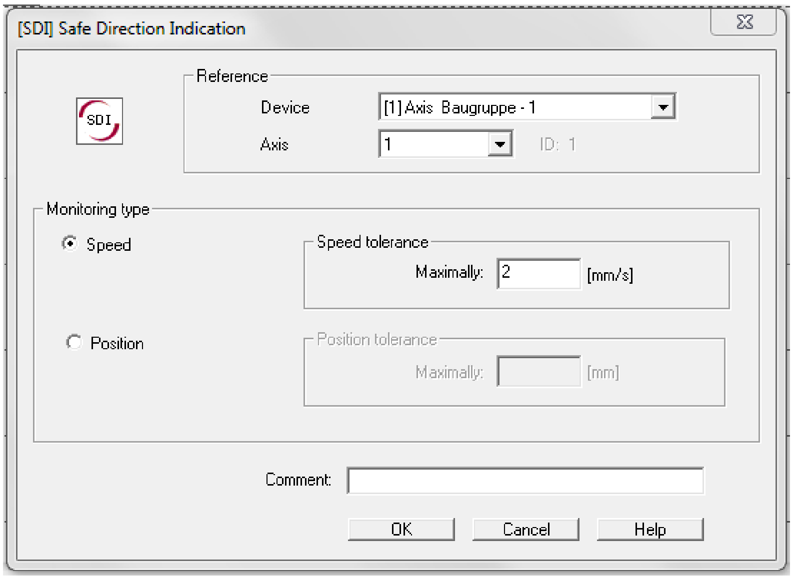


Figure 106: SDI direction monitoring

Monitoring type

Here you can select whether speed or position is to be monitored for movement in the opposite direction.

Maximally

Tolerance threshold for position or speed in the opposite direction.

Activation example

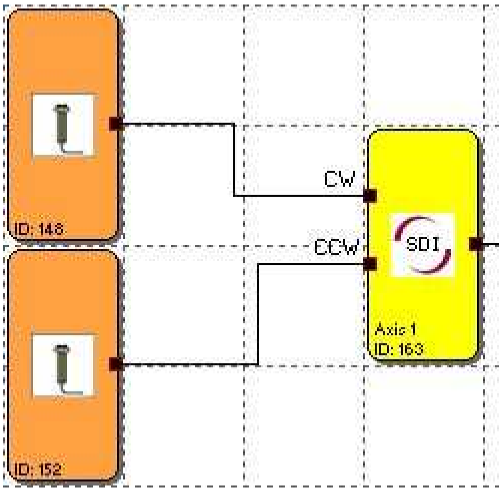


Figure 107: Activation example SDI

NOTE: Both input signals at “1” is detected as an impermissible state and will result in an alarm message.

19.7 SLS (Safely Limited Speed).

Monitoring a maximum speed

Number: 48

Access ID: Identification of the function element

Axis assignment: Any

Function: Monitoring a maximum speed

Input: Normalised position signal X from the encoder interface

Function description:

Monitoring the maximum velocity or speed of a drive. Calculation of the actual speed V from the position, or digital speed signal X. Comparison of the actual speed with the configured speed threshold. Monitoring a speed transition from fast to slow.

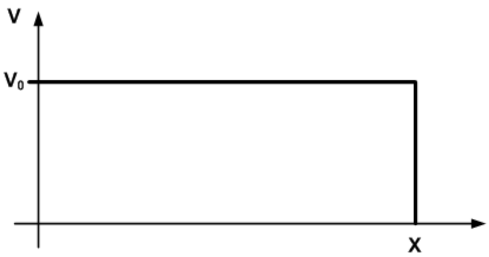


Figure 108: SLS function

Output function:

Range	HI	LO
V < V0	X	
V >= V0		X

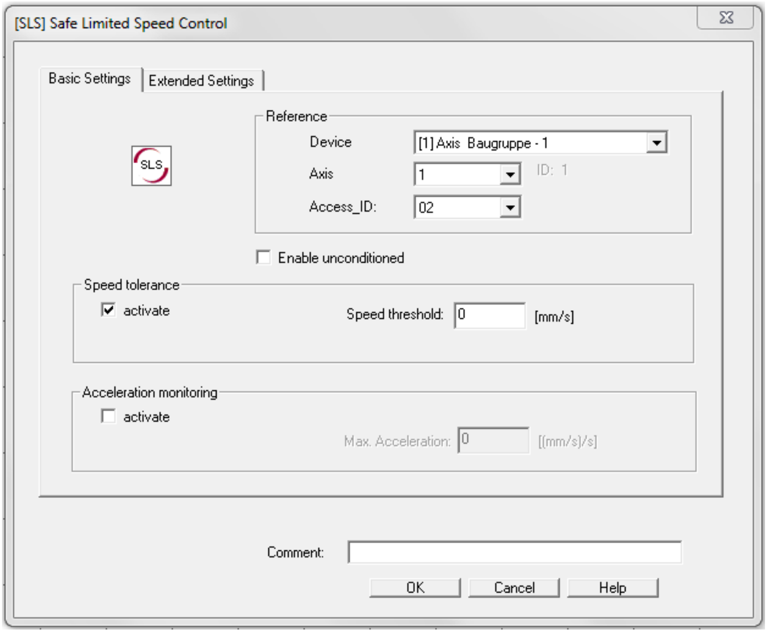


Figure 109: SLS maximum speed monitoring

Basic Settings

Enable unconditioned

The monitoring function is always active and has no input connector.

Speed tolerance

Activation of the speed monitoring.

Speed threshold

Definition of the maximum velocity, alternatively the maximum speed.

Acceleration monitoring

Activation of the acceleration monitoring.

NOTE: The acceleration monitoring corresponds to the SLA (Safe Limited Acceleration) function

Maximum Acceleration

Definition of the maximum acceleration.

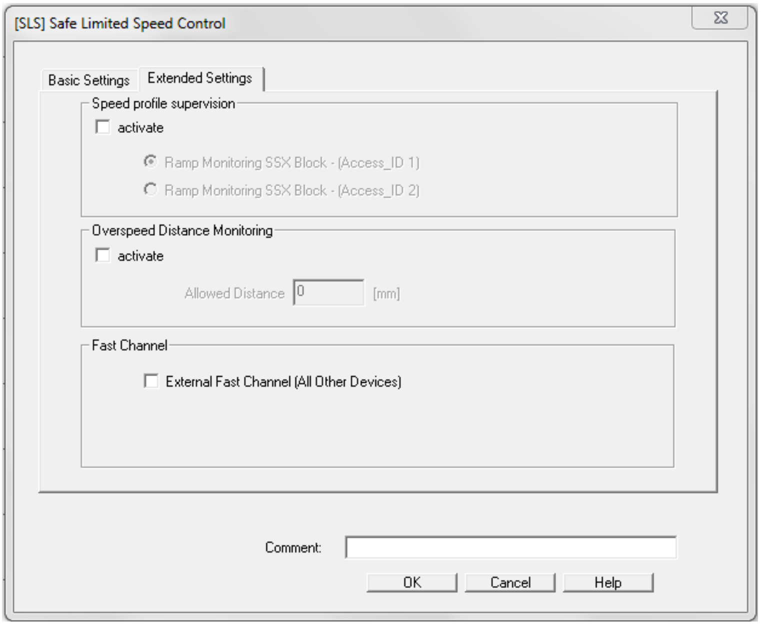


Figure 110: SLS maximum speed, monitoring, Extended Settings

Extended Settings

Speed profile supervision, activate

This option monitors the speed transition from fast to slow using an SSX function (Safe Stop 1 / Safe Stop 2).

Speed monitoring and speed profile monitoring in accordance with SSX must be activated. Here the SSX used must be already added to the project and configured. A transition from a fast to a slower speed can be monitored (parameter: Maximum Speed).

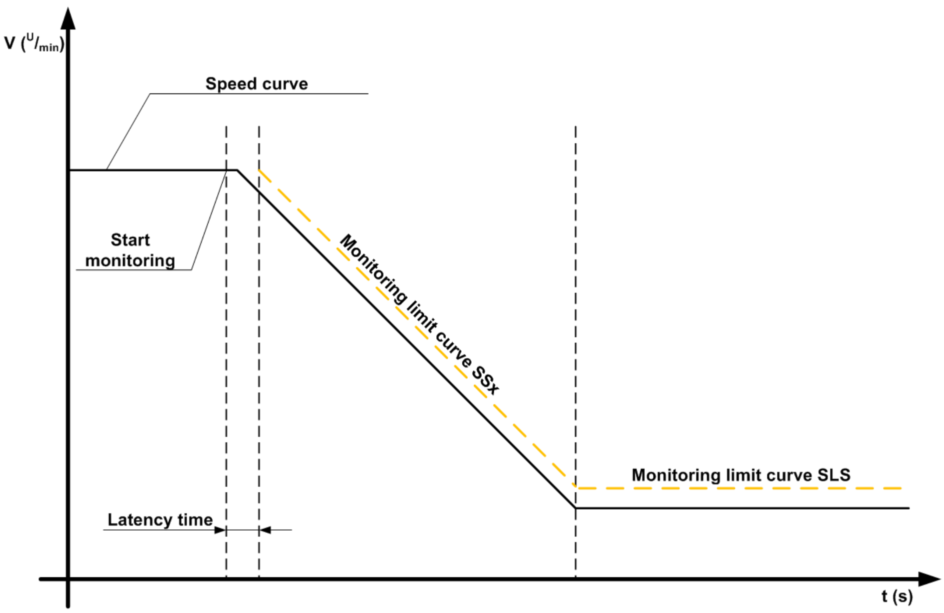


Figure 111: Speed profile monitoring with transition to SLS

If SLS is activated, then the SSX configured is automatically activated via the SLS. The SSX monitors the ramp profile for the speed. If the actual speed is lower than the SLS threshold, the SLS takes over the further monitoring until the SLS is deactivated again.

Diagnostics on the ramp profile can be undertaken as a diagnostics function with the SCOPE monitor.

NOTE: If the SSX used is activated during the "SLS ramp monitoring" (i.e. normal EMERGENCY STOP function via SSX enable), the SSX processing configured always has priority.

The SSX function is then always activated by the SLS if, on the request for SLS, the actual speed is greater than the SLS threshold.

The threshold for the SLS must be greater than 0!

If during the transition from fast to slow the calculated speed profile is exceeded, this situation is saved in both monitoring functions, SLS and SSX.

If several SLS functions with ramp monitoring are activated, the smallest configured SLS threshold is always used as the threshold for the SSX ramp.

Overspeed Distance Monitoring

This additional functionality makes it possible to filter peaks in the speed during uneven operation (peaks in the speed peaks on the signal).

Starting from the difference between the actual speed and the configured speed monitoring value the travel integral is calculated and compared with the value entered. If the value entered is exceeded, the monitoring function is triggered.

The function can only be activated if the acceleration monitoring is disabled.

Example for overspeed distance monitoring:

The illustration shows an example of overspeed distance monitoring. A drive exceeds the threshold "V_{limit}" configured in the SLS function. When the speed is exceeded the speed above the threshold is integrated. If the actual speed drops below the threshold again, the integral also drops. In the rest of the profile the speed increases again and remains above the threshold "V_{limit}" configured. As a consequence the integral also increases again and the shuts down on exceeding the overspeed distance (= integrated speed portion). The process in the integrator can be displayed using the SCOPE function.

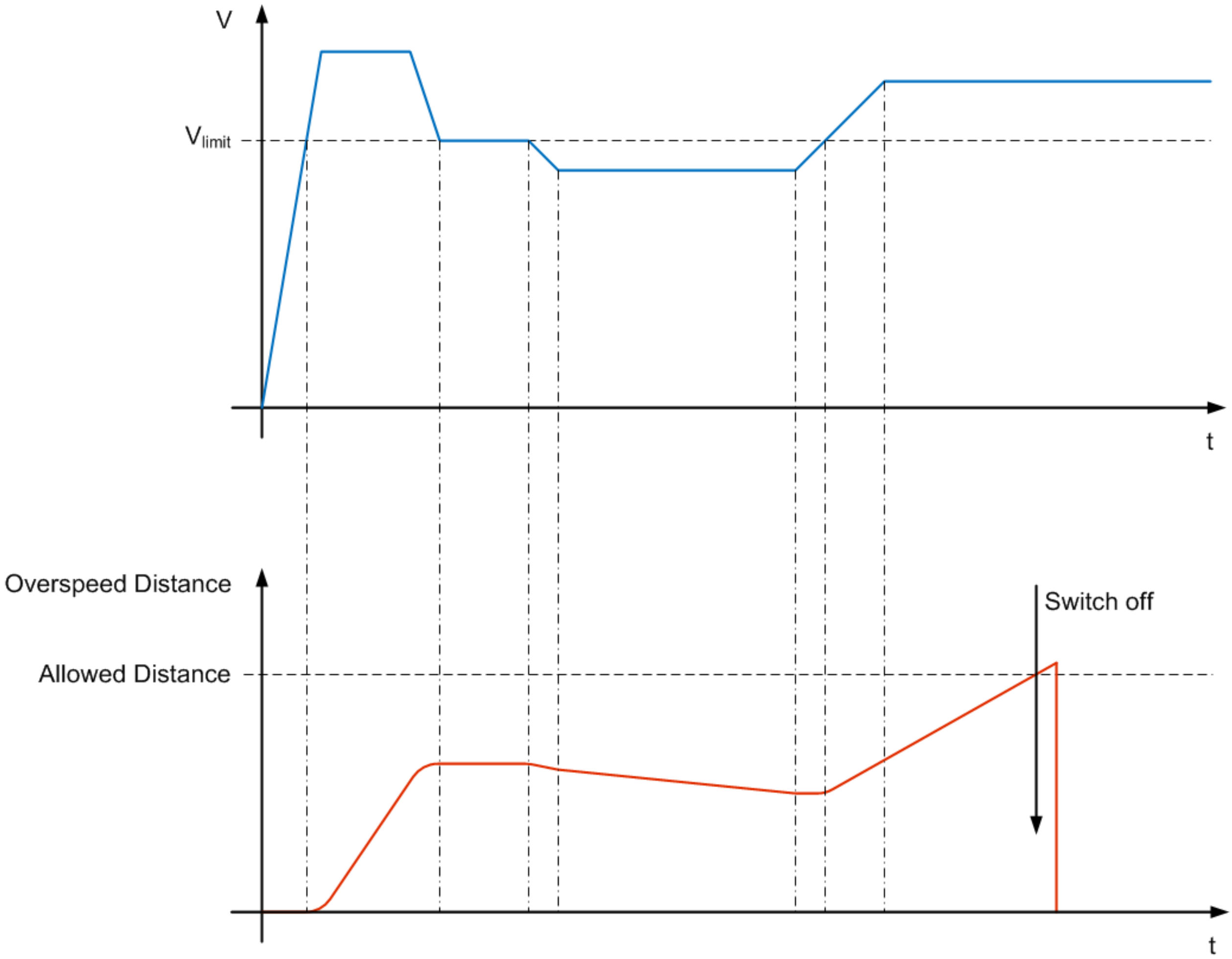


Figure 112: Overspeed Distance Monitoring

CAUTION: On the usage of this function the behaviour of the application changes. On this topic it is essential you pay attention to the information given in the installation manual!

Using fast channel

The system reaction time can be reduced with the "Fast Channel" option (see chapter "19.13 Fast Channel").

19.8 SOS (Safe Operating Stop)

Standstill monitoring

Number: 6

Access ID: Identification of the function element

Axis assignment: Maximum 1 function per axis

Function: Monitoring standstill

Input: Normalised position/speed signal V and X from the encoder interface

Function description:

Standstill monitoring on the drive at the actual position with drive enabled or possibly with position controller activated. Calculation of the actual speed V from the position, or digital speed signal X. Comparison of the actual speed with the configured monitoring window.

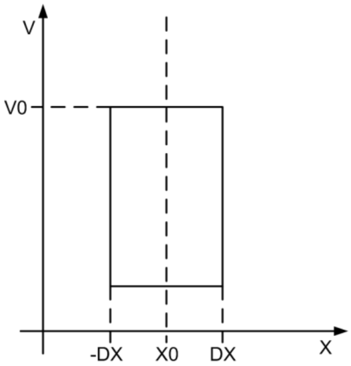


Figure 113: SOS function

Output function for position monitoring:

Range		HI	LO
X > (X0 – DX) X < (X0 + DX)	AND	X	
X <= (X0 - DX) X >= (X0 + DX)	OR		X

Output function for speed monitoring

Range	HI	LO
V < V0	X	
V >= V0		X

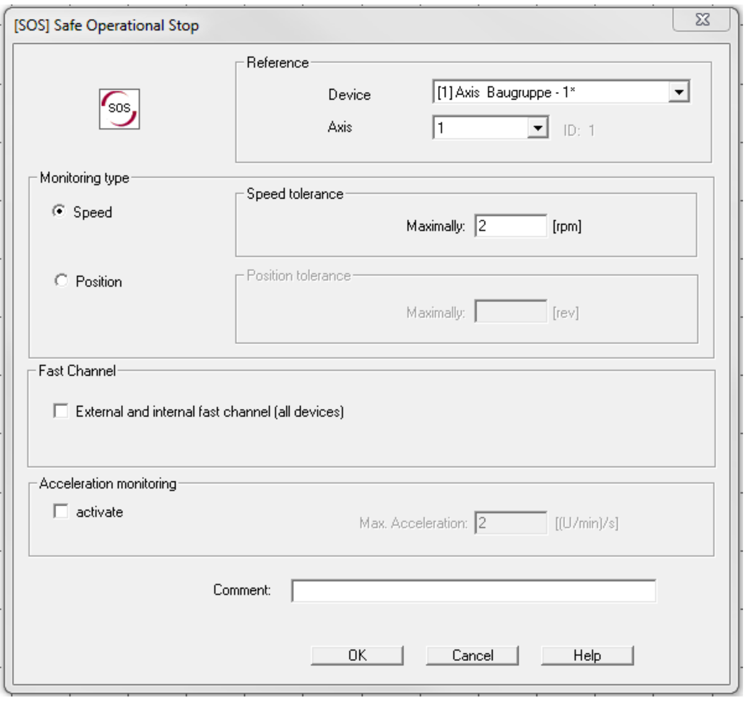


Figure 114: SOS standstill monitoring

Monitoring type

Definition of the monitoring type for standstill based on a minimum speed threshold or a position window.

NOTE: "Creeping" is possible with the "speed threshold" monitoring type for SOS if a position window is not activated.

Maximally

Minimum speed or permissible relative deviation from the actual position at the time of the activation of the SOS functionality.

Fast Channel

The system reaction time can be reduced with the "Fast Channel" option (see chapter "19.13 Fast Channel").

Acceleration monitoring

Optional maximum value for acceleration monitoring while the SOS function is active.

Example entry 1

In a manufacturing machine during specific manual processes the speed is to be monitored for a safe reduced value. The active movement to be monitored represents a rotary movement. Drive is provided using an electric motor with integrated encoder and an intermediate gearbox.

Monitoring type

It is sufficient to monitor the speed in the application.

Selection: Speed

Maximum speed tolerance

A standstill window of 5 rpm is adequate in the application.

Entry: 5 rpm

Example entry 2

On a manufacturing machine the access to the working area for manual insertion or set-up is to be enabled at certain positions of the main feed axis. The drive remains active in this position and is only monitored for standstill. The active movement to be monitored represents a linear movement. A sin/cos encoder is positively and directly connected to this main drive axis as a linear travel measuring system. Drive is provided using an electric motor with encoder and an intermediate gearbox.

Monitoring type

To prevent a creeping movement in the hazardous direction, the relative position must be monitored in the application.

Selection: Position tolerance

Maximum position tolerance

Monitoring for a relative position of 10 mm is adequate in the application.

Entry: 10 mm

19.9 STO (Safe Torque Off)


Safely shutdown torque

Number: 6

Access ID: Identification of the function element

Axis assignment: Maximum 1 function per axis

Function: The flow of power from the servo drive to the motor is interrupted.

Function description:

The block activates and deactivates the safety function "Safe Torque Off (STO)". The supply of power to the drive is safely interrupted (no electrical isolation) on the activation of this safety function. The generation of torque (or force on a linear motor) and therefore a hazardous movement is prevented. The "STO" function complies with stop category 0 as per IEC/EN 60204-1.

Activation example

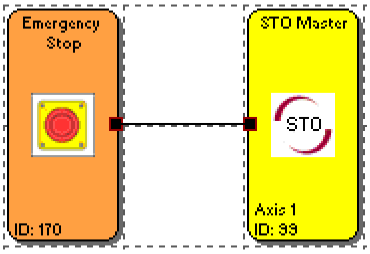
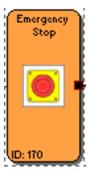


Figure 115: Activation example STO

State of the emergency stop	State of the STO	State of the servo drive
		
Emergency stop not actuated, both normally closed contacts are closed. The function block provides a logical **"1"** on the output.	The safety function STO is **deactivated** by the logical **"1"** on the input of the **STO** function block.	The servo drive indicates the state **"ready to start"**, provided a quick stop is not present and the supply voltage is in the operational range.
Emergency stop actuated, both normally closed contacts are open. The function block provides a logical "0" on the output.	The safety function STO is **activated** by the logical "0" on the input of the **STO** function block.	The servo drive indicates the state **"start inhibit"**.

NOTE: Protection against unexpected restarting after the restoration of the power supply is not provided in the activation example shown. If STO is deactivated on the restoration of the supply of power, the axis may start if autostart is programmed. It is to be ensured that the servo drive (the SRP/CS) can achieve or maintain the safe state of the machine.

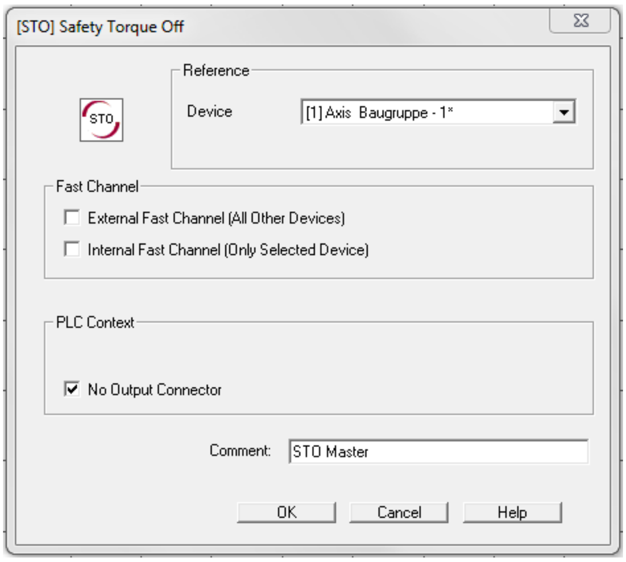


Figure 116: STO Safe Torque Off

Fast Channel

The system reaction time can be reduced with the "Fast Channel" option (see chapter "19.13 Fast Channel").

PLC Context

Here you can remove the output connector from the function block.

19.10ECS (Encoder Supervisor)

User-defined evaluation of the encoder status

Number: 6

Access ID: Identification of the function element

Axis assignment: Maximum 1 function per axis

Function: Evaluation of the encoder status via PLC function

Function description:

The acquisition of the safe speed and position is based on numerous measures and different error reactions in the form of alarm messages. On the detection of a speed/position error the operating system switches the MSDFS from the RUN state to the ALARM state if an ECS element is not used. All safe outputs are inhibited immediately.

On the addition of an ECS element in the function block diagram this state change is suppressed and the operating system remains in the RUN state. The PLC program must now trigger the required measures to prevent dangerous states in the application via the status of the ECS element. Alarm messages from the encoder interface are output with the same reference number with the prefix "E".

NOTE: This function can have a significant effect on the safety of an application. It must be ensured that no safety-critical situations are caused by the usage of the ECS function!

Activation example

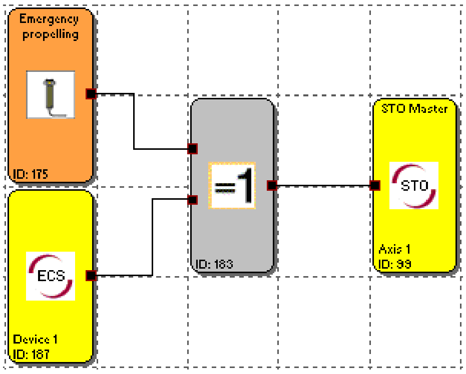


Figure 117: Activation example ECS

Normal state:

The system is operating without any errors and is in the "Run" state.

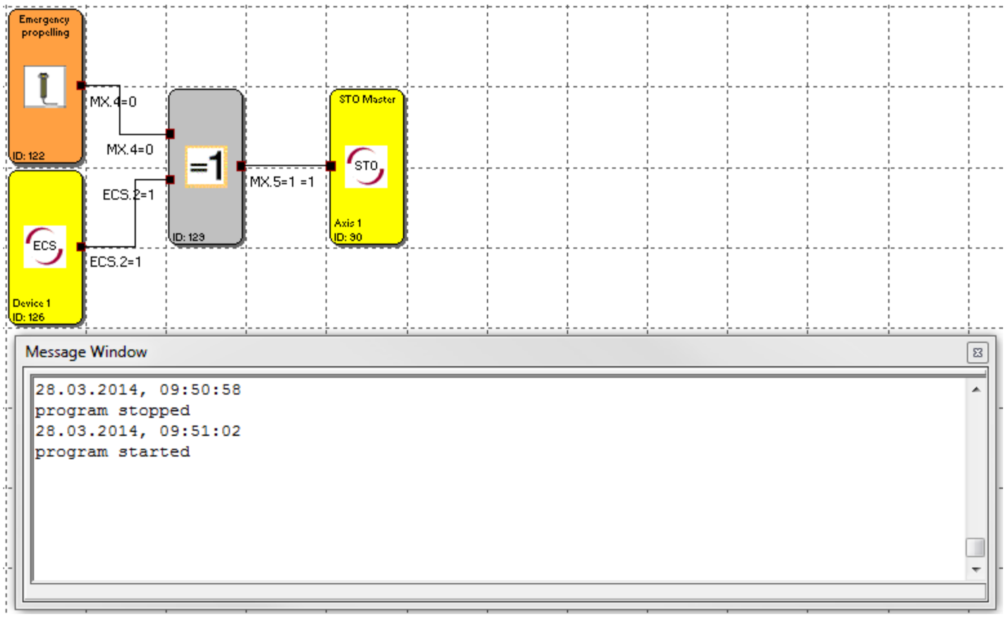


Figure 118: Normal state

Error state:

The system has detected an encoder error, the state of the ECS block changes from 1 to 0. The system is still in the "RUN" state, as the encoder error is suppressed by the ECS block. With the aid of the status message, suitable measures must now be taken or programmed as a reaction to the encoder error.

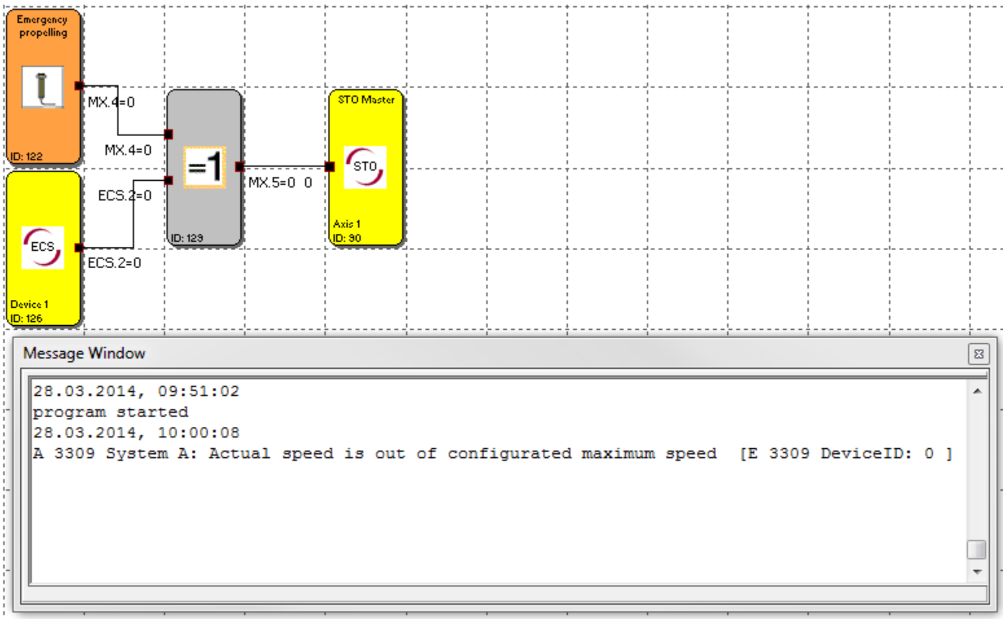


Figure 119: Error state

Enable emergency movement:

Encoder errors are still present and suppressed by the ECS block. With the aid of a control (Emergency propelling) and an EXCLUSIVE OR block emergency movement is possible in this example.

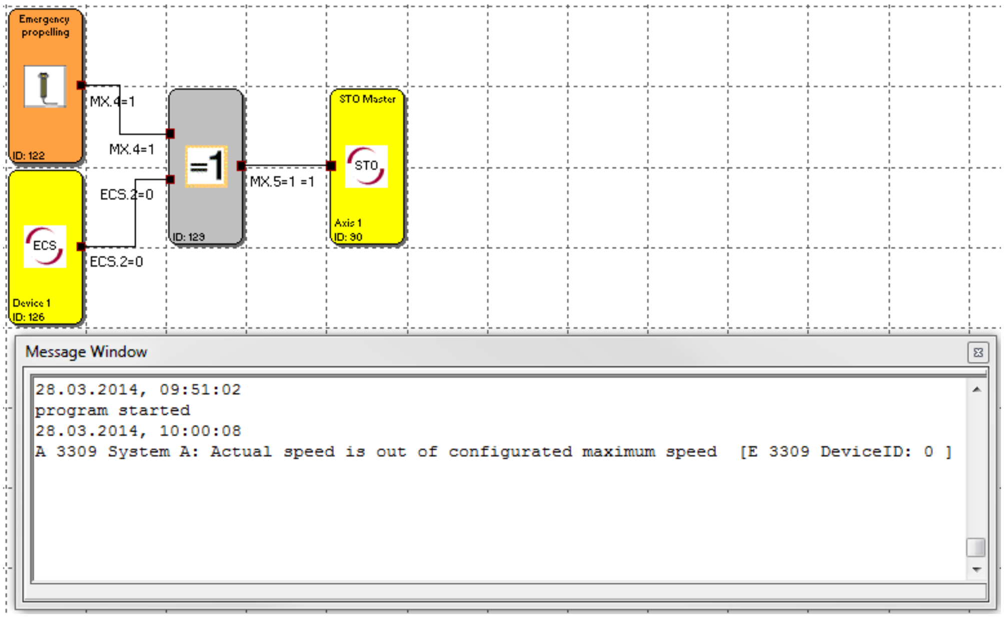


Figure 120: Enable emergency movement

Error rectified:

Encoder error has been rectified and a reset undertaken. The system is operating without errors again. The status of the ECS block is then 1 again.

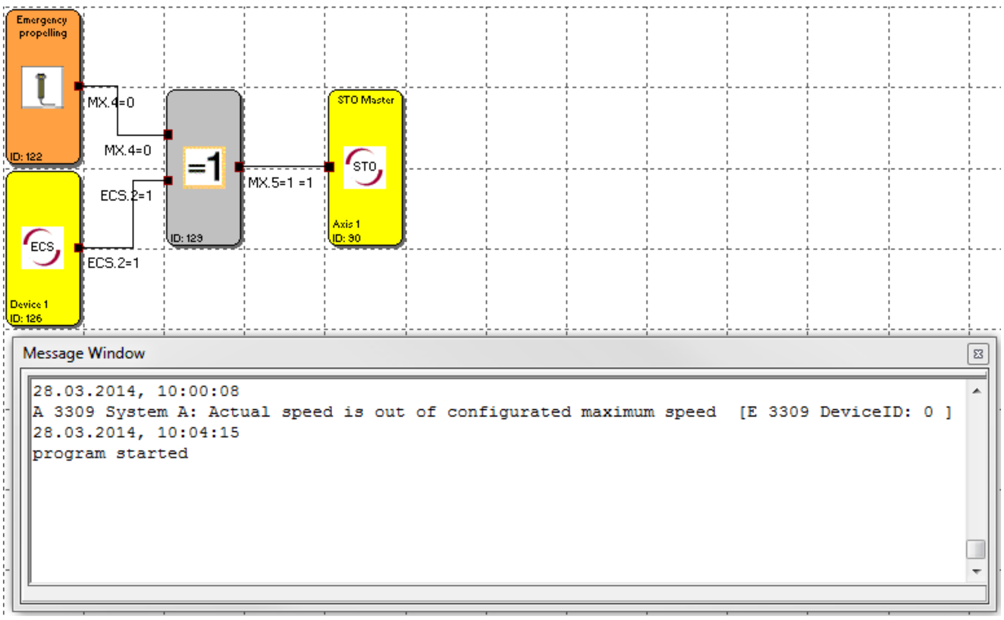


Figure 121: Error rectified

NOTE: The ECS block is permanently activated in the activation example. As standard the ECS block has an input connector with which it is activated. If the ECS block is activated (input connector = 1) and there are no errors present (see list "Suppressed alarms") the ECS block outputs a logical 1. If an error is detected the state of the output changes from 1 to a 0. If the ECS block is not activated (input connector = 0), errors are not suppressed and the corresponding alarm message is output and all outputs rendered passive.

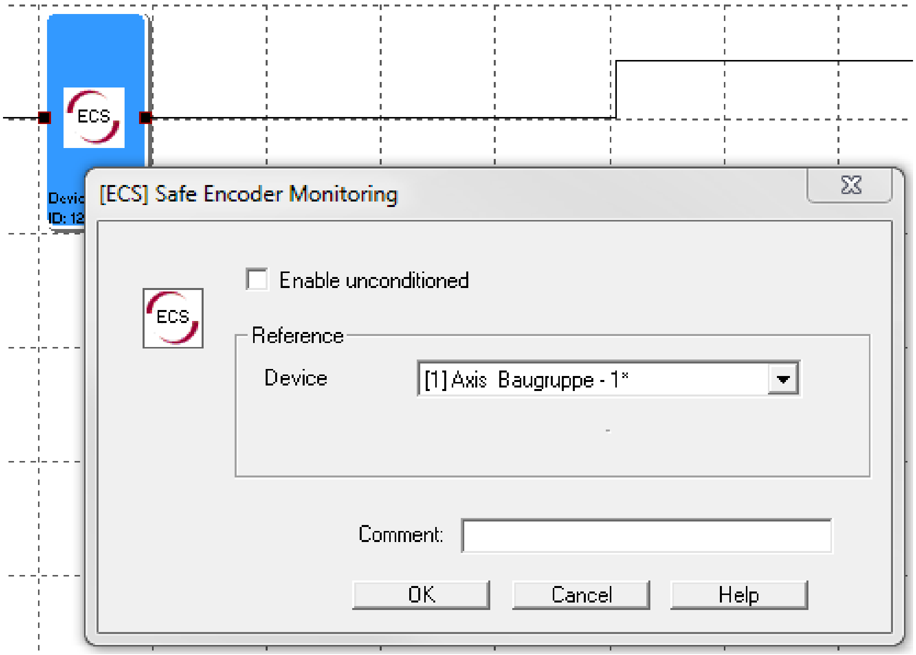


Figure 122: ECS block with input connector

Suppressed alarms

System A	System B	Diagnostic function
3309	3310	Diagnostics, speed check on the maximum speed (1st axis)
3329	3330	Diagnostics, speed check on the maximum speed (2nd axis)
3301	3302	Speed check (comparison) on the two sensors (1st axis)
3321	3322	Speed check (comparison) on the two sensors (2nd axis)
3303	3304	Position check (comparison) on the two sensors (1st axis)
3322	3323	Position check (comparison) on the two sensors (2nd axis)
3307	3308	Check on the section length for valid range (1st axis)
3327	3328	Check on the section length for valid range (2nd axis)

19.11 EOS (Encoder Offset Supervisor)

EOS Set the encoder position to a configured position

Quantity: 6

Access ID: Identification of the function element

Axis assignment: 1 function per axis and absolute value encoder

Function: Calculation of an offset for position sensors based on a set position configured and the actual sensor position. On the activation of the EOS function, the actual position is adjusted to a preset value configured by re-calculating and setting the offset. The offset is saved in non-volatile memory.

Function description:

The function is activated with a rising edge on the input for the function. The EOS function can only be used if the position processing has been activated and an absolute encoder has been configured on the sensor channel selected (SSI encoder).

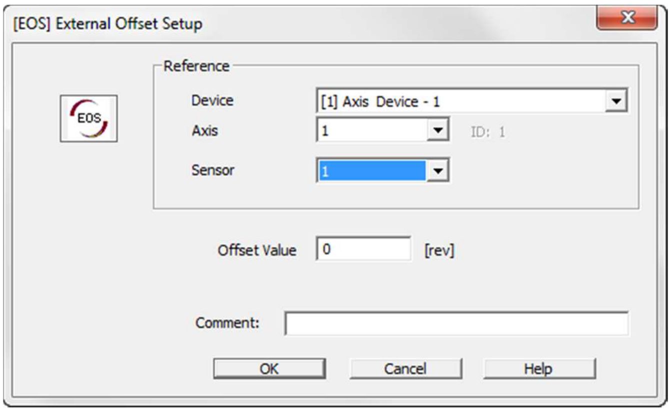


Figure 123: EOS External Offset Setup

Reference

Device

Select here the servo drive to which the EOS function is to relate.

Axis

Select here the axis to which the EOS function is to relate.

Sensor

Selection of the encoder, where 1 = encoder A and 2 = encoder B.

Offset Value

Preset value (setpoint) for the required sensor.

NOTE: It is only possible to use a maximum of one EOS function for an absolute value encoder.The activation of the EOS function during operation is to be excluded.

The function is used for maintenance and service purposes. This usage is to be ensured by selecting suitable means to trigger the function. Suitable means includes, e.g., a key switch that can only be accessed by qualified maintenance and service personnel. It is to be ensured by organisational measures that the physical position of the axis matches the set position. The offset calculated is saved in the device in non-volatile memory. To be able to run the EOS function correctly, the ECS function must be activated while the EOS function is activated.

19.12ESM (Encoder Standstill Monitoring)

Monitoring of the standstill time

Number: 6

Access ID: Identification of the function element

Axis assignment: Maximum 1 function per axis

Function: Monitoring of an adjustable standstill time

Function description:

Errors that are not detected at standstill can be detected by moving the axis. To achieve a high level of diagnostics coverage (DC), it is therefore necessary to move the axis within a period of time. The monitoring function block ESM monitors whether the assigned axis has been moved within an adjustable period.

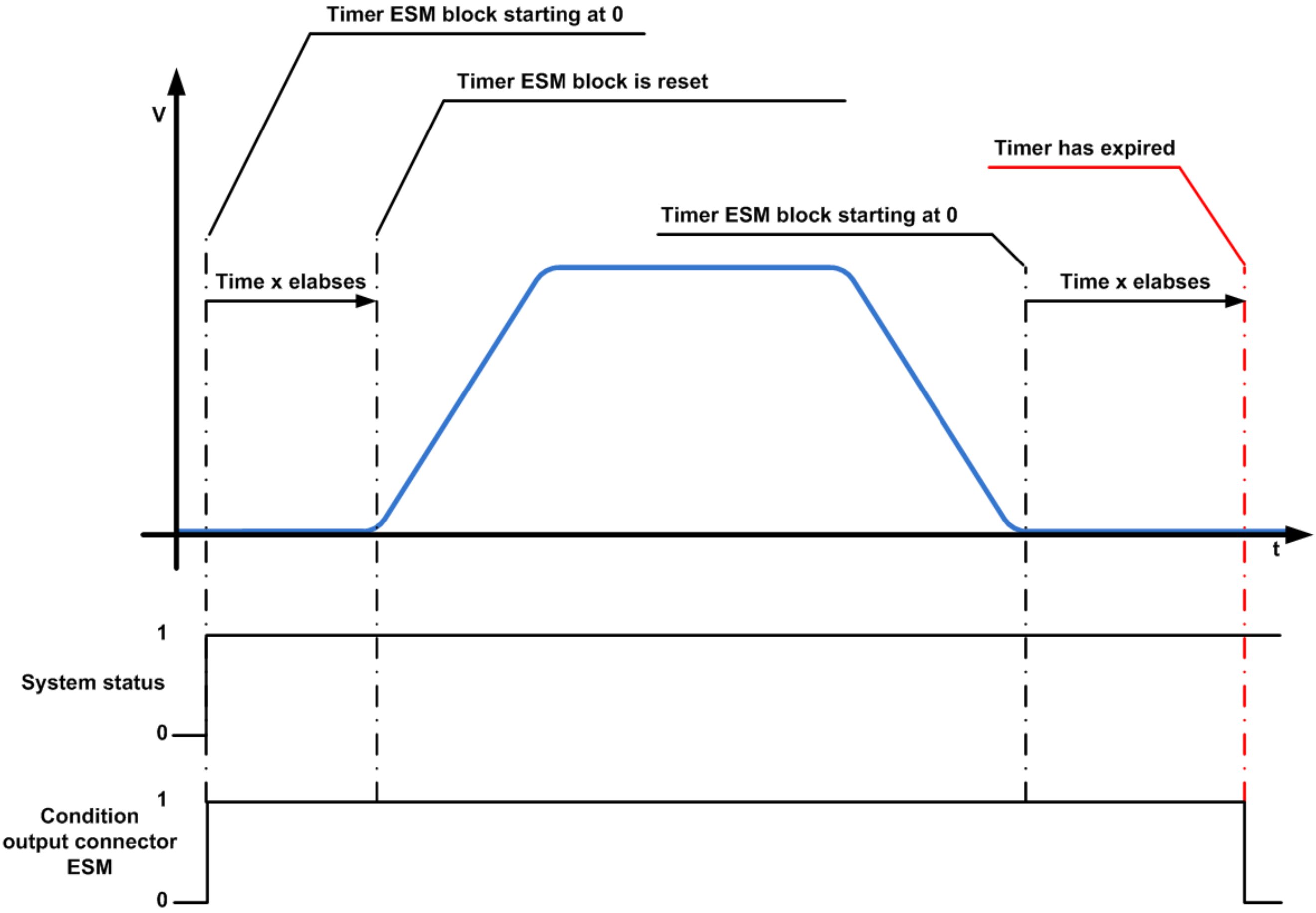


Figure 124: Diagram ESM

Activation example

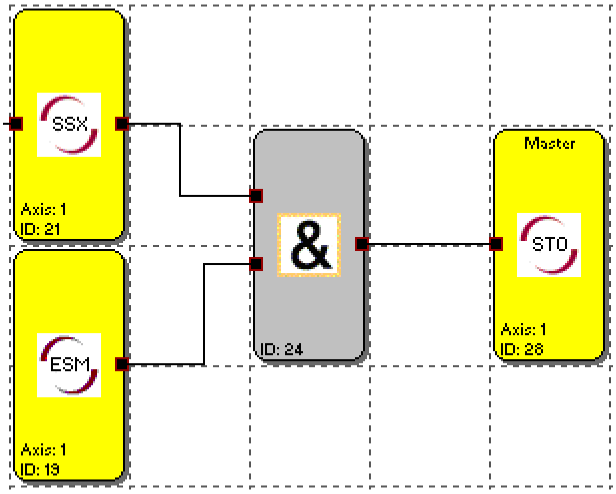


Figure 125: Activation example ESM

NOTE: If an encoder system is configured for an axis and an ESM monitoring function block is <u>not</u> used for this axis, the system automatically switches to the safe state after 8h and an error is generated.

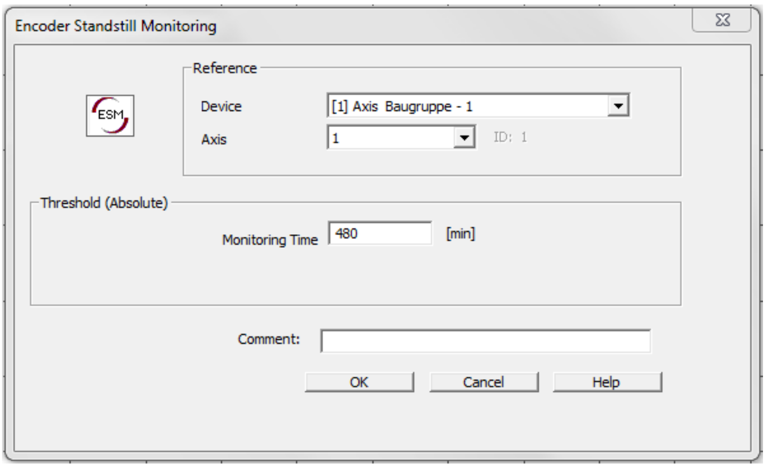


Figure 126: ESM Encoder Standstill Monitoring

Monitoring Time

Enter here the time for which the axis is allowed to be at standstill. The time is entered in minutes where 0 min is the minimum that can be entered and 1440 min (24 h) the maximum.

CAUTION: If "0 min" is entered, the monitoring is deactivated! This setting will result in the loss of the safet function!

19.13Fast Channel

Fast channel refers to the property of the devices to react more quickly than is possible on processing the safety programs in the normal cycle. For this purpose the fast channel is generated from a selected input function block and/or a monitoring function block.

The following can be selected for the generation of the fast channel:

- Function blocks that can be connected to the safe digital inputs ISSD00 to ISSD03, except the "start/reset switch" and the "functional input"
- Monitoring function blocks SLS and SOS

The fast channel generated acts on selected output function blocks and/or STO. The following can be selected for the action of the fast channel:

- Function blocks that can be connected to the safe digital outputs OSSD00 to OSSD03, except as pulse outputs, or functional outputs on function blocks configured.
- Brake output

To use the fast channel, select between "External Fast Channel (All Other Devices)" and "Internal Fast Channel (Only Selected Device)" on the related function blocks. Simultaneous selection of the external and internal fast channel is also possible.

Figure 127: Fast channel selection

NOTE: A triggered fast channel must be acknowledged, independent of whether the internal or external fast channel has been triggered. An acknowledgement is only effective if the triggering event has been reset.

External Fast Channel

If an external fast channel is generated, this channel acts on all function blocks on which the external fast channel is selected. These can be function blocks for the device in which the external fast channel was generated and function blocks for all devices in the axis group in which a fast channel has been selected.

NOTE: The external fast channel can only be used if there are at least two servo drives (master and slave) in the group

Internal Fast Channel

If an internal fast channel is generated, it acts only on the function blocks for the device in which the fast channel has been generated or is selected.

19.14 Acknowledging alarms and monitoring functions

The MSDFS offers various functions for acknowledging a monitoring function, or alarms. These can be realised both by safe and non-safe inputs. As a rule the acknowledgement is to be considered an independent safety function, see chapter **"14.8 Start / reset switch"**. A non-safe input is only allowed to be used for acknowledgement if the risk assessment for the application shows that a hazardous state cannot be caused and the acknowledgement is therefore not to be evaluated as a safety function.

The start behaviour of protective devices, for instance after a device reset, is an independent safety function and is covered in chapter "14.8 Start / reset switch".

On the topic of the acknowledgement of monitoring functions and alarms, the following descriptions apply.

In principle a differentiation can be made between the following MSDFS shutdowns:

Shutdown types and their acknowledgement		
Shutdown type	**Description**	**Possible methods of acknowledging**
Errors in the servo drive	Reaction to errors in non-safe part of the servo drive, for instance, overcurrent in the power stage or overtemperature on the motor.	▪ Non-safe digital input on the affected device (master or slave) ▪ Safe digital input with corresponding function on the affected device (master or slave) ▪ DRIVEADMINISTRATOR 5on the affected device (master or slave) ▪ Field bus system on the affected device (master or slave) ▪ Mains reset on the affected device (master or slave) **NOTE:** The reaction of the servo drive to errors in the non-safe part can be configured.

Alarms in the safe part (SMC)	Reaction to alarm in the safe part of the servo drive (SMC), for instance wrong pulse (signature) on a safe digital input. **NOTE:** Alarms can be reset by an acknowledgement on the master axis.	▪ Start, reset switch (via a safe digital input) on the master ▪ Non-safe digital input on the master ▪ DRIVEADMINISTRATOR 5 on the master ▪ Field bus system on the master ▪ Mains reset on the master **NOTE:** Only after prior risk assessment!
Errors in the safe part (SMC)	Reaction to errors in safe part of the servo drive (SMC), for instance, errors in the supply of power to the encoder. **NOTE:** Errors can only be reset by an acknowledgement on the triggering axis. If the triggering axis is not the master axis, it must also be acknowledged.	▪ Start, reset switch (via a safe digital input) on the master ▪ Non-safe digital input on the affected device (master or slave) ▪ DRIVEADMINISTRATOR 5 on the affected device ▪ Field bus system on the affected device ▪ Mains reset on the affected device **NOTE:** Only after prior risk assessment!
Shutdown of a monitoring function	Reaction to infringement of the monitoring limits set in the monitoring block. During this process the output connector on the monitoring block changes from a logical "1" to a logical "0". **NOTE:** Can only be acknowledged on the master.	▪ Start, reset switch (via a safe digital input) ▪ Non-safe digital input on the master ▪ DRIVEADMINISTRATOR 5 ▪ Field bus system **NOTE:**_Only after prior risk assessment!
Error in the safe part with the message "ER." at the start	Internal error in safe part of the servo drive, for instance, errors in the process image or internal plausibility checks.	▪ Mains reset

As listed in the "Table: "Shutdown types and their acknowledgement" the error can be acknowledged in the following ways:

Start, reset switch

Designation:	Acknowledgement via start, reset switch
Location of connection:	Any safe digital input from ISSD00 to ISSD03 on X38
Parameter configuration:	See chapter "14.8 Start / reset switch".
Signal evaluation:	Rising edge
Acknowledgement of:	• Alarms and errors in the safe part • Shutdown of a monitoring function

NOTE: Alarms and errors in the safe part as well as shutdowns of a monitoring function can only be acknowledged from the master.

As the safe part also sends the error to the non-safe part of the servo drive, an alarm message must also be acknowledged in the servo drive part (see Acknowledgement via digital input)

The alarm reset input can be operated with a continuous voltage of 24 V (without pulses) and is edge controlled.

Safe digital input with corresponding function

Designation:	Acknowledgement via safe digital input
Location of connection:	Any safe digital input from ISSD00 to ISSD03 on X38 on the servo drive.
Parameter configuration:	Selected digital input must be set via the DRIVEADMINISTRATOR to "RSERR(13) = Reset alarm".
Signal evaluation:	Rising edge
Acknowledgement of:	• Errors in the servo drive • Alarms and errors in the safe part • Shutdown of a monitoring function

NOTE: Alarms and errors in the safe part as well as shutdowns of a monitoring function can only be acknowledged from the master.

The alarm reset input can be operated with a continuous voltage of 24 V (without test pulse) and is edge controlled.

Along with their safe function in the safe part of the servo drive, safe digital inputs can be assigned a functional function at the same time, for example "Reset alarm" or "Quick stop".

Non-safe digital input

Designation:	Acknowledgement via digital input
Location of connection:	Any digital input from ISD00 to ISD05 on X38 on the servo drive.
Parameter configuration:	Selected digital input must be set via the DRIVEADMINISTRATOR to "RSERR(13) = Reset alarm".
Signal evaluation:	Rising edge
Acknowledgement of:	▪ Errors in the servo drive ▪ Alarms and errors in the safe part ▪ Shutdown of a monitoring function
NOTE: Alarms and errors in the safe part as well as shutdowns of a monitoring function can only be acknowledged from the master. The alarm reset input can be operated with a continuous voltage of 24 V (without pulses) and is edge controlled.	

The other possible ways of acknowledging are, apart from the location from which the acknowledgement is made, identical. Other possible ways are:

- DRIVEADMINISTRATOR 5
- Field bus system
- Mains reset

19.14.1 Acknowledgement behaviour of monitoring functions

Not all monitoring functions require an acknowledgement after triggering. There are also different requirements for acknowledgement. The following table provides an overview of the differences:

Monitoring function	Acknowledgement behaviour	Comment
SEL (Safely Emergency Limit)	Acknowledgement possible in the activated and deactivated state.	Only in the activated state if axis again within the configured limit.
SLP (Safely Limited Position)	Acknowledgement possible in the activated and deactivated state.	Only in the activated state if axis again within the configured limit.
SCA (Safe Cam)	No acknowledgement necessary.	Is automatically reset in the good state (position or speed within the configured limits).
SSx (SS1 or SS2)	Acknowledgement only possible in the deactivated state.	If the ramp monitoring AND the SOS monitoring have triggered for SS2 monitoring, acknowledgement must be undertaken twice. The acknowledgement behaviour of the SOS monitoring is described in the row SOS.
SLI (Safely Limited Increment)	Acknowledgement only possible in the deactivated state.	
SDI (Safe Direction)	Speed monitoring:Acknowledgement possible in the activated and deactivated state. Position monitoring:Acknowledgement only possible in the deactivated state.	
SLS (Safely Limited Speed)	Acknowledgement possible in the activated and deactivated state.	
SOS (Safe Operating Stop)	Speed monitoring:Acknowledgement possible in the activated and deactivated state. Position monitoring:Acknowledgement only possible in the deactivated state.	Position monitoring:Acknowledgement also possible in the activated state if axis again within the configured limit.
STO (Safe Torque Off)	No acknowledgement necessary.	
ECS (Encoder Supervisor)	Acknowledgement possible in the activated and deactivated state.	In the activated state the ECS block must be acknowledged 2x, if movement was not previously in a permissible range!
ESM (Encoder Standstill Monitoring)	No acknowledgement necessary.	If an ESM monitoring function block is not used, the system automatically switches to the safe state after 8h. Acknowledgement can then only be undertaken with a mains reset.
EMU (Emergency Unit)	Always active (no input connector), must be acknowledged after triggering.	If the EMU monitoring is triggered on the master, the block only changes to "0" and the related output is rendered passive. If, on the other hand, the EMU monitoring on a slave axis triggers, an alarm is generated and the entire system switches to the safe state.
Fast Channel	Acknowledgement possible in the activated and deactivated state.	
Activated state = input connector for the monitoring function at logical "1" *Deactivated state = input connector for the monitoring function at logical "0"*		

Appendix 1 Encoder combinations

ID	Encoder A		Encoder B		Possible safety functions	Fault exclusion
	Type	Interface	Type	Interface		
-	Not Connected	-	Not Connected	-	-	-
57	Proxy Switch 2 Counter (2ZP) / HTL	X4	Resolver	X6	SCA, SS1, SS2, SLI, SDI, SLS, SOS	-
16	Resolver	X6	Not Connected	-	SCA, SS1, SS2, SLI, SDI, SLS, SOS	Fault exclusion mech. shaft fracture, shape-based or force-based encoder shaft connection required.
2	Sinus Cosine	X7	Not Connected	-	SCA, SS1, SS2, SLI, SDI, SLS, SOS	Fault exclusion mech. shaft fracture, shape-based or force-based encoder shaft connection required.
6	Sinus Cosine	X7	Proxy Switch 2 Counter (2ZP) / HTL	X4	SCA, SS1, SS2, SLI, SDI, SLS, SOS	
52	Sinus Cosine	X7	TTL	X8 1)		
68	Sinus Cosine	X7	Sinus Cosine	X8 1)		
8	SSI	X7	Proxy Switch 2 Counter (2ZP) / HTL	X4		

63	SSI	X7	Sinus Cosine	X7		
67	SSI	X7	SSI	X8 [1)]		
69	SSI	X7	Sinus Cosine	X8 [1)]		
70	SSI	X7	TTL	X8 [1)]		
1	TTL	X7	Not Connected	-		
5	TTL	X7	Proxy Switch 2 Counter (2ZP) / HTL	X4	SCA, SS1, SS2, SLI, SDI, SLS, SOS	
51	TTL	X7	TTL	X8 [1)]		
58	TTL	X7	Resolver	X6	SCA, SS1, SS2, SLI, SDI, SLS, SOS	
	NOTE: You will find the safety level that can be achieved is the MSD model description "Functional safety" (chapter "3.8.1 Safe encoder evaluation"). [1)] Option 2 requires the Safety Technology Option second safe axis monitor (Sin/Cos) or second safe axis monitor (SSI), depending on the encoder type.					

Appendix 2 Functions of the functional outputs

Setting	Designation	Function
(0)	OFF	Input shut down
(1)	ERR	Collective error message
(2)	BRAKE	Output becomes active corresponding to the holding brake function
(3)	ACTIV	Power stage and control active
(4)	S_RDY	Output becomes active when the device is initialised after power on
(5)	C_RDY	Output becomes active if the device is "ready to start" due to setting the "ENPO" signal and there is no error message present. Device ready for operation - ReadyToSwitchOn flag set in DriveCom status word (in the states 3, 4, 5, 6, 7)
(6)	REF	The stipulated setpoint has been reached (dependent on type of control)
(7)	HOMATD	Homing complete
(8)	E_FLW	Tracking error
(9)	ROT_R	Motor is in the standstill window for clockwise
(10)	ROT_L	Motor is in the standstill window for counterclockwise
(11)	ROT_0	Motor is in the standstill window, dependent on the actual value
(12)	STOP	The drive is in the "quick stop" state
(13)	HALT	Indication "System ist im Zustand HALT" (System is in the HALT state), activated via CiA402 profile, input or PROFIBUS IntermediateStop, SERCOS. Reaction is as per HALT option code (**P 2221 MPRO_402_HaltOC**).
(14)	LIMIT	Output is set if a setpoint has reached its limit.
(15)	T_GT_Nx	T is greater than Nx with Nx = value in **P 0741 MON_Torque/forceThresh**

(16)	N_GT_Nx	N is greater than the value in **P 0740 MON_SpeedThresh**
(17)	P_LIM_ACTIV	Position setpoint limited (e.g. with software limit switches configured)
(18)	N_LIM_ACTIV	Speed setpoint limit active
(19)	T_LIM_ACTIV	Torque limit active
(20)	not defined	Not defined
(21)	ENMO	Motor contactor output (motor wiring via contactor)
(22)	MSD PLC	MSD PLC sets the output
(23)	WARN	Warning, collective message
(24)	WUV	Warning, undervoltage in the DC link
(25)	WOV	Warning, overvoltage in the DC link
(26)	WIIT	Warning, I^2xt power stage protection reached
(27)	WOTM	Warning, motor temperature
(28)	WOTI	Warning, inverter heat sink temperature
(29)	WOTD	Warning, temperature inside the inverter
(30)	WLIS	Warning, current threshold reached
(31)	WLS	Warning, speed threshold reached
(32)	WIT	Warning, I^2xt motor protection threshold
(33)	WLTQ	Warning, torque limit reached
(34)	TBACT	Table positioning in the "AUTO" state and activated
(35)	TAB0	Significance 2^0
(36)	TAB1	Significance 2^1

(37)	TAB2	Significance 2^2
(38)	TAB3	Significance 2^3
(39)	COM_1MS	Set output via field bus in 1 ms cycle
(40)	COM_NC	Set output via field bus in the NC cycle
(41)-(54)	not defined	Not used

TAKE A CLOSER LOOK

Moog solutions are only a click away. Visit our worldwide Web site for more information and the Moog facility nearest to you.

Moog GmbH
Hanns-Klemm-Straße 28
D-71034 Böblingen
Phone +49 7031 622 0
Telefax +49 7031 622 100

www.moog.com/industrial
drives-support@moog.com

Technical alterations reserved.

The content of our documentation have been compiled with the greatest care and in compliance with our present status of information.

Nevertheless, we would like to point that this document cannot always be updated in parallel to the technical further development of our products.

Information and specifications may be changed at any time. For information on the latest version please refer to drives-support@moog.com.

ID no.: CB78095-001, 02/2018, Rev. 1.3