rexroth
A Bosch Company

Smart Function Kit - Software

R320103208, 2026-03, en

The information in this manual is for product description purposes only. Any information on how to use the product is only an example and a recommendation. Catalog information is not binding. The information in this manual does not release the user from exercising their own judgment and conducting their own tests. Our products are subject to natural wear and aging.

The cover depicts a sample configuration. The actual product may vary.

The original manual is in German.

Table of Contents

1 About this documentation

1.1 Identification of text types

This chapter lists all the text types found in this document and their respective identification.

1.1.1 Body text

All explanations are presented in continuous text. Individual items and topics that are not continuous are represented in the form of lists.

1.1.2 List

- List, first level
  - List, second level
    - List, third level

1.1.3 Instruction for action

→ Step of an instruction for action without chronological order or of individual instructions for action

1. Step of an instruction for action with defined order
   ✅ Intermediate result of an instruction for action or unexpected action of the device
2. Step of an instruction for action with defined order End

✅ result of an instruction for action

1.1.4 Reference

◊ Cross-reference or reference to documentation

1.1.5 Notice

> **Notice**
> Important information or notice regarding handling

1.1.6 Additional information

💡 Additional information about the previous action step

1.1.7 Console input

test heading

```
helloWorld();
```

1.1.8 Code Snippet/code Files

test heading

```
helloWorld();
```

1.1.9 Inline Code

All `InlineCodes();` within a continuous text or list are highlighted.

1.2 Abbreviations

The following abbreviations are used in this documentation:

Abbreviation	Meaning
API	Application Programming Interface
EMC	**E**lectro **M**echanical **C**ylinder
HCS01	Drive controller
HMI	**H**uman **M**achine **I**nterface (graphical user interface)
HTTPS	**H**ypertext **T**ransfer **P**rotocol **S**ecure
ID	**Id**entification number
I/O	**I**nput/**o**utput
IP	**I**nternet **P**rotocol
JSON	**J**ava**S**cript **O**bject **N**otation
NOK	Not okay rating
OK	Okay rating
OM	Operating mode
OPC UA	**O**pen **P**latform **C**ommunications **U**nified **A**rchitecture
PC	**P**ersonal **C**omputer
PR21	Industrial PC

Abbreviation	Meaning
REST	**Re**presentational **S**tate **T**ransfer
SFK	**S**mart **F**unction **K**it
SMC	**S**equential **M**otion **C**ontrol
SMO	**S**afe **Mo**tion
PLC	**P**rogrammable **L**ogic **C**ontroller
STO	**S**afe **T**orque **O**ff
SYNC	**Sync**hronization
XF5, XF6	Network interfaces at the industrial PC
X26	Network interface at the drive controller
X31	I/O connector strip at the drive controller

Table 1 Abbreviations

2 Safety

2.1 Intended use

The Smart Function Kit is used to be installed in or joined to machinery, incomplete machinery or equipment.

Use the product only as intended.

Intended use also includes compliance with the operating, maintenance, and servicing requirements prescribed by the manufacturer.

Any use beyond or other than that is considered improper. The manufacturer shall not be liable for damage resulting thereof. Any associated risk shall be borne by the operator.

2.2 Safety instructions

This documentation contains safety instructions regarding any actions that involve a risk of personal injury or damage to property. The safety precautions described must be adhered to.

The safety instructions are structured as follows:

⚠ SIGNAL WORD	
	**Type and source of hazard!** Consequences if ignored. → Hazard prevention measure.

- **Warning sign:** draws attention to the hazard
- **Signal word:** Indicates the severity of the hazard
- **Type and source of hazard:** indicates the type or source of the hazard
- **Consequences:** describes the consequences that may occur if precautions to avoid the hazard are not taken
- **Hazard prevention measure:** indicates how to avoid the hazard

The safety instructions cover the following hazard classes. The hazard class describes the risks involved if the safety instruction is not complied with.

Warning sign, signal word	Meaning
⚠ DANGER	Indicates a hazardous situation which will result in death or serious injury if not avoided.
⚠ WARNING	Indicates a hazardous situation which may result in death or serious injury if not avoided.
⚠ CAUTION	Indicates a hazardous situation which may result in minor or moderate injury if not avoided.
NOTICE	Property damage: The product or surroundings may be damaged.

Table 2 Safety instructions hazard classes acc. to ANSI Z535.6

2.3 Residual risks

⚠ DANGER

Danger of death due to live components.

Touching live components will result in death or severe physical injury.

→ Always disconnect the machine from the mains during cleaning, maintenance and repair work.

→ Any work at electrical components/assemblies, e.g. motor, hydraulic system, and all electrical lines, may only be performed by qualified professionals according to the rules of electrical engineering.

→ Immediately remedy any identified defects at electrical parts.

→ Only work according to the circuit diagram.

→ Do not use defective parts.

⚠ DANGER

Uncontrolled motion due to software malfunction or incorrect I/O wiring

A software malfunction or incorrect I/Os wiring can cause an uncontrolled movement of the SFK. Serious injuries due to impacts, crushing, clamping or throwing away of components can result.

→ Do not reach into the working area of moving components.

→ If present, do not manipulate the protective enclosure.

→ Keep the protective door closed.

⚠ DANGER

Uncontrolled movement due to remote activation (HMI, fieldbus)

Remote activation can cause uncontrolled movement of the SFK. Serious injuries due to impacts, crushing, clamping or throwing away of components can result.

→ Do not reach into the working area of moving components.

→ If present, do not manipulate the protective enclosure.

→ Keep the protective door closed.

⚠ CAUTION

Indirect contact with live components

Indirect contact with live parts (e.g. in case of faulty insulation) can lead to injuries.

→ Check live parts for damage.

→ Repair the damage immediately.

3 Commissioning

The software-guided commissioning of the SFK takes place in several steps. The language, the password and hardware-related settings can be made.

After successful commissioning, the system is immediately ready for operation.

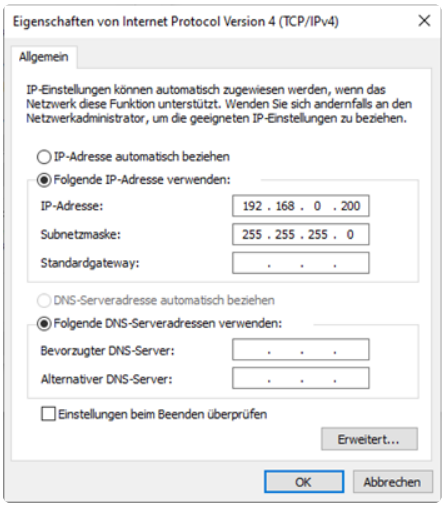
Notice
In order to be able to use the safety-specific functions of the HCS01, the safety technology must be put into operation (see chapter Commissioning with safety option S4).
Commissioning of the SFK must be performed before commissioning of the safety technology.

Notice
The SFK requires a force sensor for reliable force measurement and control (see Chapter Force sensor). Carry out mounting and connection before commissioning.

3.1 Initial commisioning

For initial commissioning, the system must be completely pre-wired and switched on.

After switching on, it takes approx. 90 seconds for the system to start up.

Screens/comments	Action steps
 Factory set network settings for the system: • IP address: 192.168.0.1 • Subnet mask: 255.255.255.0 • Gateway address: 0.0.0.0 Required network configuration of the commissioning PC: • IP address static • IPv4 address: 192.168.0.x with x > 1 (e.g. 192.168.0.200) • Subnet mask: 255.255.255.0	→ Ensure that your HMI control unit or a commissioning PC is connected with the network of the SFK.

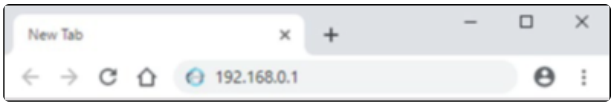
Screens/comments	Action steps
	→ Check the network settings of your HMI control unit.
	→ Open your web browser and enter the IP address of the SFK (default: 192.168.0.1). ✅ The web browser is connected to the SFK. The commissioning dialog window is displayed. **Notice** The web browser or the web browser security settings may generate a security warning during connection setup, which must be acknowledged. **Notice** If the SFK web interface is not displayed: → Check whether there is a network connection to PR21 (e.g. by means of PING test). → Check whether the CMOS-Industry PC Battery PR21 needs to be replaced.

Table 3 Initial commisioning

3.1.1 Login

Log in at the system:

1. Enter the default password *admin* or the new password, if already changed.
2. Confirm with Enter or click the *Login* button.
3. To change to the next commissioning step: Click the *Next*.

3.1.2 Changing language selection, user accounts and password

Language selection

The system can be operated in German, English and Chinese.

→ Select the desired language in the selection menu.

User account

The system provides the following user accounts: Administrator and operator account

- The administrator account is password protected and is used for:
  - system configuration
  - the creation of programs
  - the definition of the quality assessment
  - export of data
- The operator account is not password-protected and is the same as the logged-out condition. The operator account is used during operation and only provides limited access to the system.

Changing a password

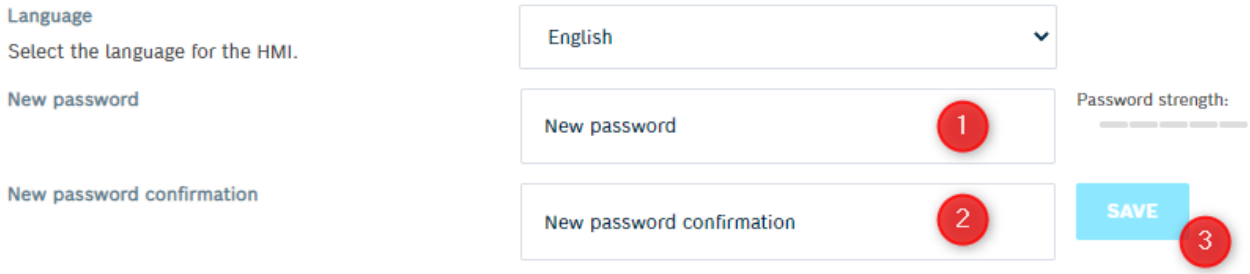


Figure 1 Changing a password

To change the password for the administrator account during commissioning:

1. Go to System Settings.
2. Enter a new password into the field (1).
3. Confirm the new password in the field below (2).
4. To save the new password: Click *SAVE* (3).

- ✅ Changing the password overwrites the default password.
- ✅ Login is only possible with the new password.

> **Notice**
> If no new password is assigned, the default password remains *admin* and can be used for login with the administrator account. The password can be changed at any time.

NOTICE

Unauthorized access to system and data

The default password does not comply with common safety standards and does not offer sufficient protection of the system.

→ Change the default password.

→ Use a secure password.

NOTICE

Loss of system settings and system data

The administrator password cannot be reset. If the password gets lost, system access is not possible.

→ Note down the new password and keep it in a safe place.

3.1.3 System check

Notice

The SFK can be ordered with or without a force sensor.
If the SFK is ordered without a force sensor, a suitable sensor must be added.
◊ For further information, see chapter Customized force sensor.

During the system test, the system will self-diagnose and verify that all components are properly connected and functional.

Commissioning can only be continued if all required components could be recognized by the system check.

→ To change to the next commissioning step: Click the *Next*.

If a defective component is detected during the system test, the *Next* button is disabled.

Commissioning cannot be directly continued.

→ To continue commissioning: Check the following points:

- Proper connection of the HCS01 with the PR21.
- Both devices are switched on.
- The network settings on the HCS01.
- The software is properly installed on the PR21.

◊ For further information on the individual points, see chapter Troubleshooting and fault clearance.

3.1.4 Installation of the software

NOTICE

Impairments in system behavior

Shutting off the PR21 or drive controller during the update will cause impairments in the system behavior.

→ Do not switch off the PR21 or the drive controller during the update process.
Exception: The software requires you to do this.

→ Always carry out any software update in full.

→ If the update process was interrupted, carry out the update again.

→ To start the installation: Click the *Start* button.

The installation progress is indicated by a progress bar.

→ To change to the next commissioning step: After completion of the installation, click the *Next* button.

In case of any error during the software installation, carry out the following steps:

→ Click on *Repeat* as soon as an error message is displayed.

The last (faulty) installation step is carried out again.

If the process is repeated several times and the error persists:

→ If the error messages appear in the title bar (see the chapter Title bar), acknowledge them.

✓ If the error messages cannot be acknowledged or they reappear, the cause of the error is still present.

In this case, the cause of the error must be corrected first:

→ Check the system for correct connection.

→ Check additional add-on modules, such as the safety zone module, for correct wiring and configuration.

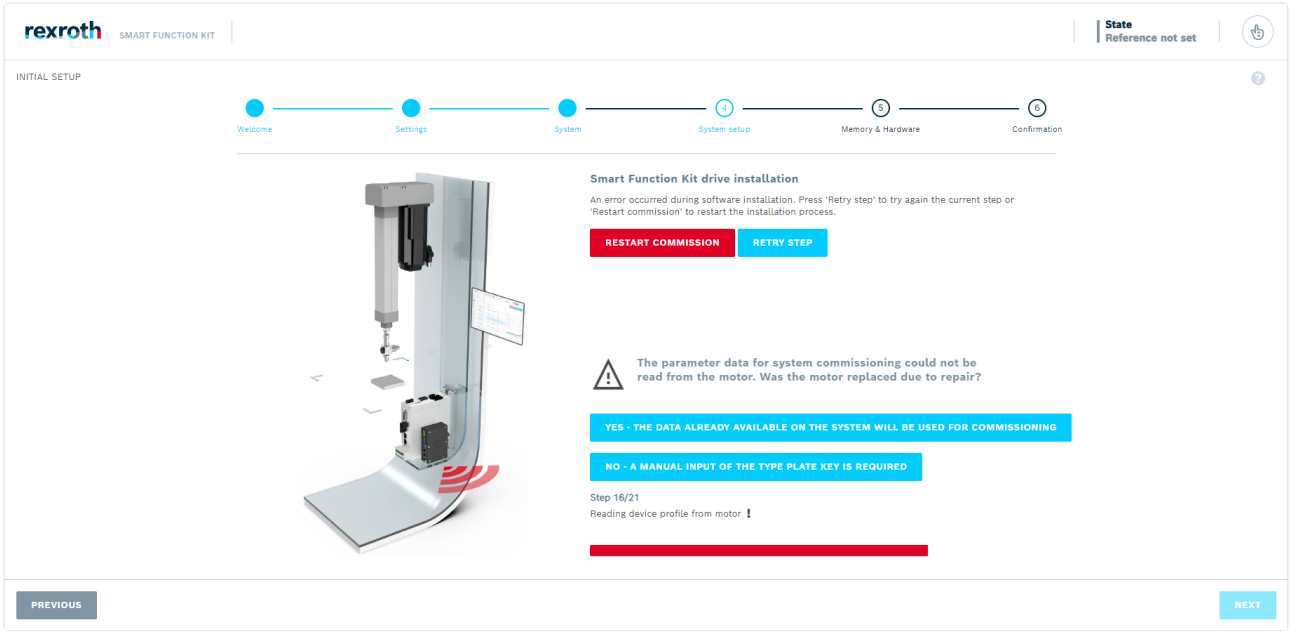
→ Click the *Restart* button.

✓ The entire installation is restarted.

◊ For more information, please see the chapter Troubleshooting and fault clearance.

In step 4 of commissioning, the axis parameters stored in the motor encoder memory are transferred to the drive controller.

If the parameters cannot be read from the motor encoder memory (e.g. because a replacement motor has been installed on which no parameters are stored) the following message appears:



→ Click *YES* if there has already been a commissioning before and the parameters available on the drive controller are still correct (no changes have been made to the hardware configuration).

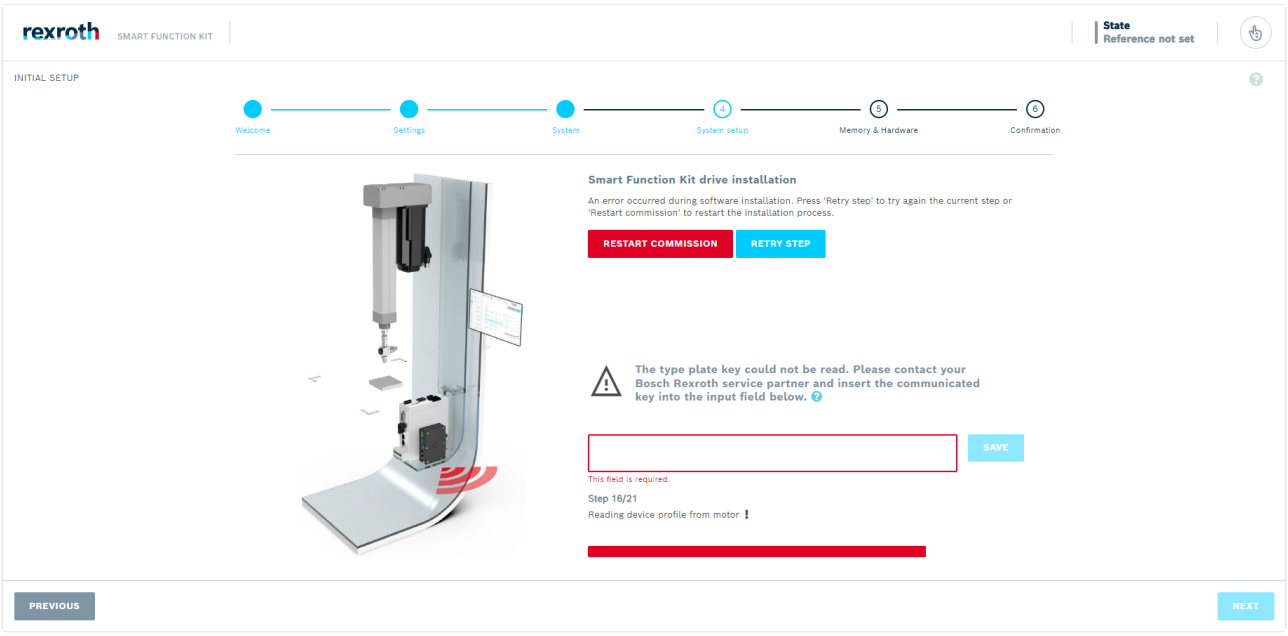
✓ The commissioning is continued with these parameters.

→ Click *NO* if there has not been any commissioning before or if changes have been made to the hardware configuration (e.g. different motor).

✓ An entry field will appear for the type code.

→ Contact your Bosch Rexroth service partner to obtain the appropriate type code and enter it in the input field.

✓ The commissioning is continued.



During commissioning, the drive controller changes operating mode several times. In step 16 of the system setup, a phase change from PM (parameter mode) to OM (operating mode) is performed. An incorrect parameter will result in an error in this step.

If this error occurs during software installation, follow these steps:

→ Click on *Repeat* as soon as an error message is displayed.

✅ The last (faulty) installation step is carried out again.

If the process is repeated several times and the error persists:

→ Restart the system.

→ Repeat the step.

If the error still occurs, perform the following steps:

→ Determine the error number using the information text.

→ Determine the corresponding remedial measures from chapter System errors.

→ Follow the corrective measures to solve the problem.

→ If necessary, restart the system.

→ If necessary, restart the installation.

✅ The commissioning is continued.

NOTICE

Commissioning may be interrupted in step 16 of system setup by the following errors:
- "Switching from PM to OM failed"
- "Switching from PM to OM timed out"
- "ClearError Command failed"

These errors can generate different error numbers.

3.1.5 Memory and hardware configuration

Memory settings

In the memory settings the circular buffer can be configured and activated.

◊ For more information about the memory settings, see chapter Settings > System configuration.

Hardware configuration

The hardware used in the system is automatically recognized by the Smart Function Kit.

NOTICE	
	**Damage to the system** Incorrect system parameters and changed default parameters may result in damage to the system. → Check the system parameters for correctness.

→ Check the displayed components and system parameters for correctness.

→ To change to the next commissioning step: Click the *NEXT*.

If the displayed parameters do not comply with the configuration as per the order, they can be changed manually.

→ To do so, click the SHOW EXTENDED CONFIGURATION button and then EDITING.

◊ More information about extended configuration can be found in chapter Settings > Hardware configuration.

3.1.6 Homing

→ To set the system zero point: In the last step of commissioning, you can optionally carry out a homing.

→ To carry out the homing in the system settings: Alternatively, finalize the commissioning.

⚠ **WARNING**	
	**Risk due to moving parts** Moving parts can cause severe lacerations, stabbing injuries and crushing. → Do not reach into the working area of moving components.
	**Risk due to moving axes** Moving axes can cause damage to the product and the system. → Enable faultless retraction of the cylinder.

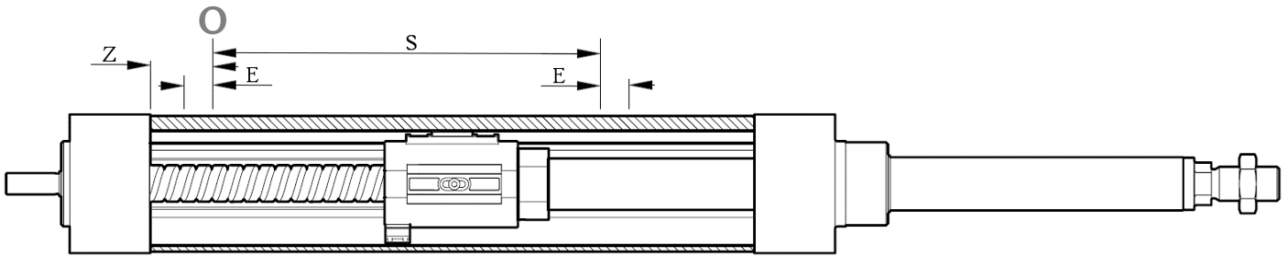


Figure 2 Position of software limit switches

- The mechanical end stop Z is at -3 mm from the zero point (0).
- The negative software limit switch is at -1 mm from the zero point (0).
- The maximum usable stroke range of the system is S and corresponds to the dimension S_{MAX} of the ordered configuration.
- The positive software limit switch is at S+1 mm from the zero point (0).

→ To finish commissioning directly without any homing: Click the *Finish* button.

✅ During homing procedure, the cylinder moves in up to the motor-side mechanical end stop.

✅ The position of the mechanical end stop is used to determine the system zero point and the respective software limit switches.

→ The homing can be repeated at any time in the system settings.

> **Notice**
> The homing may only be started in unloaded condition.
> → Move the SFK to a load-free state, if this is not the case already.
> → Use the jog operation in the "Relative movement" mode.

→ Acknowledge the system errors displayed in the title bar.

✅ Faultless retraction of the cylinder is enabled.

◊ For more information, please see the chapter Troubleshooting and fault clearance.

→ To start the homing: Click the *Homing* button.

3.2 Commissioning with safety option S4

For the commissioning of the safety technology in a HCS01 with safety option S4, you also need the IndraWorks DS software.

◊ It can be downloaded from the Bosch Rexroth AG homepage following this link: https://www.boschrexroth.com/media-details/fe4412ab-adf5-497c-ae43-55ff0a1ab127

◊ For further information on the supported safety functions, the connection of safety-relevant peripheral devices and the safety-relevant behavior of the software, see chapter Safety technology.

1. Open the IndraWorks DS software on your engineering device (e.g. laptop) and establish a connection to the HCS01.
   💡 For this purpose, you can either use port X24/X25 (standard network address: 192.168.0.x) or X22/X23 (standard network address: 192.168.2.x). Port X26 will still remain connected to PR21.
2. Deactivate the Safe Motion function in IndraWorks if it is activated.

> **Notice**
> As long as the Safe Motion function is deactivated, the connection between the HCS01 and the safety zone module HZS01 must be disconnected.

3. Make sure that the connection between the safety zone module HSZ01 and the HCS01 is disconnected .

> **Notice**
> If the safety zone module HSZ01 is connected to the HCS01 when the SFK is commissioned, the error message F7035 is displayed permanently. The commissioning cannot be carried out.

4. Release the intermediate circuit voltage at the HCS01.
5. Activate the functional commissioning.

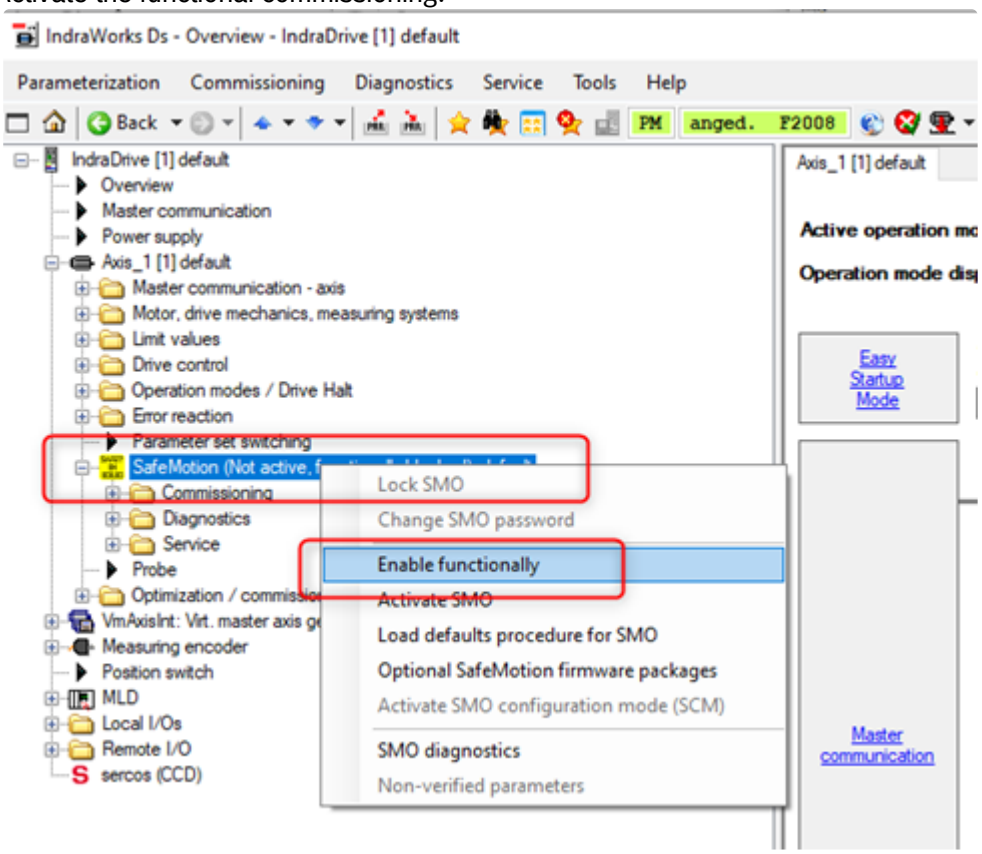


Figure 3 Activation of the functional commissioning (1)

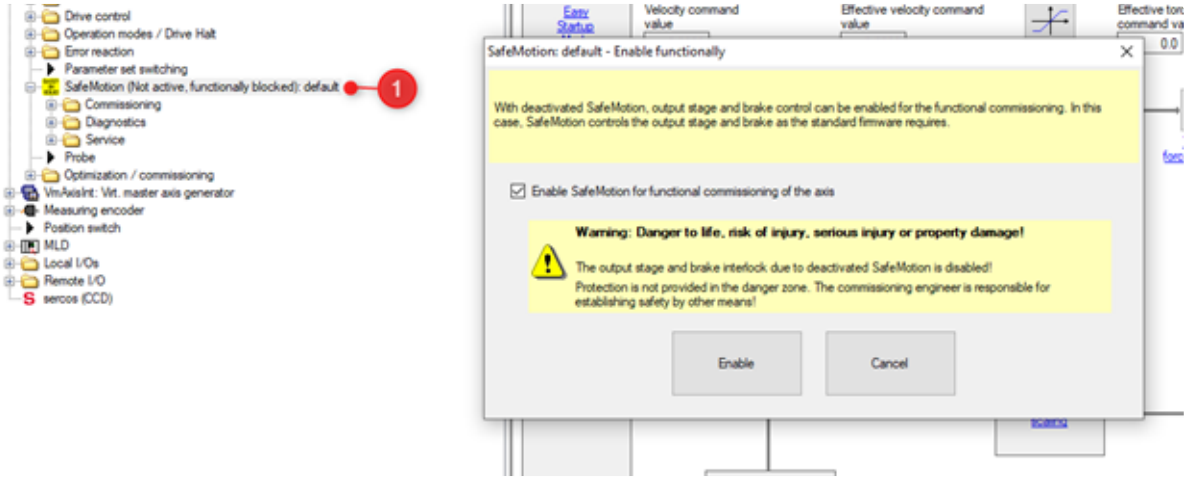


Figure 4 Activation of the functional commissioning (2)

6. Commission the SFK according to chapter Commissioning.
7. Perform homing after completion of the commissioning.

> **Notice**
> → To be able to approach absolute positions and start programs: Perform the homing.

8. In order to be able to use the drive-integrated oscilloscope for the machine acceptance: In the SFK software, activate the acceptance test mode. The setting necessary for this purpose can be found at *Configuration → General → Activate acceptance test mode*.

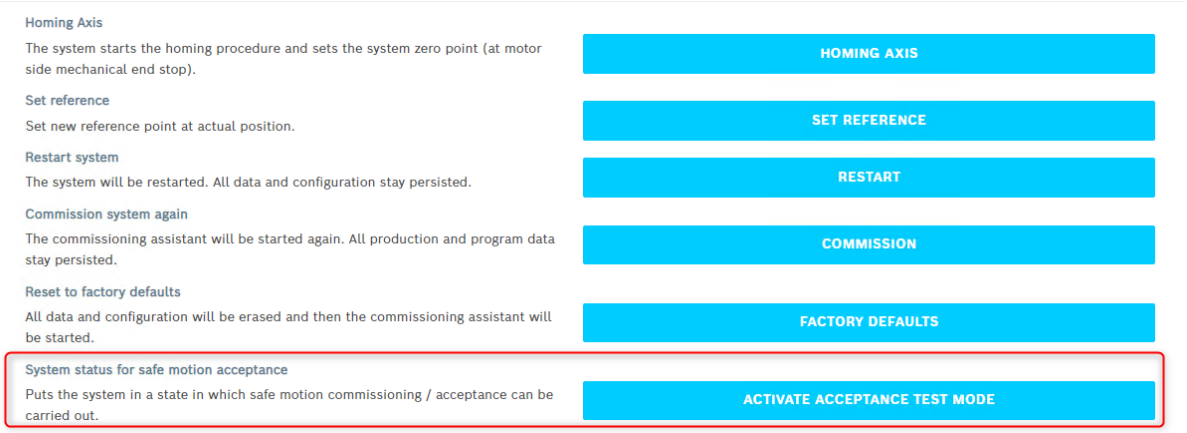


Figure 5 Activation of Acceptance Test Mode

9. Switch off the HCS01 and connect it to the safety zone module HSZ01 via the bus cable according to the installation instructions (R320103194).
10. Restart the system.
11. Start up the “Safe Motion” safety technology with the IndraWorks DS software. To do so, activate the SMO.

Notice
→ Note the safety functions listed in chapter Safety option S4 which are currently supported by the SFK. Selection of unsupported safety functions can lead to incorrect system behavior. Carry out commissioning of the safety technology in accordance with the instructions “IndraDrive Integrated Safety Technology Safe Motion (from MPx-18)” (R911338919) Chapter 9, see www.boschrexroth.com/medienverzeichnis.
→ Also observe the information on the SFK below.

Please note the following points when commissioning the safety technology:
Changes to the safety technology can only be made in parameter mode.
→ Before initial commissioning: Stop the PLC under *MLD* → *Configuration* → *STOP*.
→ Then switch to parameter mode.

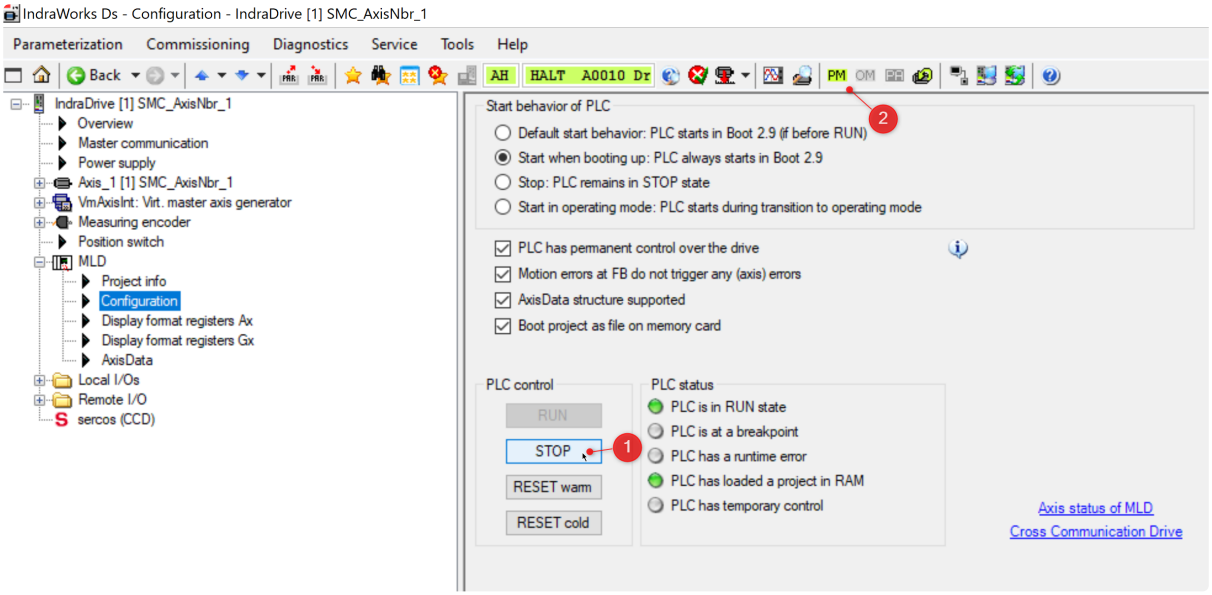


Figure 6 Stopping of PLC and switching to parameterization mode

In the “encoder settings” step, the encoder data from the standard firmware must be applied in the SMO.

Notice
Any deviating configuration is not permissible for the SFK.

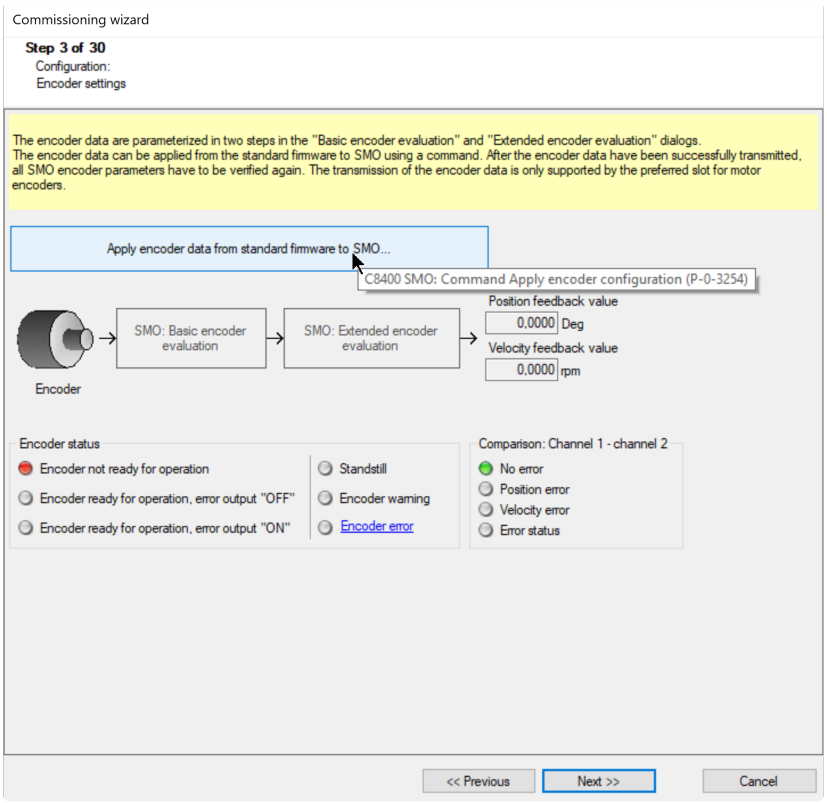


Figure 7 Applying encoder data to SMO

In the “scaling settings” step, the scaling data from the standard firmware must be applied in the SMO.

> **Notice**
> Any deviating configuration is not permissible for the SFK.

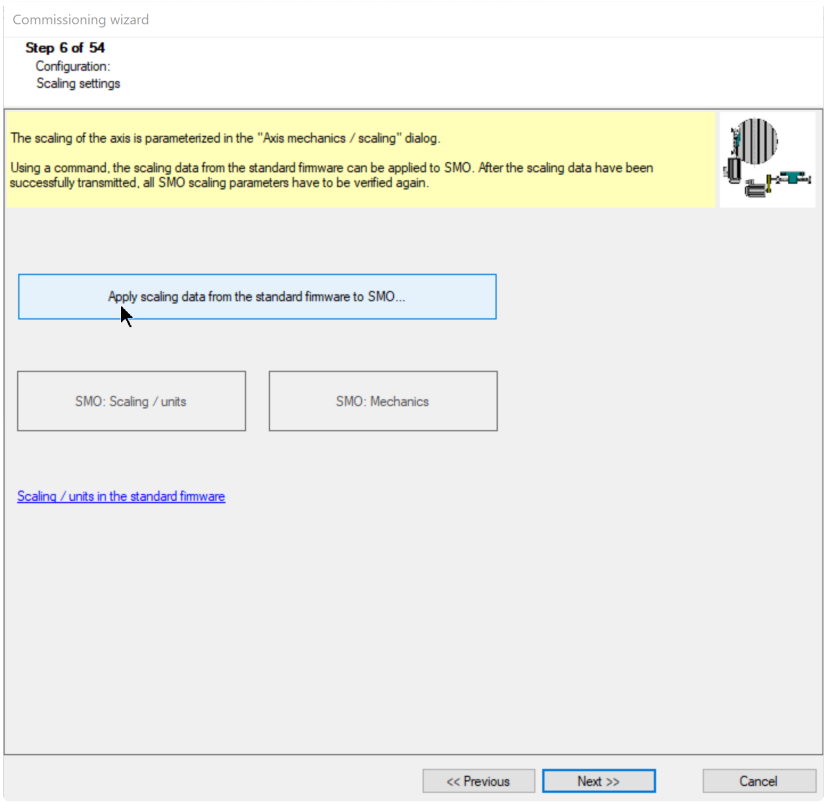


Figure 8 Applying scaling data to SMO

Notice
Safe referencing is not possible with the current SFK version.
The axis must be configured as zone master in the “Safety zone and door locking” step.
When using PROFIsafe or FSoE, "Axis acknowledges safety independently (no zone node)" can be selected.

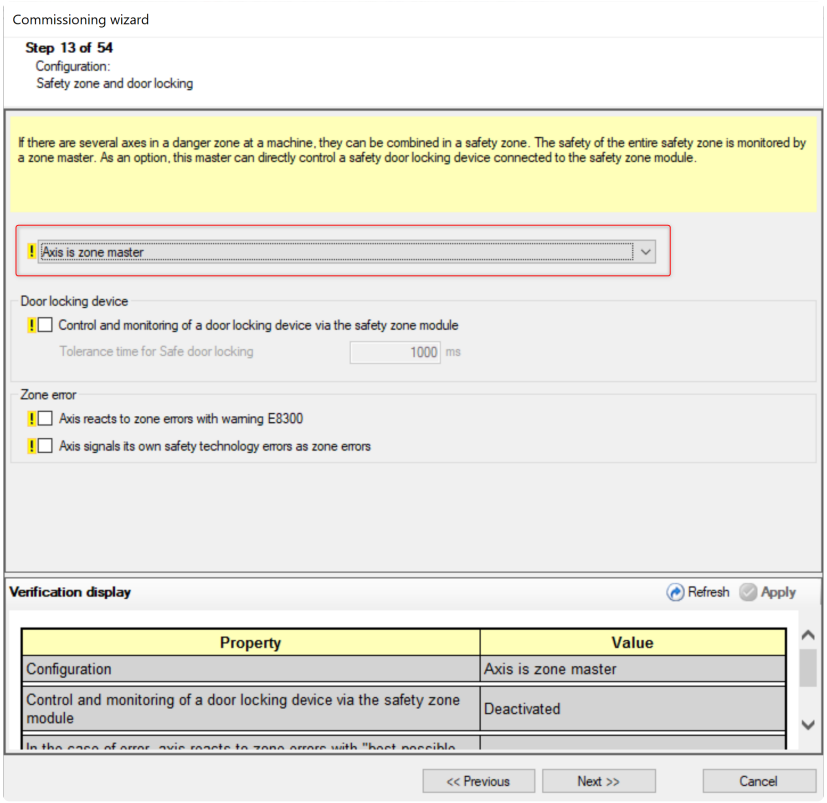


Figure 9 Configuration of the axis as zone master

In the “Change in operating state change” step, “NC-controlled” must be selected for transfer to standstill as an integrated control is used in the drive.

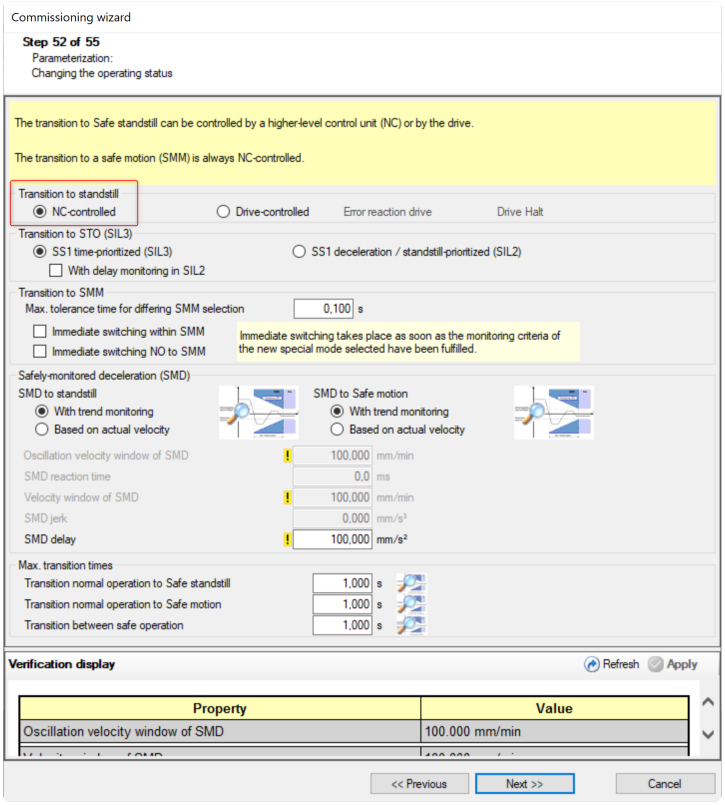


Figure 10 Transition to standstill

Commissioning of safety technology is completed after preparation of the SafeMotion report.

12. Carry out the axis evaluation. To do so, use the “Drive-internal command value input (command value box) and drive-internal enabling (*Easy Startup* mode)”.
    💡 Please observe that the PLC is still stopped.
13. After axis evaluation with the *Easy Startup* mode, the parameters must be reset to their state before the axis evaluation: To do so, click on *Yes*.

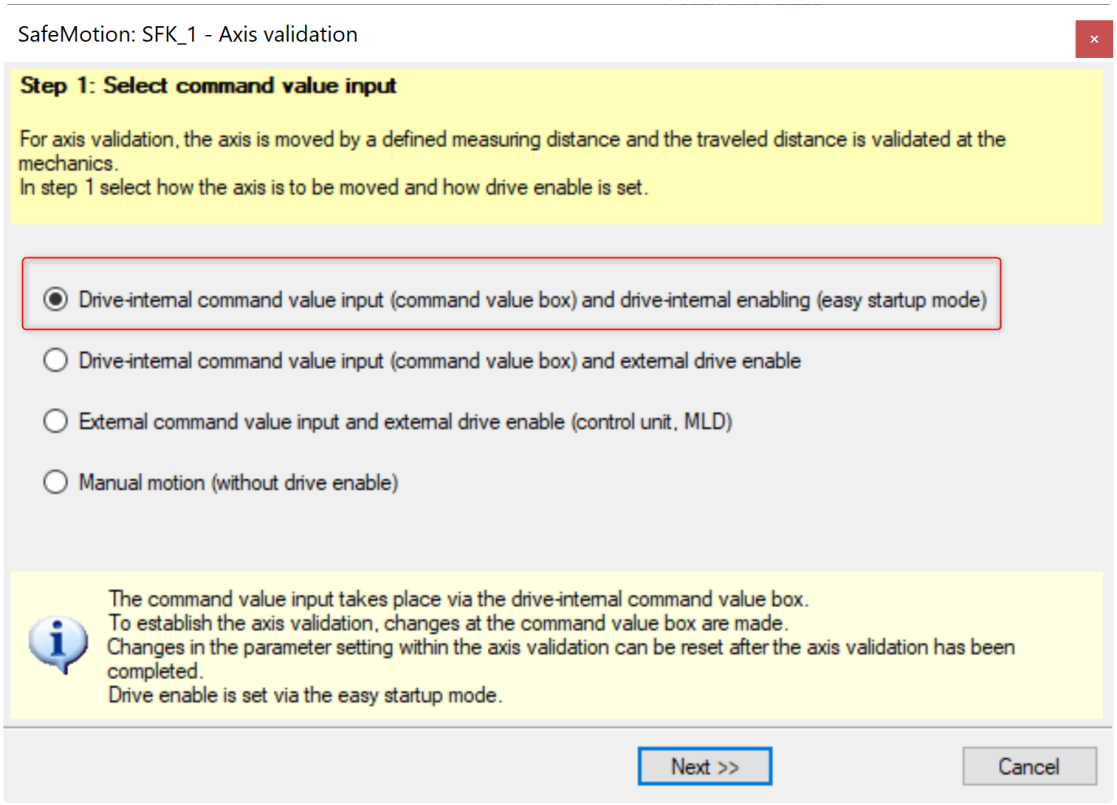


Figure 11 Axis evaluation (1)

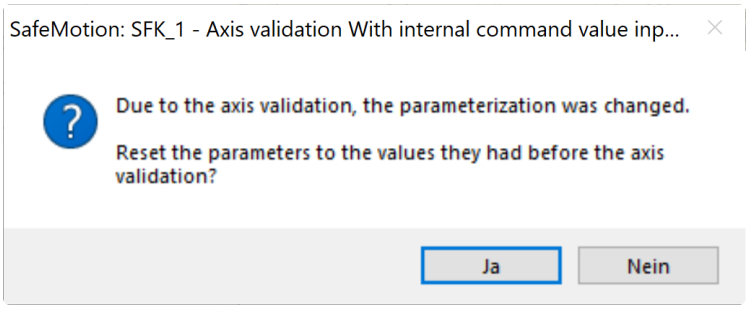


Figure 12 Axis evaluation (2)

14. After axis evaluation, reactivate the PLC under *MLD* → *Configuration*: First, click on the *RESET warm (1)* button and afterwards on the *RUN (2)* button.

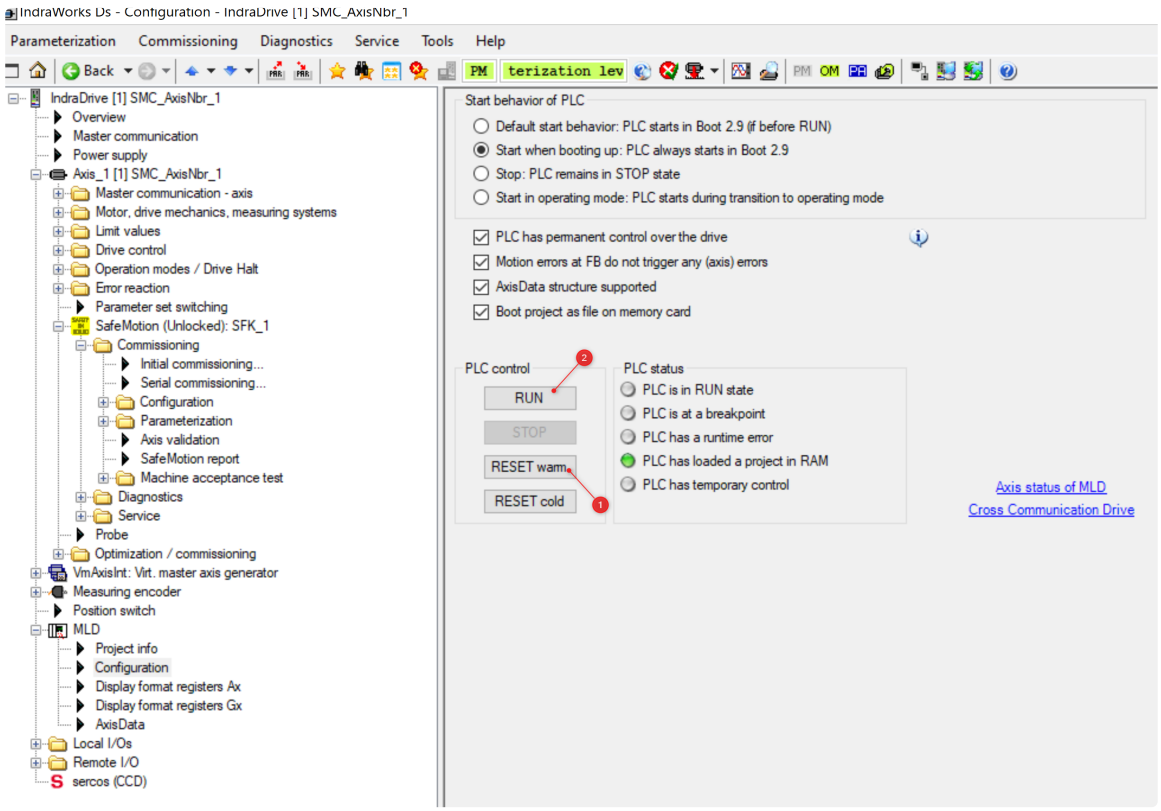


Figure 13 Start of PLC after axis evaluation

◊ For more information on commissioning the safety technology, please see the “IndraDrive Integrated Safety Technology Safe Motion (from MPx-18)” instructions (R911338919) chapter 9, see www.boschrexroth.com/medienverzeichnis. In addition, observe the notes specific to the SFK in the paragraph above.

Notice
Contact the service if the error message “F8304 SMO: System Error” is displayed.

3.2.1 Configuration instructions for safe fieldbus communication

The integrated safety technology in the "Safe Motion" version supports not only discrete selection and acknowledgment but also the selection and acknowledgment of the safety functions via safe channels of the master communication (safe bus communication). For this purpose, the standard protocol of the respective master communication is extended by a safe protocol so that, in addition to standard communication, safe information can also be transmitted to and from a safety master.

◊ For further information on safe fieldbus communication, see the manual "IndraDrive Integrated Safety Technology Safe Motion" (from MPx-18) (R911338919).

◊ For more information on the supported safety functions, the connection of safety-relevant peripheral equipment and the safety-relevant behavior of the software, please refer to the chapter Safety technology.

1. Carry out steps 1 to 8 inclusive according to chapter Commissioning with safety option S4.
2. Activate the desired fieldbus protocol in the System configuration.

> **Notice**
>
> If the safe fieldbus communication is already enabled in the master control, the error message "F4012 Incorrect I/O length" is displayed.

3. Continue commissioning from step 11 (see chapter Commissioning with safety option S4).
   a. Select the safe fieldbus communication in the corresponding dialog window:

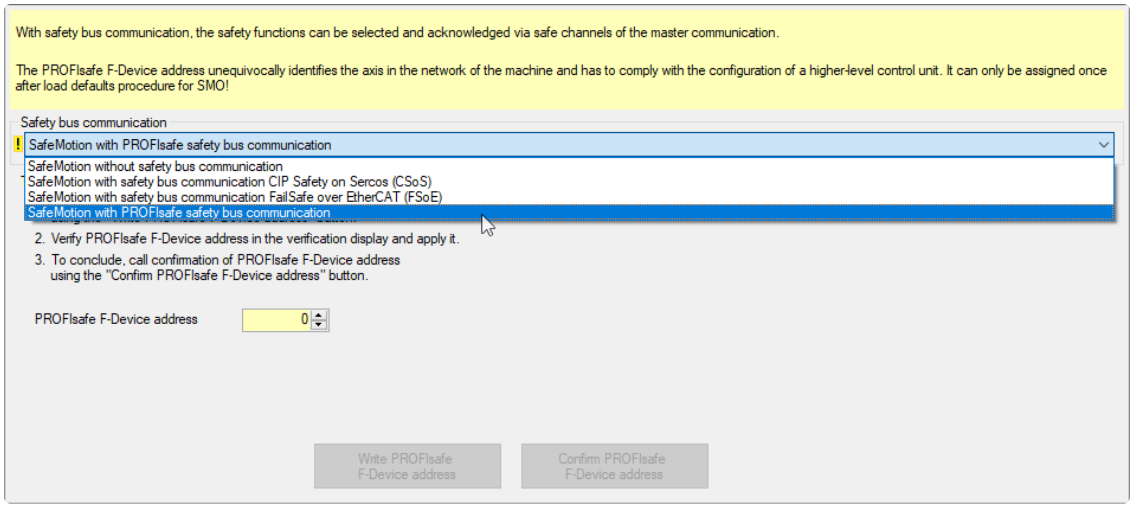


Figure 14 Selection safe fieldbus communication

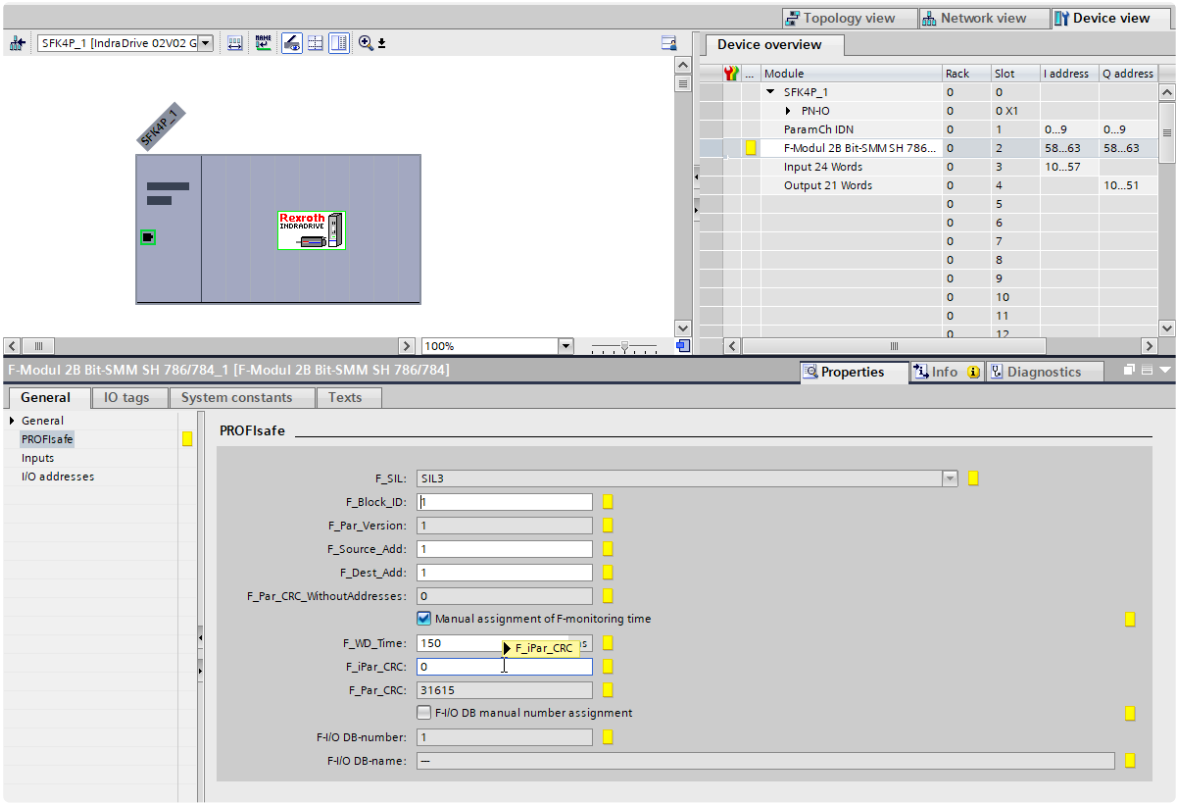
   b. Follow the configuration instructions after selecting the appropriate safety protocol.

Notes on using PROFIsafe in conjunction with the TIA Portal

1. First, configure the HCS01.
   a. Assign an F-Device address in the device (see screenshot above Selection safe fieldbus communication).
   b. Note the first four elements of the parameter S-0-1800.0.15 (elements 0..3) as a checksum.
2. In the TIA Portal, select the F-module of the HCS01 to be configured.

F-Parameter in the F-Host	Parameter description
F-target address "F_Dest_Add"	The F-device address of the drive is configured in the F-parameter "F_Dest_Add". This F-device address must match the F-device address configured in the drive and be unique in the PROFIsafe network.
F-Source address "F_Source_Add"	The address of the F-Host is assigned in "F_Source_Add". It must be unique in the PROFIsafe network.
PROFIsafe-Watchdog "F_WD_Time"	The "F_WD_Time" is transferred with the other F-parameters from the F-Host to each F-Device and is used to monitor the time of the cyclic PROFIsafe telegrams. If at least one new PROFIsafe telegram is not received within the "Watchdog Time", the PROFIsafe driver sets the data content of the assemblies to "0" (Fail Safe) and triggers the error in the drive F3461. The maximum monitoring time is 200 ms.

F-Parameter in the F-Host	Parameter description
Checksum "F_iPar_CRC"	The checksum "F_iPar_CRC" is intended to ensure that a connection is only established with an F-device with a specific safety configuration and parameterization. For this, the value from the parameter S-0-1800.0.15 (elements 0..3) of the drive are adopted into this parameter. When establishing the connection, "F_iPar_CRC" is compared with the value from S-0-1800.0.15. If the values are not identical, the drive reports the error F3460. If the value "F_iPar_CRC" is "0", no check is performed.



3. To verify the settings: Follow the configuration instructions of the safety controller.

◊ For further details please see the following document: https://www.boschrexroth.com/de/de/media-details/e4778763-fdaf-4bdb-990d-8b75923be974

3.2.2 Machine acceptance

After successful commissioning of the safety technology and axis evaluation, machine acceptance testing can be performed.

→ To carry out the desired axis movements with the required dynamics: For acceptance tests, create programs in the SFK software.
At the same time, you can record the desired parameters (e.g. actual velocity) using the oscilloscope in the IndraWorks DS software.

◊ For more information on the machine acceptance, refer to the instructions “IndraDrive Integrated Safety Technology Safe Motion (from MPx-18)” (R911338919).

Notice
In order to be able to use the oscilloscope in IndraWorks DS, the acceptance test mode in the SFK software must be activated.
→ Activate the acceptance test mode in the navigation area under *Configuration* → *General*.

→ Once you have commissioned the system and it is ready for production operation, deactivate the acceptance mode under *Configuration* → *General* → *Deactivate Acceptance Test Mode*. This enables data recording during a program run.

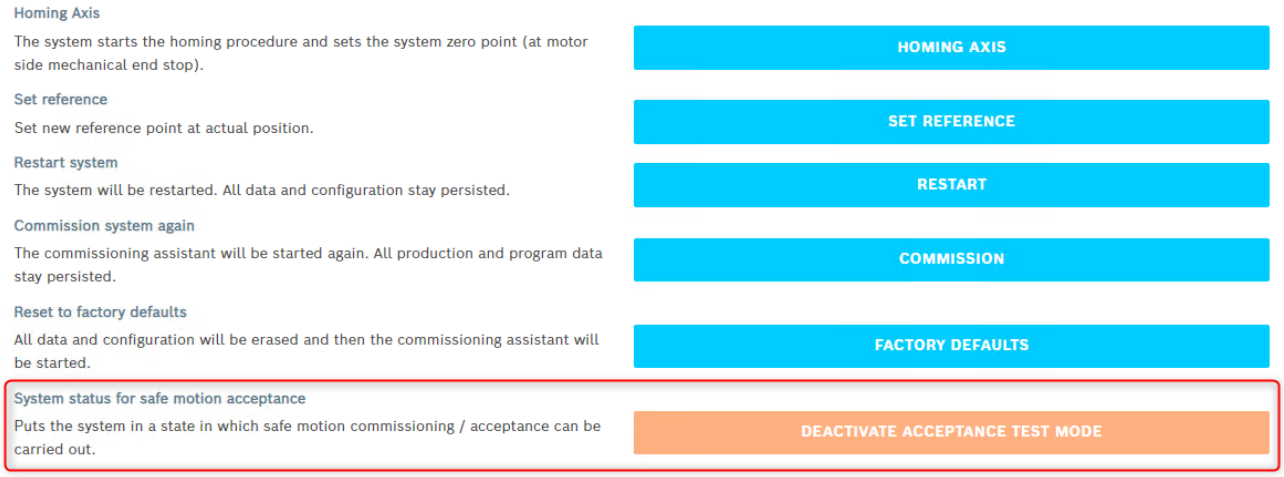


Figure 15 Deactivation of Acceptance Test Mode

Notice
→ When using the safety function "Safe maximum speed (SMS)" enter a speed in the parameter "Y1004 - Maximum speed" [in mm/min] in the extended configuration.
💡 Enter a speed that is slightly lower than the specified speed to avoid generating an error even in the event of overshoot.
This prevents a higher velocity from being parameterized in the component and thus the occurrence of error F7020 - Safe maximum speed exceeded.

3.3 Force sensor

Notice

The SFK can be ordered with or without a force sensor.
If the SFK is ordered without a force sensor, a suitable sensor must be added.

For reliable force measurement and control, the SFK requires a force sensor. Carry out mounting and connection before commissioning.

◊ For more information on wiring of peripheral devices with the SFK, please see the assembly instructions (R320103194) and the exemplary circuit diagram, see https://www.boschrexroth.com/product/cls_eCommerce_lx_1555747.

3.3.1 Included force sensor

If the included force sensor is used, no further configuration is required in the system.

All parameters are automatically configured during commissioning. Thanks to calibration at the factory, the force sensor does not require any individual adjustment of parameters.

Notice
It is recommended to have the force sensor regularly recalibrated by the manufacturer at intervals of approx. 24 months. Contact your sales partner and provide the serial number of the force sensor (see chapter Help).

3.3.2 Customized force sensor

If a customized force sensor is used, the system configuration parameters must be adapted to the sensor characteristics. These parameters can be set during commissioning. The parameters can be adjusted later as well.

Procedure for configuration:

1. First wire the force sensor according to the assembly instructions (R320103194) and the example circuit diagram.
2. >Call up the Extended configuration.
   The procedure for editing parameters is described in chapter Hardware configurations.
3. Activate the force sensor by setting of: **Y1049** "Activation of analog force sensor", value: 1.
4. Set the lead characteristics of the force sensor in [N/V]: This value is specified in the data sheet of the force sensor or any signal conditioners used.
   The lead characteristics in the tension and pressure range are typically identical.
   - Pull: **Y1050** “Sensor characteristic (Pull)", value in [N/V]
   - Pressure: **Y1053** “Sensor characteristic (Push)", value in [N/V]
5. Set the zero passage in [V]: This value is specified in the data sheet of the force sensor or any signal conditioners used.
   The zero passage corresponds to the output voltage in non-loaded condition. **Y1054** "Sensor zero signal", value in [V].
6. Set the permissible overload in [N] for the "Join to force" module.
   As a guideline, 45% of the numerical value of the gradient characteristic can be selected. **VFR002** "'Join to force': Maximum overload", value in [N].
7. Save the extended configuration.
   Changes are applied immediately.
8. Carry out the taring via the control bar (see chapter Tare force sensor).

Notice
Only force sensors with an output signal of -10 to +10 V can be used.

NOTICE

Incorrect sensor evaluation

Incorrect configuration can lead to incorrect evaluation of the sensor. This may result in damage to the workpiece.

→ Check the entered parameters for plausibility.

3.3.3 Verification

Verification of the measured force can be carried out as part of a machine acceptance tests or other methods. The verification determines deviations from a comparison measurement. The test can be carried out after commissioning.

For successful measurement, ensure the following:

- The external force sensors have received valid calibration and are operated in compliance with the applicable instructions.
- A suitable positioning of external force sensors without tilting to the cylinder.

- The force sensor is installed and wired according to assembly instructions (R320103194) and example circuit diagram.

Proceed as follows:

1. Tare the force sensors to 0 N.
2. Prepare a sequence (see the chapter Programming) with the “Join to force” module.
3. Use a low speed with high acceleration to achieve the required force.
4. Read the force value from the external sensors directly after leaving the “Join to force” module. Later reading leads to a deviating value due to settling of the mechanics.

Notice
Fixed deviations over the entire measuring range of the external force sensors can be compensated by taring (see chapters Tare force sensor and Library: "Tare" module).
Linear deviations the deviation of which is dependent on the force indicate an impact of the measurement setup.
→ Check the measurement setup.

3.4 Recommissioning after standstill

When restarting the system after a shutdown, recommissioning is not required. All relevant information is saved in the system. Immediate resumption of the production process is ensured.

4 Operation

The SFK web interface can be displayed via a web browser. The web interface is divided into three main areas:

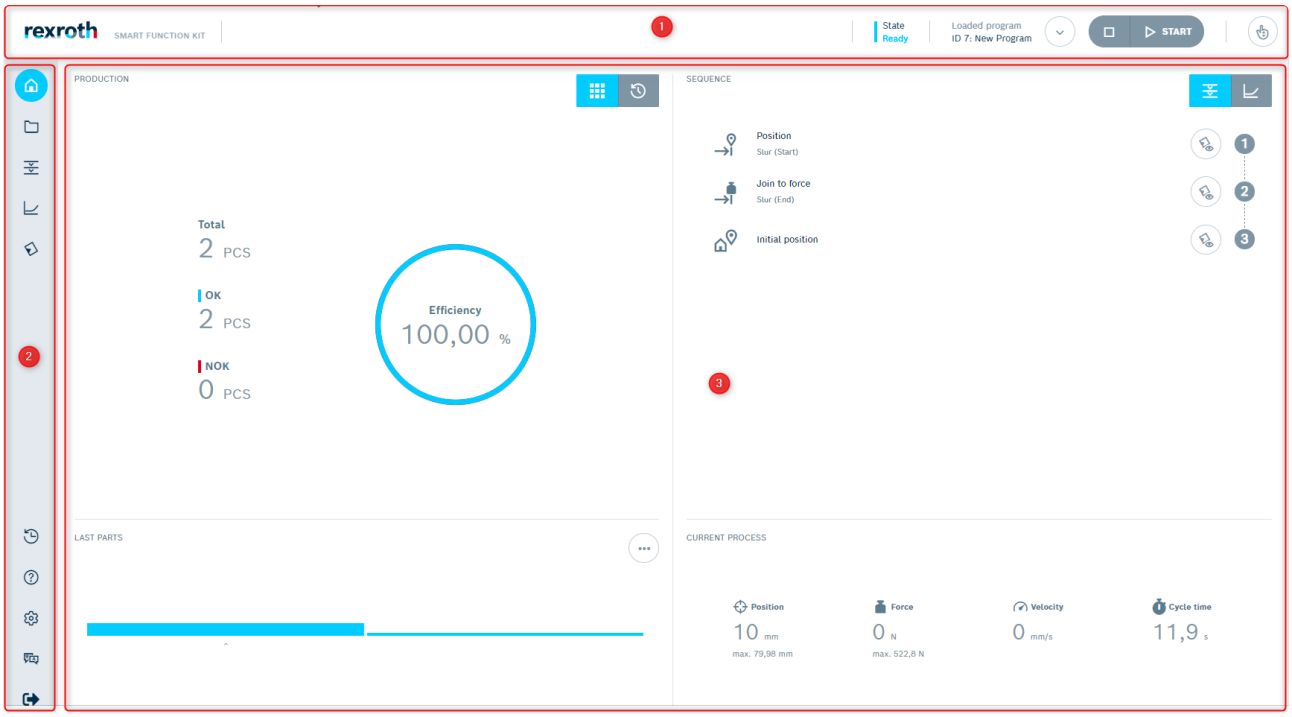


Figure 16 User interface overview Smart Function Kit

1. Title bar: Display of status information
2. Navigation bar: Navigation through the web interface
3. Content area: Display of the selected program

> **Notice**
> The SFK web interface is not suitable for being open simultaneously in several browser tabs or on different computers. If opened several times, display errors may occur. For information on the troubleshooting, see chapter Display errors.

4.1 Title bar

The title bar displays the system status and provides the possibility of changing and manually starting and stopping the program.

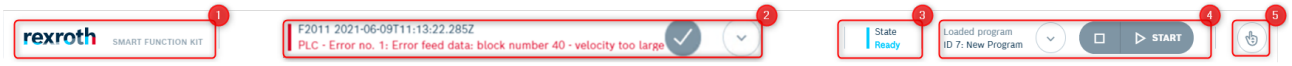


Figure 17 Title bar

Position	Designation	Description
1	Icon and title	Display of the Bosch Rexroth AG logo with product name (here: Smart Function Kit). → To switch to full screen mode: Click on the Bosch Rexroth AG *logo*.
2	Messages and errors	Display of the system messages and errors. If several errors occur, the error with the highest priority is always displayed. → To acknowledge (delete) the error messages: Click the *Acknowledge* button ( ). ✅ The last error is always acknowledged. → To acknowledge (delete) several pending errors: Click the *Acknowledge button several times* ( ). → To change to the diagnostics area: Click the *Info* button ( ). ✅ The diagnostics area provides more information on the errors.
3	System status	Displays whether or not the system is ready for operation. A distinction is made between the following states: **Ready:** The system is ready for operation. **Error:** An error is pending. Operation is not possible. **Parameter mode:** The system is in parameter mode, e.g. for writing system parameters. Switching between parameter and operating mode is performed automatically. **Power missing:** The intermediate circuit voltage has not yet been enabled on the drive controller. **Reference missing:** The reference point of the system has not yet been set.
4	Program selection	Display to select the program. → To open the selection control element: Click the arrow. ✅ A list of all programs created and saved in the program management is displayed. The programs are sorted according to the fieldbus id. → Select the required program and click *START*. ✅ The selected program is started and performed once. **Notice** The button *START* can only be operated when the system is not performing any other action. → To stop all system movements: Click the *STOP* button. ✅ The running program is aborted and the produced part is rated NOK. The cycle time of the program run is set to -1 second.

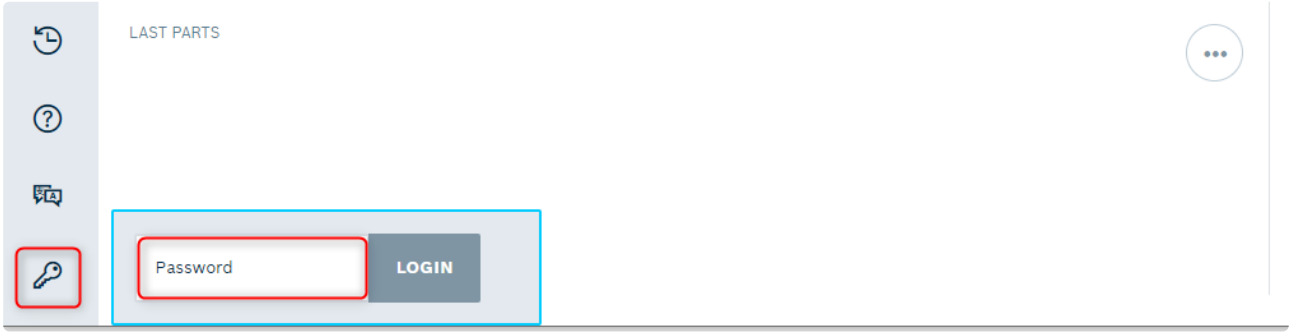
Position	Designation	Description
5	Control panel	Button to manually operate the system. • To show or hide the control panel: Click the *Control Panel* button. ◊ For more information about the control bar, see chapter Control panel.

Table 4 Title bar

4.2 Navigation bar

The navigation bar offers the administrator in the administrator account the possibility to log into the system with the password "admin".

If the password was changed during commissioning, login is only possible with the password you chose yourself.



4.3 Content area

The content area displays the area currently selected in the navigation menu.

In the content areas Edit program, Reference curves and curve evaluation, only one individual program is edited at a time.

The program to be edited is selected via the selection element in the content area.

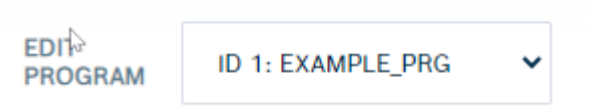


Figure 18 Content area Selection element Program

This selection element can also be used to quickly switch between different programs without having to first open the program management.

4.4 Full-screen mode

The web interface of the SFK can be displayed in full-screen mode. For this purpose, the navigation elements of the web browser are hidden and the entire screen is used for display of the web interface.

→ To switch to full screen mode: Click the Rexroth logo in the title bar.

✓ Full-screen mode is activated.

→ To adjust the width of the navigation bar: Click the Rexroth logo again.

→ To deactivate, follow the instructions of the web browser and/or the operating instructions of the mobile device.

5 Dashboard

The dashboard is displayed in the content area and provides an overview of the system and production status. The four tiles provide information on production statistics, quality evaluation of the parts produced last, on the current program progress and the current machine values.

The dashboard is automatically displayed after start-up of the system and can be alternatively called up at any time via the *Home* button in the navigation bar.

All information always refers to the program selected in the navigation bar. As soon as programs are switched in the status bar, the contents of the dashboard are changed accordingly.

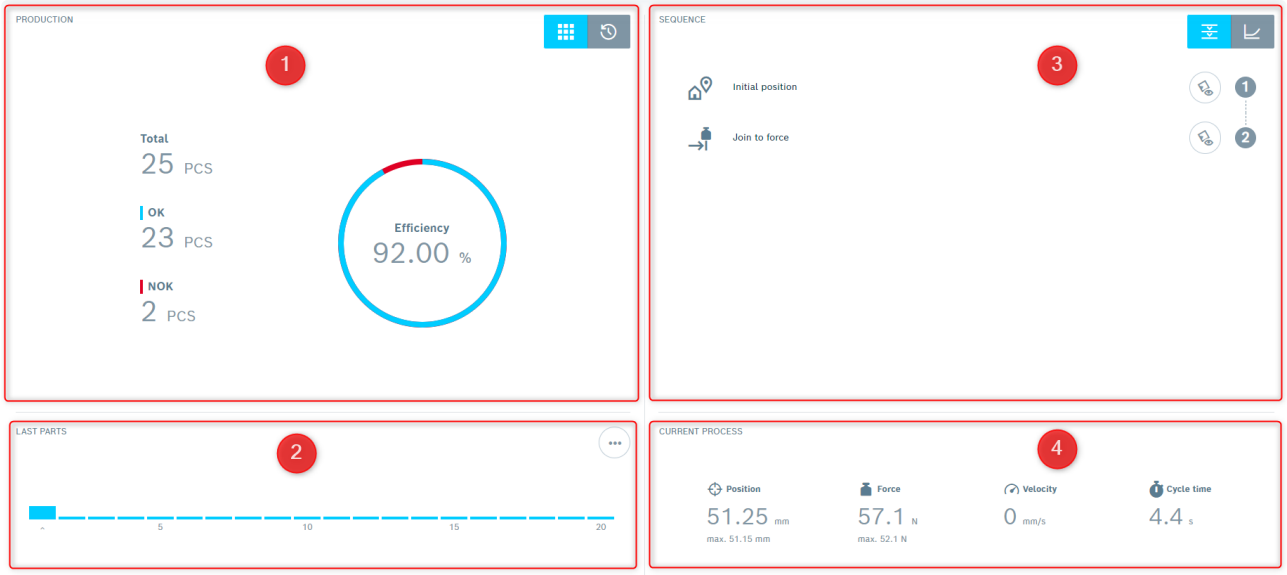


Figure 19 Dashboard

1. Production statistics
2. Last parts
3. Sequence
4. Current process

5.1 Production statistics

Production statistics show, among other things, the number of parts produced.

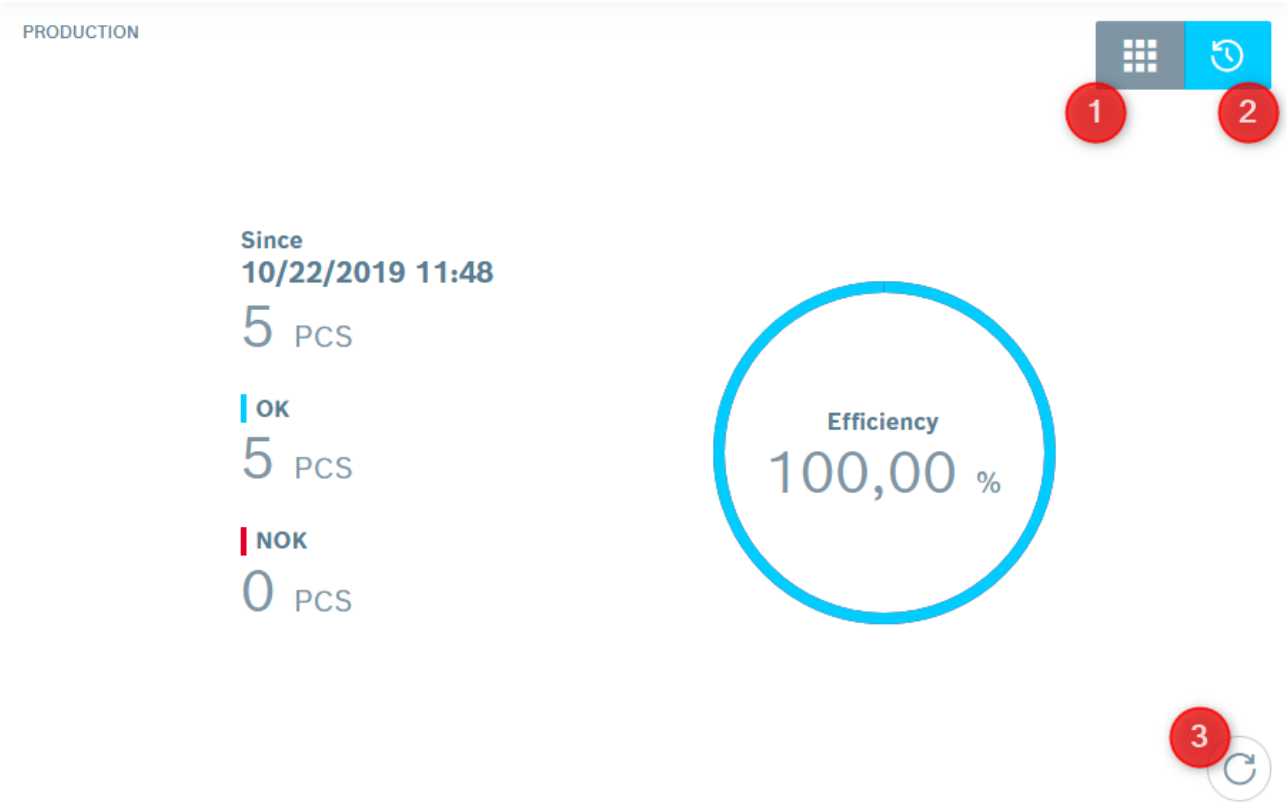


Figure 20 Production statistics

→ Click the statistics icon (1) to display:

- the total number of parts produced in the currently selected program
- the total number of parts with quality evaluation OK and NOK
- the percentage of OK and NOK parts

Notice
As soon as the production sequence has been started via the title bar or externally, the part located in the machine is evaluated as a produced part.
Recording of reference curves does not count as produced parts.
The values of the production statistics are individually recorded for every program and are not reset upon change of program.

→ Click the clock icon (2) to display:

- the number of parts produced in the current shift,
- the production data from the last reset, here “From 11.06.2021 12:25”.

→ To reset the statistics: Click the *Reset* button (3).

5.2 Last parts

In Last parts, the quality evaluation of the last 20 parts is displayed graphically in the selected program.

Figure 21 Last parts

→ To change to the *Production history* area and view all production data of the selected program: Click the ... *button* (1).

→ To display detailed information about the produced part: Click one of the *evaluations* (2).

5.3 Sequence

In Sequence, the individual steps of the selected program are listed. The view can be used to check whether the loaded program is the required production program.

Sequence can be displayed as sequence of steps or as force-displacement or force-time curve.

Sequence of steps

Figure 22 Sequence of steps

→ To display the production program: Click the *Program* button (1).
✔ The active production step is highlighted.

Force-displacement curve

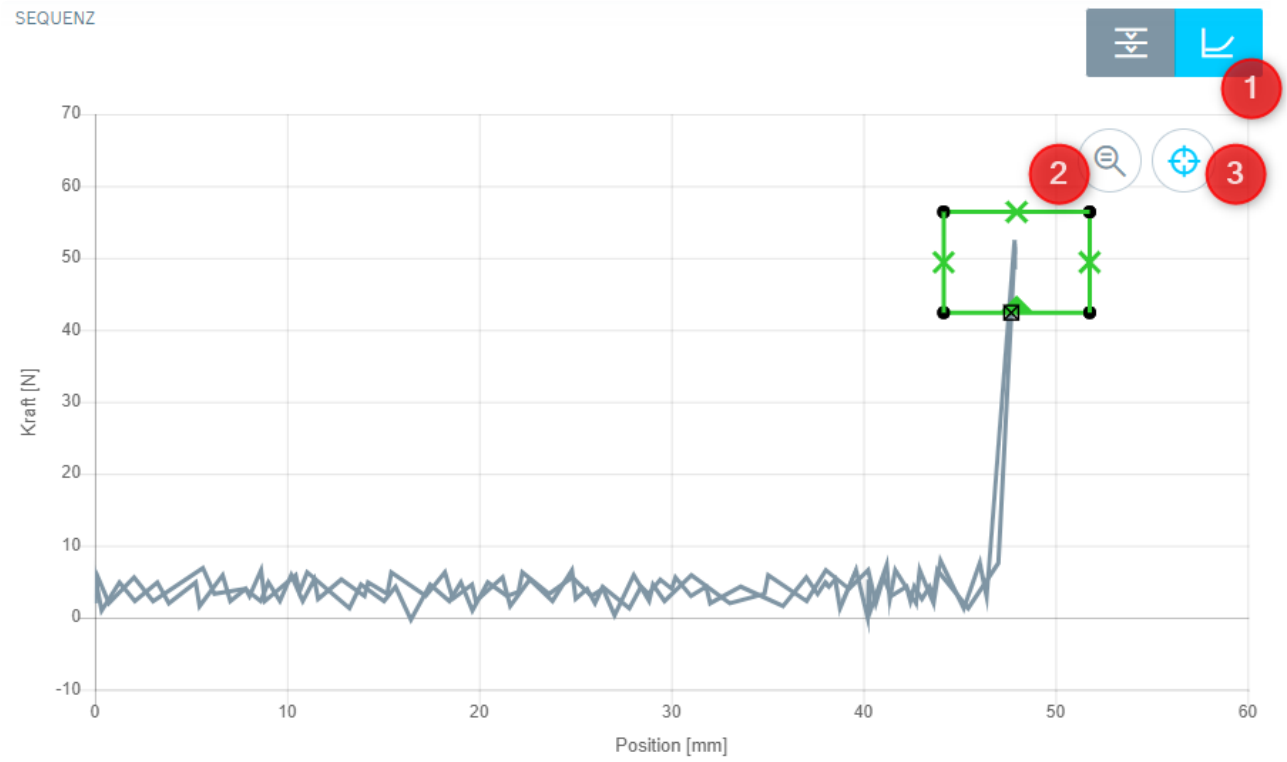


Figure 23 Force-displacement curve

→ To have the production program displayed as a force-displacement curve: Click the *Curve* button (1).

✓ The force-displacement curve is set up live while the production program is running.

✓ The evaluation elements and the quality evaluation are displayed.

> **Notice**
> The display may differ from the actual values as the data of the force-displacement curve are not processed in real time.
> If no production program is running, the last recorded curve including quality evaluation is displayed.

→ To set the zoom: Use the mouse to select the curve and scroll in or out with the mouse wheel.

→ To reset the zoom: Click the *magnifying glass* (2).

→ *Position symbol* (3): The position is displayed on the X-axis. Clicking the button displays the time on the X-axis. By clicking again, you can switch back to the position.

> **Notice**
> You cannot create or display evaluation elements in the force-time curve.

5.4 Current process

Current process displays the current values for position, force, velocity and cycle time. The values are updated live during the production program.

Figure 24 Current process

Notice

The force value is not displayed in real time and the force values are filtered for better representation. There may be slight deviations between the displayed and the real force value. For forces up to 100 N, two decimal places are shown. For higher forces, one decimal place is displayed. The accuracy of the displayed force value depends on the measurement uncertainty and the measuring range of the force sensor.

Notice

The position value is not displayed in real time and the position values are filtered for better representation. There may be slight deviations between the displayed and the real position value. Three decimal places are provided for the position. The accuracy of the displayed position value is predominantly dependent on mechanical influences.

The maximum values achieved during the production of a part are displayed below the current actual values. If no program is produced, the maximum values of the last part produced with this program are displayed.

6 Control panel

→ To show or hide the control panel: Click the *Control bar button in the title bar* .

◊ For more information on the title bar, please see the chapter Title bar.

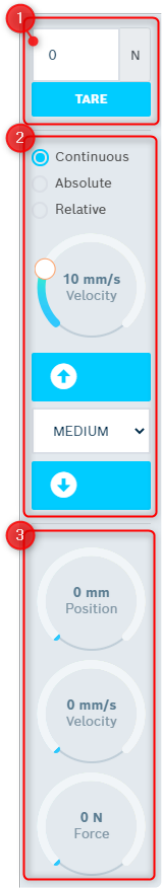


Figure 25 Control panel

The following functions can be called up via the control panel:

- Tare force sensor (1)
- Manual movement (2)
- Displaying current values for position, force, and travel speed (3)

6.1 Tare force sensor

The control panel offers the option of taring the force sensor to any desired value. This makes it possible to compensate for the weight of a tool, for example.

Figure 26 Tare force sensor

→ Enter the required value in the input field (1).

→ If required, adjust the value upwards or downwards using the arrow buttons.

→ Click the *TARE* button (2).

✅ The current sensor value is equated with the tare value.

✅ The tare value is saved and kept until next taring.

> **Notice**
> Taring applies to all force values (e.g. in program steps, curve recording, quality evaluation).

6.2 Manual movement

Three modes for manual movement are available via the control panel.

- Continuous
- Absolute
- Relative

They can be selected via the button. Depending on the selected mode, the display of the control panel and/or the behavior of the system will change.

6.2.1 Continuous

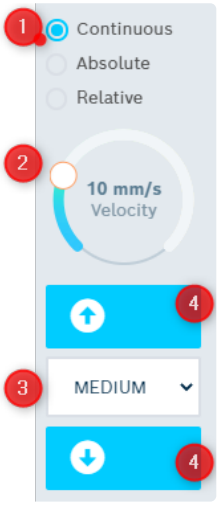


Figure 27 Manual movement - Continuous

→ Select the *Continuous* (1) mode.

→ To set the velocity (mm/s) with which the SFK is to move: Move the controller (2).

→ Open the selection element (3) and set the sensitivity (high/medium/low) with which the system is to respond to an interruption of communication.

- HIGH: Immediate system stop in case of interruption of communication.
- MEDIUM: Preselection. No immediate system stop in case of temporary interruption of communication.
- LOW: Shutdown of the system in case of extended interruption of communication.

→ Press one of the two arrow buttons *(up/down) (4)* and keep the corresponding button pressed.

✅ The SFK will move for as long as the arrow button is kept pressed.

Notice
To make sure that the system is put to a standstill in case of interruption of the communication, cyclic status monitoring is performed.
If status monitoring is interrupted, the system is stopped for safety reasons.
A new motion command must be started via the arrow button (4).

NOTICE

High lag of the system with sensitivity set to low

High lag of the system if the sensitivity is set to low may result in damage to the system.

→ Use the *low sensitivity (low)* setting only at low velocities.

6.2.2 Absolute

In *Absolute* mode, a defined target position is approached absolutely. This SFK detects whether the specified target position is in positive or negative direction and thus only the corresponding arrow button will be available for selection.

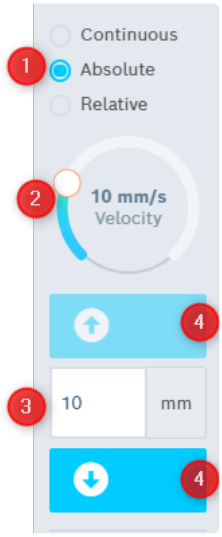


Figure 28 Manual movement - Absolute

→ Select the *Absolute* (1) mode.

→ To set the velocity (mm/s) with which the SFK is to move: Move the controller (2).

→ Enter the required target position in the input field (3) in mm.

✅ The system verifies the entered target position and automatically blocks movement to values that are outside the system limits.

> **Notice**
> A note is displayed below the input field if the entered target position cannot be approached.

→ Adjust the value in the input field (3) according to the note.

→ Press the arrow button *(up or down) (4)* that is activated and keep it pressed.

✅ The SFK will move for as long as the arrow button is kept pressed.

6.2.3 Relative

The Relative mode allows for inching mode with variable jog velocity and speed.

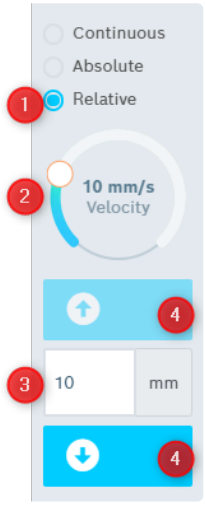


Figure 29 Manual movement - Relative

→ Select the *Relative* (1) mode.

→ To set the velocity (mm/s) with which the Smart Function Kit is to move: Move the controller (2).

→ Enter a defined jog velocity (mm) in the input field (3).

→ Press one of the two arrow buttons *(up/down)* (4).

✅ Each time you press one of the arrow buttons (4), the pre-defined jog velocity (3) is traveled at the defined velocity (2).

6.3 Displaying current values for position, force, and velocity

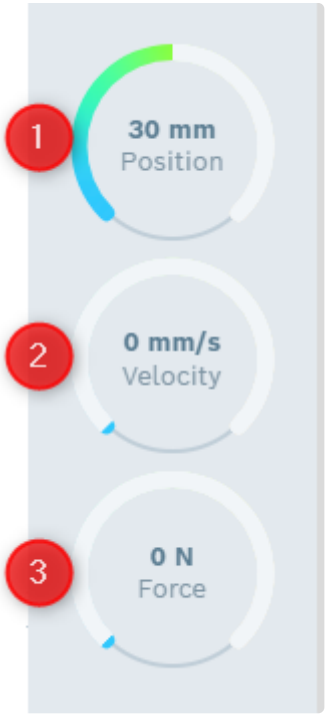


Figure 30 Displaying current values for position, force, and velocity

The following current values are displayed in the lower area of the control panel:

- Position (1)
- Velocity (2)
- Force (3)

The force value is not displayed in real time and the force values are filtered for better representation. There may be slight deviations between the displayed and the real force value.

7 Program management

All programs available in the system can be managed in *program management*.

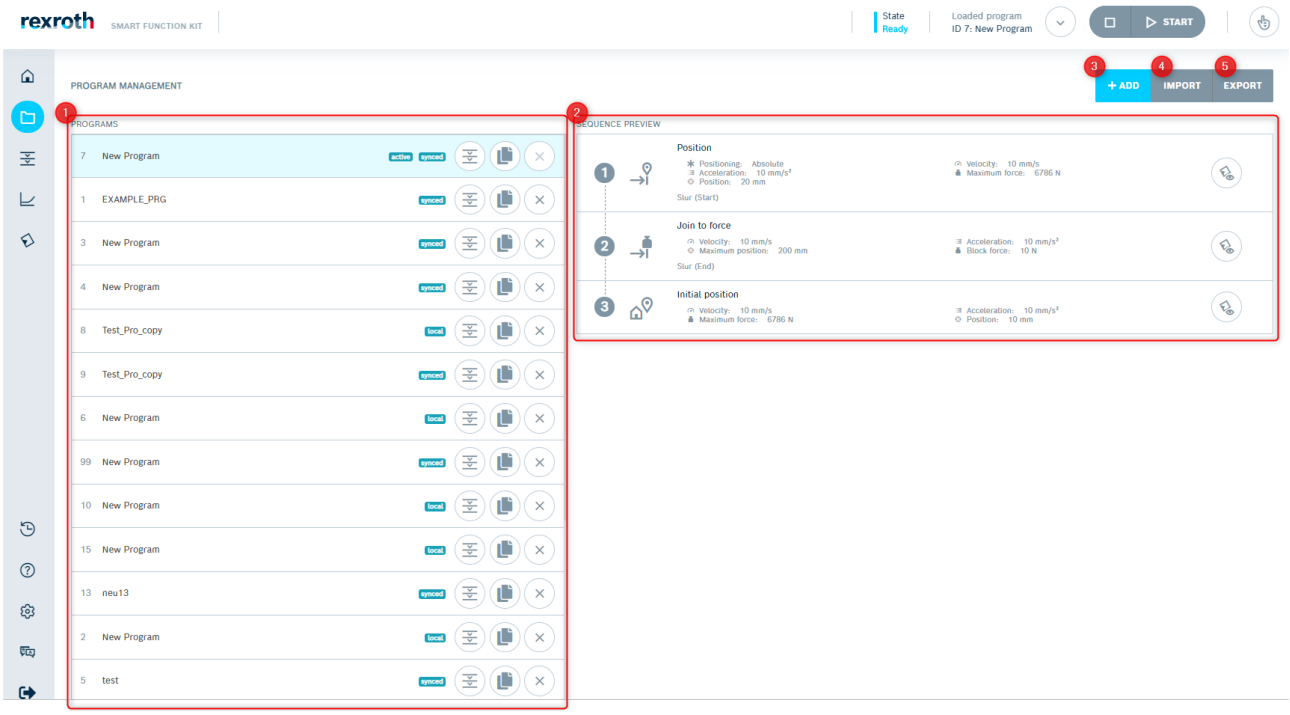


Figure 31 Program management

Position	Designation	Description
1	Programs	List of all programs, including program name and program fieldbus ID. The programs are sorted according to the last usage time, so that the last used programs appear at the top of the list. **Program status** • active: This is the program currently loaded in the title bar. • synchronized: The program version on the PR21 and the motor controller is the same. The program can be selected and started. • local: A program has been newly created, imported, or copied and has not yet been saved. The program cannot be selected and started in the title bar. → Save the program using the *Save* button to set it to the “synchronized” status. • obsolete: The program has been changed or not saved yet. The program cannot be selected and started in the title bar. → Save the program using the *Save* button to set it to the “synchronized” status. **Editing a program** → Click the *Edit* button . ✅ The view changes automatically to the *Edit program* area. 💡 Here, changes can be made to the program: Adjustment of the program sequence with all detailed parameters as well as the program name and the program fieldbus ID. **Copying a program** → Click the *Copy* button . ✅ The view changes to the *Copy Program* area. ✅ The program, including the reference curves and the evaluation elements is duplicated and can be changed in the Edit program area. 💡 A copied program is automatically assigned a new program name and a new program fieldbus ID. Both values can be manually changed in the *Edit program* area. → Save the duplicated program once in the *Edit program* area. ✅ The original program is not affected by this. **Deleting a program** → Click the *Delete* button . ✅ The program is permanently removed from the program list. 💡 However, the program is kept in the data base as well as all related production data and entries in the history.
2	Sequence preview	Preview of the program selected from the list. The preview contains the program sequence and the detailed parameters of the individual program steps.

Position	Designation	Description
3	Adding a program	→ Click the *+ ADD* button. ✅ A new program is created. ✅ The view changes automatically to the *Edit program* area. 💡 The program name is automatically preassigned as New program and the program fieldbus ID is the next free fieldbus ID. → Manually change the two values in the *Edit program* area.
4	Import program	→ Click the *IMPORT* button. → In the *File selection dialog* navigate to the storage location of previously exported programs and select the desired program. → Click *Open*. ✅ The program is imported. ✅ During import, you will be automatically redirected to the *Edit program* area. → Save the imported program once in the *Edit program* area. ✅ Import is completed. 💡 When attempting to save the imported program, the system checks whether the program is compatible with the system and, e.g., all limit values are complied with. If individual settings are not compatible, they can be manually changed before saving.
5	Export program	→ Click the *EXPORT* button. ✅ The selected program is exported to the local memory of the HMI control unit. 💡 The export is done in JSON file format and contains the program, the reference curves and the definition of the evaluation elements. 💡 The export can be used as a data backup or to transfer a program to another system.

Table 5 Program management

8 Programming

Programming is carried out by graphically arranging modules from the library in a program sequence.

The individual arrangement and parameterization of the modules result in the desired program sequence.

The *Start data recording* (4) and *End data recording* (5) components are always part of the sequence and can be moved, but not deleted.

These components define the area in which the pressing curve is recorded. All modules that fall outside this range will not be recorded or evaluated. Only one measuring range can be defined per sequence.

The maximum supported execution time is 15 minutes per curve / pressing process.

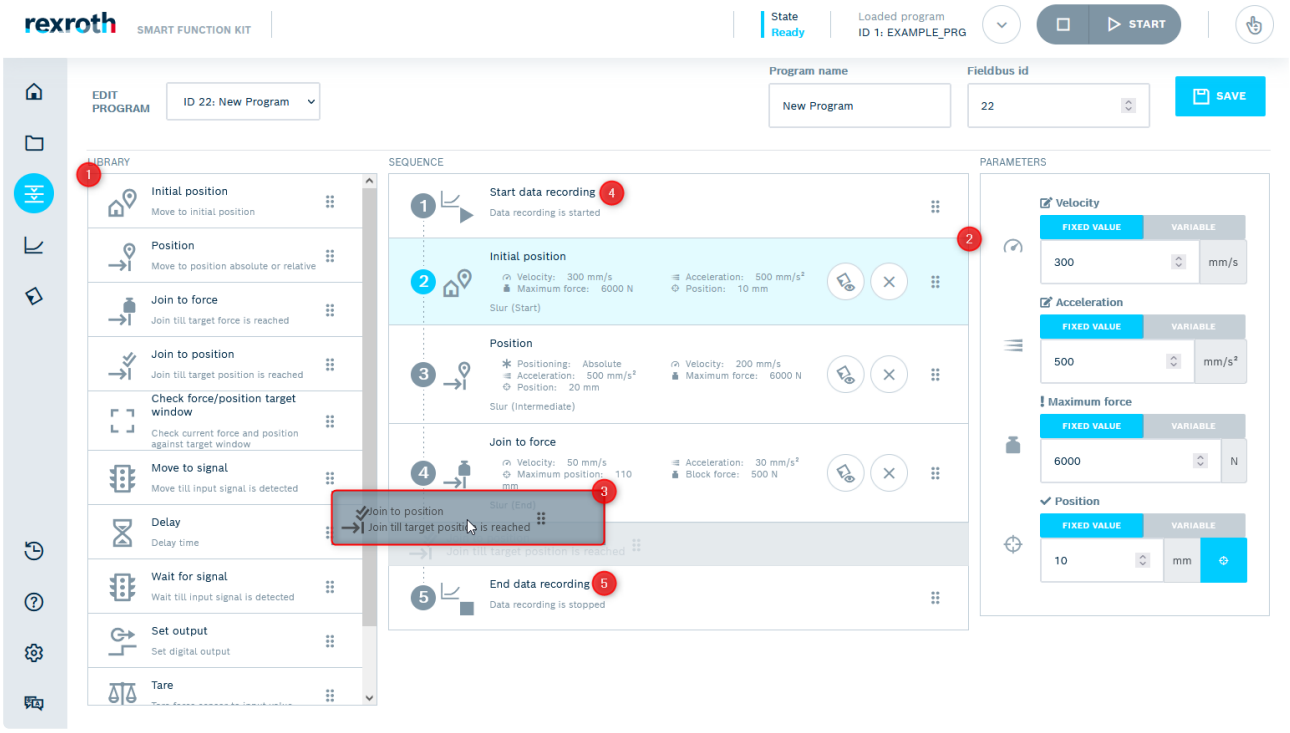


Figure 32 Programming

→ Place the required component (3) from the library (1) in the correct position in the program sequence (2) by using the drag-and-drop function.

→ Move the component *Start data recording* and *End data recording* to the desired positions

→ Change the program name to match your sequence and assign a fieldbus ID.

The *fieldbus ID* field can be used to assign a unique identification number for the program. When a program is newly created, copied or imported, the system automatically assigns the next free fieldbus ID.

→ Click the *SAVE* button to save the program. Then select the program in the title bar to set the changes active.

8.1 Module properties

The SFK records the corresponding force-displacement curve for the program sequence. It can be set for every module whether the module is going to be included in the evaluation. In case that a module is excluded from the evaluation, it can be additionally defined whether or not it is going to be included in the force-displacement curve.

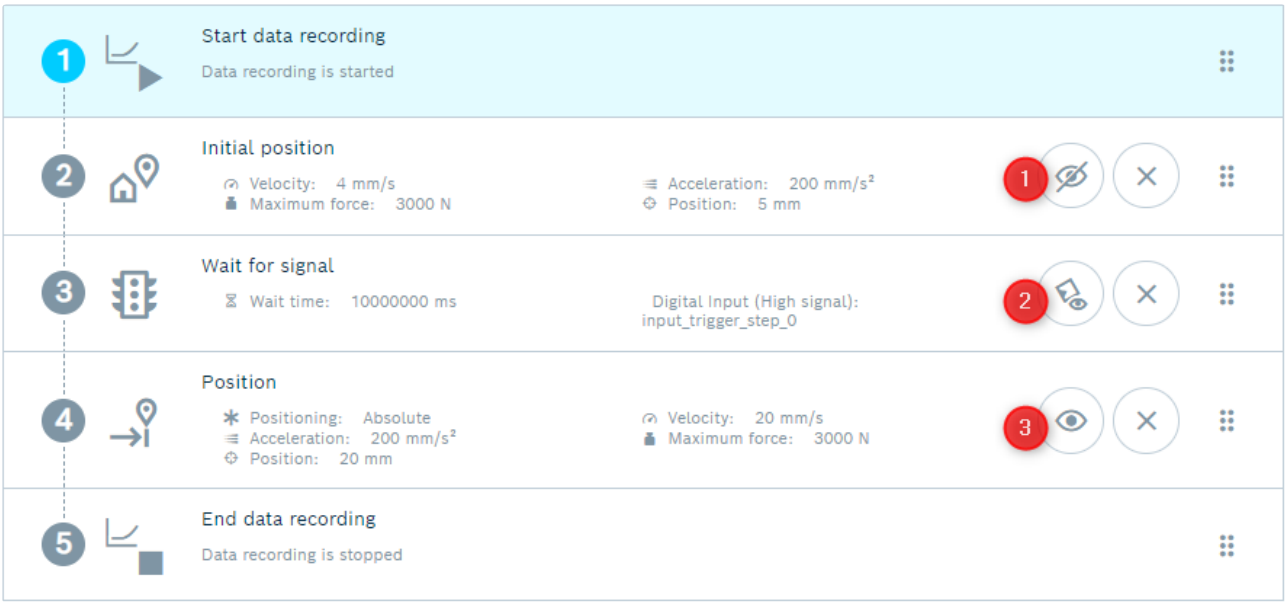
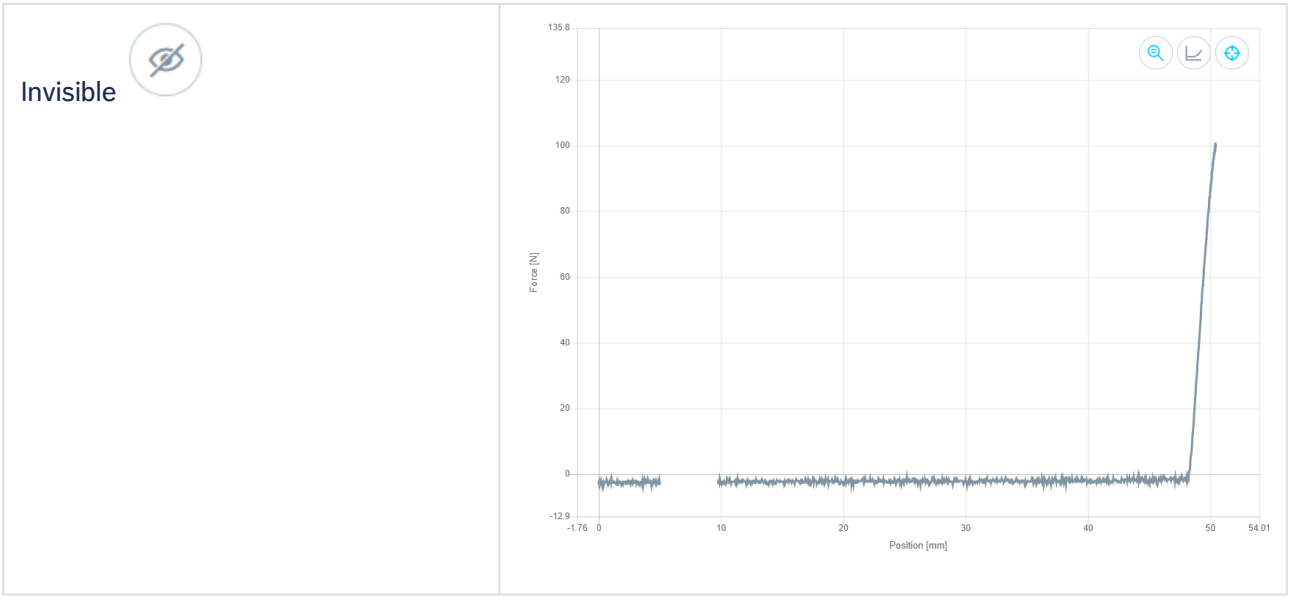


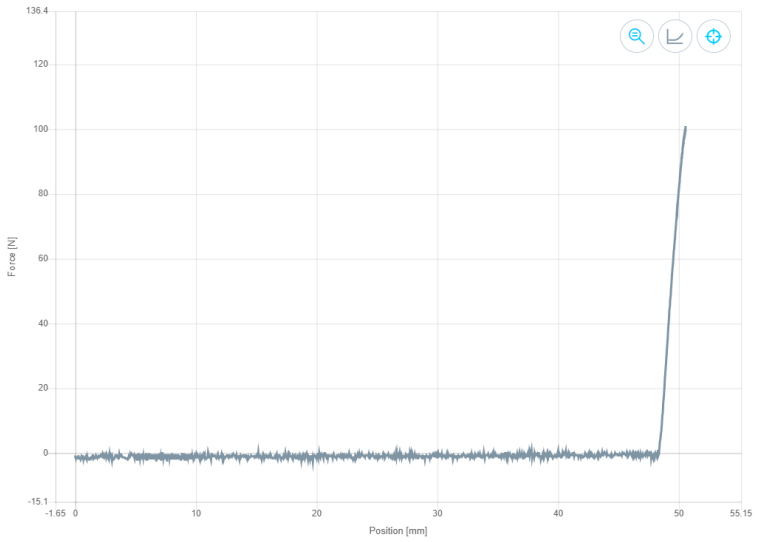
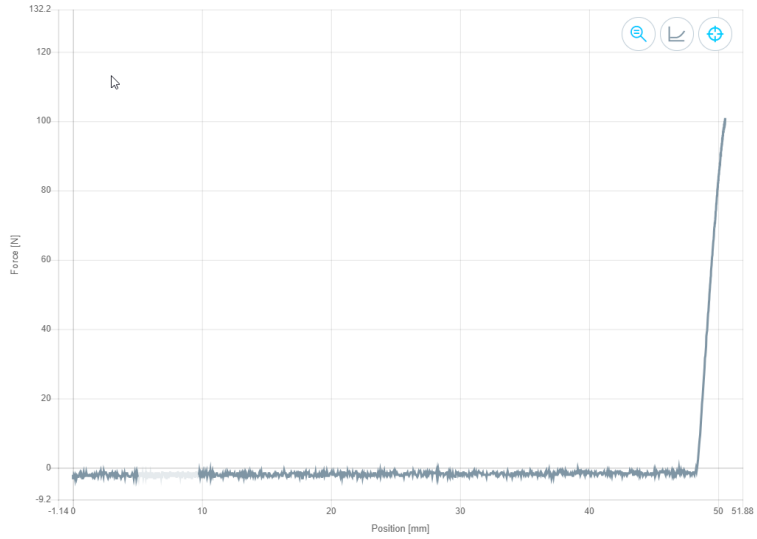
Figure 33 Module properties

The properties can be activated or deactivated via a drop-down menu in the program module.

- Invisible (1): Curve section is not shown and not evaluated.
- Evaluated (2): Curve section is shown and evaluated.
- Visible but not evaluated (3): Curve section is displayed but not evaluated.

Notice
To allow for visual differentiation: Modules and/or their force-displacement curves that are not included in the evaluation but still are to be shown are displayed with a reduced color intensity.



Evaluated 	
Visible but not evaluated 	

8.2 Library

The system provides a library of ready-made modules on the left side of the screen. You can integrate the required modules from this library into the sequence by dragging and dropping or by clicking on them.

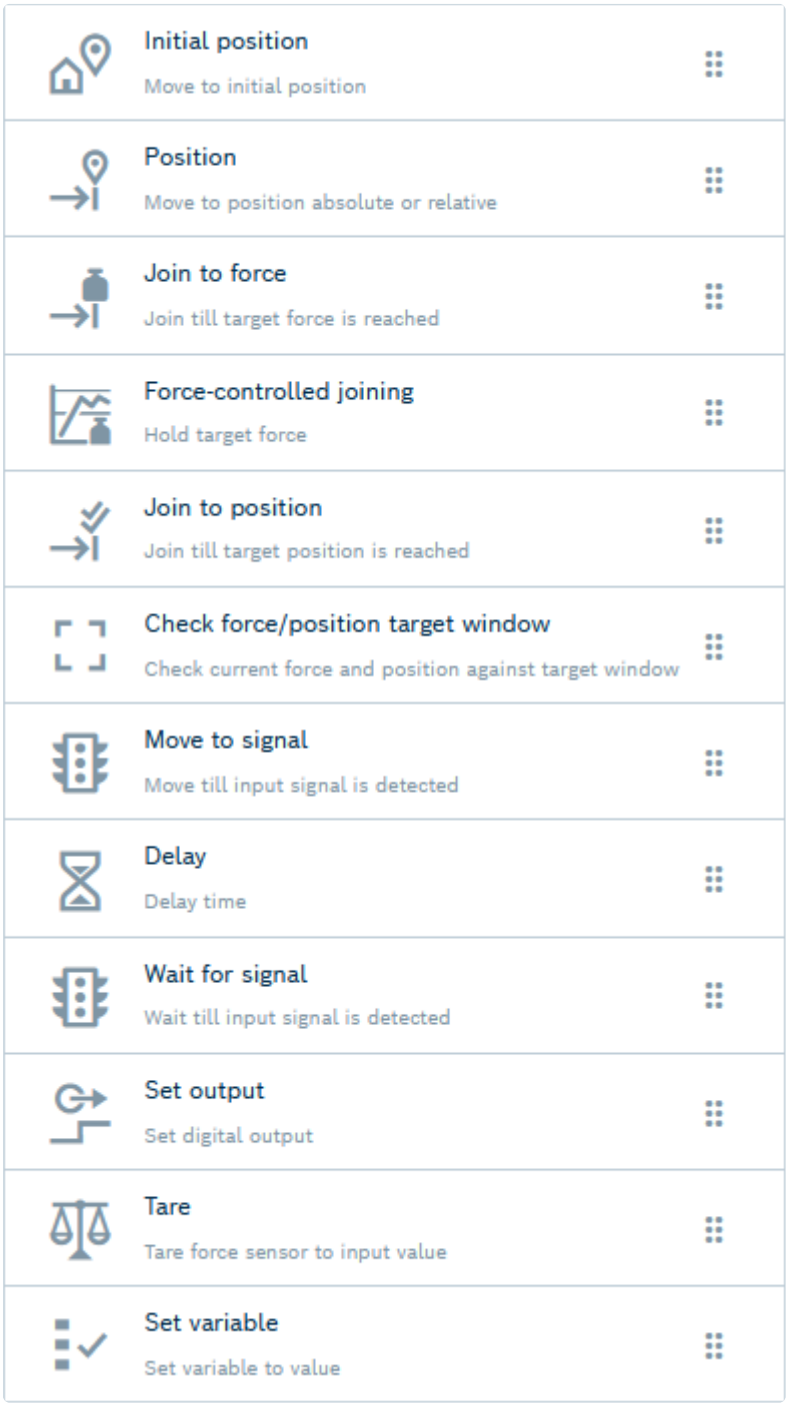
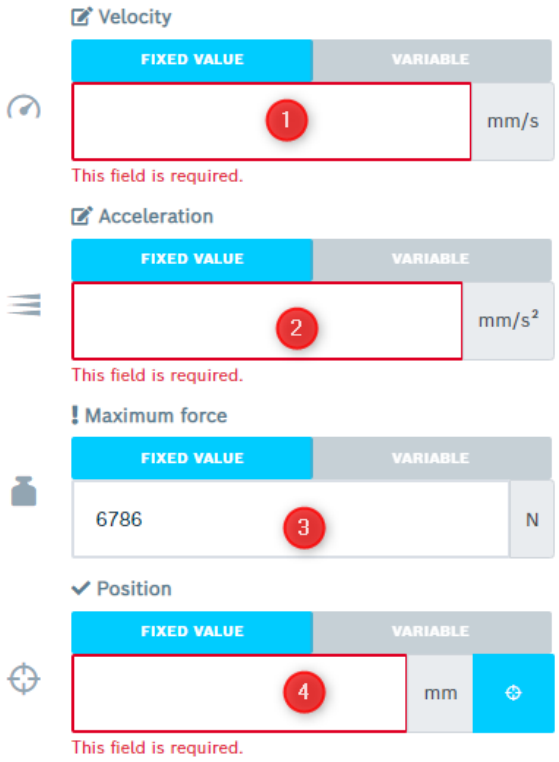
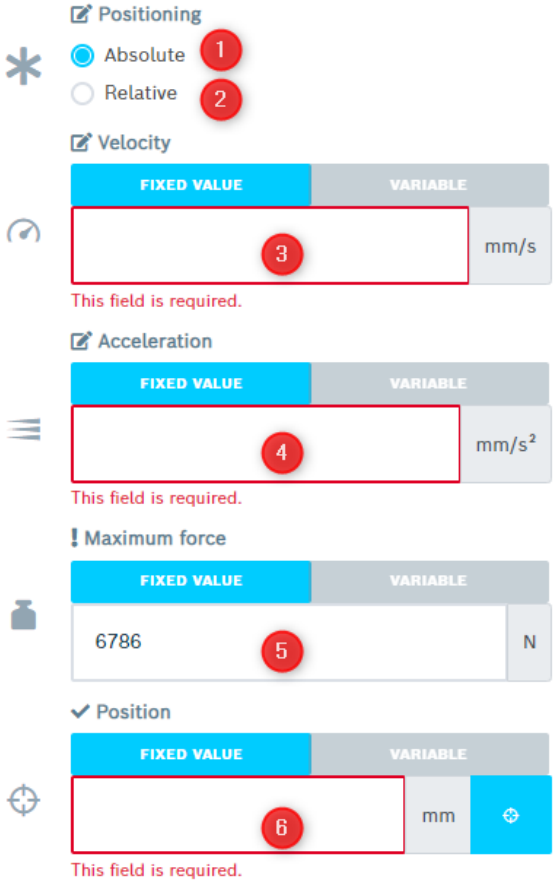
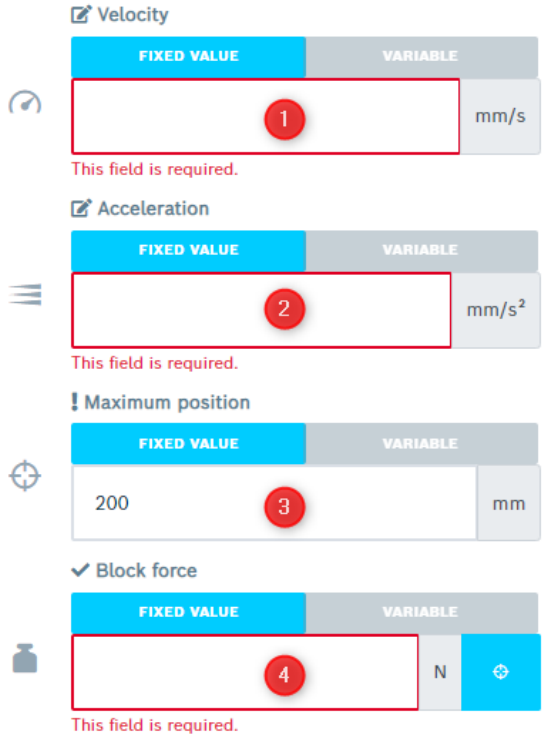


Figure 34 Library

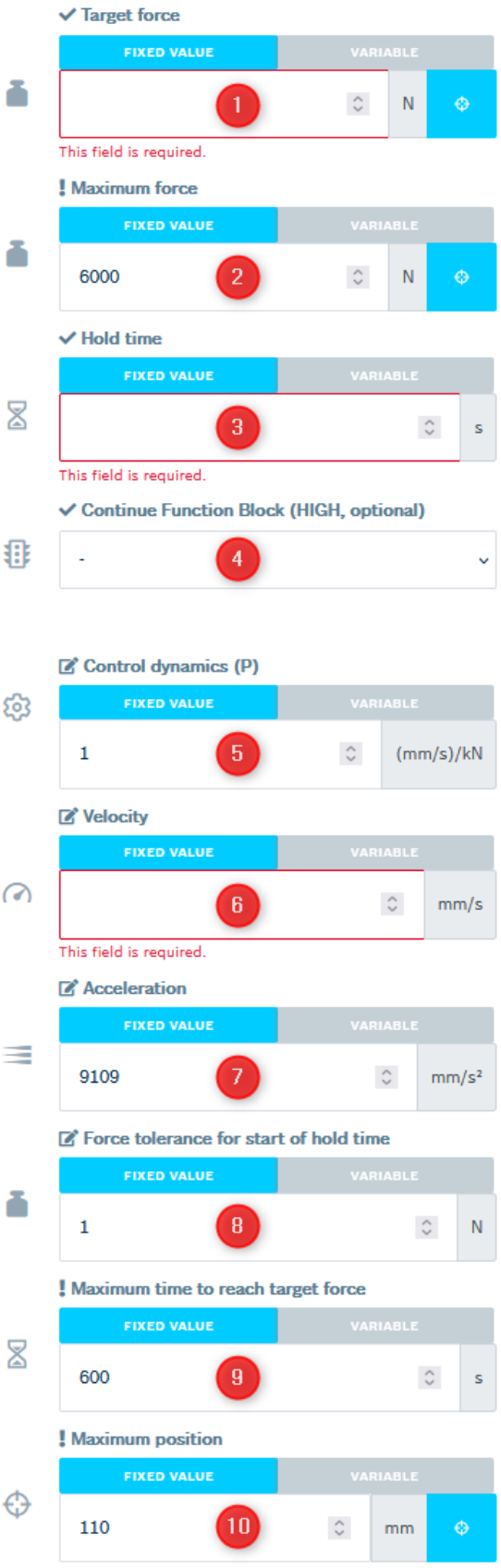
The following table describes the prefabricated modules in more detail:

Screens/comments	Action steps
	**Initial position** The module *Initial position* is used to move to a defined starting position. This position describes the starting point of the joining sequence. Once the position is reached, the system switches to the next module in the program sequence. → Enter the velocity (1) [in mm/s] and the acceleration (2) [in mm/s^2] with which the system is to move. ✓ Reaching of the velocity with the selected acceleration is not checked. → Specify the maximum force (3) [in N]. ✓ If no force is specified, the maximum force of the system is automatically used. → Enter the target position (4) [in mm]. ✓ While approaching the target position (4), the current force is monitored by the force sensor. ✓ To prevent damage to the system or the tools: If the actual force exceeds the specified maximum value, the system is stopped immediately. ✓ The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. **Notice** Instead of manually entering the fixed value: → In the respective input fields (1-4), alternatively click the *VARIABLE* tab and select the predefined variable from the selection element.

Screens/comments	Action steps
	**Positioning** The module *Positioning* is used to move to a defined position. → Select absolute (1) or relative (2) approach of the position. With absolute positioning, the target position is directly approached. With relative positioning, the specified distance is traveled, starting at the current position. Once the position is reached, the system switches to the next module in the program sequence. → Enter the velocity (3) [in mm/s] and the acceleration (4) [in mm/s^2] with which the system is to move. Reaching of the velocity with the selected acceleration is not checked. → Specify the maximum force (5) [in N]. → Enter the target position (6) [in mm]. If no force is specified, the maximum force of the system is automatically used. While approaching the target position (6), the current force is monitored by the force sensor. To prevent damage to the system or the tools: If the actual force exceeds the specified maximum value, the system is stopped immediately. The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. **Notice** Instead of manually entering the fixed value: → In the respective input fields (3-6), alternatively click the *VARIABLE* tab and select the predefined variable from the selection element.

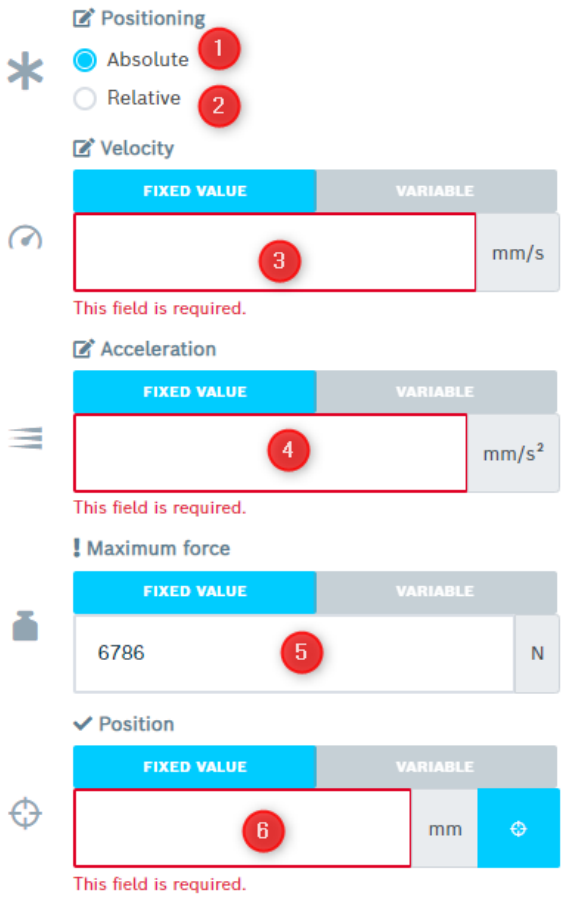
Screens/comments	Action steps
	**Join to force** The *Join to force* module moves the system until a defined target force is reached at the force sensor. The system will then switch to the next module in the program sequence. → Enter the velocity (1) [in mm/s] and the acceleration (2) [in mm/s^2] with which the system is to move. ✅ Reaching of the velocity with the selected acceleration is not checked. → If required, enter the maximum position (3) [in mm]. ✅ If no position is specified, the maximum position of the system is automatically used. ✅ While approaching the block force (4) (target force), the current position is monitored. ✅ To prevent damage to the system or the tools: If the actual position exceeds the specified maximum value, the system is stopped immediately. ✅ The exceeded position causes the program sequence to be aborted and the currently produced part to be rated NOK. → Enter the target force (4) [in N]. ✅ If the actual force is higher than the target force when entering the module, the system switches to the next module in the sequence. **Notice** Instead of manually entering the fixed value: → In the respective input fields (1-4), alternatively click the *VARIABLE* tab and select the predefined variable from the selection element. 💡 In addition, it is possible to adjust the performance of the component via parameter *VFR005 Minimal relative velocity* in the Hardware configuration.

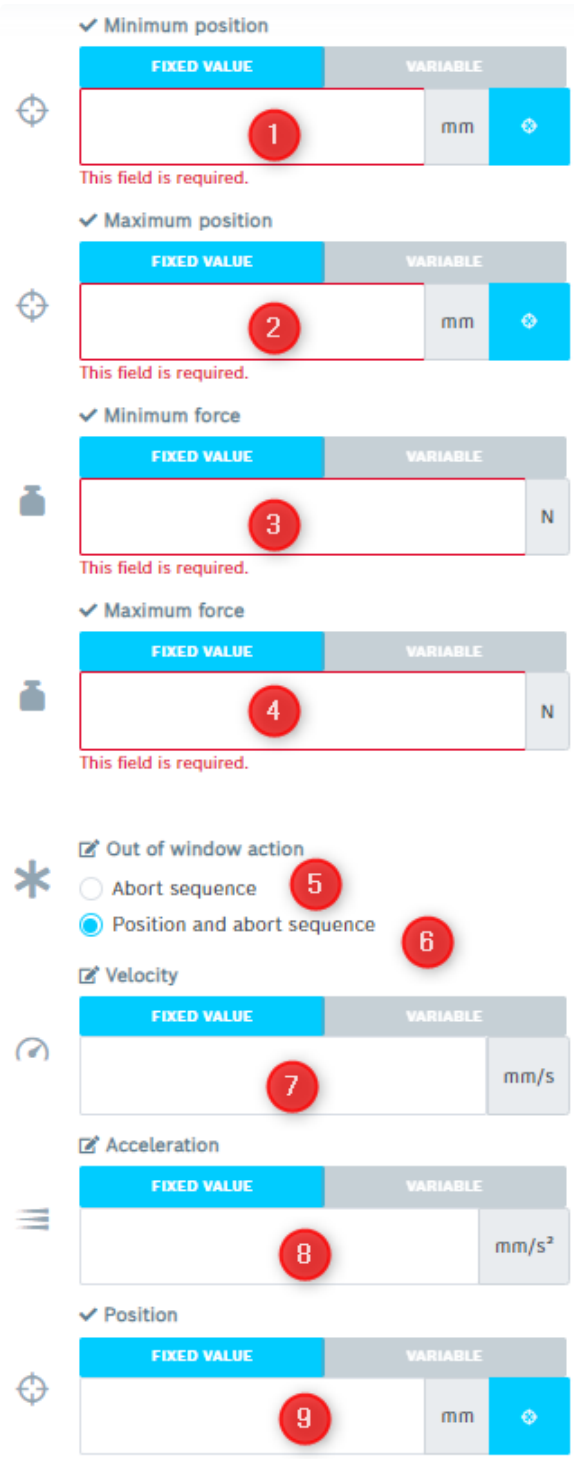
Screens/comments	Action steps
	NOTICE  **Exceedance of the target force by high kinetic energy** Excessively high joining speed may cause the target force to be exceeded due to the high kinetic energy. If the values are significantly exceeded, the error message *Error 41004 – Program error: Max force exceeded* is displayed (see chapter Error catalog). This may result in damage to the workpiece. → Reduce the speed. → If necessary, increase the maximum overload with parameter VFR002. **Notice** **No operating case without force sensor** An operating case without a force sensor is not permitted. → Add a customer-specific force sensor if the SFK was ordered without a force sensor. → See chapter Customized force sensor.

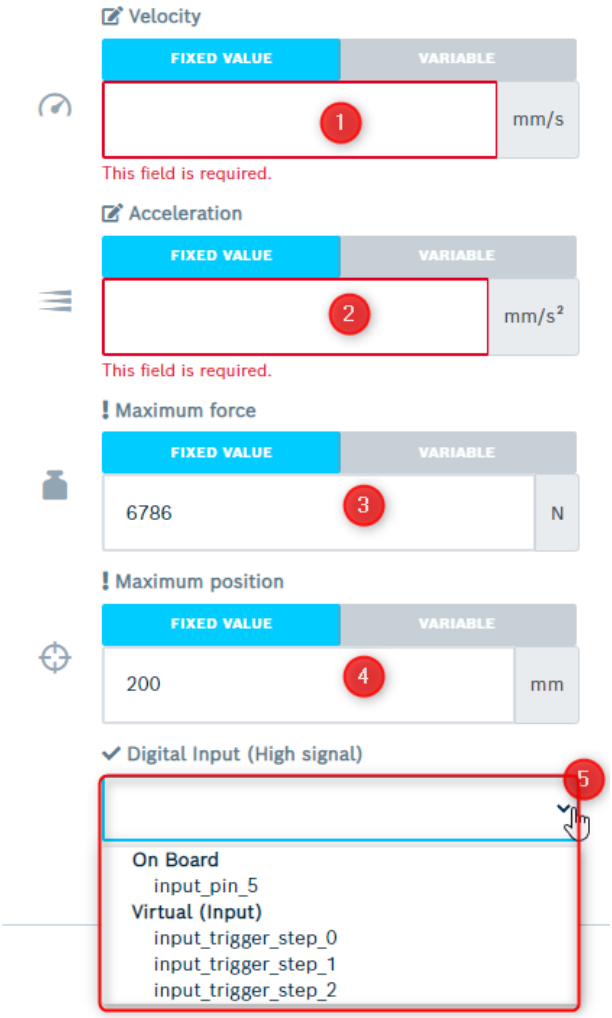
Screens/comments	Action steps
	**Force-controlled joining** The module *Force-controlled joining* moves the system to a defined target force at the force sensor until a defined time has elapsed or the defined input for "Continue Function Block" (4) is HIGH or TRUE. The system will then switch to the next module in the program sequence. → Enter the target force (1) [in N] and the maximum force (2) [in N]. ✓ When specifying fixed values: The system checks whether the maximum force value is greater than the target force value. When specifying variables: Make sure that the maximum force value is greater than the target force value. ✓ If no maximum force is specified, the maximum force of the system is automatically used.

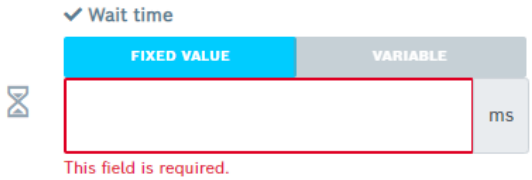
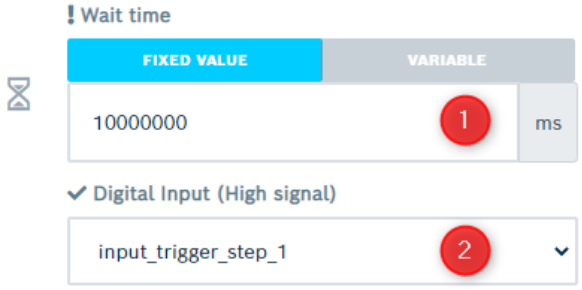
Screens/comments	Action steps
	NOTICE  **Exceedance of the target force by high kinetic energy** Excessively high joining speed may cause the target force to be exceeded due to the high kinetic energy. If the defined maximum force (2) is exceeded, the program will be aborted and error message 41004 is displayed (see chapter Error catalog). This may result in damage to the workpiece. → If necessary, increase the maximum force (2). → Reduce the control dynamics (5). → Reduce the maximum velocity (6). → Enter the hold time (3) [in s] (max. 600 seconds). ✓ If the hold time has expired, the program continues to the next module in the sequence. ✓ Once the hold time starts, the maximum time for reaching the target force (9) is no longer taken into account. → Optional: Select an Input for "Continue Function Block" (4). ✓ In the drop-down menu, you can choose from Integrated and Virtual Inputs. ✓ Only inputs that have previously been assigned an alias under Configuration → I/O Configuration will be displayed. ✓ If the defined input (4) is HIGH or TRUE upon entering the function block or during the function block's execution, the next function block will start immediately. → Specify the control dynamics (5) [in (mm/s)/kN]. ✓ The system checks whether the control dynamic is greater than 0 (zero).

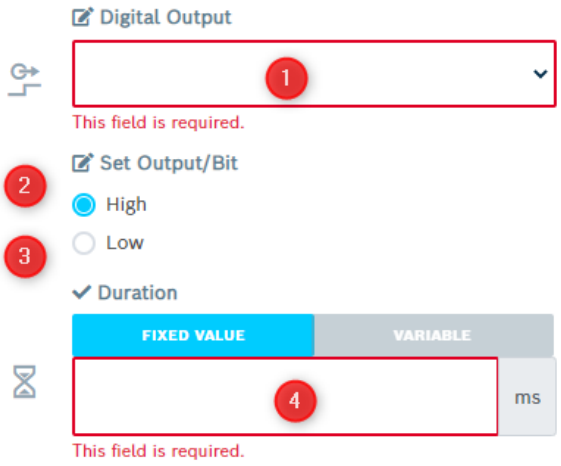
Screens/comments	Action steps
	Notice To accelerate the approach to the target force: → Increase the control dynamic in terms of magnitude. To reduce (over)oscillation: → Reduce the control dynamic in terms of magnitude. → Enter the maximum velocity (6) [in mm/s]. Reaching the velocity depends on the control dynamics and target force deviation in the process. → Enter the acceleration (7) [in mm/s^2]. → Specify the force tolerance for the start of the holding time (8) [in N]. As soon as the target force with the defined force tolerance has been reached once during module execution, the holding time starts. Leaving the tolerance range later has no influence. → Enter the maximum time for reaching the target force (9) [in s] in seconds. If the target force with the force tolerance is not reached within this time, the program sequence is aborted and the currently produced part is evaluated as NOK. If the actual force is within the tolerance range of the target force when entering the module, the holding time starts immediately. If no time is specified, a maximum time of 600 seconds is automatically used. → If required, enter the maximum position (10) [in mm]. If no position is specified, the maximum position of the system is automatically used. While approaching the target force (1), the current position is monitored. To prevent damage to the system or the tools: If the actual position exceeds the specified maximum value, the system is stopped as quickly as possible. Exceeding the position causes the program sequence to be aborted and the currently produced part is evaluated as NOK.

Screens/comments	Action steps
	Notice For more information about the force control feature, see the document "Smart Function Kit Pressing - Expert Note Force Control", document number R320103264.
	**Join to position** The module *Join to position* is used to join a product with a specific target position. → Select absolute (1) or relative (2) approach of the position. With absolute positioning, the target position is directly approached. With relative positioning, the specified positive distance is traveled, starting at the current position. For relative negative positioning, use the "Positioning" function block. Once the position is reached, the system switches to the next module in the program sequence. → Enter the velocity (3) [in mm/s] and the acceleration (4) [in mm/s^2] with which the system is to move. Reaching of the velocity with the selected acceleration is not checked. → Specify the maximum force (5) [in N]. → Enter the target position (6) [in mm]. If no force is specified, the maximum force of the system is automatically used. While approaching the target position (6), the current force is monitored by the force sensor. To prevent damage to the system or the tools: If the actual force exceeds the specified maximum value, the system is stopped immediately. The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. **Notice** Instead of manually entering the fixed value: → In the respective input fields (3-6), alternatively click the *VARIABLE* tab and select the predefined variable from the selection element.

Screens/comments	Action steps
	**Check force/position target window** The module *Check force/position target window* checks the actual values of force and displacement against specified target values during the sequence execution. The actual values are determined at the same time when leaving the previous module, taking into account the system tolerances. If the actual values are within the target value range, the system switches to the next module in the program sequence. If at least one target value is violated, the program sequence is aborted with an error message (error 41010 – 41024, see chapter Error catalog). → Enter the minimum position (1) [in mm]. → Enter the maximum position (2) [in mm]. → Specify the minimum force (3) [in N]. → Specify the maximum force (4) [in N]. → Select an action to be executed in case the value is out of target window: • Abort sequence (5) • Position and abort sequence (6) With optimum positioning: → Enter the speed (7) [in mm/s] and the acceleration (8) [in mm/s^2] with which the system is to move. ✓ Reaching of the velocity with the selected acceleration is not checked. ✓ The target position (9) is directly approached. **Notice** When setting the window size, system tolerances must be observed.

Screens/comments	Action steps
	**Move to signal** The module *Move to signal* moves the system until a specific digital or virtual input delivers a high signal. The system will then switch to the next module in the program sequence. The module enables integration of an external sensor and probing of a workpiece surface with this sensor. → For example to integrate additional process check options or measuring systems: Control the module via a higher-level control using virtual inputs. → Enter the velocity (1) [in mm/s] and the acceleration (2) [in mm/s^2] with which the system is to move. ✓ Reaching of the velocity with the selected acceleration is not checked. → Enter the maximum position (4) [in mm]. ✓ If no position is specified, the maximum position of the system is automatically used. → Specify the maximum force (3) [in N]. ✓ If no force is specified, the maximum force of the system is automatically used. ✓ While approaching the target position, the current force is monitored by the force sensor. ✓ To prevent damage to the system or the tools: If the actual force exceeds the specified maximum value, the system is stopped immediately. ✓ The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. **Notice** Instead of manually entering the fixed value: → In the respective input fields (1-4), alternatively click the *VARIABLE* tab and select the predefined variable from the selection element. → From the selection element (5) select an input that is to deliver a high signal. Define the inputs in the I/O configuration before using them.

Screens/comments	Action steps
	Notice Input signals must be present for at least 40 ms.
	**Delay** The module *Delay* causes the system to wait for a defined time before executing the next program step. → Enter a waiting time [in ms] (max. 600000 milliseconds). **Notice** Instead of manually entering the fixed value: → Click the *VARIABLE* tab and select the predefined variable from the selection element. The system will then switch to the next module in the program sequence.
	**Wait for signal** The module *Wait for signal* causes the system to wait until a defined digital or virtual input has a high signal. The system will then switch to the next module in the program sequence. The command enables timely coordination of the system with other machine parts, (e.g. clamping element). → Enter a maximum time (1) [in ms]. **Notice** Instead of manually entering the fixed value: → Click the *VARIABLE* tab and select the predefined variable from the selection element. Upon expiry of this period, the program sequence is aborted. The currently produced part is evaluated as NOK. → From the selection element (2) select an input that is to deliver a high signal. Define the inputs in the I/O configuration before using them.

Screens/comments	Action steps
	**Set output** The module *Set output* sets a defined digital or virtual output to the defined level or state for a defined period of time. With this command, a signal can be sent to another part of the machine (e.g. a conveyor belt, signal lamp). → From the selection element (1), select an output that is to be set. Define the outputs in the I/O configuration before using them. → Select whether the bit is to be set *high* (2) or *low* (3). → Specify the time period [in ms] (4) for which the output should be set to the defined level. **Notice** If the program is canceled during the specified period, the value of the output is not reset. ✅ After the time period has elapsed, the output is switched. ✅ The system will then switch to the next module in the program sequence. → To set the signal for an indefinite time: Set the value to 0 ms. ✅ The system will then switch to the next module in the program sequence. The output remains in the defined state **Notice** Instead of manually entering the fixed value: → Click the *VARIABLE* tab and select the predefined variable from the selection element.

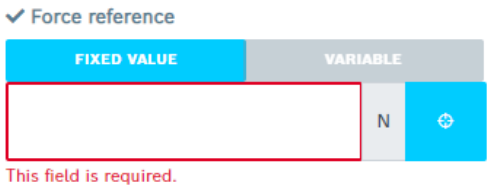
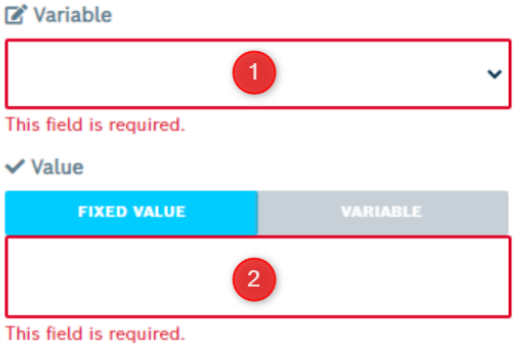
Screens/comments	Action steps
	**Tare** The module *Tare* tares the force sensor to any value (e.g. for compensating a tool weight). Taring applies to all force values (e.g. in other programs, curve recording, quality evaluation). → Enter the force value in N to which the sensor is to be tared. **Notice** Instead of manually entering the fixed value: → Click the *VARIABLE* tab and select the predefined variable from the selection element.
	**Set variable** The module *Set variable* sets a defined variable to a fixed value or the value of another variable. → From the selection element (1), select a predefined variable that is to be changed. → Enter a fixed value (2) with which the variable is to be redefined. **Notice** Instead of manually entering the fixed value: → Click the *VARIABLE* tab and select the predefined variable from the selection element. **Notice** When the program is started, all used variables are saved in the data base for documentation (variable table). Changes to variable values made by the command *Set variable* are only saved in the variable table after the program has been restarted (see chapter Variables configuration).

Table 6 Library

Notice
Exceeding limit values (e.g. maximum force, maximum position or maximum time) during the process leads to a NOK result.

8.3 Slur groups

A slur group comprises at least two consecutive modules of a sequence. The modules of this group are considered as a single unit.

The movements of these modules are consolidated and their cycle time is automatically optimized. The modules are provided with the addition “Slur”.

Effects

Constituents of a slur group perform continuous movement without standstill upon module change. Instead of a short standstill between individual modules, the system performs a brief transition. During this period, the system performs a transition from the previously applied speed to the speed of the following module. The speed adjustment is carried out with the acceleration value of the target module and is achieved when it enters it.

This results in reduced cycle times.

Requirements

During sequence processing, the modules used and their respective parameters are constantly analyzed. Various requirements need to be met to make slurring possible.

The following modules can initiate slurring. They are marked with “Slur (Start)” in the sequence:

- Positioning
- Join to position

The following modules can be used as intermediate steps as often as required. Their respective target positions must always be between the start and end positions of the slur group and must always be consecutive. They are marked with the addition “Slur (Intermediate step)”:

- Positioning
- Join to position

The slur group must end with one of the following modules. It is marked with “Slur (End)”:

- Positioning
- Join to position
- Join to force
- Force-controlled joining
- Move to signal

💡 In addition, the absolute position for the first module of the slur group must be known.

→ Make sure that the target velocity can be reached by sufficient acceleration.

💡 If this is not observed, the step may be switched prematurely and the curve may be evaluated incorrectly.

💡 Position values must be specified by fixed values.

Interrupting a slur group

A slur group typically leads to an improvement in cycle time. In some applications, short-term standstill of the axis is desired. In such cases, the groups can be broken up:

→ Use the *Initial position* module instead of the modules *Join to position* or *Position*.

💡 This module is never part of a slur group.

→ Add one of the following modules to the position that is to be interrupted:

- Delay (wait time 0 ms)
- Tare
- Set output

9 Reference curves

In the *Reference curves* area, you can record one or more reference curves for a program. A reference curve is a force-displacement curve that is targetedly recorded upon set-up of the joining process. Recording is mostly performed with sample parts.

Creation of several reference curves is an optional step and not required for the simple program sequence.

At least one reference curve is required for defining the evaluation elements and setting up the automatic quality evaluation.

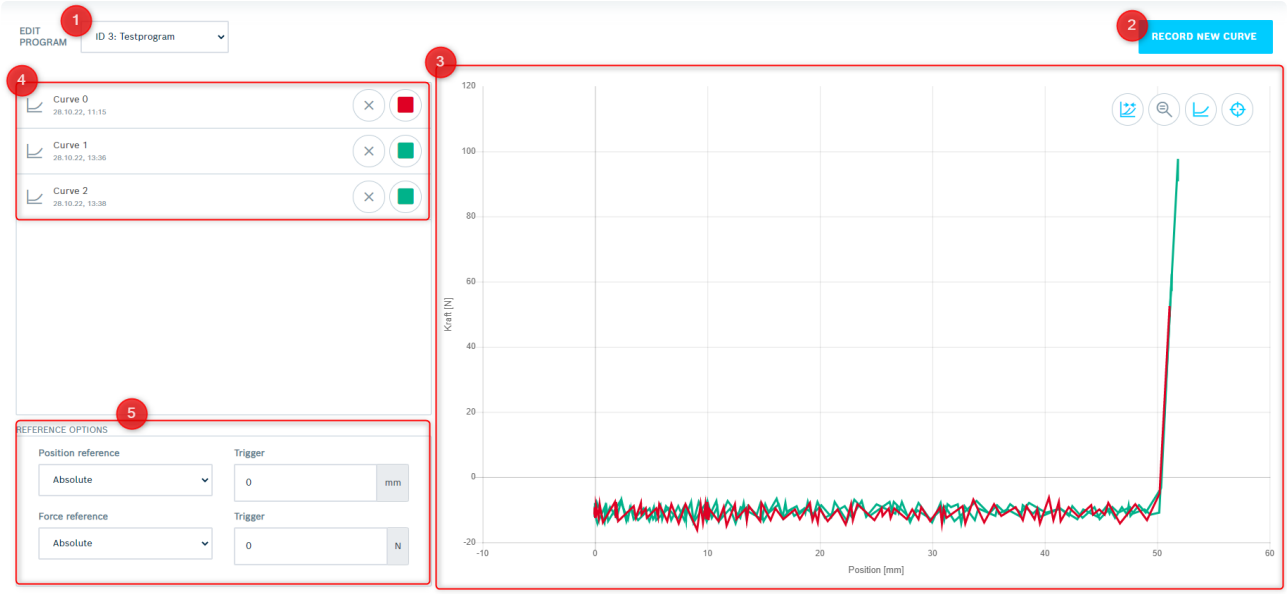


Figure 35 Reference curves

Position	Designation	Description
1	Selection element program	Drop down box with available programs → To record a curve or view curves that have already been created: Select a program from the selection element.
2	*RECORD NEW CURVE*	Button for recording the force-displacement curve ◊ For more information, see chapter Recording.
3	Curve diagram	Diagram for displaying the force-displacement curve Recording of the curve is displayed live. If several curves are recorded at the same time, the curves are displayed in the curve diagram as an array of curves.
4	Curves list	List of all recorded curves ◊ For more information, see chapter Curves list.

Position	Designation	Description
5	Reference options	Setting the dynamic displacement of the force-displacement curve ◊ For more information, see chapter Reference options.

Table 7 Reference curves

9.1 Recording

Recording of a reference curve can only be performed if no program for parts production is currently running in the system.

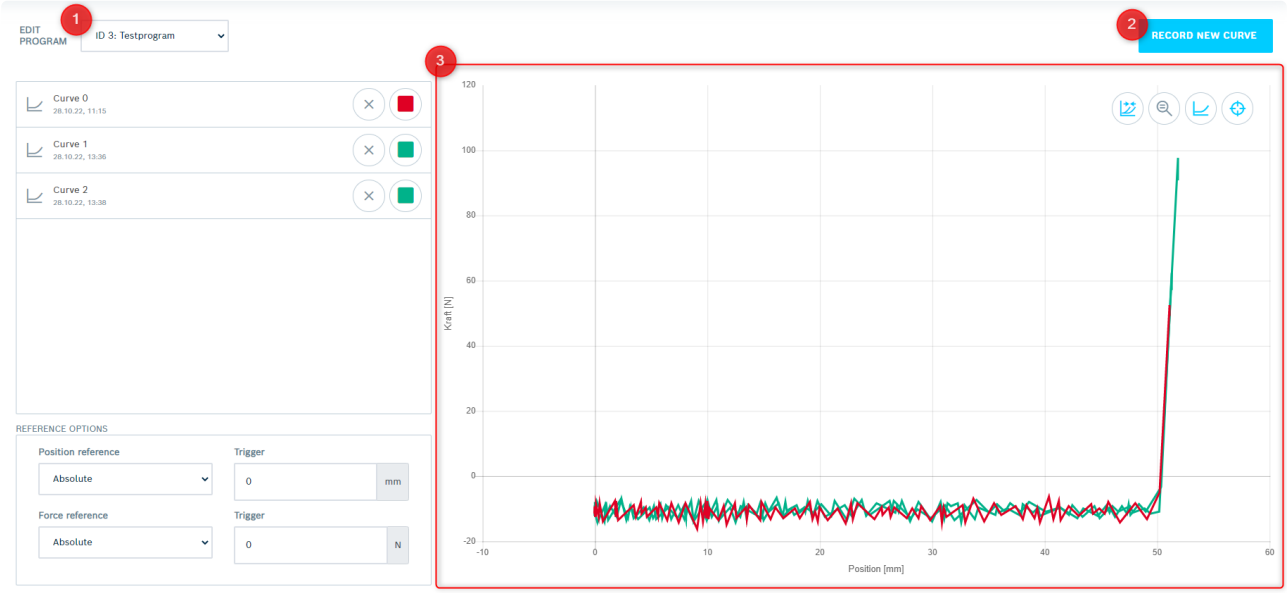


Figure 36 Recording

→ From the selection element (1), select a program.

→ Click the *RECORD NEW CURVE* (2) button.

✅ The program runs once and the force-displacement curve is displayed live in the curve diagram (3).

✅ If several curves are recorded at the same time, the curves are displayed in the curve diagram (3) as an array of curves.

💡 Recording of reference curves is not rated as production. The production data in the data base and the production statistics will not be affected by this.

NOTICE

Misinterpretation when setting evaluation elements

Any reference curves that are already present are not automatically deleted when modifications are made to the program. This may result in machine damage.

→ Manually check the sequence of programs and evaluation properties if they have been modified.

→ If required, re-record the sequences.

9.2 Curves list

Once recording of a curve is completed, the curve is displayed as an entry in the table view.

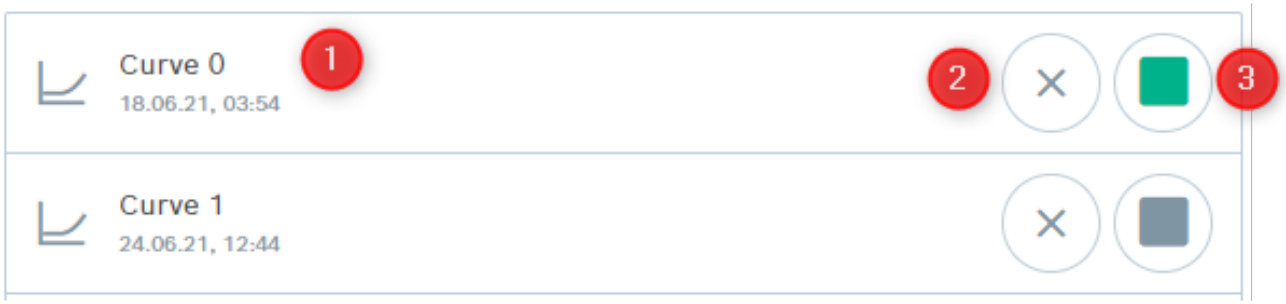


Figure 37 Curves list

Position	Designation	Description
1	Curve name (Renaming)	The reference curves are given an automatically generated name. → To make the name editable: Click on the name. → Enter the desired curve name and confirm your input by pressing *Enter* or by exiting the respective input field. 💡 Defining individual names can make assigning reference curves to the respective sample parts easier.
2	Delete	Is used to delete the reference curve. → Click the *Delete* button. ✅ The reference curve will be deleted from the table. ✅ If a reference curve is deleted, it is removed from the data base and cannot be restored.

Position	Designation	Description
3	Evaluation	→ Click the *Evaluate* (3) button. ✅ A selection element is displayed. → Select one of the three marking options: • Inactive • Good reference • Bad reference ✅ Curves marked with Good reference are shown in the diagram in green and Bad reference in red. ✅ Curves that are marked inactive are hidden in the diagram and not considered in the evaluation. → To have the curve displayed in the diagram again: Change the marking of an inactive curve to Good reference or Bad reference.

Table 8 Reference curves

9.3 Reference options

Under Reference options, the dynamic displacement of the force-displacement curve can be set.

9.3.1 Position reference (X-axis)

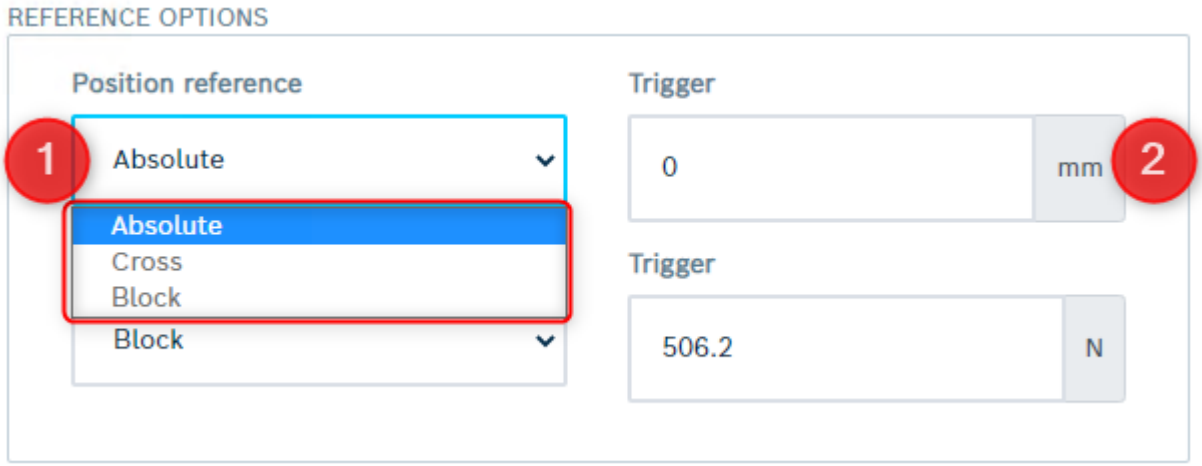


Figure 38 Position reference (X-axis)

Absolute

The *Absolute* position reference setting is the default.

With *Absolute* position reference, the X-coordinates of all evaluation elements are predefined. The positions of the evaluation elements in the direction of the X-axis are at fixed distances from the coordinate origin.

→ Select the *Absolute* position reference in the selection element (1).

✅ The distance is defined when creating the evaluation element.

💡 The *trigger* (2) cannot be changed manually.

Cross

With the *Cross* position reference setting, all evaluation elements are dynamically shifted along the Y-axis. The positions of the evaluation elements along the Y-axis are at fixed distances from a trigger point.

The distance from the trigger point is defined when creating the evaluation element.

→ Select the *Cross* position reference in the selection element (1).

→ Enter the position value in the *Trigger* input field (2).

✓ The first reference curve marked as good reference (see chapter Evaluation) is considered the reference curve.

✓ The selected trigger position is marked on the X-axis and represented in the plot by means of a vertical dashed line. The force of the reference curve applied there is the reference force.

✓ All subsequent curves are shifted along the Y-axis so that the force applied at the trigger position is equal to the reference force. This shift is performed graphically. The shift offset is saved together with the data set.

Block

With the *Block* position reference setting, all evaluation elements are dynamically shifted along the Y-axis. The positions of the evaluation elements along the Y-axis are at fixed distances from the block point. The distance from the block point is defined when creating the evaluation element.

→ Select the *Block* position reference in the selection element (1).

✓ This has the following effect: The first reference curve marked as good reference (see chapter Evaluation) is considered the reference curve.

✓ The block position (maximum position) of the reference curve is marked on the X-axis and represented in the plot by means of a vertical dashed line. The force applied there is the reference force.

✓ All subsequent curves are shifted along the Y-axis so that the force applied at their maximum position is equal to the reference force. This shift is performed graphically. The shift offset is saved together with the data set.

9.3.2 Force reference (Y-axis)

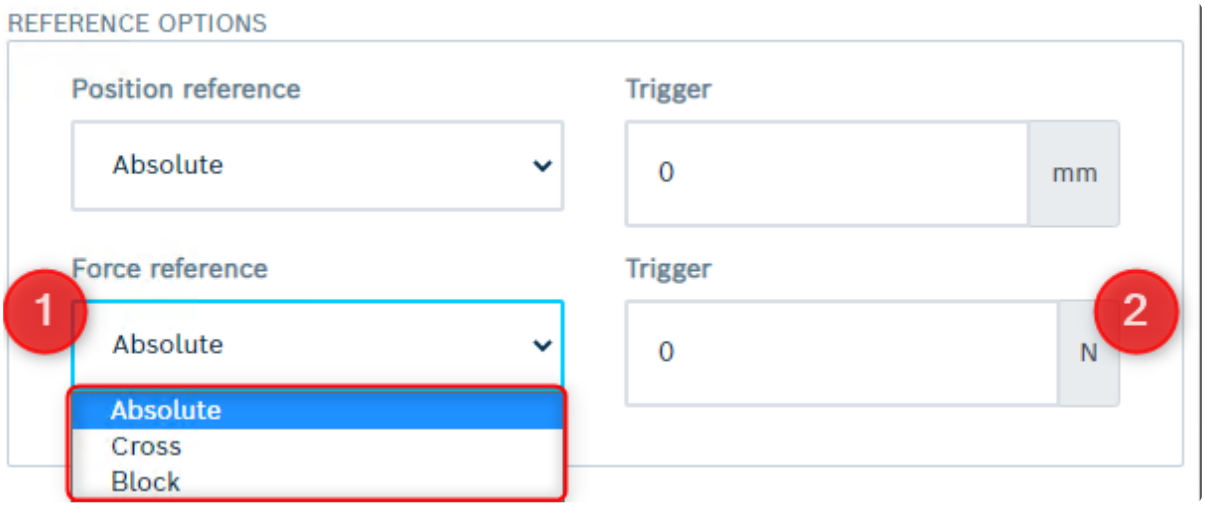


Figure 39 Force reference (Y-axis)

Absolute

The *Absolute* force reference setting is the default.

With *Absolute* force reference, the Y-coordinates of all evaluation elements are predefined. The positions of the evaluation elements in the direction of the Y-axis are at fixed distances from the coordinate origin.

→ Select the *Absolute* force reference in the selection element (1).

✅ The distance is defined when creating the evaluation element.

💡 The *trigger* (2) cannot be changed manually.

Cross

With the *Cross* force reference setting, all evaluation elements are dynamically shifted along the X-axis. The positions of the evaluation elements along the X-axis are at fixed distances from a trigger point.

The distance from the trigger point is defined when creating the evaluation element.

→ Select the *Cross* force reference in the selection element (1).

→ Enter the position value in the *Trigger* input field (2).

✅ The first reference curve marked as good reference (see chapter Evaluation) is considered the reference curve.

✅ The selected trigger position is marked on the Y-axis and represented in the plot by means of a vertical dashed line. The force of the reference curve applied there is the reference force.

✅ All subsequent curves are shifted along the X-axis so that the force applied at the trigger position is equal to the reference force. This shift is performed graphically – The shift offset is saved together with the data set.

Block

With the *Block* force reference setting, all evaluation elements are dynamically shifted along the X-axis. The positions of the evaluation elements along the X-axis are at fixed distances from the block point. The distance from the block point is defined when creating the evaluation element.

→ Select the *Block* force reference in the selection element (1).

✅ The first reference curve marked as good reference (see chapter Evaluation) is considered the reference curve.

✅ The block position (Position Maximum) of the reference curve is marked on the Y-axis and represented in the plot by means of a vertical dashed line. The force applied there is the reference force.

✅ All subsequent curves are shifted along the X-axis so that the force applied at their maximum position is equal to the reference force. This shift is performed graphically. The shift offset is saved together with the data set.

9.3.3 Diagram representation

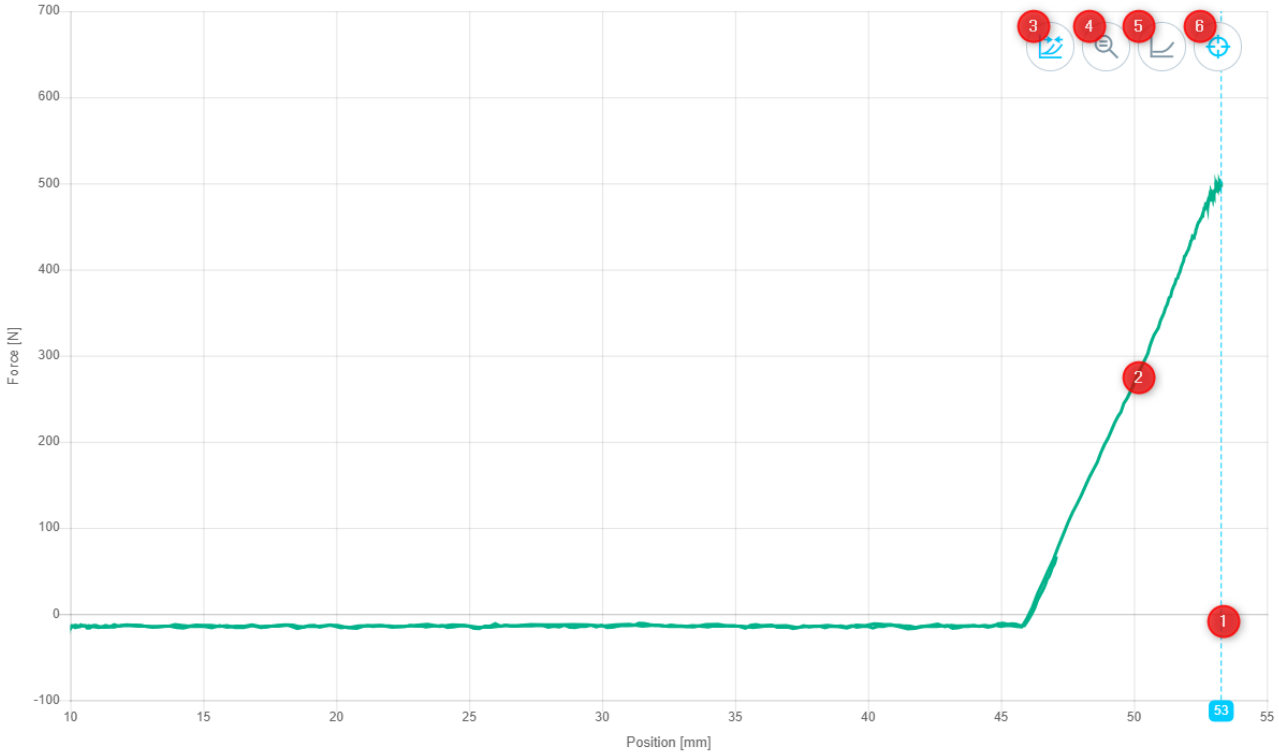
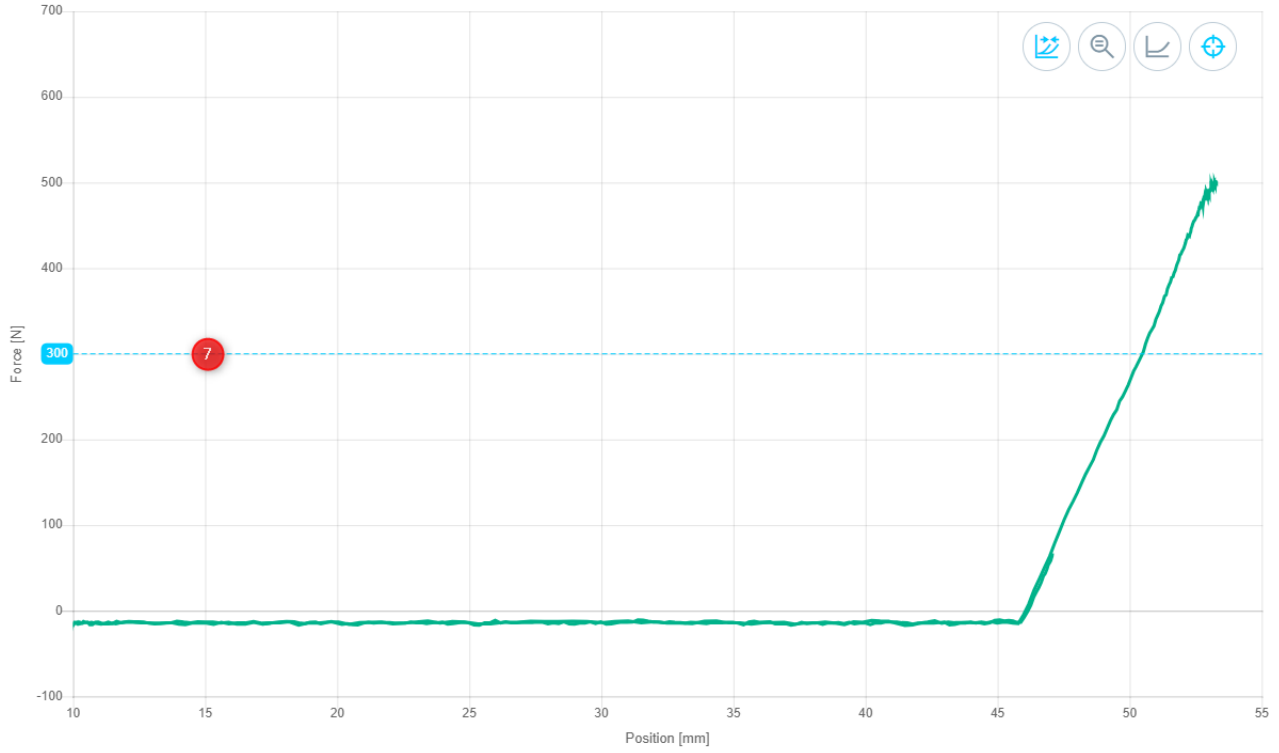


Figure 40 Diagram representation



Position	Designation	Description
1	Position reference	Depending on the selected position reference (cross or block), the block position (Position Maximum) of the reference curve or the trigger position is marked on the X-axis and represented in the plot by means of a vertical dashed line.
2	Curve representation	If the reference option is set to something other than *Absolute* position reference or *Absolute* force reference, representation of the curves in the diagram will change. The changed representation takes into account dynamic positioning by displaying the reference curves with an appropriate shift.
3	Curve consolidation	Button to show/hide the customized curve representation in the diagram
4	Zoom	Button to reset the zoom to the default view
5	Show and hide curve details	Button for displaying the x and y values of the entire curve
6	Show time axis/position axis	Button to display the X-axis as time or position axis
7	Force reference	Depending on the selected force reference (cross or block), the block position (Position Maximum) of the reference curve or the trigger position is marked on the Y-axis and represented in the plot by means of a vertical dashed line.

Table 9 Diagram representation

10 Evaluation

In the *evaluation elements* area you can define evaluation elements for every program. The elements are used for automatic quality assessment and based on their evaluation, the system determines whether a produced part is OK or NOK after the process has been completed.

Notice
Not only evaluation elements can lead to a NOK result, but also the exceeding of limit values (e.g. maximum force, maximum position or maximum time) during the process.
An OK evaluation is only made if the program has been run through completely (i.e. no termination due to limit values, drive errors, CANCEL command) and all criteria of all evaluation elements are met.

Notice
Evaluation elements can only be created in the force-displacement representation of the diagram.

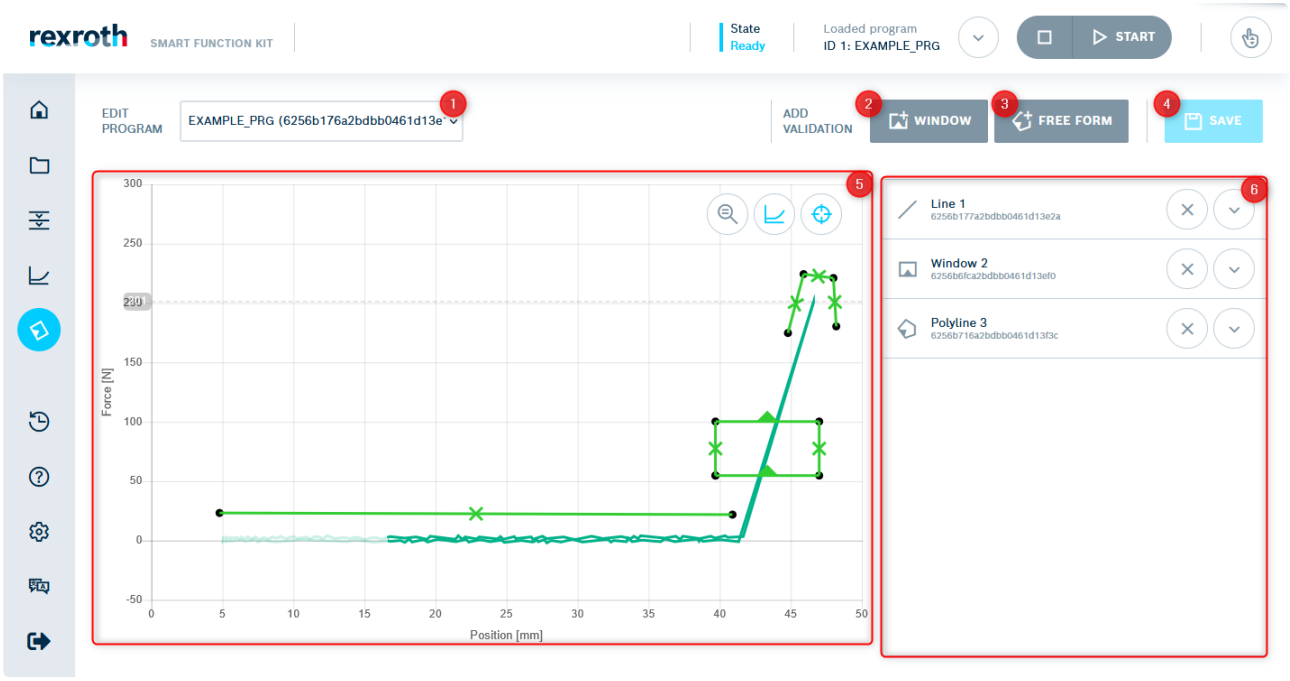
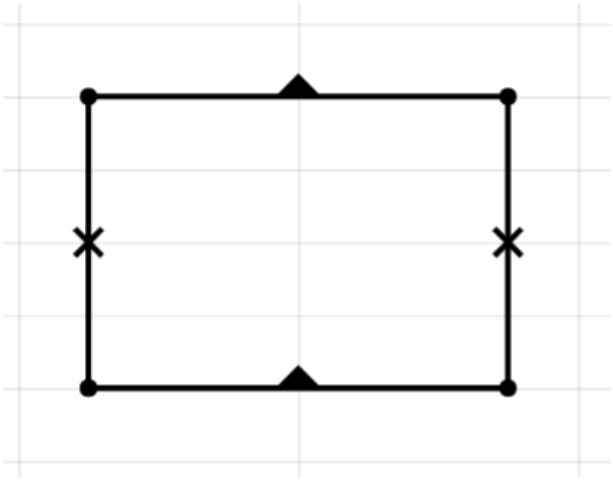
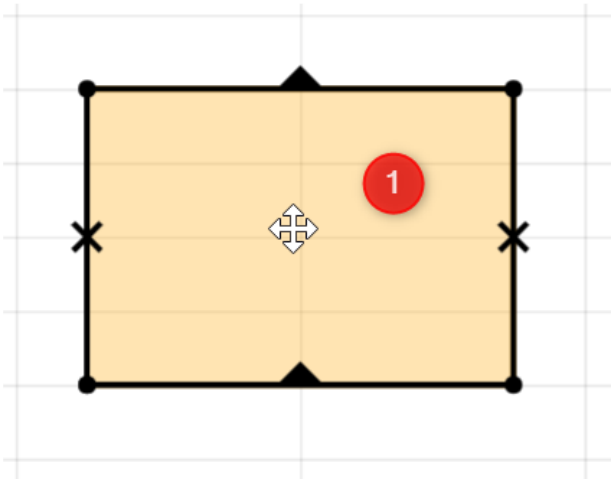


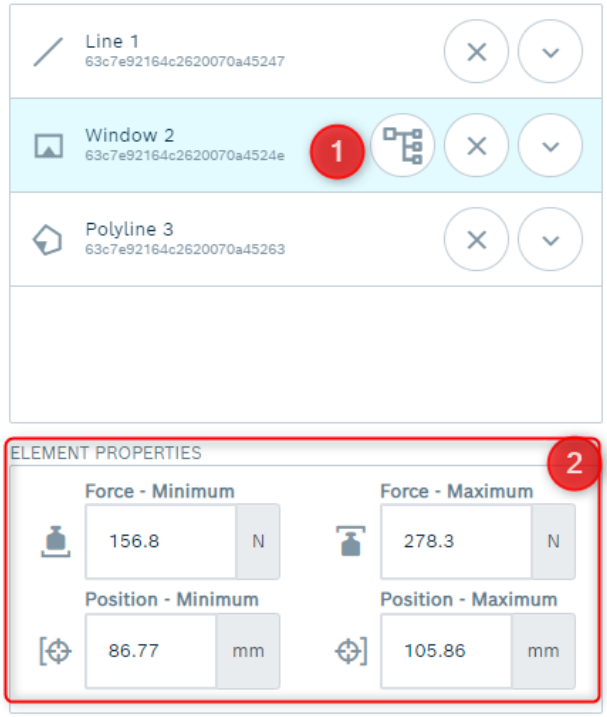
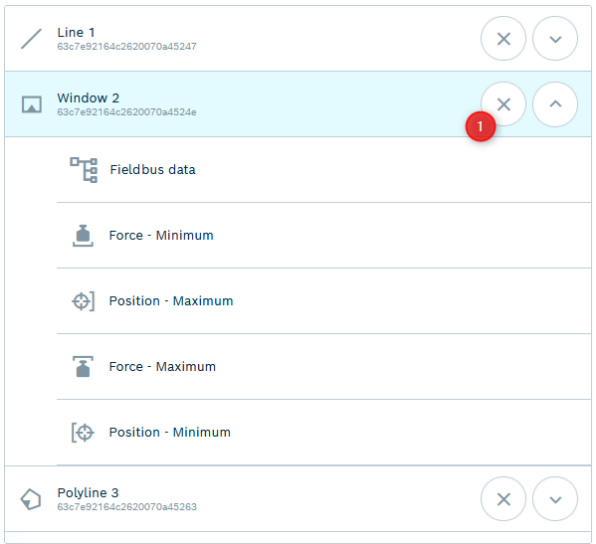
Figure 41 Reference curves

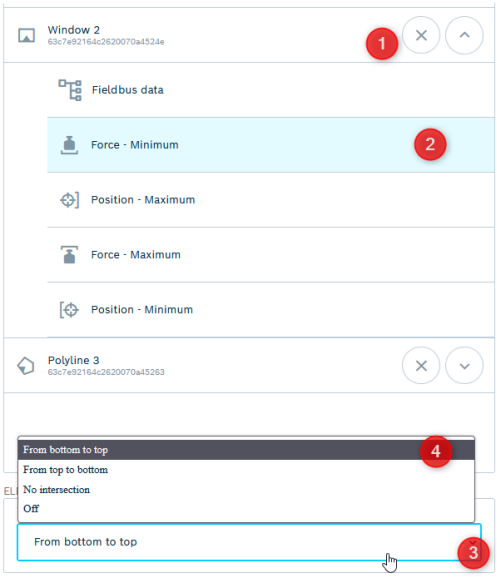
Position	Designation	Description
1	Selection element program	Selection element with available programs → To evaluate a curve: Select a program from the selection element.
2	*WINDOW*	Button to create an evaluation window

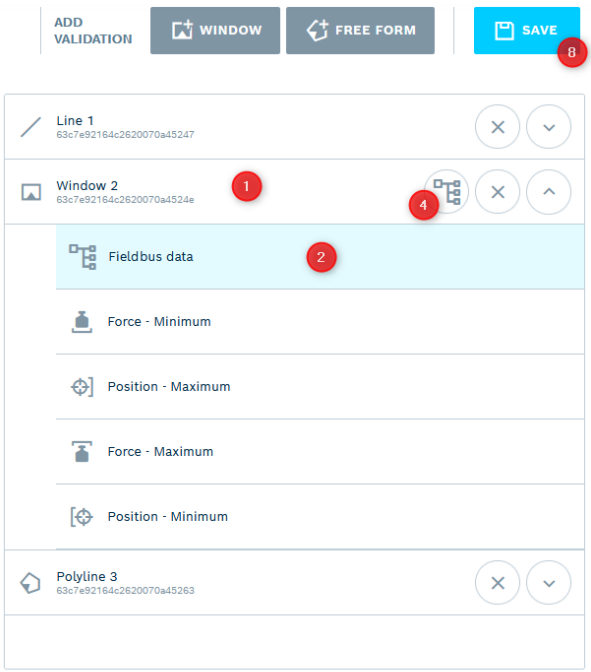
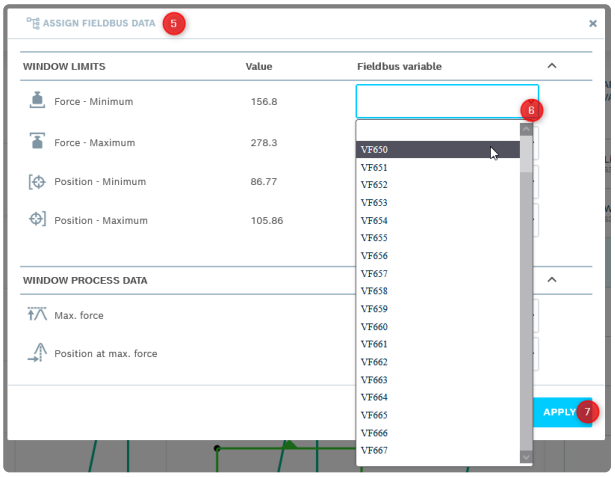
Position	Designation	Description
3	*FREE FORM*	Button to create an evaluation line or an evaluation polygon (e.g. an envelope curve)
4	*SAVE*	Button to save: • Any changes to evaluation windows and evaluation elements are applied. • The evaluation of the changed evaluation windows and evaluation elements is updated. **Notice** Saving is only possible if: • There are any (unsaved) changes to windows and evaluation elements. • Rules are not violated (indicated by red exclamation mark). • *FREE FORM* (3) is not active. • The force-displacement representation is displayed. **Notice** When you leave the page and changes have not been saved, a message box appears asking whether you want to discard the changes: → Confirm to irrevocably discard the changes. → Cancel the process to keep the editor page open.
5	Diagram view	Drawing area for evaluation elements **Notice** In the force-time view, evaluation elements cannot be displayed, created or edited.
6	List of evaluation elements	• List of the created evaluation elements. • Determination of the position of an evaluation element (window, line, polygon).

Table 10 Evaluation

Screens/comments	Action steps
	**Creating evaluation windows** → Click the *WINDOW* (2) button. ✅ A new evaluation window with standard settings is created. It is placed in the center of the diagram (5).
	**Moving evaluation windows** → To move the evaluation window: Click in the center of the evaluation element (1). ✅ The evaluation element is marked and a cross-hair cursor is displayed. → With the cross-hair cursor click on the target position in the drawing area. → To move a line: Click the line. ✅ The selected line is marked yellow and a cross-hair cursor is displayed. → With the cross-hair cursor click on the target position in the drawing area.

Screens/comments	Action steps
	**Detailed positioning of evaluation windows** The position of the sides of an evaluation window can be defined in the list of evaluation elements. → Click the respective evaluation window (1). → Enter the minimum and maximum values of force and position into the input fields (2). The attributes of the evaluation window can also be customized via the API interface (see chapter Window evaluation element).
	**Deleting evaluation windows** → To remove an evaluation window from the list of evaluation elements: Click the *Delete* button (1). 💡 An evaluation window can only be completely deleted. Deleting individual components (e.g. min. force) is not possible.

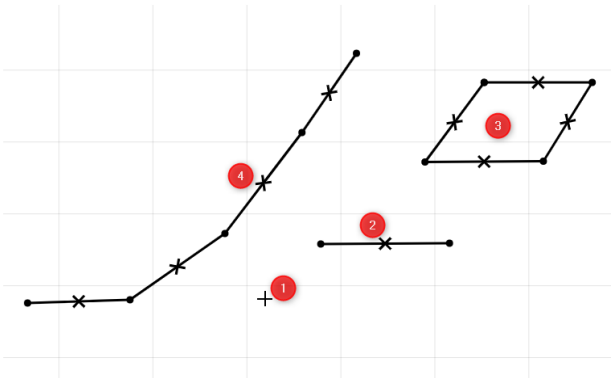
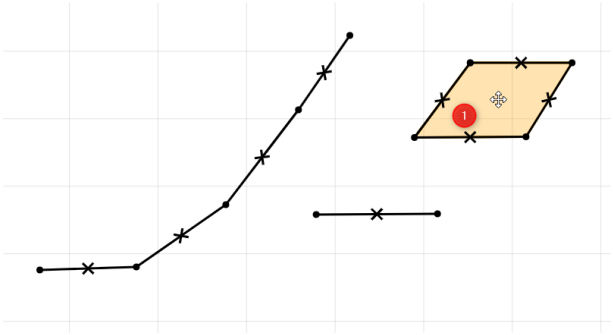
Screens/comments	Action steps
	**Properties of the sides of an evaluation window** An evaluation window is used for the automatic quality evaluation of the produced parts. For this purpose, the sides of the evaluation window are given the following properties: • From bottom to top/From left to right • From top to bottom/From right to left • No crossing • OFF (no evaluation of this side) → Adjust the properties of the side directly in the diagram. → Click on the middle of the side. ✓ Every mouse click moves the selection to the next property in the menu. → Alternatively, open the evaluation window in the list (1) and select the corresponding side (2). → Open the selection element (3). → Select a property (4).

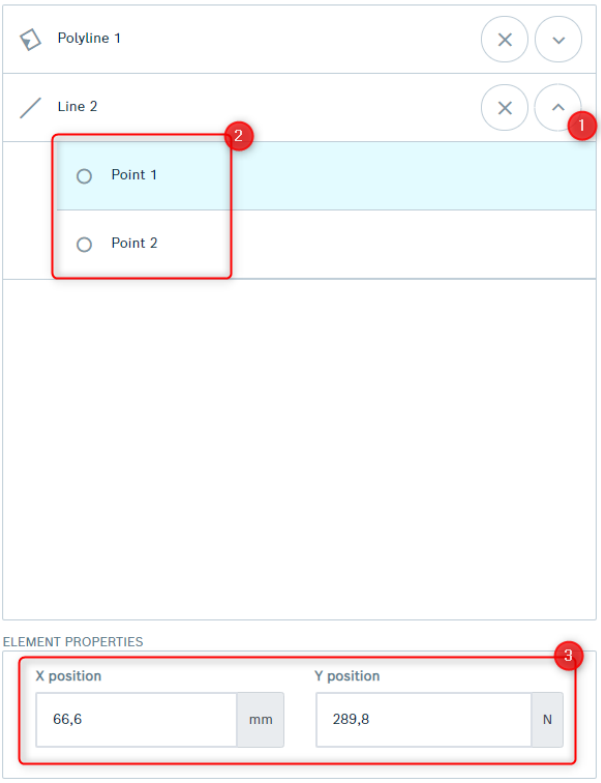
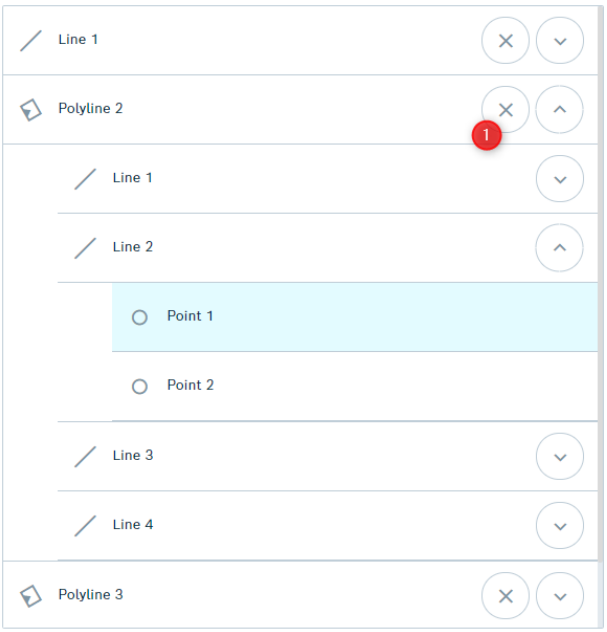
Screens/comments	Action steps
	**Fieldbus data allocation for evaluation windows** **Notice** The fieldbus data allocation is only available window-type evaluation elements. Various window limit values and window process data can be made available via the fieldbus: • Window limit values ◦ Force minimum ◦ Force maximum ◦ Position minimum ◦ Position maximum • Window process data ◦ Max. force: The maximum force that has been reached in the width of the window. The value can also be outside the window if no measuring point is within the window. ◦ Position at max. force: The position where the maximum force in the width of the window occurred. The process data and window limit values to be transmitted can be set separately for each evaluation window. → Open the desired evaluation window (1) and select the “Fieldbus data” (2) property.
	**Notice** The *ASSIGN FIELDBUS DATA* button (3) is only selectable if a fieldbus log for the system has been selected in the settings. In addition, the fieldbus icon (4) also indicates the availability of the function. → Click the *ASSIGN FIELDBUS DATA* button (3) or on the fieldbus icon (4) to open the pop-up window (5). → Open the selection element of the desired window value (6) and select the desired VF variable.

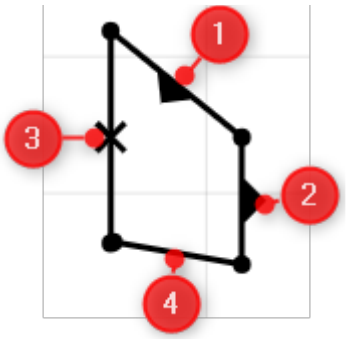
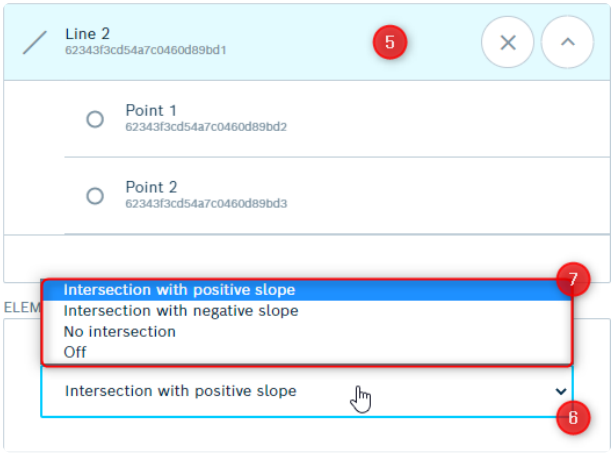
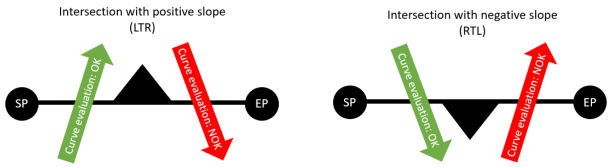
Screens/comments	Action steps
	Notice VF variables used in the current pop-up window are grayed out for all other window values. VF variables that have already been assigned to other evaluation windows are not displayed in the selection element. → Once you have completed the configuration of the fieldbus data of the evaluation window: Click the *APPLY button* (7). → To save the changes: Click the *SAVE button* (8). **Data availability** At program start, all VF variables in the range from 650 to 799 and the process variables for the last press curve VF600 to VF602 (max. position, max. force and cycle time) are set to the initial value 999999999. As soon as the evaluation of the program run is completed (see chapter Notifications bit 4 and bit 5), the data of the process and the evaluation windows are available. If, despite the completed evaluation (OK/NOK result is available), the initial value -999999999 for variables in the range 650 to 799 is still read out, the cause may be that the force-path curve has not reached the range of the window. **Notice** When the initial value of -999999999 is retrieved using the Fieldbus command "Read Variable" (Command 14, see chapter Requests and responses), it will be read as -2147483648.

Screens/comments	Action steps
	After saving, the diagram displays a live preview of the evaluation. All curves (1) rated good (marked green) are considered in the evaluation. The evaluation result is displayed by an appropriate color for every line. • Red line (2): The progress of the curves rated good violates the criterion. • Green line (3): The criterion for all curves rated good is valid. The curves rated bad (4) are not included in the evaluation and are only for visual orientation. For curves with bad evaluation, no evaluation (5) is displayed. **Notice** A live evaluation will always use the current program and its properties with regard to the evaluation. The reference curves which do not correspond to this configuration may cause unexpected behavior in the representation.

Table 11 Window evaluation element

Screens/comments	Action steps
	**Creating evaluation lines or polygons** → Click on *FREE FORM*. ✅ The drawing area of the diagram view is activated. → Click on any point in the diagram. ✅ A new point (1) is created. → To create another point: Click somewhere else. ✅ Points created one after the other are automatically connected by a line (2). 💡 Any number of points can be set when defining evaluation elements. → To draw a closed polygon (3): Place the end point on the start point. 💡 Polygons that are not closed can also be used as valid evaluation elements. → To finish the drawing process, there are 3 options: • Click the *FINISH FORM button*. • Click the complete drawing Right-click on the drawing area. • Draw a closed polygon. → To add another evaluation line or an evaluation polygon: Click on *FREE FORM*. ✅ The drawing mode is activated again.
	**Moving evaluation elements** → To move the evaluation element: Click in the center of the evaluation element (1). ✅ The evaluation element is marked and a cross-hair cursor is displayed. → With the cross-hair cursor click on the target position in the drawing area. → To move a line or point: Click the line or point. ✅ The selected line or point is marked yellow and a cross-hair cursor is displayed. → With the cross-hair cursor click on the target position in the drawing area.

Screens/comments	Action steps
	**Detailed positioning of evaluation line or polygon** The position of the points of an evaluation line or an evaluation polygon can be specified in the list of evaluation elements. → Open the menu of the corresponding evaluation element (1). → Mark the individual points (2). → Enter the coordinates of the points in the input fields (3).
	**Deleting evaluation lines or polygons** → To remove an evaluation line or evaluation polygon from the list of evaluation elements: Click the *Delete* button (1). 💡 An evaluation line or polygon can only be completely deleted. Deleting individual components (e.g. line of a polygon) is not possible.

Screens/comments	Action steps
 	**Properties of an evaluation line or an evaluation polygon** An evaluation line or an evaluation polygon is used for the automatic quality evaluation of the produced parts. For this purpose, the evaluation line and/or the lines of the evaluation polygon are given one of the following properties: • Crossing with positive lead (1). • Crossing with negative lead (2). • No crossing (3). • OFF (no evaluation of this line) (4). Each line has a starting point (SP) and an end point (EP). The crossing of the line by the curve (example see figure Reference curves) is defined as follows:  The force-displacement curve of a produced part is automatically checked for compliance with the criteria. If one of the criteria is violated, the produced part is rated NOK. 💡 The drawing mode must not be active during quality assessment. → Adjust the property of the evaluation line and/or the evaluation polygon directly in the diagram. → Click on the middle of the line. ✅ Every mouse click moves the selection to the next property in the menu. → Alternatively, open the evaluation element in the list (5) and select the desired line. → In the lower section open the selection element (6). → Select a property (7).

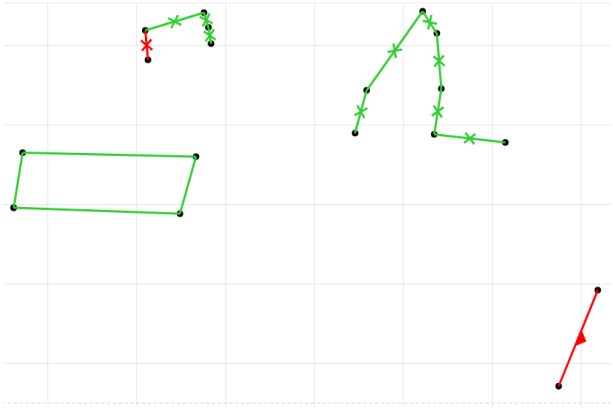
Screens/comments	Action steps
	After saving, the diagram displays a live preview of the evaluation. All curves rated good (marked green) are considered in the evaluation. The evaluation result is displayed by an appropriate color for every line. • Red line: The progress of the curves rated good violates the criterion. • Green line: The criterion for all curves rated good is valid. The curves rated bad are not included in the evaluation and are only for visual orientation. **Notice** A live evaluation will always use the current program and its properties with regard to the evaluation. The reference curves which do not correspond to this configuration may cause unexpected behavior in the representation.

Table 12 Evaluation element line and polygon

11 Log files

11.1 Press history

During production, all production data are saved in a data base and can be viewed later in the *Press history* area.

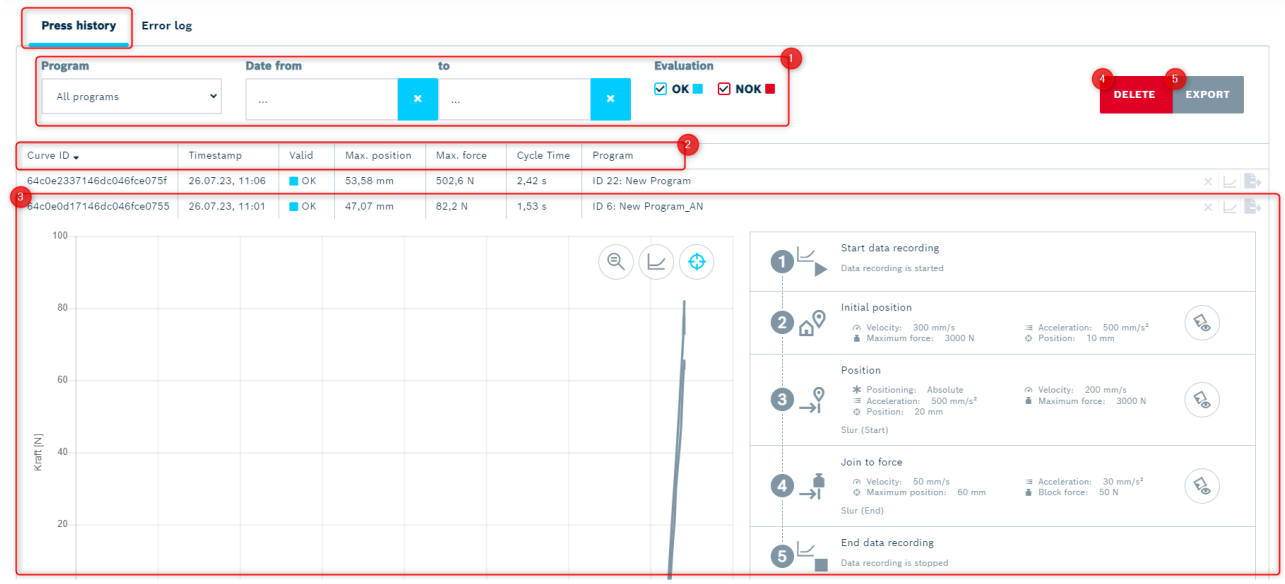
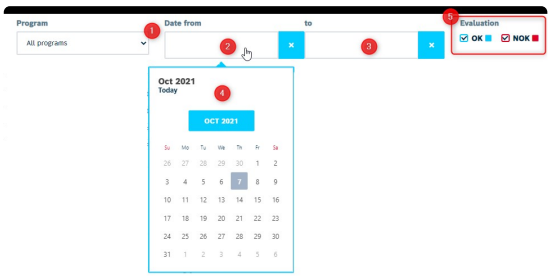
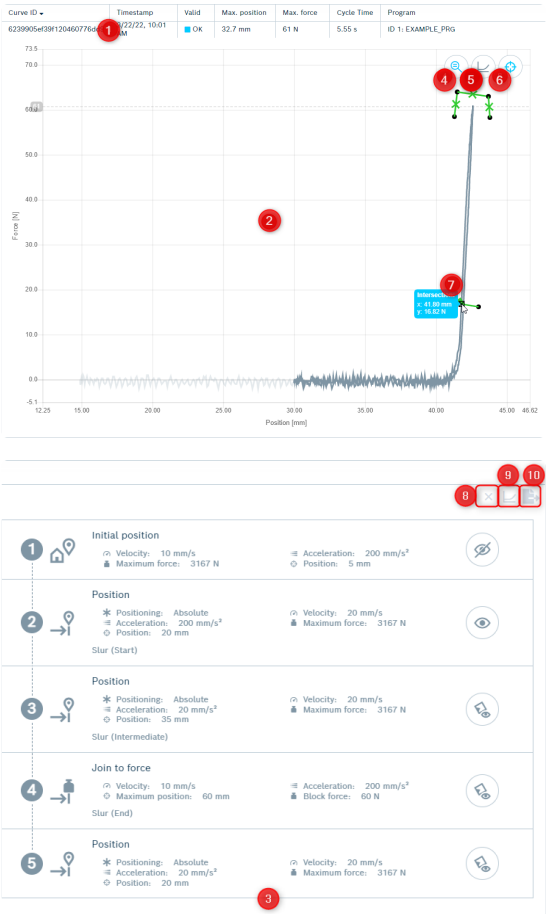


Figure 42 Press history

Position	Description/image	Description
1	Filter options 	→ To filter the table by the required program: From the selection element (1), select a program. → Click the *Date from* (2) or *to* (3) input field. ✓ A calendar (4) is displayed. → Select a period using the calendar. → To filter the table by the result of the quality evaluation: Select or deselect the fields *OK* or *NOK* (5). ✓ The table will be filtered according to the selected criteria.

Position	Description/image	Description
2	Production data	The *press history* shows the production data in the form of a table. The table can be sorted by the curve ID. Production data include: • Curve ID • Timestamp • Result of quality evaluation • NOK Cause • ◦ Evaluation OK: - ◦ Evaluation NOK: Error number (Details on the NOK cause and the sequence step that triggered the NOK evaluation can be seen in the detailed view) ◦ Reason not available: n/a • Maximum position • Maximum force • Cycle time (duration of sequence execution) • Serial number (optional, see the chapter Fieldbus)

Position	Description/image	Description
		• Program (name) **Notice** The force value is not displayed in real time and the force values are filtered for better representation. There may be slight deviations between the displayed and the real force value. For forces up to 100 N, two decimal places are shown. For higher forces, one decimal place is displayed. The accuracy of the displayed force value depends on the measurement uncertainty and the measuring range of the force sensor. **Notice** The position value is not displayed in real time and the position values are filtered for better representation. There may be slight deviations between the displayed and the real position value. Three decimal places are provided for the position. The accuracy of the displayed position value is predominantly dependent on mechanical influences. **Notice** A response value of "-1 s" for the cycle time indicates that the sequence was stopped due to an abort command.

Position	Description/image	Description
3	Detailed view  	→ Click on a program within a row of the table (1). ✓ The detailed view with the diagram of the force-displacement curve (2) and the individual program steps (3) is displayed. → Click on the shown diagram and move the view by drag-and-drop. → To reset the diagram zoom: In the diagram, click the *Zoom* button (4). → To show or hide the details in the diagram: Click the *Curve Details* button in the diagram (5). The *position symbol* (6) indicates that the X-axis is selected as position. → To switch to the time range: Click on the position symbol (6). ✓ The position symbol (6) changes to a clock symbol. → To switch back to position view: Click on the clock symbol. → To collapse the diagram and return to the table view: Click the program line (1) again. → To view further details: Hold the mouse pointer over a curve point, an evaluation element or an intersection (7). ✓> A pop-up window with further details will appear (e.g. ID, x and y coordinates). For curve points, time, position, force and corresponding step are displayed. → To delete the selected data set: Click the *Delete* button (8). → To make the production curve available as a reference curve: In the diagram, click the *Set curve as reference* button (9). → To export the selected data set: Click the *Export* button (10). ✓ A dialog window for file selection is opened in the web browser and the storage location in the HMI control unit can be selected.

Position	Description/image	Description
4	Delete	→ To delete production data and free memory capacity: Click the *Delete* button. 💡 If a filter is active, only the filtered results are deleted. 💡 Deleting data is only possible if the user is logged in as administrator. **Notice** If *press history* data are deleted, they are permanently removed from the system. → To avoid data loss: Export the data before they are deleted.

Position	Description/image	Description
5	Export	→ Click the *Export* button. ✅ A dialog window for file selection is opened in the web browser and the storage location in the HMI control unit can be selected. ✅ All currently filtered data sets are exported as collective export. 💡 Export is performed in JSON data format and includes all data from the production history (Production history) for the selected data set. 💡 Exporting data is only possible if the user is logged in as administrator. ***NOTICE*** ⚠ **Delays and aborts due to unstable network connection** During export (System → Client), data are written via a continuous data stream. If the network connection is unstable, this may cause delays and/or the writing of the data may be aborted. → Regularly check the network lines for damage. → Contact the network provider. → Ensure a stable network connection, preferably via LAN. → Ensure that the bandwidth is sufficient. → Perform the export without simultaneous activities (e.g. pressing). This also applies to other web browsers.

Position	Description/image	Description
		Notice **Impairment of the download speed in case of parallel production operation** If the export function is activated during production operation, the download speed may be considerably reduced. **Long duration for export of large amounts of data** Exporting large amounts of data may take considerably longer, since a continuous flow of data from the internal data base needs to be built up.

Table 13 Press history

11.2 Error log

The *Error log* area presents information messages, warnings and errors of the SFK in the form of a table.

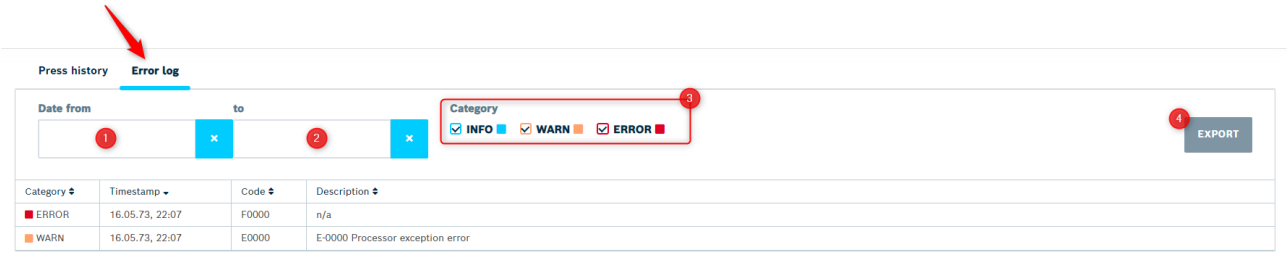


Figure 43 Error log

→ Click the *Date from* (1) or *to* (2) input field.

✓ A calendar is displayed.

→ Select a period using the calendar.

→ To filter the table by category: Select or deselect the *INFO, WARN* or *ERROR* (3) fields.

✓ The table will be filtered according to the selected criteria.

💡 The individual columns of the table can be sorted in ascending or descending order.

→ To export the error log: Click the *EXPORT* button (4).

✓ A dialog window for file selection is opened in the web browser and the storage location in the HMI control unit can be selected.

✓ Diagnostics are exported in JSON file format.

◊ For detailed information on the individual steps, please see the chapter Troubleshooting and fault clearance. Please also observe the documentation of the individual products.

12 Help

The help area contains instructions, links to product information, contact information, and fieldbus function blocks for connecting the Smart Function Kit to higher-level controls.

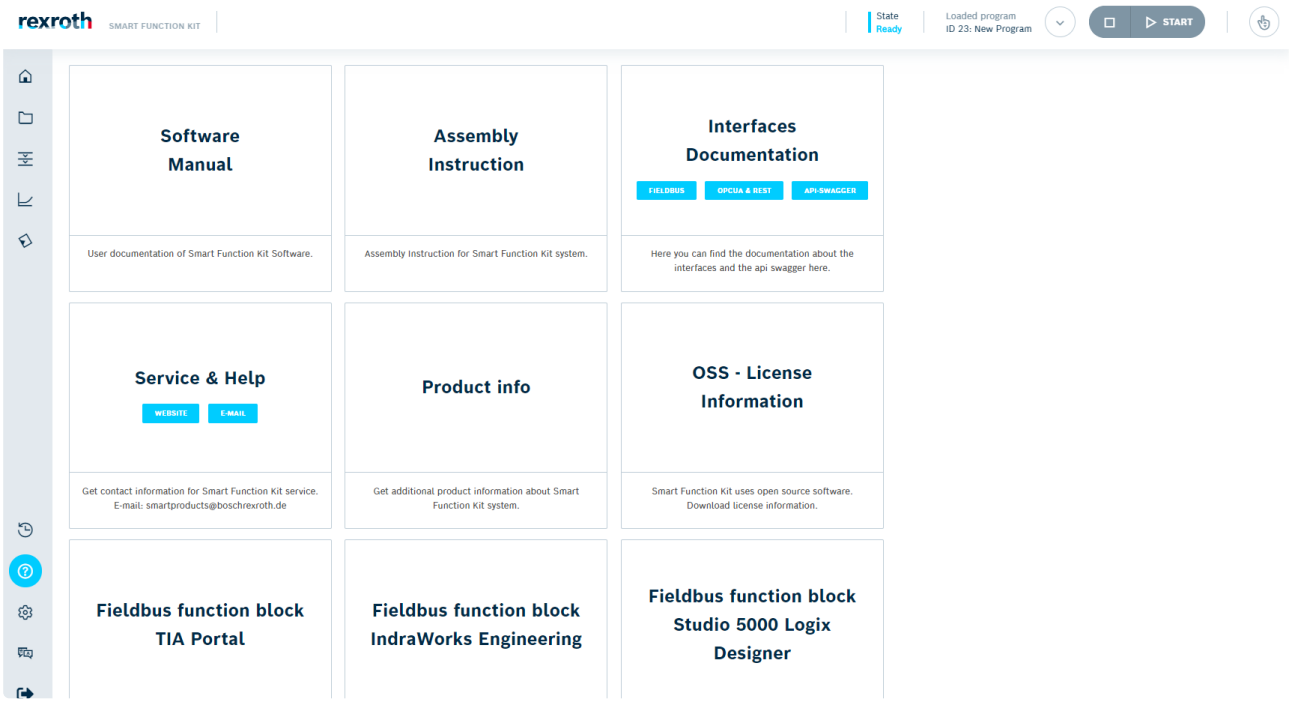


Figure 44 Help

→ To open a document: Click the relevant tile (e.g. Software manual).

✓ The selected document will be displayed in PDF format.

→ To open a link: Click the relevant tile (e.g. service or product info).

✓ The link is opened (here: list of contact persons, etc.).

→ To download a file: Click the relevant tile (e.g. fieldbus function block ...).

✓ The file will be downloaded.

13 Settings

Various system settings can be made at the SFK in the *Configuration* navigation area.

Figure 45 SFK settings

By clicking on the tabs, the following areas can be called up:

- General configuration
- System configuration
- Hardware configuration
- I/O Configuration
- Variables configuration

13.1 General configuration

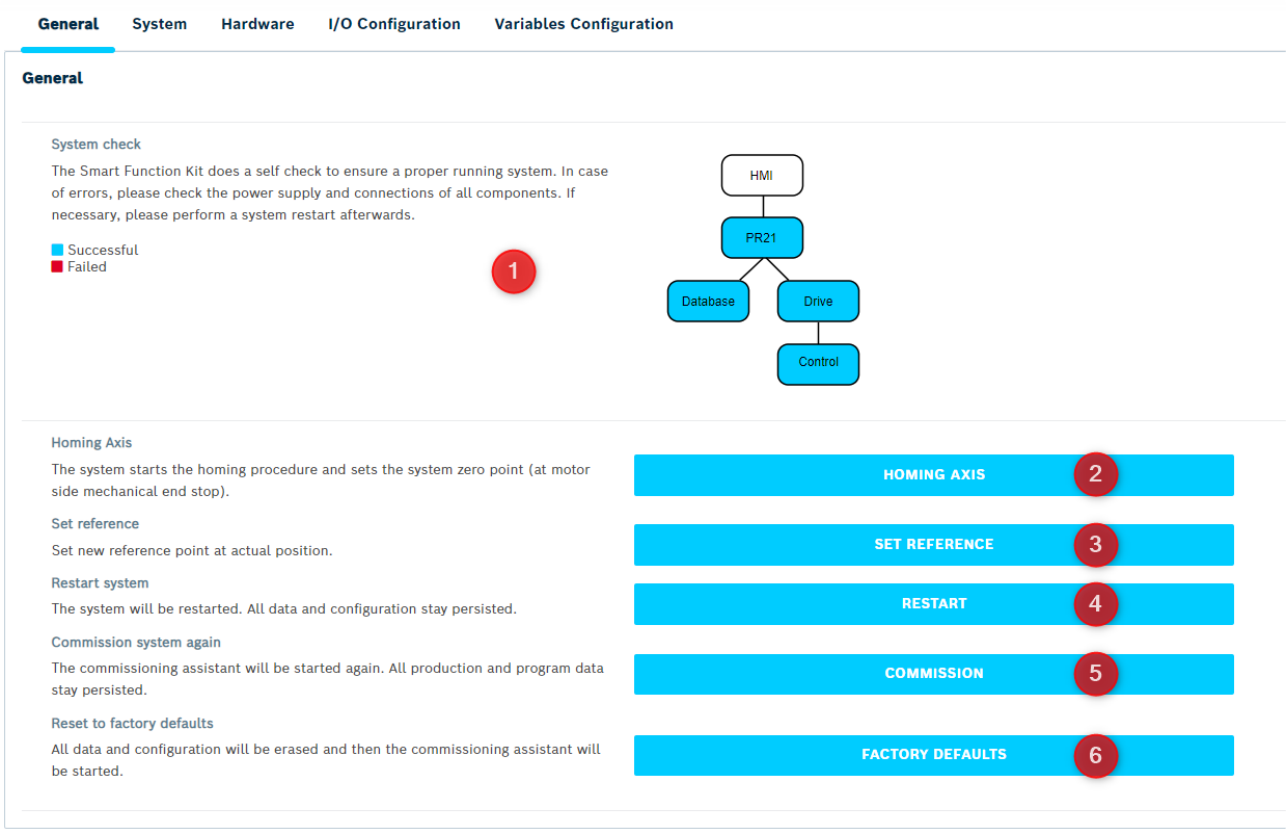


Figure 46 General configuration

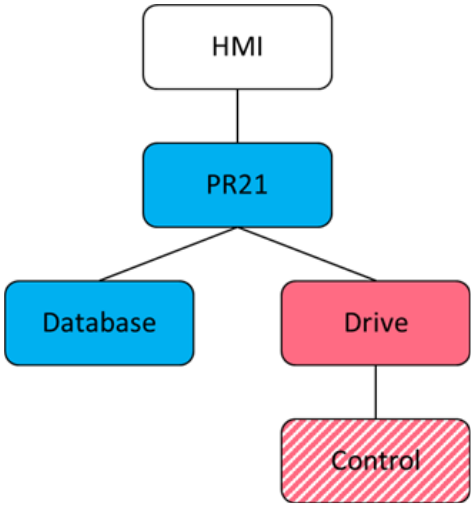
System check (1)

The *General configuration* tab displays the current status of the system. For this purpose, the system performs a self-diagnosis and checks that all components are correctly connected and ready for operation.

Status display:

- Blue: Component ready
- Red: Component not ready or connection error

The system can only be operated if all components are ready for operation.



→ If a faulted component is detected in the system check, proceed as follows:

1. Restart the system (main switch of the system).
2. Please make sure that the HCS01 is correctly connected with the PR21.
3. Please make sure that both devices are switched on.
4. Check the network settings at the HCS01.
5. Perform recommissioning of the system.
6. Check that the software is correctly installed on the PR21.

◊ For further information on the individual points, see Troubleshooting and fault clearance.

Homing axis (2)

◊ Information on the homing can be found in chapter Homing.

Set reference (3)

NOTICE	
	**System damage and functional limitation** The *Set reference* function means that the system limits and the software limit switches are not automatically adjusted to the new system zero point. Faulty software limit switches may result in damage to the system. The functional limitation of the system may result. → Use the *Set reference* function only if absolutely necessary. → Adjust the parameter Y-1044 *Max. position* in the extended *configuration* according to the new zero point.

→ To set the system zero point at the current position of the cylinder: Click the *SET REFERENCE* button.

Restart (4)

→ To restart the entire system: Click the *RESTART* button.

During this process, a note is displayed in the web interface.

Notice
Restart may take up to 90 seconds. Do not close the web browser window during this period. The web interface is automatically displayed after restart and the SFK can be operated as usual.

Commission (5)

If recommissioning of the system is required, the commissioning wizard can be called up via the *COMMISSION* button.

NOTICE

Resetting the system settings to default settings in case of recommissioning

In case of recommissioning, all system settings will be reset to the default values. All production data and programs are retained. This may result in functional limitations of the system.

→ Back up all settings before the recommissioning. For more information on the commissioning within the scope of a software update, please see the chapter Software Update.

Factory defaults (6)

→ To completely reset the system: Click the *FACTORY DEFAULTS* button.

NOTICE

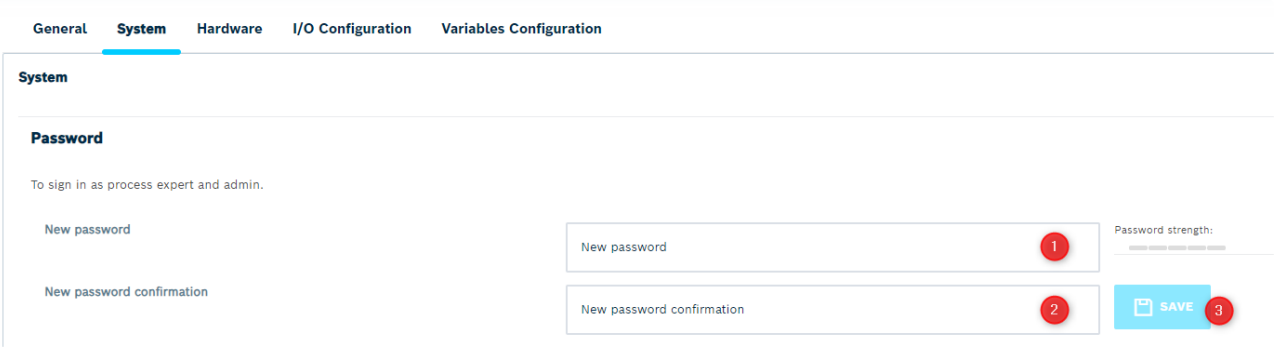
Data loss when resetting system to factory defaults

When resetting the system to factory defaults, all settings and production data will be lost.

→ Perform a data backup before reset to factory defaults.

13.2 System configuration

Changing a password



The password for logging into the administrator account can also be changed after commissioning.

1. Call up the system settings by the tab *Configuration*.
2. Enter a new password into the field (1).
3. Confirm the new password in the field below (2).
4. To save the new password: Click *SAVE* (3).

✅ Changing the password will overwrite the existing password.

✅ Login is only possible with the new password.

Network settings

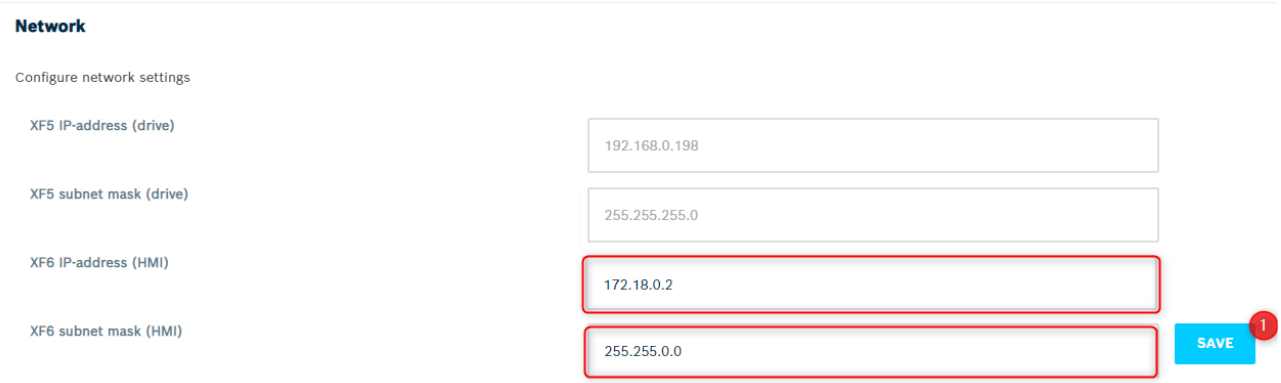


Figure 47 Network settings

To integrate the SFK into a local network, adjust the IP address and subnet mask as follows:

1. Enter a valid IP address and subnet mask in the input mask.
2. To apply the settings: Click *SAVE* (1).

💡 The address range 192.168.1.X cannot be used for the SFK.

→ Make sure that the HMI control unit is in the same subnet, as otherwise communication with the SFK cannot take place.

The following network settings are factory set for the system:

- IP address: 168.0.1
- Subnet mask: 255.255.255.0

- Gateway address: 0.0.0.0

The drive controller (X26) is directly connected with the PR21 (XF5). This connection forms a separate network that is not visible from the outside. For security reasons, the network settings for this network cannot be changed.

NOTICE

Loss of connection between drive controller and PR21

If the connection between the drive controller and the PR21 is lost, the SFK cannot be used.

→ Do not manually change the IP address of the drive controller (X26). The drive controller (X26) has a fixed IP address 192.168.1.1.

System information

Figure 48 System information

Synchronization of the system time is performed automatically during start-up. During synchronization, the system time of the HMI operator panel is read out and adopted as the system time in the SFK. All future production data will receive the new system time as a timestamp.

→ To synchronize the system time: Click the *SYNC button* (1).

NOTICE

Subsequent modification of the system time

Subsequent modification of the system time will cause inconsistent timestamps in the production data. For records that have already been saved, the timestamp is not adjusted, so the time history is not correct.

→ Make sure beforehand that the system time is correct when initially setting it.

→ If the system time is incorrect, login is not possible.

The Component information (2) area shows the version and/or hardware identification code of the software and hardware used.

- SFK software version
- Interpreter Firmware version (SMC)
- Drive controller firmware version

- Drive controller hardware version

Memory settings

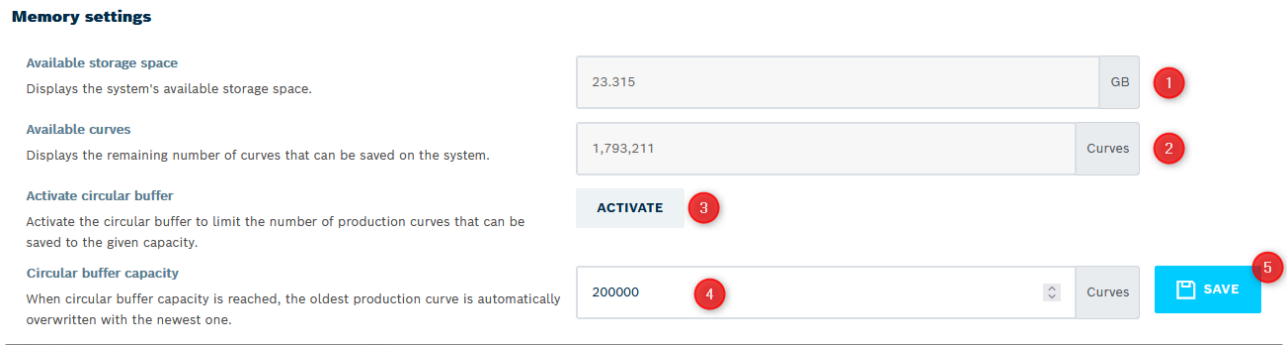


Figure 49 Memory settings

The *Available space* display field (1) shows the available storage capacity of the PR21. In addition, the number of storable curves is limited to the values displayed in the field *Available curves* (2). As soon as the storage space in GB or the storage capacity in curves is used up, no more new production data can be saved.

In the delivery state there are about 25 gigabytes or about 1.7 mio curves available as storage space.

NOTICE

Data memory on the PR21 is full

If the data memory on the PR21 is full, no new production data can be saved. You can no longer start programs.

→ Perform a backup of production data in the *Press history* area.

→ Delete production data that are no longer needed in the *Press history* area.

💡 When the remaining storage space reaches 300 MB or 10,000 curves, a warning message is issued both in the dashboard and via the fieldbus (see chapter Notifications Bit 9).

💡 If the remaining storage space is 100 MB or 0 curves, no further curves can be recorded, i.e. no programs can be started (see chapter Notifications Bit 7 and Bit 8 error code 31007). You can still execute the *Jog* command.

The button *ACTIVATE* (3) activates the circular buffer function. When activated, the size of the circular buffer is set to the defined *circular buffer capacity* (4).

→ Adjust the *circular buffer capacity* (4) to the desired value and accept it using the *SAVE button* (5) before activating the circular buffer.

→ If the circular buffer is already activated and the circular buffer capacity should be adjusted: Set the circular buffer capacity in the field *Circular buffer capacity* (4) and save the setting using the *SAVE button* (5).

NOTICE

Data loss due to activation of the circular buffer

If the number of production curves is greater than the desired capacity for the set circular buffer capacity, data loss will occur. The oldest curves are deleted until the number of curves in the memory matches the defined capacity.
This process may take some time depending on the number of curves to be deleted and cannot be canceled.

To create a backup of the curve data before activating the circular buffer, the following procedure is recommended:

→ Perform a backup of production data in the *Press history* area.

→ Delete production data that are no longer needed in the *Press history* area.

→ Set the desired circular buffer capacity.

→ Activate the circular buffer.

💡 The system cannot be used while adjusting the circular buffer capacity. Programs cannot be started (see chapter Notifications bit 7 and bit 8 error code 31008).

Exporting/importing configurations

Figure 50 Exporting/importing configurations

The SFK provides the option of saving the entire system as backup.

The backup includes programs, reference curves, evaluation elements, production data, configurations and device profile parameters.

→ To back up the system: Click the *DOWNLOAD CONFIGURATION* button (1).

→ Select a storage location for the backup on the HMI control unit.

Notice
→ To completely restore the system after reset to factory defaults: Load a backup.
By means of the backup, it is also possible to transfer settings from one system to another identical system.

→ To select the storage location of the backup on the HMI control unit: Click the button ... (2).

→ To restore a backup: Click *UPLOAD CONFIGURATION* (3).

Notice
After loading a backup, a system commissioning must be performed for safe operation.
The following settings are not included in the backup and must be transferred manually: Fieldbus, Extended Configuration and I/O Configuration.
◊ For more information, see chapter Software Update.

WARNING

Delays or aborts due to unstable network connection

During a backup (System → Client) or restore (Client → System), data are read and/or written via a continuous data flow. If the network connection is unstable, this may cause delays and/or the writing of the data may be aborted.

→ Regularly check the network lines for damage.

→ Contact the network provider.

→ Ensure a stable network connection, preferably via LAN.

→ Ensure that the bandwidth is sufficient.

Selection of fieldbus

Figure 51 Selection of fieldbus

The SFK can be operated as a subsystem with a fieldbus master.

→ Open the menu (1) and select one of the following fieldbuses available for the SFK:

- Sercos III
- EtherCAT (SoE)
- Profinet
- Ethernet/IP

→ To ensure compatibility with previous projects of software version 4: Adjust the telegram configuration if necessary. For this purpose, you can choose between compatibility mode compact (NUMBERS) and compatibility mode extended (CHARACTERS) (2).

→ Enter the device name to be used in the fieldbus for the SFK in the input field (3).

Input is optional and depends on the fieldbus selected in the menu (1).

→ To save the fieldbus selection: Click *SAVE* (4).

The drive is restarted.

◊ For further information on the fieldbus, see chapter Fieldbus.

13.3 Hardware configurations

The hardware configuration area provides an overview of the SFK hardware.

During commissioning, the hardware configuration is automatically read and all parameters and limit values are automatically set.

Extended configuration

The extended configuration offers the option of manually changing parameters and limit values.

→ To display the extended configuration: Click *SHOW EXTENDED CONFIGURATION*.

→ To close the extended configuration: Click *HIDE EXTENDED CONFIGURATION*.

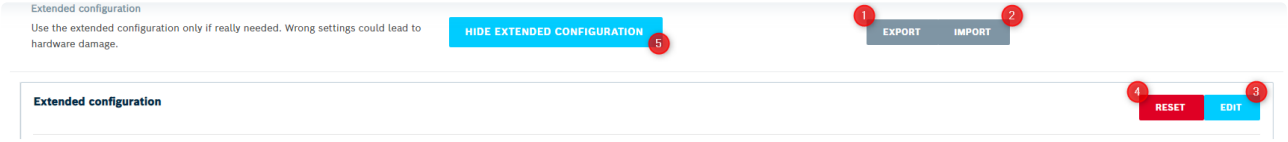


Figure 52 Extended configuration

→ To export the current values: Click *EXPORT* (1).

→ To import previously set extended configurations: Click *IMPORT* (2).

→ To enable manual input in the input fields of the extended configuration: Click *EDIT* (3).

✓ After clicking, this button changes to the *SAVE* function.

→ To save the inputs made: Click *SAVE* (3).

✓ After clicking, this button changes to the *EDIT* function.

→ To reset any inputs made: Click *RESET* (4).

NOTICE

Invalid system parameters due to changes in extended parameters

Invalid system parameters due to changes in extended parameters may result in production damage and function loss.

→ Only change standard parameters if absolutely necessary.

→ After any change, carefully observe and check the system behavior.

The SFK uses various components which have their own configuration interface via parameters.

◊ For further information on the parameters, please refer to the documentation for the individual products.

S and P parameters: Configuration of the IndraDrive drive controller.

Y parameter: Configuration of the SMC technology package.

VFR variables: Configuration of the interpreter firmware (double).

MFR variables: Configuration of the interpreter firmware (boolean).

List of system parameters:

ID	Name
S-0-0052	Reference distance [mm]
S-0-0121	Translation input
S-0-0122	Translation output
S-0-0123	Spindle pitch [mm]
S-0-0057	Position window [mm]
P-0-0001	Switching frequency of the power output stage [Hz]
P-0-0217	Time constant analog input filter [ms]
P-0-0279	Trending rate [ms]
P-0-1320	StrictMode
P-0-1321	Force limitation homing [%]
Y0015	E-Stop active
Y1045	Min. position [mm]
Y1044	Max. position [mm]
Y1004	Max. velocity [mm/min]
Y1006	Max. acceleration [mm/s^2]
Y1007	Max. drive torque [%]
Y1011	Position negation
Y1049	Analog sensor
Y1050	Force sensor characteristic (Pull) [N/V]
Y1053	Force sensor characteristic (Push) [N/V]
Y1054	Sensor zero crossing (pull/push) [V]

ID	Name
VFR002	“Join to force”: Maximum overload [N]
VFR004	Force limit of positioning command [N]
VFR005	“Join to force”: Minimal relative velocity
VFR006	Max. force
MFR001	Endless mode
MFR002	Filter actual values
MFR003	SafeMotion
MFR004	SMO: Pause program
MFR005	Cycl. documentation of variable values
MFR006	Program persistent after reboot
MFR007	Data recording deactivated
MFR008	Stop and torque off

Table 14 List of system parameters

S-0-0052 Reference distance

In this parameter, the required actual position value of an axis position (reference position) can be entered. The value entered in parameter S-0-0052 becomes effective as the actual position value at the reference position via the commands *Homing* or *Set absolute reference*.

S-0-0121 Translation input

Gearbox input rate. Together with S-0-0122, the parameter forms the transmission ratio.

S-0-0122 Translation output

Gearbox output rate. Together with S-0-0121, the parameter forms the transmission ratio.

S-0-0123 Spindle pitch

The parameter displays the distance the axis is moving when the gear output shaft or motor shaft makes one revolution.

S-0-0057 Position window

The parameter describes the maximum permissible deviation from the target position for a positioning command, by default 0.1 mm. Smaller values improve the repeatability, but can increase the cycle time of a program run.

P-0-0001 Switching frequency of the power output stage

In this parameter, the switching frequency of the power output stage can be set. A higher value reduces generation of perceivable noise of the drive controller and results in lower mechanical efficiency. The parameter can only be assigned with the following values: 4000, 8000, 12000, 16000.

P-0-0217 Time constant analog input filter

The signal of the force sensor can be captured with a digital low pass filter.

In this parameter, the filter time constant can be set in milliseconds (0.25 ms to 60 ms).

A value of 0.25 ms deactivates filtering.

P-0-0279 Trending rate

In this parameter the time value can be set. In the interval, the measured values for force and position are recorded, as standard every 4 ms. The parameter can only be assigned with the following values: 2, 4, 6, 8, ...

P-0-1321 Force limitation homing

This parameter limits the force generated during the homing at the mechanical end stop. If the specified force is reached, the mechanical end stop is detected and the reference point is set. The value is specified as a percentage of the nominal motor force.

This parameter is preset at the factory to 50% for systems with a nominal force <= 30 kN, or 30% for systems with a nominal force >= 45 kN.

Y0015 E-stop active

In this parameter, it can be set whether or not the signal for E-Stop is to be evaluated at the connection X31, pin 3. The parameter can be set to *TRUE* or *FALSE*.

The signal for E-Stop is processed as negative signal, this means as soon as X31, pin 3 no longer receives the signal, the error F4034 for E-Stop active is set.

Activation of parameter “MFR008 Stop and Torque Off”: Parameter Y0015 is automatically set to the value 0.

Deactivation of parameter “MFR008 Stop and Torque Off”: Parameter Y0015 is not automatically set to the value 1.

Y1045 Min. position

In this parameter, the position of the software limit switch in the negative direction can be set in mm, by default with the value -1 mm. When a program is created, the input values are evaluated to a value ≥ (Y1045 + 1 mm).

Y1044 Max. position

In this parameter, the position of the software limit switch in the positive direction can be set in mm, by default with the value S_{Max} +1 mm. When a program is created, the input values are evaluated to a value ≤ (Y1044 – 1 mm).

The travel range can be limited in the following cases:

- In case of interference contours

- If the *Set absolute reference* function is used to set a zero point for the system that differs from the standard (mechanical end stop).

Y1004 Max. velocity

This parameter can be used to set the maximum velocity of the system [in mm/min]. When a program is created, the input is evaluated to this value. The value must be greater than 0.1 mm/s.

Y1006 Max. acceleration

In this parameter, the maximum acceleration of the system can be set [in mm/s²]. When a program is created, the input is evaluated to this value.

Y1007 Max. drive torque

In this parameter, the maximum permissible drive torque of the system can be set. The value is given in the percent of the rated motor torque.

Y1011 Position negation

In this parameter, the operating direction of the drive can be set if the direction of rotation is reversed by a gearbox or a belt side drive. The parameter can be set to *TRUE* (negation enabled) or *FALSE* (negation disabled).

For a system with belt side drive the parameter must be set to *TRUE*.

Y1049 Analog sensor

In this parameter, the force sensor can be set active (value =1) or inactive (value = 0). The value *99* is reserved for internal purposes.

Force sensor inactive: All force values in the system are generated on the basis of the drive torque and the process data are recorded.

Y1050 Force sensor characteristic (pull)

This parameter is used to convert the analog value of the force sensor into the unit Newton. The parameter is only applied if the parameter Y1049 is set to TRUE.

The sensor characteristic value is entered in N/V.

Y1053 Force sensor characteristic (push)

This parameter is used to convert the analog value of the force sensor into the unit Newton. The parameter is only applied if the parameter Y1049 is set to TRUE.

The sensor characteristic value is entered in N/V.

Y1054 Sensor zero crossing (pull/push)

The parameter is used to indicate which signal the force sensor supplies at a force value of 0 N.

The parameter is moreover used to convert the analog value of the force sensor into the unit Newton. The parameter is only applied if the parameter Y1049 is set to TRUE.

The zero signal is entered in V.

VFR002 “Join to force”: Maximum overload

In this parameter the maximum overload for the "Join to force" module can be set. If this overload value is exceeded, error 41004 (see Error catalog) is issued. The sequence is aborted and rated NOK.

The parameter is specified in Newton.

VFR004 Force limit of positioning command

In this parameter, the maximum force of the positioning command can be set.

When the maximum force is reached: The travel movement is stopped and error 40304 is displayed (see chapter Error catalog). By default, this feature is disabled with the value 0 (zero).

When the maximum force is exceeded: A reverse movement (negative direction of travel) can be performed.

To allow for subsequent extension: Follow one of the following procedures.

→ Set the value of the force limit above the current force value.

→ Disable the function by setting the value to 0 N.

The limit applies both to positioning via the fieldbus and via the control panel.

The parameter is specified in Newton.

VFR005 “Join to force”: Minimal relative velocity

This parameter can be used to define the minimum velocity (in % of the press-in velocity defined in the component) at which the system moves towards the target force. During motion, the velocity of the system is reduced to the minimal relative velocity. The smaller the value, the more accurately the target force is reached, but the slower the movement becomes.

Example: A specified velocity of 60 mm/s at 2% minimal relative velocity results in an absolute minimal velocity of 1.2 mm/s.

By default, the value is 1%. If the value is set to 0%, there is no minimal velocity, which means that the speed progress is continuously proportional to the difference of time and actual force. Values above 100% also result in deactivation of the functionality.

VFR006 Max. force

Using this parameter, the maximum force of the system can be parameterized.

The actual force is monitored for the parameterized value.

When the maximum force is reached, the movement is stopped and error 20104 is displayed (see chapter Error catalog).

If the parameter is set to 0, the monitoring is disabled. The limit value of the drive controller (Y1007) applies.

By default, the maximum force value is adjusted to the entire system. This takes into account the limit values of mechanics, drive, controller and force sensor.

MFR001 Endless mode

The parameter is used to repeat a program indefinitely. After the last program step has been completed, the first program step is called automatically and the program is repeated.

Input value=0 (default setting): Endless mode is disabled.

Input value=1: Endless mode is enabled.

To finish the endless machining: Click the *STOP* button.

Notice

No data recording in endless mode active
No data is recorded and stored in endless mode. The current values are displayed and updated in the dashboard. If an endless cycle is manually aborted, an entry is made in the database but without the corresponding curve data.

MFR002 Filter actual values

This parameter can be used to enable or disable filtering of the current force, position, and velocity.

Input value=0 (default setting): The filter function is disabled. The force, position and velocity values are permanently updated on the dashboard and on the activated fieldbus.

Input value=1: The filter function is enabled. The force, position and velocity values are updated on the dashboard and on the activated fieldbus during program execution.

MFR003 SafeMotion

The parameter is used to activate the SafeMotion functionality of an S4 drive controller. The assignment of the safe input pairs must correspond exactly to the specification.

MFR004 SMO: Pause program

Activation of the SafePause system function (*Settings → Extended configuration*) causes a change in the system behavior:

- If SMST is selected, the program is interrupted and set to the pause state.
- If SMST is not selected, the program seamlessly resumed.

The subsequent evaluation is carried using the evaluation elements.

MFR005 Cycl. documentation of variable values

The parameter is used to activate or deactivate the cyclic saving of variable values.

Input value=1 (default setting): The values of the variables used are read upon program start and saved in the data base (press history).

Input value=0: No variable values are documented for the program sequence.

MFR006 Program persistent after reboot

The parameter is used to enable or disable loading of the last active program after a system restart.

Input value=1 (default setting): Loading the last active program after system restart is enabled.

Input value=0: The last active program must be loaded manually after system restart.

MFR007 Data recording deactivated

The parameter is used to activate or deactivate the data recording during program execution. Data recording is performed via the drive-internal oscilloscope.

Input value=0 (default setting): Data recording during program execution is activated. The drive-internal oscilloscope is automatically configured with the necessary parameters upon program start.

Input value=1: Data recording during program execution is disabled. An evaluation with evaluation elements cannot be performed. The drive-internal oscilloscope is not utilized by the SFK.

MFR008 Stop and Torque Off

The parameter is used to activate the Stop and Torque Off function.

Input value=0 (default setting): The Stop and Torque Off function is disabled.

Input value=1: Parameter “Y0015 Activation of E-Stop functionality” is set to 0. If the signal edge is falling at input IX31.3, the Stop and Torque Off functionality is called up.

13.4 I/O configuration

The SFK is equipped with integrated and virtual I/Os.

13.4.1 Integrated (digital) inputs and outputs

The digital interface on the drive controller, connector X31 has 7 digital inputs and 1 digital output.

The integrated inputs and the output (1) can be freely assigned to system and program functions.

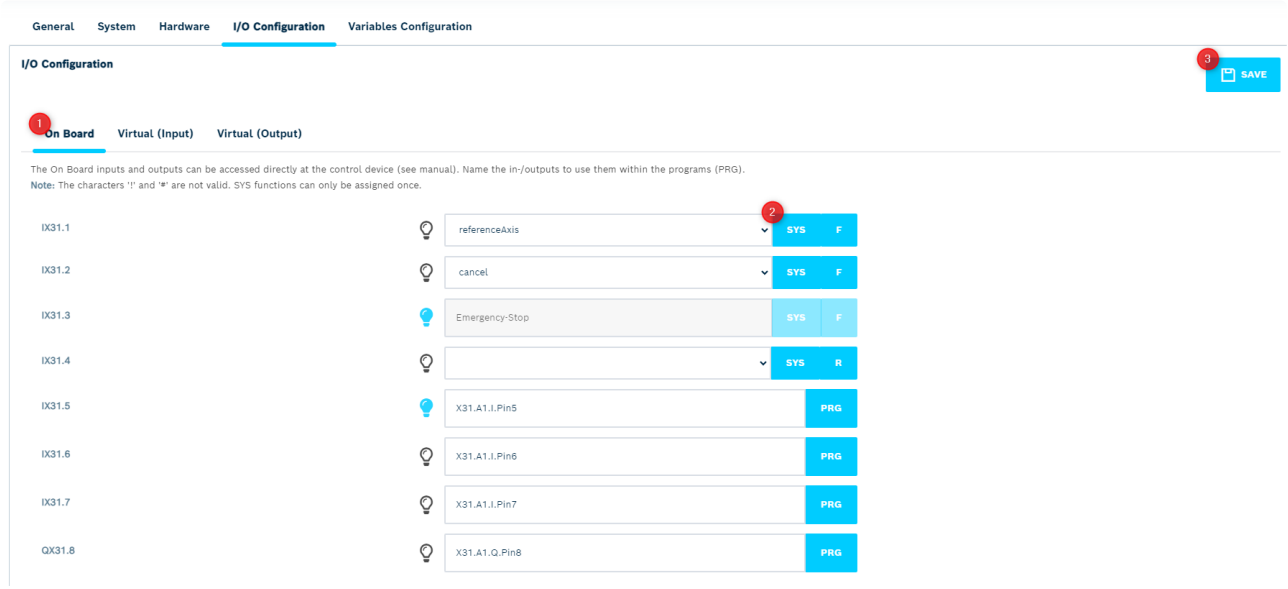


Figure 53 I/O configuration - On Board

System functions

The respective input is linked to the configured system function. The output provides the configured system status.

→ To be able to use a system function: With the selection button (2), select the *SYS* function.

✅ The text box changes to a selection list. The following functions can be selected:

Inputs (IX31...):

- Start current program (start)
- Cancel current program/command (cancel)
- Jog negative (jogNegative)
- Jog positive (jogPositive)
- Start homing (referenceAxis)
- Set absolute reference (setReference)
- Tare force sensor (tare)
- Clear error (clearError)

Output (QX31...): High level is on if

- program is running (programRunning)
- parameterization mode is active (parameterMode)
- power is switched on (powerOn)
- reference is set (referenceSet)
- evaluation takes place after one cycle (validating)
- evaluation result = OK, low level = NOK (validated)
- drive error is present (error)

For the inputs, it can be set whether a rising or falling edge is to be reacted to.

→ To switch between rising edge (R) and falling edge (F): Click the *R* or *F* button.

→ To activate all changes: Click *SAVE* (3).

Notice
Input signals that control system functions must be present for at least 40 ms.
Make sure, that input signals are not be dynamized (light-dark test).
A system function can only be assigned once.

Program functions

The respective input or output can be used as a signal in a program (e.g. in the module *Wait for signal* or *Set output*).

→ To be able to use a program function:

- With the selection button (2), select the *PRG*
- Create the input or output by entering an individual alias name

The Input or output with the corresponding alias name can be used in the sequence modules.

Inputs always respond to a rising edge.

→ To activate all changes: Click *SAVE* (3).

HINWEIS

System malfunction due to change of settings

The settings for inputs and outputs are globally valid. Changing the settings may cause the system to malfunction.

→ Manually adapt programs with an older configuration to the new settings.

13.4.2 Virtual inputs and outputs

The virtual inputs and outputs are connected with the master controller control via fieldbus parameters. The bits can be freely assigned and integrated in the program sequence.

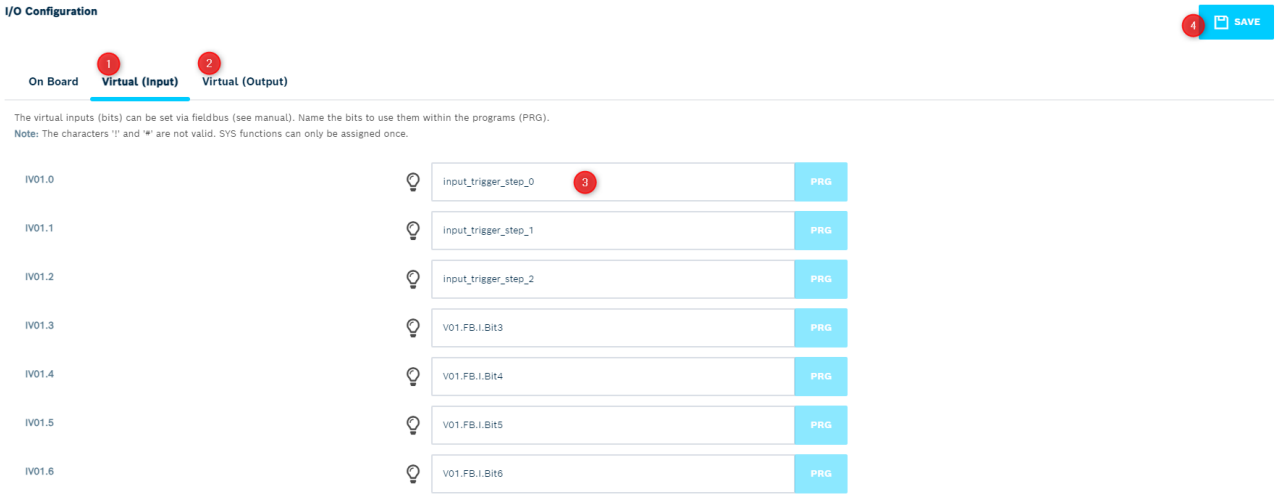


Figure 54 I/O KonfiI/O configuration - Virtual inputs and outputs

The virtual inputs (1) and outputs (2) can be used in a program (e.g. in the module *Wait for signal* or *Set output*).

→ For this purpose, create the input or output by entering an individual alias name (3).

✅ Input or output with the corresponding alias name can be used in the sequence modules.

→ To activate all changes: Click *SAVE* (4).

13.5 Variables configuration

In the *Variables configuration* area, variables can be assigned for use in press programs.

Program variables can only be used in the configured state. The variable range starts at VF500. A maximum of 100 variables are available.

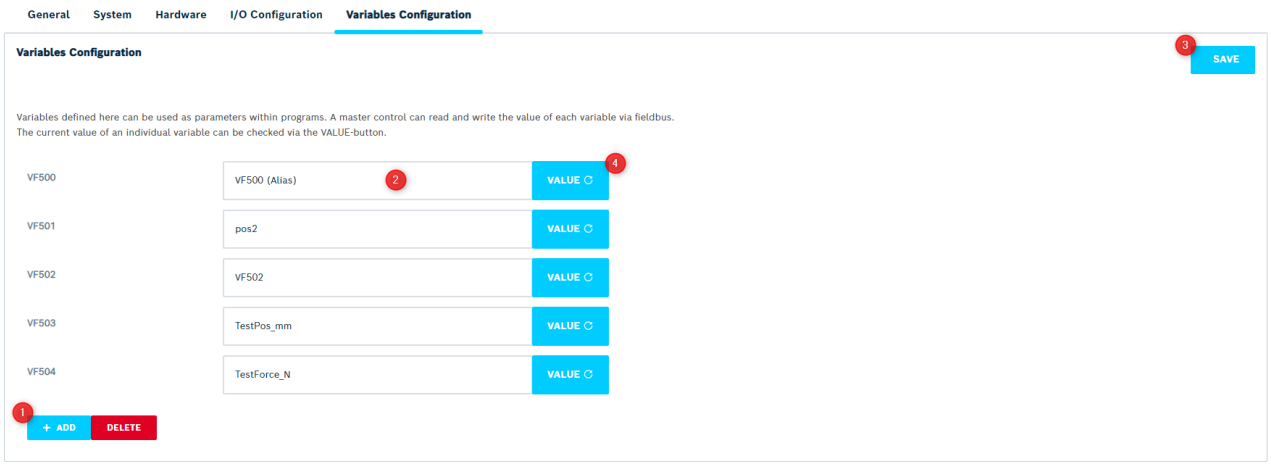


Figure 55 Variables configuration

→ To add further variables: Click *+ ADD* (1).

→ Enter the alias in the corresponding input field (2).

> **Notice**
> The alias can be assigned to any variable. During subsequent processing of the program step parameters, the alias can be used as an alternative to a fixed value (e.g. “pos1”, see figure).

→ To save the configuration: Click *SAVE* (3).

→ To read the current variable value: Click the *VALUE* button (4) at the appropriate variable.

Notice
This function is helpful e.g. for checking the value assignment via fieldbus.

NOTICE

System malfunction due to change of settings

The settings for inputs and outputs are globally valid. Changing the settings causes the system to malfunction.

→ Manually adapt programs with an older configuration to the new settings.

The program variables used and the related values are saved in the press history for every recorded curve. These values are recorded on program start and while the program is running.

This feature can be deactivated in the Extended configuration.

◊ See parameter MFR005.

✔ The variable table is no longer displayed.

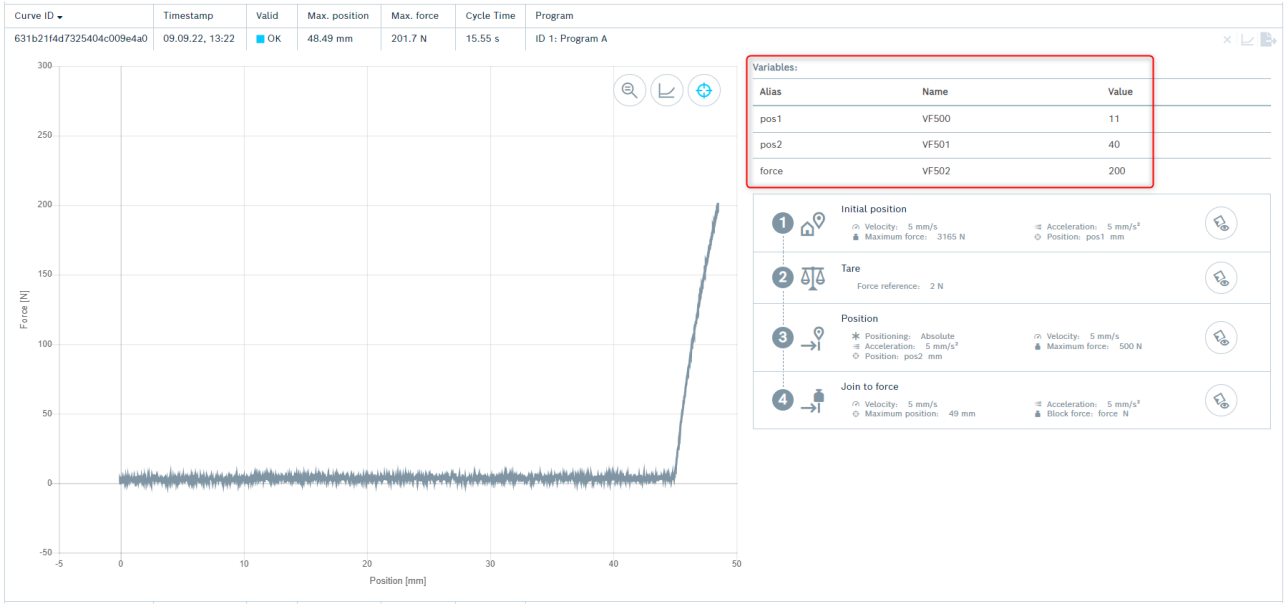


Figure 56 View in the press history

NOTICE

Invalid variables values

During the runtime, the variables values are subjected to a plausibility check. Invalid values cause a program abort. The error message F2011 / F2012 “PLC - Error” is displayed.

→ Make sure that the variables values for positions are within the valid travel range.

→ Make sure that the variables values for velocities are specified as positive values and smaller than or equal to the maximum velocity.

→ Make sure that the variables values for accelerations are specified as positive values and smaller than or equal to the maximum acceleration.

→ Make sure that the variables values for times are specified as values higher than or equal to 0.

Notice
Program variables can be described via the fieldbus, OPC-UA and via the SFK web interface. Program variables retain their value when the program changes. They are not reset to 0 until the system restarts.
◊ For more information, see chapter Library.

14 Variable ranges

The Smart Function Kit provides different variable ranges for different functions. Here, you can distinguish between

- Program variables
- Process variables
- System variables

14.1 Program variables

Program variables can be used as parameters in press programs. The variable range that can be used for that purpose ranges from VF500 to VF599. This variable range can be written as well as read. For more information on how to use these variables, see the Variables configuration in this instruction.

14.2 Process variables

Process variables can be used to read data about the last curve.

Process variable	Function
VF600	Max. position
VF601	Max. force
VF602	Cycle time

Table 15 Process variables

In addition, in the range VF650 to VF799, window limit values and window process data can be read out.

- For further information, see the chapter Evaluation → Fieldbus data allocation for evaluation windows.

This variable range is set to its initial value 999999999 at program start. The values are available as soon as the evaluation of the pressing process is completed.

This variable range can only be read. All writing operations are prevented.

14.3 System variables

System variables influence the behavior of the pressing system. These variables can be written or read via the extended configuration. Further information on this can be found in the section Hardware configurations → Extended configuration.

Notice
Quick reading and writing of variables via fieldbus is possible using the SFK_Fast_Variable_Access.

15 Safety technology

The SFK software supports safety functions of EN 61800-5-2.

- See the type key of the drive controller for the safety option of your SFK (safety option L3 or S4).

Notice
Observe standard EN ISO 16092.

15.1 Safety option L3

The HCS01 with the safety option L3 supports the safety function “Safe torque off” (STO).

With safety function STO, a safety level up to category 4, PLe, SIL3, can be achieved.

◊ For more information on the functionality of the safety function STO, please see the application description “IndraDrive Integrated Safety Technology Safe Torque Off (from MPx-16)” (R911332633), see https://www.boschrexroth.com/product/cls_eCommerce_lx_1555747.

15.1.1 Stop and torque off function

The Stop and Torque Off function triggers the following system reaction to a falling signal edge at input IX31.3.

The behavior differs as follows:

- Emergency stop (standard):
  Program stop and shutdown of the system via the E-Stop drive functionality. The displayed error message must be acknowledged for further operation.
- Stop and torque off:
  Program stop and shutdown of the system with switching to “Drive ready” (Ab) mode. In this mode, there is no torque at the drive and no axis motion can take place.
  Instead of an error message, the activated EMERGENCY STOP generates a warning. In the web interface this is shown as state *Safety Warning*.
  After a rising signal edge at input IX31.3 (e.g. unlocking of the Emergency OFF), the drive is automatically switched back to “Drive stop” (AH) mode. The *Safety warning* message disappears and the system condition changes to *Ready*. The axis can be moved again.

Figure 57 Stop and torque off function

Configuration of the system for use of the stop and torque off function

Configuration is carried out in the extended configuration via parameter MFR008.
0: Emergency stop (standard)
1: Stop and torque off

If activated, parameter Y0015 is set to FALSE.

→ Set parameter Y0015 to TRUE again if you want to go back to Emergency Stop.

Notice
The maximum acceleration rate of the system (see parameter Y1006) has a direct effect on the shutdown time.

15.2 Safety option S4

The HCS01 with the safety option S4 supports the “Safe Motion” (SMO) safety technology.

Notice
→ Take note of the instructions in chapter Commissioning with safety option S4 for the commissioning of the safety technology.

In connection with the SFK software, the following safety functions can be provided:

	Safety function	Achievable Performance Level (PL) acc. to EN ISO 13849-1 and Safety Integrity Level (SIL) acc. to EN 62061
Safe standstill	Safe torque off (STO)	Category 4, PL e, SIL 3
	Safe brake piloting (SBC)	
	Safe operating stop (SOS)	Category 3, PL d, SIL 2
Safe shutdown	Safe stop 1 – time priorization (SS1-t)	Category 4, PL e, SIL 3
	Safe stop 1 – standstill priorization (SS1-r)	Category 3, PL d, SIL 2
	Safe stop 2 – standstill priorization (SS2-r)	
	Safely monitored deceleration (SMD)*	
Safe motion	Safe limited speed (SLS)	
	Safe maximum speed (SMS)*	

Table 16 Safety option S4

* not defined in EN 61800-5-2

◊ Further information on the functionality of the SMO safety technology can be found in the application description "IndraDrive Integrated Safety Technology Safe Motion (from MPx-18)" (R911338919), see downloads on www.boschrexroth.com.

Notice
When safety option S4 is selected in the product configuration, the SFK will be delivered with the HSZ01 safety zone module.
If the SFK is to be integrated into a global safety system, e.g. using PROFIsafe or FSoE, the safety zone module is not required. A suitable SFK system configuration without a safety zone module is available on request.

Notice
A cyclic test of the braking system according to DIN EN ISO 13849-1, Category 2 is currently not supported.

15.2.1 Pause program

The behavior of the software depends on the selected safety option of the HCS01 and can be adjusted by configuration (e.g. whether the program is paused or canceled).

Behavior without pause function

→ Set the parameter MFR004 in the extended configuration to the value = 0 (default setting).

- ✅ When the safety technology is triggered, the SFK cancels pending commands.
- ✅ The running program is immediately aborted.

✅ The press curve is rated NOK.

→ Acknowledge any pending errors.

→ Make sure that the safety technology is no longer active.

✅ When the program is restarted, the first program step is called up.

Behavior with pause function

→ Set the parameter MFR004 in the extended configuration to the value = 1 (only applies to the drive controller with safety option S4).

- ✅ When the safety technology is triggered, the SFK interrupts the program.
- ✅ The program remains paused while the safety technology is triggered.

✅ After the triggering of the safety technology has been terminated, the program sequence is continued at the interrupted position.

💡 A new start command is not required.

💡 An interrupted program can be evaluated with OK.

15.2.2 SLS - Safely Limited Speed

If the movement speed (specified via fieldbus or Control panel) is below the defined limit speed, movement is possible. If it exceeds this limit, the system stops the movement with an error message.

The program start button (see Title bar) is deactivated in manual mode, i.e. no programs can be started.

15.3 Connection to peripherals

The safety options S4 and L3 enable the connection of safety-related peripheral devices to the SFK (e.g. light barriers, emergency stop buttons, two-hand controls).

Safety option L3

The peripherals are pre-wired directly on the HCS01.

→ Make sure that the STO input is triggered after the E-stop input with a time delay.

✓ In case the safety technology is triggered, the axis is put to a standstill before the energy supply of the motor is interrupted.

✓ This prevents coasting of the axis.

Safety option S4

- **Safety option S4 with HSZ**: The delivery includes the HSZ01 safety zone module. Peripheral devices are connected to the safety zone module. This is connected to the HCS01 so that the signals from the peripheral devices are forwarded to the drive controller.

  ✓ Shut down of the drive is performed via the configured transition function (SS1-t, SS1-r or SS2-r). Additional triggering of the E-stop input is not required.

  → Make sure that when the SMO functionality is used, parameter MFR003 has the value = 1 (default setting).

- **Safety option S4 without HSZ**: The delivery does not include the HSZ01 safety zone module. The peripheral devices are connected to the global safety system (usually the safe PLC / higher-level control). This is connected to the HCS01 via safe bus communication so that the signals from the peripheral devices are forwarded to the drive controller.

  → Follow the instructions in chapter Configuration instructions for safe fieldbus communication to configure the safe bus communication (e.g. PROFIsafe, FSoE FailSafe over EtherCAT).

  ✓ Shut down of the drive is performed via the configured transition function (SS1-t, SS1-r or SS2-r). Additional triggering of the E-stop input is not required.

  → Make sure that when the SMO functionality is used, parameter MFR003 has the value = 1 (default setting).

◊ For further information on the wiring of peripheral devices with the Smart Function Kit, please see the assembly instructions (R320103194) and the example circuit diagrams at www.boschrexroth.com/product/cls_eCommerce_lx_1555747.

16 Fieldbus

16.1 Introduction

The fieldbus interface allows for the SFK to be used as subsystem in an automation plant.

The following fieldbus protocols are available for the SFK:

- SercosIII
- EtherCAT (SoE)
- Profinet
- Ethernet/IP

The minimum supported cycle time (IO cycle update time) is 4 ms.

Notice
To ensure error-free operation as a subsystem of a master control, data exchange must be carried out according to the instructions described here.

The fieldbus protocol can be selected via the *Selection list* in the *System configuration* area, see following chapter.

16.2 Hardware

16.2.1 Topology

The following figures show the physical integration of the SFK (consisting of HCS01 and PR21) into a higher-level automation system.

Notice
→ For error-free communication between the participating devices: Ensure that the appropriate device ports are correctly cabled and used.

In the figures, a Rexroth XM21 control is used as the fieldbus master. This control version has an integrated SercosIII-Master fieldbus connection. For the deployment as a Profinet master, the XFE01.1-FB-03 expansion module is used.

Notice
Third-party hardware can be used for the fieldbus master.
→ Please refer to the third-party provider's documentation.

Structure/ wiring of a Profinet fieldbus

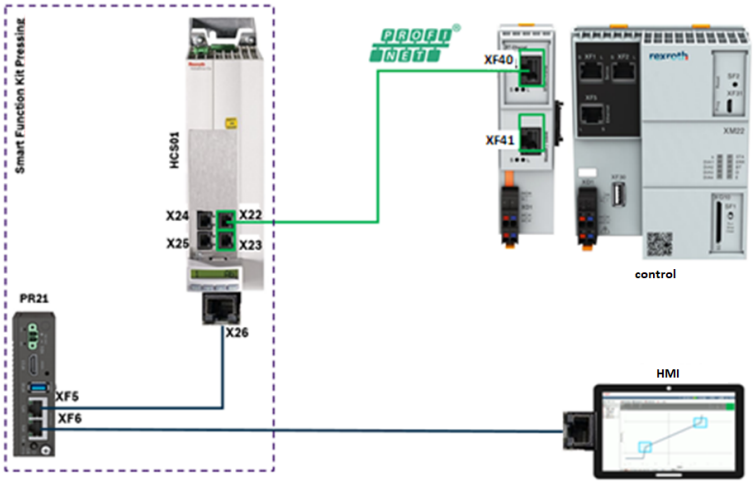


Figure 58 Structure/wiring of a Profinet fieldbus

Structure and/or cabling of a SercosIII fieldbus

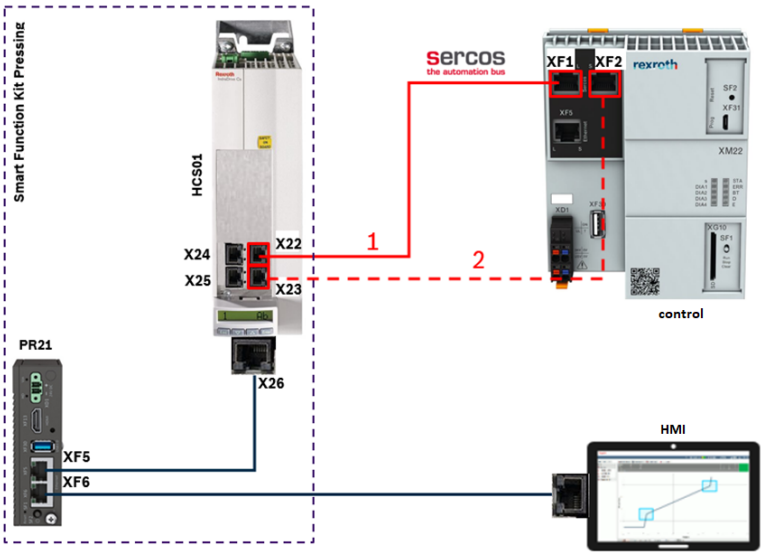


Figure 59 Structure and/or cabling of a SercosIII fieldbus

The SercosIII fieldbus can be designed as linear or ring topology.

You can achieve a higher fault tolerance against EMC interference and cable defects by setting up the SercosIII master as a ring topology (cable 1 + 2). This way, redundancy of data is ensured.

16.2.2 Settings

General requirements

The user communicates with the SFK web interface via the HMI (e.g. tablet) and the Industrial PC PR21 (port XF6). An error-free network connection is required.

The following requirements apply to the connection of the HMI (e.g. tablet) to the SFK:

- Web browser with HTML5 and Java Script support (e.g. Firefox version 52 or higher, Chrome version 74 or higher)

- Screen diagonal 10 inches (format 16:9) or higher
- Screen resolution 1280 x 720 px or higher

IP address configuration

The correct configuration of the IP addresses and subnet marks for the individual device ports is a prerequisite for error-free communication.

The default settings are as follows:

1. IndraDrive HCS01 – Port X22/X23 (FKM Engineering via IP):
   - IP address: 192.168.2.1
   - Subnet mask: 255.255.255.0
2. IndraDrive HCS01 – Port X26 (ET):
   - IP address: 192.168.1.1
   - Subnet mask: 255.255.255.0
3. Industrial PC PR21 - Port XF5 (cannot be changed):
   - IP address: 192.168.1.100
   - Subnet mask: 255.255.255.0
4. Industrial PC PR21 - Port XF6 (can be changed via web interface):
   - IP address: 192.168.0.1
   - Subnet mask: 255.255.255.0

Change IP addresses

1. IndraDrive HCS01 – Port X22/X23 (MT):

The port X22/X23 essentially fulfills two tasks:

- Interface for communication via the fieldbus: Exchange of data packets with other fieldbus participants, especially the fieldbus master.
- Interface for engineering via IP: Communication with the drive, e.g. via the engineering tool "IndraWorks Ds".

The physical port X22/X23 has its own IP address for each of these two communications. The respective IP addresses for communication via fieldbus and for engineering via IP **must differ from each other.**

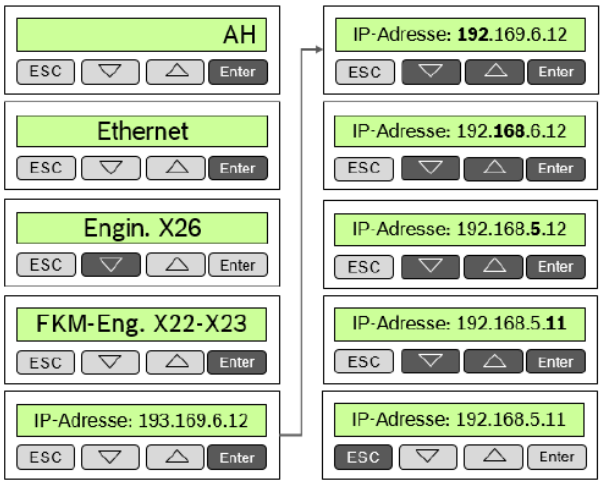


Figure 60 Adjusting the IP address of X22/X23 - Engineering over IP

The IP address can be adjusted and changed on the HCS01 if necessary.

NOTICE

Change of IP address of IndraDrive HCS01 - Port X22/X23 X24/X25

The IP address range 192.168.1.x is used internally by port X26. The system will no longer function if ports X22/X23 or X24/X25 are also occupied in this IP address range.

→ Do not change the IP address to the address range 192.168.1.x.

Notice

Configuration via Profinet fieldbus master (e.g. TIA Portal)
→ For communication via Profinet: Assign the Profinet IP address exclusively via the fieldbus master (e.g. the TIA Portal).
The engineering IP address of port X22/X23 must not match the Profinet IP address!

2. IndraDrive HCS01 – Port X26 (ET):

NOTICE

Changing the IP address of the IndraDrive HCS01 - Port X26 (ET)

If the IP address of the IndraDrive HCS01 - Port X26 (ET) is changed, the system will no longer be functional.

→ Do not change the IP address.

3. Industrial PC PR21 - Port XF6:

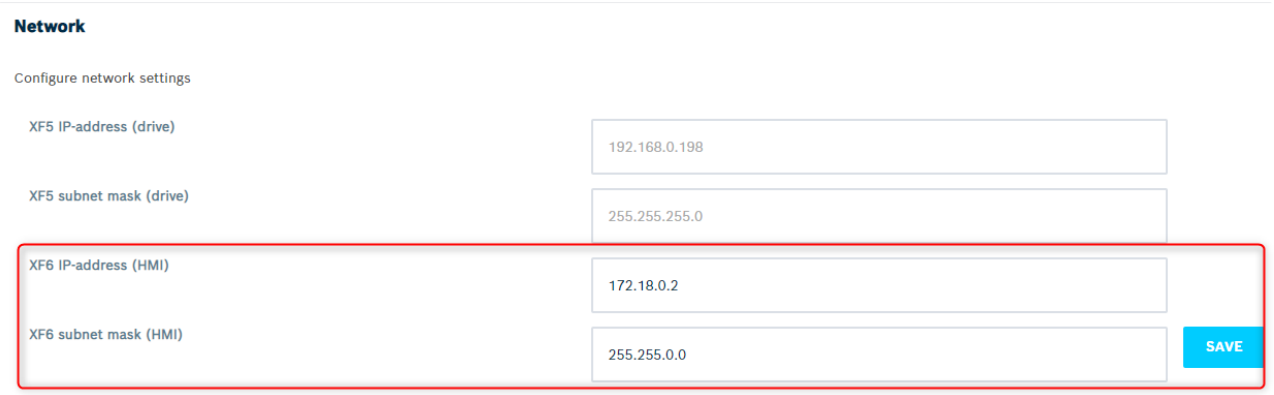


Figure 61 Adjusting the IP address of the PR21 Industrial PC – Port XF6

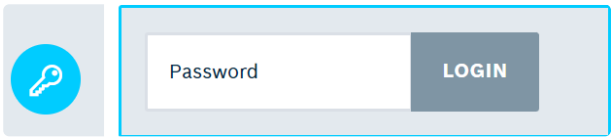
→ Open the Configuration.

→ In the *System* tab under *Network*, change the XF6 IP address (HMI) and the XF6 subnet mask (HMI).

Fieldbus activation

In order for the SFK to communicate with or to enable control via a fieldbus, the corresponding fieldbus protocol must be selected in the web interface.

Proceed as follows:

Screens/comments	Action steps
	→ Open the web browser in the HMI. → In case of connection problems: Empty the cache and clear the browser history, cookies and other data. **Notice** Clearing the cache or clearing the history is not necessary for web browsers with enabled private mode. → Enter the IP address in the address bar. *Standard IP address: 192.168.0.1* The SFK web interface is called.
	→ Enter the password *admin* (if not changed). → Sign in.

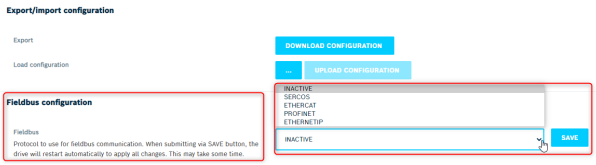
Screens/comments	Action steps
 Specification of a device name is possible with Profinet and Ethernet/IP. By default, the name is set to “SFK”. Other device names must meet the following properties: • Max. 240 characters (letters, numbers, hyphen or period). • The length of the letter sequence between two dots should not exceed 63 characters. • Do not use special characters (umlauts, quotation marks, underline, slash, space, etc.). Exception: Hyphen. • No hyphen as first and last character of the designation. • No n.n.n.n (n = 0...999) as format of the designation. • No port-xyz (x,y,z = 0...9) at the beginning of the designation.	→ Scroll down under *Configuration* → *System*. → Select the corresponding type under *Fieldbus configuration*. → If required, change the device name. → Click *SAVE*.

Table 17 Fieldbus activation

16.3 Fieldbus configuration

To establish communication, the master control must be configured. The configuration of the cyclic inputs and outputs depends on the development environment of the master control used.

Depending on the protocol used, a specific configuration file must be used in the master control for communication. This describes the device-specific communication specifications.

→ Ensure that the IP addresses in the drive controller are set appropriately beforehand (see chapter Settings).

Protocol	Title	Link	Notice
Profinet	DeviceDataSheets PROFINET IndraDrive	https://www.boschrexroth.com/de/de/media-details/58a413bb-290d-4dc3-90c0-881ca3f97738	GSDML-V2.1-Bosch Rexroth AG-011F-Indradrive_02V02-20180110.xml
EtherCAT	DeviceDataSheets EtherCAT SoE IndraDrive	https://www.boschrexroth.com/de/de/media-details/34b5c88a-56c2-4082-b7d2-869f3a90c067	BoschRexroth_IndraDrive_ECAT_SoE_01V27.XML

Protocol	Title	Link	Notice
Ethernet IP	DeviceDataSheets EtherNet_IP IndraDrive	https://www.boschrexroth.com/de/de/media-details/37f7a1c4-149e-4fe3-beb9-4fe4af722017	IndraDrive_EIP_MPx18.eds
Sercos	DeviceDataSheets SERCOS_III IndraDrive	https://www.boschrexroth.com/de/de/media-details/75eed618-d3e7-402f-9045-77e8bccde257	

For communication between the SFK and the higher-level control, data are transmitted in the form of WORDS via the fieldbus. These are recorded by the fieldbus master as inputs or outputs.

The following describes which drive parameters are assigned to the variables. The correct assignment in the higher-level control is necessary, for example, for the fieldbus protocols Sercos III and EtherCAT.

Fieldbus variable list

The cyclic data channel contains 24 input words and 21 output words.

Notice
For compatibility with the previous project from software version 4, the appropriate compatibility mode must be selected in the telegram configuration.
◊ For further information on selecting the fieldbus, see chapter System configuration.

With the configuration, up to 7 REAL values per command can be sent to the SFK (e.g. a serial number with up to 28 characters at program start).

◊ For more information about the serial number, see chapter Requests and responses.

WORD number	ET-Master (Input 24 WORDs) Description	Data type	Drive (Slave) Parameter Mapping
1	responseHandle	WORD	P-0-1414
2	responseStatus	WORD	P-0-1415
3	responseValue1	DWORD	P-0-1290
5	responseValue2	DWORD	P-0-1291
7	responseValue3	DWORD	P-0-1292
9	notificationID	WORD	P-0-1416
10	notificationValue	DWORD	P-0-1293
12	notificationStatus	WORD	P-0-1417

13	notificationPosition	DWORD	S-0-0051
15	notificationVelocity	DWORD	S-0-0040
17	notificationForce	DWORD	P-0-1294
19	digital IO Mapping	WORD	P-0-1418
20	virtual_io_in	WORD	P-0-1419
21	responseVariable	DWORD	P-0-1272
23	responseVariableValue	DWORD	P-0-1273

Table 18 Structure of cyclic fieldbus configuration

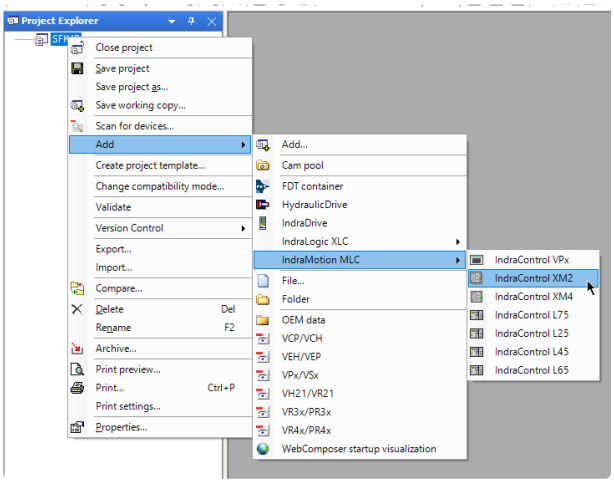
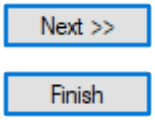
WORD number	ET-Master (Output 21 WORDs) Description	Data type	Drive (Slave) Parameter Mapping
1	requestHandle	WORD	P-0-1394
2	requestAction	WORD	P-0-1395
3	requestValue1	DWORD	P-0-1287
5	requestValue2	DWORD	P-0-1288
7	requestValue3	DWORD	P-0-1289
9	requestValue4	DWORD	P-0-1274
11	requestValue5	DWORD	P-0-1275
13	requestValue6	DWORD	P-0-1276
15	requestValue7	DWORD	P-0-1295
17	virtual_io_out	WORD	P-0-1396
18	requestVariable Type	DWORD	P-0-1270
20	requestVariableValue	DWORD	P-0-1271

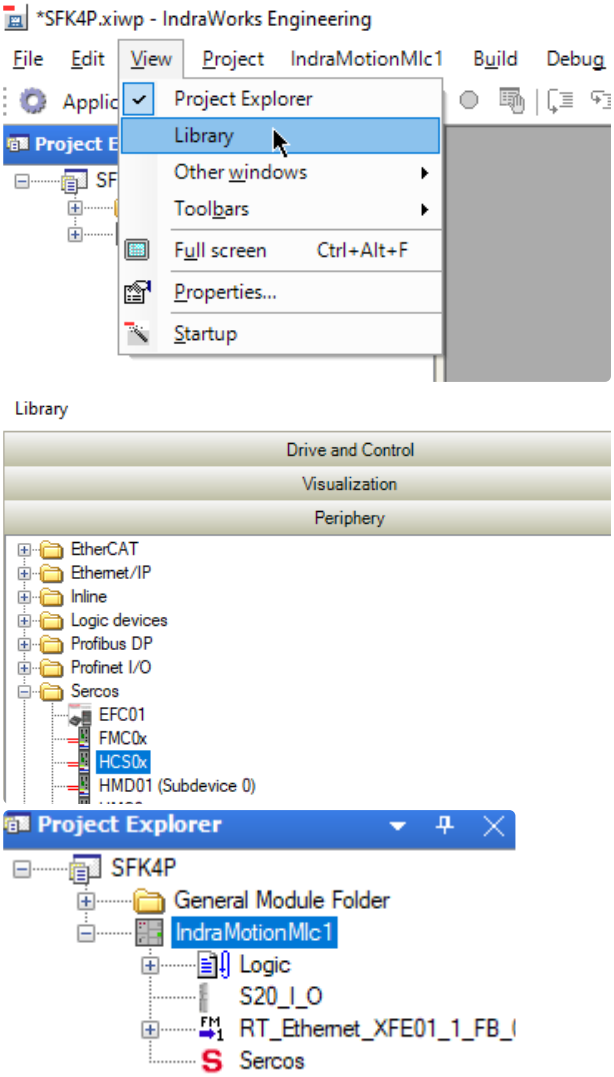
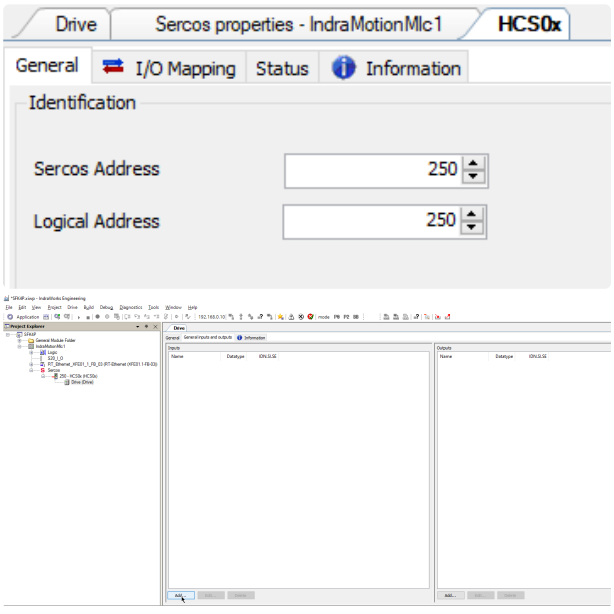
16.3.1 IndraWorks Engineering

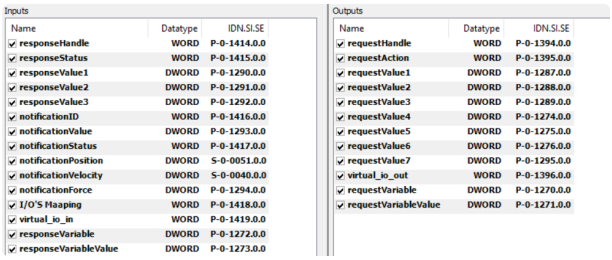
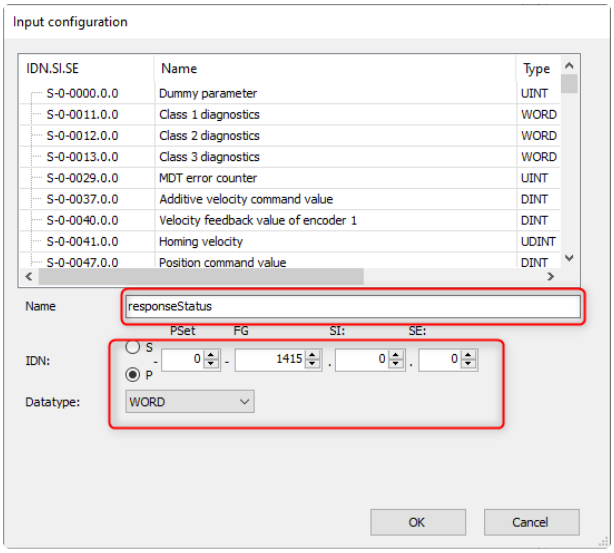
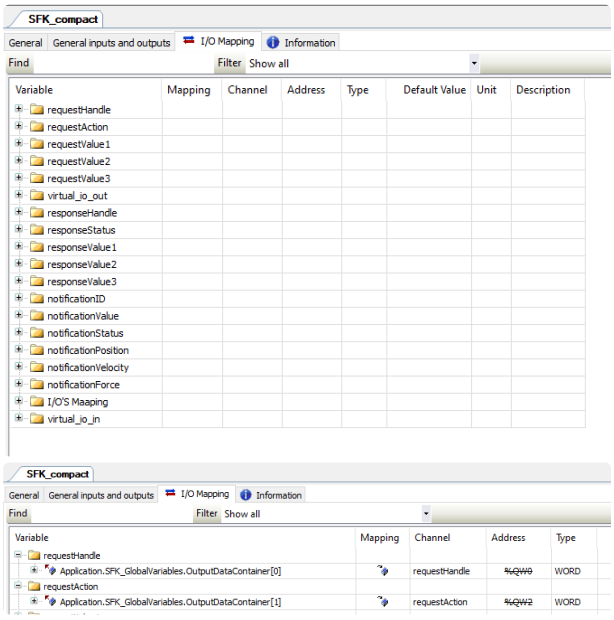
The configurations on a XM21 control system from Bosch Rexroth AG are described below. IndraWorks Engineering is used as the development environment.

Sercos III

To integrate the SFK into a XM2 control system from Bosch Rexroth AG, proceed as follows using the Sercos III fieldbus protocol:

Screens/comments	Action steps
	**First step** Add the drive of the SFK as Sercos device to the fieldbus master: → Start IndraWorks and add a new project (empty project). → Press the right mouse button. → In the menu navigate to *Add*. → Click *IndraMotion MLC*. → Select *IndraControl XM2*.
 	✅ A dialog window opens to configure the MLC control (XM2). The following configuration steps are carried out one after the other: 1. General properties 2. Hardware/communication 3. Extended settings 4. Function packages 5. Interfaces → Make the required entries in each dialog box and click *Next >>*. → When all entries have been made, click *Finish*. ✅ Configuration is completed and the control is displayed in the project node.

Screens/comments	Action steps
	Add Sercos device (HCS01): → Select in the tab *View* the *library*. → In the *Library*, click the *Periphery* tab. → Under *Sercos*, select the device *HCS0x*. → Alternatively, drag the device under the Sercos node in the Project Explorer.
	**Second step** Compare and adapt the fieldbus addresses in IndraWorks with the addresses of the Sercos device: → In the project explorer, navigate to *Sercos* → *HCS0x*. → Select *Drive from the context menu.* → Specify the parameters in the tab *General inputs and outputs*.

Screens/comments	Action steps
	→ Adjust the cyclic inputs and outputs according to the example (left).
	→ Click *Add*. ✓ An input form is opened. → Enter the names of the input and output. → Select the data type and enter the parameters.
	✓ The cyclical inputs and outputs created in the *General inputs and outputs* tab are listed in the *I/O* mapping. → Assign the inputs and outputs of the fieldbus with the *+ Symbol* the internal variables via the tab *I/O mapping* . ✓ The fieldbus master is accessing the fieldbus data. **Notice** After configuration, a restart of the fieldbus may be required. This restart is performed by changing the control modes as follows: • Operating mode (BB) • Download mode (P0) • Parameter mode (P2) • Operating mode (BB)

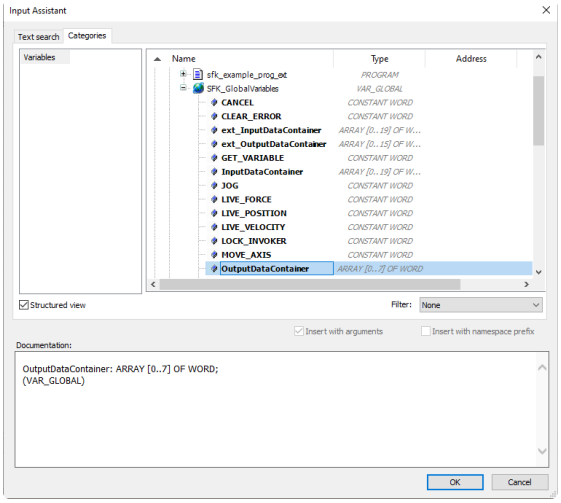
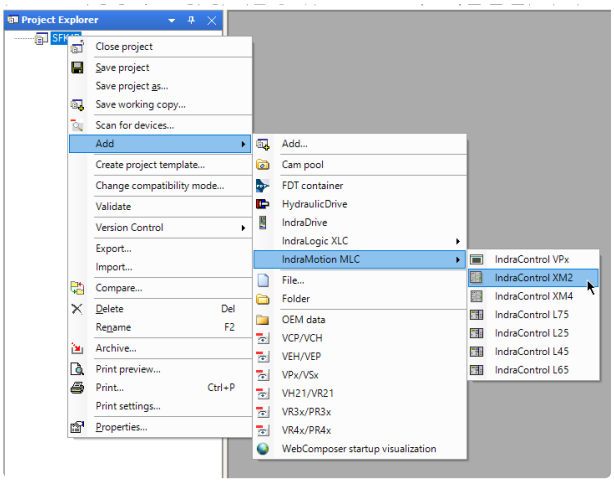
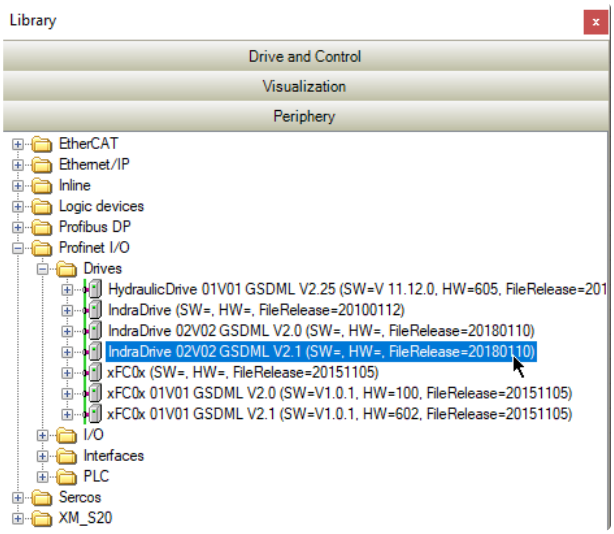
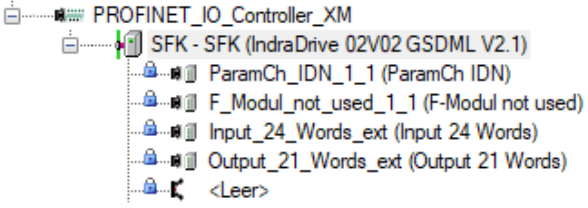
Screens/comments	Action steps
	→ Select the relevant array element of the input or output container.

Table 19 Sercos fieldbus configuration

Profinet

To integrate the Smart Function Kit into a Bosch Rexroth AG XM2 control using the Profinet fieldbus protocol, proceed as follows:

Screens/comments	Action steps
	**First step** → Start IndraWorks and add a new project (empty project). → Press the right mouse button. → Navigate in the menu to *Add* → *IndraMotion MLC*. → Select *IndraControl XM2*.
	Second step Add Profinet IO controller XM . → In the project tree, navigate to the *RT_Ethernet...* tab. → Press the right mouse button. → In the menu, navigate to *Set device*. → Select *Profinet IO Controller XM* .

Screens/comments	Action steps
	Third step Add the HCS01 as a fieldbus slave to the Profinet IO controller XM.
	First procedure: → Navigate in the project tree to *Profinet IO Controller XM*. → Press the right mouse button. → In the menu, navigate to *Add* → *Device* → *Drives*. → Select *IndraDrive 02V02 GSDML V2.1* as a fieldbus slave.
	**Second procedure:** → Select in the tab *View* the *library*. → In the *Library*, click the *Periphery* tab. → In the peripherals tree, navigate to *ProfinetIO* → *Drives*. → Drag *IndraDrive 02V02 GSDML V2.1* into the project tree.
	→ Enter in the corresponding tab *General* under station name *SFK* . **Notice** During activation of the fieldbus within the SFK software, a different station name can be specified. The case is not sensitive.
	In the project tree, *HCS01* (Profinet slave) under the *Profinet IO controller* is displayed.

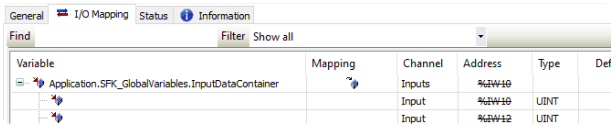
Screens/comments	Action steps
	Notice When using Profinet, the IndraDrive automatically obtains its IP address from the fieldbus master. The station name must be entered correctly. The following standard IP addresses are assigned by the fieldbus master: • PROFINET_IO_Cotroller_XM (Master): 192.168.2.1 / 255.255.255.0 • SFK (IndraDrive 02V02) (Slave): 192.168.2.2 / 255.255.255.0 In order to access the fieldbus data (inputs and outputs), the corresponding modules of the fieldbus slave must be configured or added as follows: 1. ParamCH_IDN 2. Input_24_Words 3. Output_21_Words
	**Fourth step** → Assign the inputs and outputs of the fieldbus to the internal variables via the tab *I/O mapping*. The fieldbus master is accessing the fieldbus data.

Table 20 Profinet fieldbus configuration

EtherCAT

→ To integrate the SFK into an XM2 control from Bosch Rexroth AG using the EtherCAT fieldbus protocol, refer to the Sercos III configuration described above.

Due to the standard configuration of the device description file, the two parameters "Master control word" (S-0-0134) and "Drive status word" (S-0-0135) must be added in addition to the configuration shown above - however, no mapping is necessary.

→ Leave the parameters empty. The SFK does not access these parameters.

Please refer to the figure below:

PDO Content (S-0-0016)

Index	Size	Offs	Name	Type
S-0-0135	2.0	0.0	statusword	WORD
P-0-1414	2.0	2.0	responseHandle	WORD
P-0-1415	2.0	4.0	responseStatus	WORD
P-0-1290	4.0	6.0	responseValue1	DWORD
P-0-1291	4.0	10.0	responseValue2	DWORD
P-0-1292	4.0	14.0	responseValue3	DWORD
P-0-1416	2.0	18.0	notificationID	WORD
P-0-1293	4.0	20.0	notificationValue	DWORD
P-0-1417	2.0	24.0	notificationStatus	WORD
S-0-0051	4.0	26.0	notificationPosition	DWORD
S-0-0040	4.0	30.0	notificationVelocity	DWORD
P-0-1294	4.0	34.0	notificationForce	DWORD
P-0-1418	2.0	38.0	I/OS Mapping	WORD
P-0-1419	2.0	40.0	virtual_io_in	WORD
P-0-1272	4.0	42.0	responseVariable	DWORD
P-0-1273	4.0	46.0	responseVariableValue	DWORD
		50.0		

Figure 62 EtherCAT Master input data

PDO Content (S-0-0024)

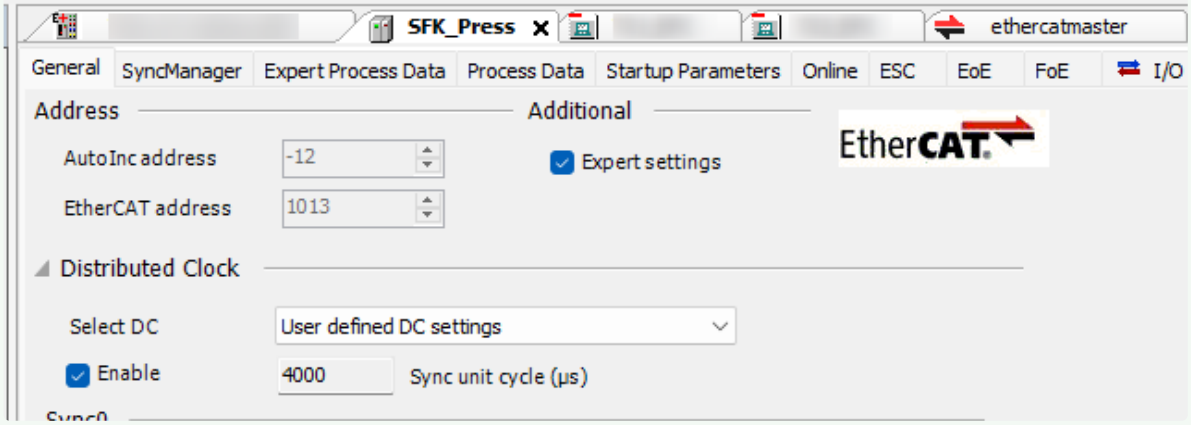
Index	Size	Offs	Name	Type
S-0-0134	2.0	0.0	controlword	WORD
P-0-1394	2.0	2.0	requestHandle	WORD
P-0-1395	2.0	4.0	requestAction	WORD
P-0-1287	4.0	6.0	requestValue1	DWORD
P-0-1288	4.0	10.0	requestValue2	DWORD
P-0-1289	4.0	14.0	requestValue3	DWORD
P-0-1274	4.0	18.0	requestValue4	DWORD
P-0-1275	4.0	22.0	requestValue5	DWORD
P-0-1276	4.0	26.0	requestValue6	DWORD
P-0-1295	4.0	30.0	requestValue7	DWORD
P-0-1396	2.0	34.0	virtual_io_out	WORD
P-0-1270	4.0	36.0	requestVariableType	DWORD
P-0-1271	4.0	40.0	requestValueValue	DWORD
		44.0		

Figure 63 EtherCAT Master output data

If you are using a third-party control, you can also use the diagram shown above as a guide.

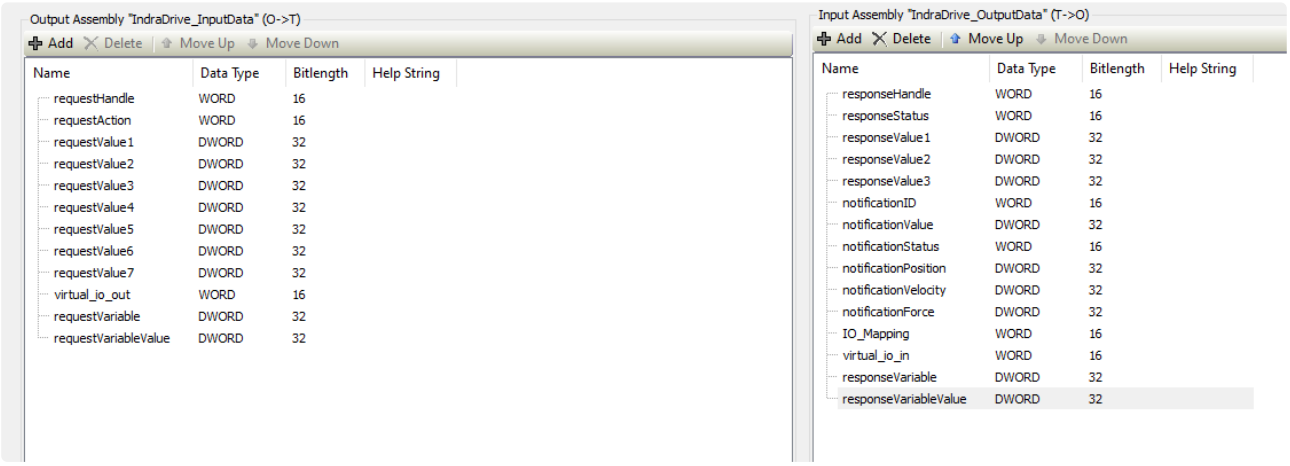
Notice

Disable "Distributed Clock" in the control; use the "Custom DC Settings" option.
Make sure that a cycle time of 4000 µs or greater is specified.



Ethernet/IP

To integrate the SFK into an XM2 control using the EtherNet/IP fieldbus protocol, it is necessary to configure the fieldbus WORDS from chapter Fieldbus configuration in the master control. As a rule, the exact designation of the parameters (S-0-XXXX or P-0-XXXX) is not necessary. The figure below shows the configuration from IndraWorks Engineering.



16.3.2 TIA Portal

Notice

→ For communication via Profinet, assign the Profinet IP address exclusively via the TIA Portal.
The engineering IP address of port X22/X23 must not match the Profinet IP address!

The head module "IndraDrive 02V02 GSDML V2.1" must be used to configure the interface. You can find the download on the Rexroth website (see chapter Fieldbus configuration). The following subordinate modules must be used:

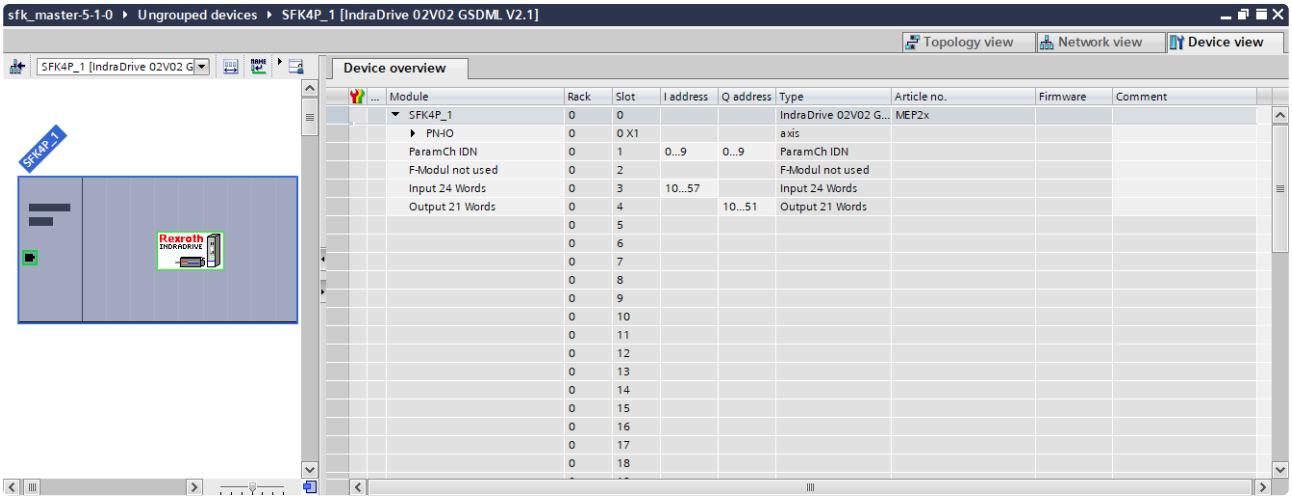


Figure 64 Module assignment of the interface

In contrast, when using ProfiSafe, the F-module must be assigned the desired configuration.

◊ For more information, see chapter Configuration instructions for safe fieldbus communication.

An example project for the Siemens TIA Portal can be downloaded from the user interface of the Smart Function Kit Pressing (see chapter Help). It contains modules and data types that facilitate communication via Profinet with the Smart Function Kit (see chapter TIA Portal).

16.4 Communication

Control via the fieldbus is carried out using the following output and input values:

Output **(Request Container)**	**Input** **(Response Container)**
Requests • Handle • Action • Input 1 - 7	Responses • Handle • Status • responseValue 1 - 3
	Notifications • notificationID • notificationValue • notificationStatus * • notificationPosition * • notificationVelocity * • notificationForce *
	Digital I/Os • digital IO Mapping *
Virtual Outputs • virtual_io_out	Virtual Inputs • virtual_io_in *

Output (Request Container)	Input (Response Container)
Fast Variable Access • request_variable_type • request_variable_value	Fast Variable Access • responseVariable • responseVariableValue

For each request, the response corresponding to the request is sent.
Notifications are divided into cyclic (*) and acyclic information. The digital I/Os and virtual inputs are also updated cyclically.

The following figure shows a schematic representation of how communication works for requests and responses.

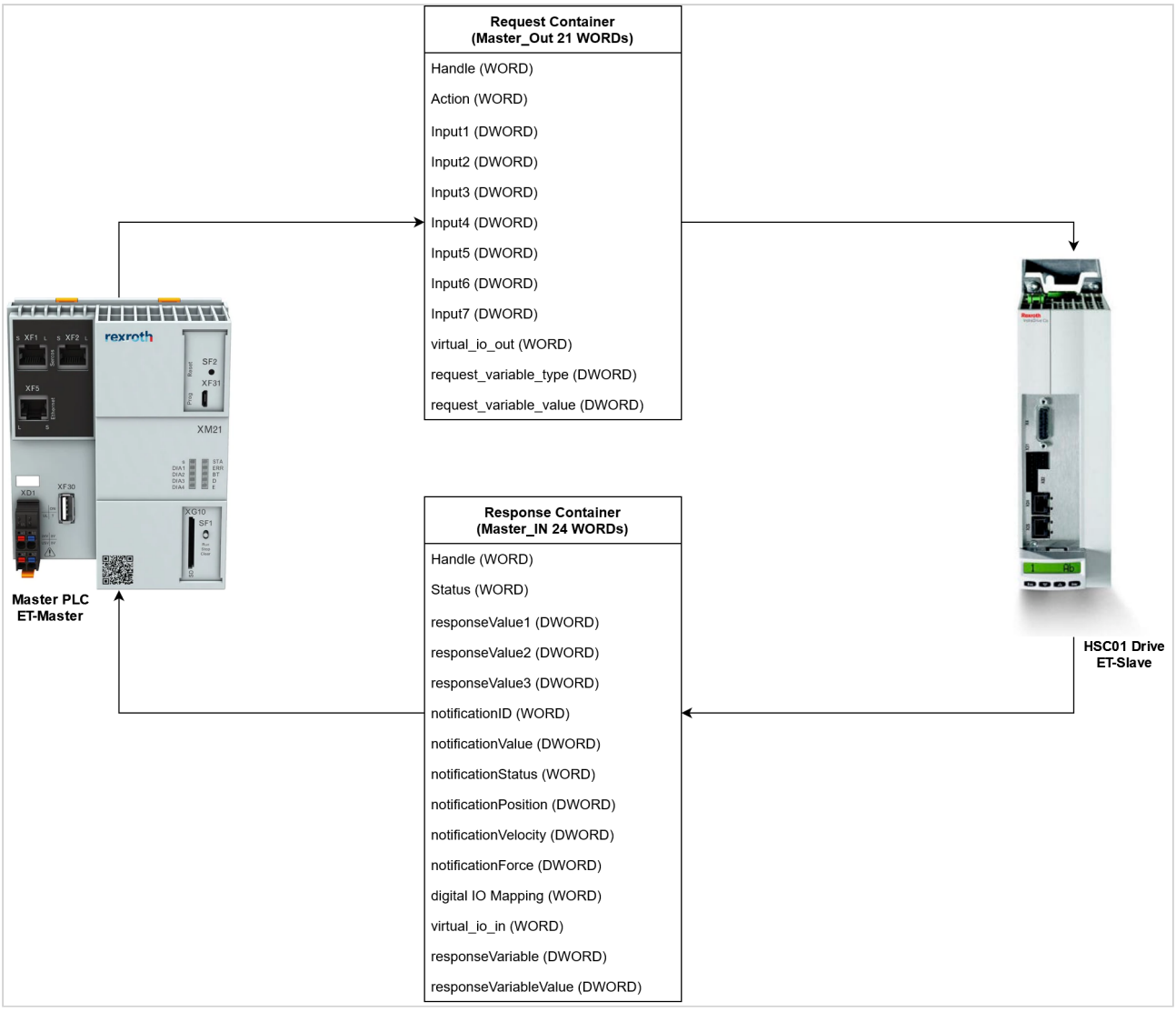


Figure 65 Schematic structure of the fieldbus communication for requests and responses

The *Request Container* is continuously evaluated by the Smart Function Kit.
If it is recognized that the *Handle* field has changed, it is checked for a valid request involving the fields *Action*, *Input 1 - 7* fields.

The Handle ID is a freely selectable integer number that is used to identify a request and the associated response.

Notice
Two consecutive commands must necessarily have a different Handle ID, otherwise no unique assignment can be made.

16.4.1 Requests and responses

The schematic structure of a request consists of the following values:

- Handle ID (**requestHandle**) for unique identification of the command
- Command ID (**requestAction**) to determine which command should be executed (see table below Requests)
- Seven transfer values (**requestValues**)

After a request has been completely processed, a response is issued. As soon as the **responseHandle** corresponds to the **requestHandle** , the answer is there. It consists of the following values:

- Handle ID identical to the respective request (**responseHandle**)
- Final status of the request (**responseStatus**). The value can be interpreted as follows:
  >responseStatus = 0 → no error during processing. In the SFK_Command block, the output "Done" is set to TRUE.
  responseStatus <> 0 → Error, the status value corresponds to the error number of the error that occurred (see chapter System errors). In the SFK_Command block, the "Error" output is set to TRUE.
- Three response values (**responseValues**)

Except for command 6 "Cancel command", only one command can be processed at a time. The interface is blocked for further commands at the time of processing (see chapter System errors → 20101 / 0x4E85).

The following table Requests contains a list of all available commands that can be sent via the fieldbus.

More information can be found in the descriptions of the commands under the table.

ID	Command	requestValue1	requestValue2	requestValue3	requestValue 4-7
1	Start program	*Serial number value 1*	*Serial number value 2*	*Serial number value 3*	*Serial number value 4-7*
2	Set program active	Fieldbus ID	x	x	x
3	Absolute positioning	Target position	Travel speed	Acceleration	x
4	Relative positioning	Distance	Travel speed	Acceleration	x
5	Clear error	x	x	x	x
6	Cancel command	x	x	x	x
7	Restart drive	x	x	x	x

ID	Command	requestValue1	requestValue2	requestValue3	requestValue 4-7
8	Reserved	x	x	x	x
9	Tare	Tare value	x	x	x
10	Start homing	x	x	x	x
11	Set variable	Type	Number	Value	x
12	Lock invoker	Invoker	Lock/ Unlock	x	x
13	Set reference	x	x	x	x
14	Read variable	Type	Variable number	x	x

Table 21 Requests

1 Start program

The command starts the currently active program. The program is completely processed once. After exiting, the corresponding response is provided. The program sequence is considered to be finished when the quality assessment has been completed.

Optionally, a separate serial number can be linked to the data set of a program sequence.
The serial number can consist of up to 28 characters. ASCII characters can be used for the serial number characters, with the exceptions listed below.

Notice
Do not use the following characters in the serial number:

- $
- "
- '
- `
- |
- ~
- {
- }

If you use the characters, the active program will not be started. The error message 10205 is displayed (see chapter System errors).

To transfer the string via Fieldbus, every character must be translated into ASCII code. The bit sequences of these characters must then be written into the REAL values 1-7 of the start command.

A REAL value consists of up to 4 characters. The translated characters of the serial number must be grouped in blocks of four and transferred to the REAL values.

Characters 1-4 are written to value 1, characters 5-8 to value 2, etc., and characters 25-28 to value 7. The bit sequences of a block of four must be written into the value from left to right (see example below).

If the alphanumeric serial number is used, we recommend using the “SFK_SerialNumberConverter” function. The function can be found in the available example projects (see chapter Fieldbus configuration) for various control platforms.

The function converts the string at the input to the 7 required REAL values. The conversion format corresponds to the sequence described above and the returned REAL values can be used without any change as values of the command “Start program”.

Example

Serial number: 'Bosch#1234'

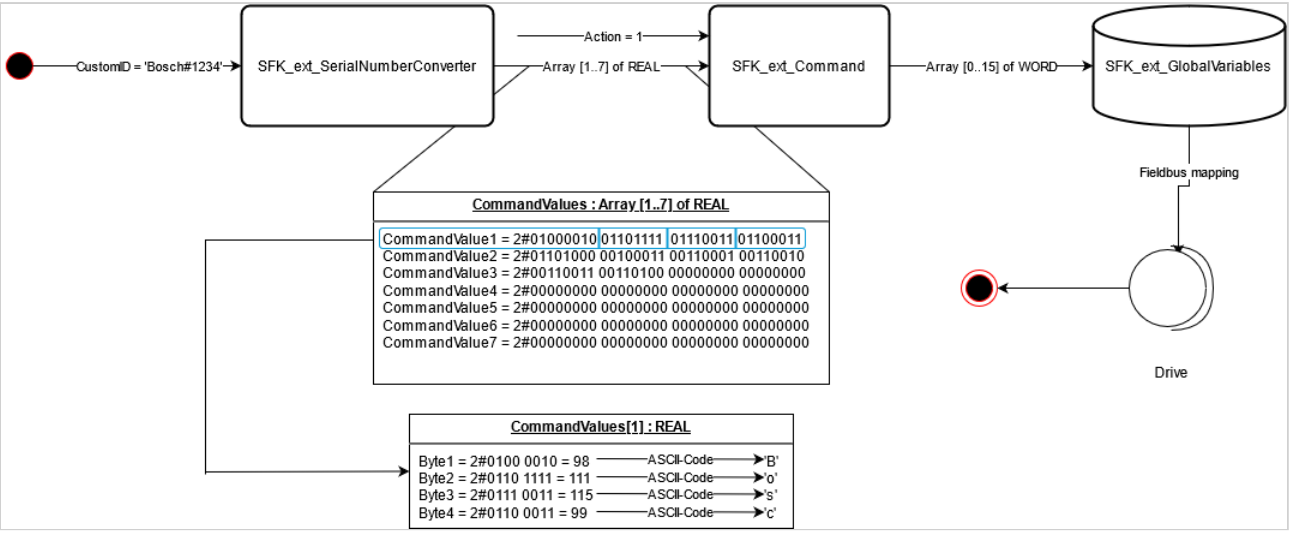


Figure 66 SFK_SerialNumberConverter

Response values (responseValues):

No response values (value = 0)

2 Set program active

The command activates the program specified in value 1. The fieldbus ID corresponds to the fieldbus ID assigned during program creation. If the program is switched, the response is provided.

Response values (responseValues):

No response values (value = 0)

3 Absolute positioning

The command enables movement to a defined target position [in mm]. By means of the parameters *Velocity [in mm/s]* and *Acceleration [in mm/s²]* the velocity profile can be adjusted. Once the target position is reached, the response is provided (see chapter Programming Examples, Example Absolute positioning).

Response values (responseValues):

No response values (value = 0)

4 Relative positioning

The command allows to travel a distance [in mm] relative to the current position. By means of the parameters *Velocity [in mm/s]* and *Acceleration [in mm/s²]* the velocity profile can be adjusted. When the target position is reached, the response is provided.

Response values (responseValues):

No response values (value = 0)

5 Clear error

The command acknowledges all errors at the SFK. Once the command has been processed by the system, the response is provided.

Response values (responseValues):

No response values (value = 0)

6 Cancel command

The command terminates any pending command. If a program is active, it is aborted and the part produced rated with NOK. The cycle time of the program run is set to -1 second. If a movement is active, it is aborted and the drive is stopped. When the drive is stopped, the response is provided.

Response values (responseValues):

No response values (value = 0)

Notice
→ To cancel a pending command: When using the function block *SFK_Command* use the Cancel input of the function block.

7 Restart drive

The command restarts the drive controller.

Response values (responseValues):

No response values (value = 0)

8 Reserved

The command is not assigned.

Response values (responseValues):

No response values (value = 0)

9 Tare

The command sets the actual force value to the specified tare value [in N]. When the command has been completed, the response is provided.

Response values (responseValues):

No response values (value = 0)

10 Start homing

The command starts the homing.

Response values (responseValues):

No response values (value = 0)

Notice
→ Please note the information in chapter Homing.

11 Set variable

The command sets values for variables. To change these values, the system does not go into parameter mode.

◊ For more information about the different variable types, see chapter Variable ranges.

The variable is composed of two arguments. The first value represents the variable type, the second value the variable number.

0: VFR	10: VF
1: MFR	11: MF

Table 22 Encoding the variable types

Notice
VFR and VF variables can only be written when the axis is at standstill (no active sequence).

Example

Writing the VF500 program variable

Value 1: 10 (type: VF)

Value 2: 500 (number: 500)

Value 3: Value (unit corresponding to the target value)

Response values (responseValues):

No response values (value = 0)

Notice
A faster writing of variables than with the Set Variable command is possible with the SFK_Fast_Variable_Access.

12 Lock invoker

The command locks or unlocks the access of an invoker to the functions of the SFK. Locking of an invoker is undone after every restart.

The code for locking or unlocking access consists of two parameters. The first parameter represents the participant, the second parameter represents the process.

Participant	Parameters	Description
Web interface	1	Users of the web interface cannot send commands to the drive. This includes, among other things, changing or starting programs or changing the extended configuration.
Fieldbus	2	No commands are accepted via fieldbus.
Physical inputs and outputs	3	Digital inputs (IX31 interface) cannot trigger commands.

Table 23 Participant parameter

Process	Parameters
Unlock participant	0
Lock participant	1

Table 24 Parameters for locking or unlocking the participants

Example

Value 1: 1 (Web interface)

Value 2: 1 (Lock participant)

Response values (responseValues):

No response values (value = 0)

13 Set reference

The command sets the reference point to the position of the drive.

Notice
The zero position is specified as default value. The value can be changed via the extended configuration globally for the system.

Response values (responseValues):

No response values (value = 0)

14 Read variable

Command reads variables.

◊ For more information about the different variable types, see chapter Variable ranges.

The variable is composed of two arguments. The first value represents the variable type, the second value the variable number.

0: VFR	10: VF
1: MFR	11: MF

Table 25 Encoding the variable types

Example

Reading the VF650 program variable

Value 1: 10 (type: VF)

Value 2: 650 (number: 650)

As response value, the SFK returns the value of the variable VF650 in ResponseValue [1] (unit corresponding to the source value).

The values are only valid after a pressing process and must be read before a new pressing process begins.

The read-out takes place via the NRT channel and may consequently take a shorter or longer time.

Response values (responseValues):

- responseValue1: *Variable value*
- responseValue2: No response value (value = 0)
- responseValue3: No response value (value = 0)

Notice
A response value of -1 signals that the value is invalid (e.g. the cycle time value after program stop).

Notice
A faster reading of variables than with the Read Variable command is possible with the SFK_Fast_Variable_Access.

16.4.2 Notifications

The status of the SFK can be read out via notifications. This enables the targeted control of the SFK via the fieldbus master and the retrieval of relevant process data.

The notifications are divided into cyclical and acyclic notifications.

Acyclic notifications

Acyclic notifications are only transmitted when a specific event occurs or when a specific request is made. The SFK provides all acyclic notifications via the "notificationValue" channel. The meaning of the value of "notificationValue" can be interpreted via the "notificationID".

The following acyclic notifications are available.

ID	Designation	Functional description
301	Active program number	The program number active in the drive is read out. The value is output in the variable "notificationValue" and is updated after a program change.

Table 26 Acyclic notifications

Cyclic notifications

Cyclic notifications are updated in each communication cycle. They include process values and binary status information.

ID	Designation	Functional description
401	Current position (**notificationPosition**)	The cyclically transmitted position is read out.
402	Current velocity (**notificationVelocity**)	The cyclically transmitted velocity is read out.
403	Current force (**notificationForce**)	The cyclically transmitted force is read out.

Table 27 Cyclic notifications - process values

Bit	Description	Value 1
0-1	Reserve	–
2	Axis in reference	1 – The axis has been referenced. 0 – There is no position reference.
3	Program active	1 – Pressing active, program is being processed. 0 – No program active.
4	Last pressing OK	1 – The last pressing yields the result Good. 0 – No result.
5	Last pressing NOK	1 – The last pressing yields the result Bad. 0 – No result.
6	Sensor tared	1 – The force sensor is tared (Offset active). 0 – The force sensor is not tared (no offset active).

Bit	Description	Value 1
7	System ready	1 – The system is ready and can be operated. 0 – Press not ready, check status.
8	Error	1 – Error active. 0 – No error active.
9	Warning	1 – Warning active. 0 – No warning active.
10	Reserve	–
11	Request possible	1 – A request can be sent to the system. 0 – No request can be sent to the system.
12	Response available	1 -– A valid response is available. 0 – Response pending.
13	Notification available	1 – A valid notification is available. 0 – Notification pending.
14	Emergency off active	1 – Emergency off is active. 0 – Emergency off is not active.
15	SMST: Safe standstill active	1 – Safe standstill is active. 0 – Safe standstill is not active.

Table 28 Cyclic Notifications - Binary Status Information

16.4.3 Status of Digital Inputs and Outputs

The current status of each digital input and output can be queried via the word "digital IO mapping" from the input container of the global variable list (see chapter I/O configuration).

The digital signal switches the bit at:

- a high level to the value "1".
- a low level to the value "0".

The assignment is shown in the table below.

Digital input/output	Fieldbus bit from the word "digital IO mapping"
IX31.1	0
IX31.2	1

Digital input/output	Fieldbus bit from the word "digital IO mapping"
IX31.3	2
IX31.4	3
IX31.5	4
IX31.6	5
IX31.7	6
QX31.8	7
Reserve	8-15

Table 29 Assignment of digital input/output to fieldbus bit

16.4.4 Virtual inputs and outputs

The SFK provides virtual inputs and outputs to enable communication between the master control and a program sequence (e.g. for dynamic stops or delays during the program sequence, waiting for external signals from measurement systems or grippers).

For this purpose, 16 bits (characters) are available per direction, which can be written and read at any time.

→ To be able to use the 16 bits: Assign the program variables to the bits in the SFK settings. The program variables can be used when creating the program (see chapter I/O configuration).

The assignment is shown in the table below.

Virtual Input	Fieldbus Bit from word " virtual_io_out"
IV03.0	0
IV03.1	1
IV03.2	2
IV03.3	3
IV03.4	4
IV03.5	5
IV03.6	6
IV03.7	7

Virtual Input	Fieldbus Bit from word " virtual_io_out"
IV03.8	8
IV03.9	9
IV03.10	10
IV03.11	11
IV03.12	12
IV03.13	13
IV03.14	14
IV03.15	15

Table 30 Virtual Inputs (SFK view)

Virtual Output	Fieldbus bit from the word " virtual_io_in"
QV06.0	0
QV06.1	1
QV06.2	2
QV06.3	3
QV06.4	4
QV06.5	5
QV06.6	6
QV06.7	7
QV06.8	8
QV06.9	9
QV06.10	10
QV06.11	11

Virtual Output	Fieldbus bit from the word " virtual_io_in"
QV06.12	12
QV06.13	13
QV06.14	14
QV06.15	15

Table 31 Virtual outputs (SFK view)

16.4.5 Variable interface "Fast Variable Access"

The variable interface "Fast Variable Access" is used to read and write certain variables quickly. For the usable variables, the function represents a faster alternative to command 11 ("Set variable") and command 14 ("Read variable"). Depending on the implementation, access time can be reduced by up to 90%.

The interface can be used in parallel to the command processing from chapter Requests and responses. Writing program variables (see chapter Variable ranges) during program execution is not provided. The variables that can be used are limited. Please note the information in the later chapters regarding the function block offered (see chapter Programming examples). If the function of the fast variable interface is to be used, the use of the provided function block is mandatory.

Notice
The use of the fast variable interface is only possible with the provided function block "SFK_Fast_Variable_Access".
→ Do not implement on your own.

Notice
Dynamic changes to program variables during program execution are not possible.
→ Write values of the program variables before the program starts.

16.5 Programming examples

16.5.1 Integrate/use example projects

For the Indra Works Engineering, TIA Portal and Studio 5000 Logix Designer engineering environment, the programming examples of the example projects provided can be used, where applicable, to integrate the functions/function blocks.

→ Please contact the service if a different engineering environment is to be used. Additional example projects are available on request.

The example projects can be downloaded from the user interface.

→ To do this, click the Help button in the dashboard.

→ Select the appropriate example project.

The example projects are also available for download in the Rexroth Store under "Further information" → "Function blocks": https://www.boschrexroth.com/product/cls_eCommerce_lx_1555747

16.5.2 IndraWorks Engineering

An example project for IndraWorks Engineering from Bosch Rexroth AG can be downloaded from the Smart Function Kit Pressing user interface. It contains function blocks and data types that facilitate communication with the Smart Function Kit.

In the example, global arrays of the Word data type are created so that the master control can access the fieldbus data. The following scheme was used:

- For the input data:
  - an array with 20 elements (cyclic mapping variables)
  - an array with 4 elements (Cyclic mapping variables for SFK_Fast_Variable_Access)
- For the output data:
  - an array with 16 elements (cyclic mapping variables)
  - an array with 4 elements (Cyclic mapping variables for SFK_Fast_Variable_Access)
  - a WORD for virtual outputs

The assignment of the input and output data of the fieldbus to the array elements (mapping) can be found in the example project provided.

The following excerpt shows a variable list taken from the global variable list *SFK_GlobalVariables* in the SFK folder.

Listing: Excerpt from global variables list "SFK_GlobalVariables"

```
VAR_GLOBAL
    // Cyclic mapping variables
    InputDataContainer: ARRAY[0..19] OF WORD;
    OutputDataContainer: ARRAY[0..15] OF WORD;

    // Fieldbus Variables
    virtualOutputs: WORD;

    // Cyclic mapping variables for SFK_Fast_Variable_Access
    InputDataContainer_SFK_Fast_Access: ARRAY [0..3] OF WORD;
    OutputDataContainer_SFK_Fast_Access: ARRAY [0..3] OF WORD;
END_VAR
```

Function blocks

SFK_Command

Der SFK_Command function block contains the abstraction layer for the Smart Function Kit.

The request and response scheme is implemented within this block to simplify operation.

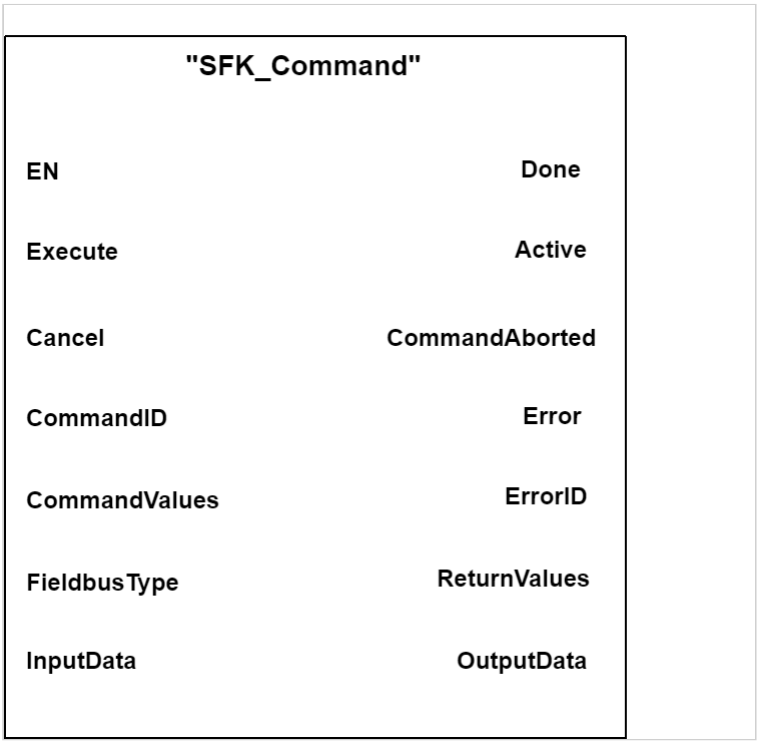


Figure 67 SFK_Command function block

Inputs:

Name	Data type	Description
EN (optional)	BOOL	Activation of the block
Execute	BOOL	Executing a command on a rising edge
Cancel	BOOL	Canceling a pending request sent via the function block
CommandID	WORD	Selected command (see chapter Requests and responses)
CommandValues	ARRAY[1..7] OF REAL	Command-specific input values
FieldbusType (optional)	UINT	Selection of the fieldbus protocol • 0 = no selection • 1 = Sercos, EtherCAT, EtherNetIP • 2 = Profinet
InputData	ARRAY[0..19] OF WORD	Cyclic data exchange container

Table 32 Inputs

Outputs:

Name	Type	Description
Done	BOOL	TRUE if the command was completely processed Reset to FALSE if Input “Execute” was set to FALSE
Active	BOOL	TRUE as long as the command is active
CommandAborted	BOOL	TRUE if the termination was successful Reset to FALSE if Input “Execute” was set to FALSE
Error	BOOL	TRUE if the current command was acknowledged with an error Reset to FALSE if Input “Execute” was set to FALSE
ErrorID	WORD	Error number (see chapter Error catalog) Reset to 0 if Input “Execute” was set to FALSE
ReturnValues	ARRAY[1..3] OF REAL	Command-specific return values Reset to 0 if Input “Execute” was set to FALSE
OutputData	ARRAY[0..15] OF WORD	Cyclic data exchange container

Table 33 Outputs

Notice
When a program starts, the "Done" output is set as soon as the program has been completely processed (including evaluation and storage).

SFK_Fast_Variable_Access

The function block "SFK_Fast_Variable_Access" is used for fast reading and writing of selected variables (e.g. for reading process data for a press-in process).

The function block is a supplement to the "SFK_Command" block.

Notice
The use of the fast variable interface is only possible with the provided function block "SFK_Fast_Variable_Access".
→ Do not implement on your own.

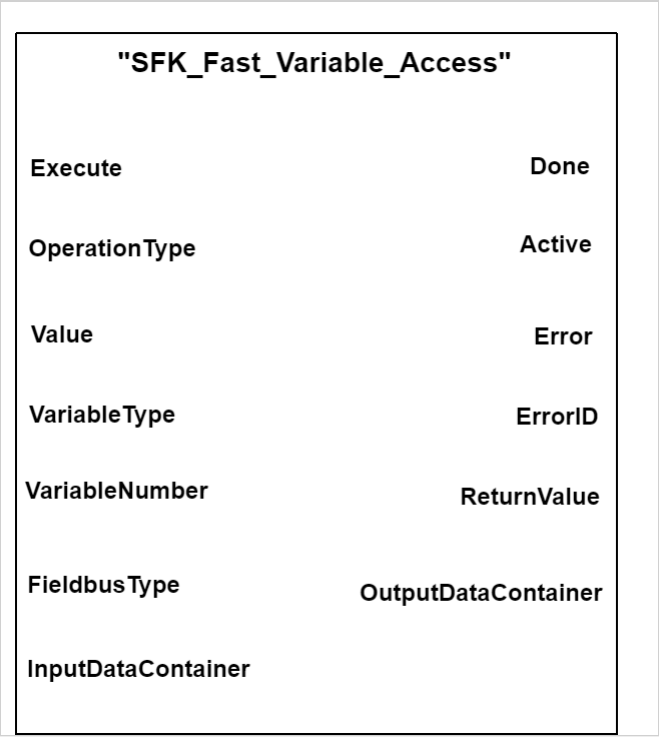


Figure 68 SFK_Fast_Variable_Access

Inputs:

Name	Data type	Description
Execute	BOOL	Used to execute a command on a rising edge
OperationType	BOOL	Used to select the desired functionality: • FALSE = Read variable • TRUE = Write variable
Value	REAL	Value that is written to the variable
VariableType	STRING	Used to select the type of variable (VF, VFR, MF, MFR)
VariableNumber	UINT	Used to select the variable number: • Read valid values: 0 - 999 • Write valid values: 200 - 999

Name	Data type	Description
FieldbusType	UINT	Used to select the fieldbus protocol: • 0 = no selection (preselection) • 1 = Sercos, EtherCAT, EtherNetIP • 2 = Profinet This input is not available in the TIA Portal because it is pre-assigned
InputDataContainer	ARRAY[0..3] OF WORD	Cyclic data exchange container

Table 34 Inputs

Outputs:

Name	Type	Description
ErrorID	WORD	Error number (see chapter Error catalog) Used to reset to 0 as soon as input "Execute" is set to FALSE
ReturnValue	REAL	Variable-specific response values Used to reset to 0 as soon as input "Execute" is set to FALSE
Active	BOOL	TRUE as long as the command is active
Error	BOOL	TRUE if the current command is acknowledged with an error Used to reset to FALSE as soon as input "Execute" is set to FALSE
Done	BOOL	TRUE if the command has been completely processed Used to reset to FALSE as soon as input "Execute" is set to FALSE
OutputDataContainer	ARRAY[0..3] OF WORD	Cyclic data exchange container

Table 35 Outputs

⚠ DANGER

Unwanted axis movement

The unauthorized writing of reserved variables can lead to unwanted axis movements.

→ Use the attached example function block.

→ Do not make any changes to the function block.

→ When using the interface manually, make sure that variables from 0 - 199 cannot be written.

Example Absolute positioning

→ To move to a target position with the SFK: Start the Positioning command via the Request channel.

The Handle ID is a freely selectable integer number that is used to identify a request and the associated response. The ID can be used to check whether the current response matches the request.

Two consecutive commands must necessarily have a different Handle ID, otherwise no unique assignment can be made.

Sending the request

Handle ID	Command	Value 1	Value 2	Value 3	Value 4-7
151	3	50000	150000	150000	-
Word 1	Word 2	Word 3 Word 4	Word 5 Word 6	Word 7 Word 8	-

Table 36 Request scheme Request Container

Example

Value 1: 50 (mm, target position)

Value 2: 150 (mm/s, velocity)

Value 3: 150 (mm/s², acceleration)

→ To execute the above command with the function block SFK_command: Assign the inputs as follows.

```
// Passing the input values
sfkCommand.Command := 3;
sfkCommand.CommandValues[1] := 50;
sfkCommand.CommandValues[2] := 150;
sfkCommand.CommandValues[3] := 150;
// Sending the request
sfkCommand.Execute := TRUE;
```

Notice
The conversion of the values 1 to 3 (multiplication by 1000) is carried out internally in the function block.

Response

If the command was successfully executed and the target position is reached, in the *(Response)container* the following response scheme is displayed:

Handle ID	Status	Value 1	Value 2	Value 3
151	0	0	0	0
Word 1	Word 2	Word 3 Word 4	Word 5 Word 6	Word 7 Word 8

Table 37 Response scheme Response container upon successful command execution

Notice
The response is returned as soon as the target position is reached.

Status = 0 → no error

Notice
For the Positioning command, the response values 1 to 3 are always 0.

The SFK_command function block returns the following response values:

sfkCommand.Done = TRUE

sfkCommand.Error = FALSE

sfkCommand.ErrorID = 0

If the command was not executed successfully or an error occurred during execution, in the *(Response)container* the following response scheme is displayed:

Handle ID	Status	Value 1	Value 2	Value 3
151	xxxxxx	0	0	0

Handle ID	Status	Value 1	Value 2	Value 3
Word 1	Word 2	Word 3 Word 4	Word 5 Word 6	Word 7 Word 8

Table 38 Response scheme Response container in case of unsuccessful command execution

Status <> 0 → Error

The status value corresponds to the error number of the error that occurred (see chapter Error catalog).

The SFK_command function block returns the following response values:

sfkCommand.Done = FALSE

sfkCommand.Error = TRUE

sfkCommand.ErrorID = xxxxxx

Notice
The function block is reset by resetting the sfkCommand.Execute bit.

Implementation on the master control

The following examples show how an implementation can be done on the master control.

The examples are intended as a guide for programming. Operational adjustments are necessary.

Example program start and termination of the program

The following programming logic shows how a program is started and aborted due to an unexpected event.

For this purpose, the Cancel input of the SFK_Command block is used. This input leads to an immediate abort of a currently pending command. This is followed by a manual return stroke as an abort reaction.

```
CASE State OF

        STATE_WAI_AH:
            // Check if the SFK is in the "Ready" state
            IF SFK_GlobalVariables.InputDataContainer[11].7 THEN
                State :=  STATE_START_PROG1;
            END_IF

        STATE_START_PROG1:
            // Command "Start program"
            sfkCommand.Command := 1;
            // Execute the command
            sfkCommand.Execute := TRUE;
            State := STATE_CHECK1;
```

```
STATE_CHECK1:
    // Check if the command was executed successfully
    IF sfkCommand.Done THEN
        // Reset the block sfkCommand
        sfkCommand.Execute := FALSE;
        // Jump to the next command
        State := STATE_START_PROG1;
    // Check whether an error occurred during the execution of the command
    ELSIF sfkCommand.Error THEN
        State := STATE_ERROR;
    // Check whether the command was aborted via the sfkCommand block
    ELSIF sfkCommand.CommandAborted THEN
        State := STATE_ABORTED;
    END_IF

STATE_ABORTED:
    // Check if the program has been completely aborted
    IF sfkCommand.CommandAborted AND NOT
SFK_GlobalVariables.InputDataContainer[11].3 THEN
        // Reset the inputs of the block
        sfkCommand.Execute := FALSE;
        sfkCommand.Cancel := FALSE;
    // Check if the block has been reset
    ELSIF sfkCommand.resetDone THEN
        // Jump to manual return stroke
        State := STATE_MOVE_BACK;
    END_IF

STATE_MOVE_BACK:
    // Command "Absolute positioning"
      sfkCommand.Command := 3;
      // absolute target position "0 mm"
      sfkCommand.CommandValues[1] := 0;
      // Speed "50 mm/s"
      sfkCommand.CommandValues[2] := 50;
      //Acceleration "500 mm/s²"
      sfkCommand.CommandValues[3] := 500;
      // Execute the command
      sfkCommand.Execute := TRUE;
```

```
        STATE_ERROR:
            // Error response
            ;

END_CASE

// Query the condition that leads to a Cancel command
IF Cancel THEN
    // Set the Cancel input to cancel a currently executed command
    sfkCommand.Cancel := TRUE;
    // Switch to cancel reaction
    State := STATE_ABORTED;
END_IF

// Cyclic call of the block
sfkCommand(
    FieldbusType := FieldbusType ,
    InputDataContainer:= SFK_GlobalVariables.InputDataContainer,
    OutputDataContainer=> SFK_GlobalVariables.OutputDataContainer);
```

Example program start with subsequent quality assessment

The following program logic shows how a program is selected and started. After starting, the program is repeated as long as the cycles are evaluated as OK. During a NOK assessment, manual positioning is performed and the process is interrupted.

```
CASE State OF

        STATE_WAI_AH:
            // Check if the SFK is in the "Ready" state
            IF SFK_GlobalVariables.InputDataContainer[11].7 THEN
                State :=  STATE_SET_PROG1;
            END_IF

        STATE_SET_PROG1:
            // Command "Set program active"
            sfkCommand.Command := 2;
            // Program ID = 1
            sfkCommand.CommandValues[1] := 1;
            // Execute the command
```

```
            sfkCommand.Execute := TRUE;
            State := STATE_CHECK1;

STATE_CHECK1:
            // Check if the command was executed successfully
            IF sfkCommand.Done THEN
                  // Reset the block sfkCommand
                  sfkCommand.Execute := FALSE;
                  // Jump to the next command
                  State := STATE_START_PROG1;
            // Check whether an error occurred during the execution of the command
            ELSIF sfkCommand.Error THEN
                  State := STATE_ERROR;
            // Check whether the command was aborted via the sfkCommand block
            ELSIF sfkCommand.CommandAborted THEN
                  State := STATE_ABORTED;
            END_IF

STATE_START_PROG1:
          // Command "Start program"
            sfkCommand.Command := 1;
            // Execute the command
            sfkCommand.Execute := TRUE;
            State := STATE_CHECK2;

STATE_CHECK2:
            // Check if the command was executed successfully
            IF sfkCommand.Done THEN
                  //Check if result is OK
                  If SFK_GlobalVariables.InputDataContainer[11].4 THEN
                     // Reset the block sfkCommand
                     sfkCommand.Execute := FALSE;
                     // Jump to the next command
                     State := STATE_START_PROG1;
                  // Check if result NOK
                  ELSIF SFK_GlobalVariables.InputDataContainer[11].5 THEN
                        // Reset the block sfkCommand
                        sfkCommand.Execute := FALSE;
                        // Jump to error handling
                        State := STATE_MOVE_BACK;
```

```
                    END_IF
                // Check whether an error occurred during the execution of the command
                ELSIF sfkCommand.Error THEN
                    State := STATE_ERROR;
                // Check whether the command was aborted via the sfkCommand block
                ELSIF sfkCommand.CommandAborted THEN
                    State := STATE_ABORTED;
                END_IF

            STATE_MOVE_BACK:
                // Command "Absolute positioning"
                sfkCommand.Command := 3;
                // absolute target position "10 mm"
                sfkCommand.CommandValues[1] := 10;
                // Speed "50 mm/s"
                sfkCommand.CommandValues[2] := 50;
                //Acceleration "500 mm/s²"
                sfkCommand.CommandValues[3] := 500;
                // Execute the command
                sfkCommand.Execute := TRUE;

            STATE_ERROR:
                // Error response
                ;

            STATE_ABORTED:
                // Abort reaction
                ;
END_CASE

// Cyclic call of the block
sfkCommand(
    FieldbusType := FieldbusType ,
    InputDataContainer:= SFK_GlobalVariables.InputDataContainer,
    OutputDataContainer=> SFK_GlobalVariables.OutputDataContainer);
```

Example program start and reading of significant curve data

The following programming logic shows how a program is selected and started. After the start, the maximum force and the maximum position of the cycle are read out. For this purpose, the Fast_Varible_Access block is used in addition to the SFK_Command block. The program will then be restarted.

```
CASE State OF

        STATE_WAI_AH:
            // Check if the SFK is in the "Ready" state
            IF SFK_GlobalVariables.InputDataContainer[11].7 THEN
                State :=  STATE_START_PROG1;
            END_IF

        STATE_START_PROG1:
            // Command "Start program"
            sfkCommand.Command := 1;
            // Execute the command
            sfkCommand.Execute := TRUE;
            State := STATE_CHECK1;

        STATE_CHECK1:
            // Check if the command was executed successfully
            IF sfkCommand.Done THEN
                // Reset the block sfkCommand
                sfkCommand.Execute := FALSE;
                // Jump to the next command
                State := STATE_READ_VALUE_FORCE;
            // Check whether an error occurred during the execution of the command
            ELSIF sfkCommand.Error THEN
                State := STATE_ERROR;
            END_IF

        STATE_READ_VALUE_FORCE:
            // Set the block input to the value for max. force
            sfk_Fast_Variable_Access.VariableNumber := 600;
            // Execute the read command
            sfk_Fast_Variable_Access.Execute := TRUE;

            // Check if the query was executed successfully
            IF sfk_Fast_Variable_Access.Done THEN
                // Writing the response value of the block into the variable maxForce
                maxForce := sfk_Fast_Variable_Access.ReturnValue;
                //Jump to program start
```

```
                State := STATE_START_PROG1;
            // Check if the query resulted in an error
            ELSIF sfk_Fast_Variable_Access.Error THEN
                State       := STATE_ERROR;
            END_IF

END_CASE

// Cyclic call of the blocks
sfkCommand(
    FieldbusType := FieldbusType ,
    InputDataContainer:= SFK_GlobalVariables.InputDataContainer,
    OutputDataContainer=> SFK_GlobalVariables.OutputDataContainer);

sfk_Fast_Variable_Access(
    FieldbusType := FieldbusType ,
    VariableType := VariableType,
    OperationType := FALSE,
    InputDataContainer := SFK_GlobalVariables.InputDataContainer_SFK_Fast_Access,
    OutputDataContainer => SFK_GlobalVariables.OutputDataContainer_SFK_Fast_Access);
----------
CASE State OF

STATE_WAI_AH:
        // Check if the SFK is in the "Ready" state
        IF SFK_GlobalVariables.InputDataContainer[11].7 THEN
            State :=  STATE_START_PROG1;
        END_IF
```

16.5.3 TIA Portal

An example project for the Siemens TIA Portal can be downloaded from the user interface of the Smart Function Kit Pressing in the Help area. It contains blocks and data types that facilitate communication via Profinet with the Smart Function Kit.

A Siemens control and an SFK have already been created in the project and the fieldbus communication has been configured. The organizational block shows an example of how the blocks are called. The input and output parameters for the basic function of the blocks are already interconnected. All other parameters can be individually linked to your own variables depending on the application.

The fieldbus communication is realized via 24 input and 21 output words. To simplify assignment, separate data types were created for the inputs and outputs, which were assigned to the respective fieldbus addresses as PLC variables in the "PLC_RawData" tab.

The fieldbus variables and their data types are described in more detail in the following tables:

Offset	Name	Data type	Description
0	Command	SFK_typeCmdResponse	Variables for communication via the function block SFK_Command
16	Reserve	WORD	Reserved
18	ProgramID	DWORD	ID of the loaded SFK program * 1000
22	SystemState	SFK_typeSystemState	SFK-Status Bits
24	Position	DWORD	Position in 1/10000 mm
28	Velocity	DWORD	Velocity in 1/1000 mm/min
32	Force	DWORD	Force in 1/1000 N
36	IOState	SFK_typeIOState	Status of the physical SFK IOs
38	VirtualOutputState	SFK_typeVirtualOutputState	Status of the virtual SFK outputs
40	FastVarResponse	SFK_typeFastVarResponse	Variables for communication via the FastVarResponse function block

Table 39 SFK output / master input

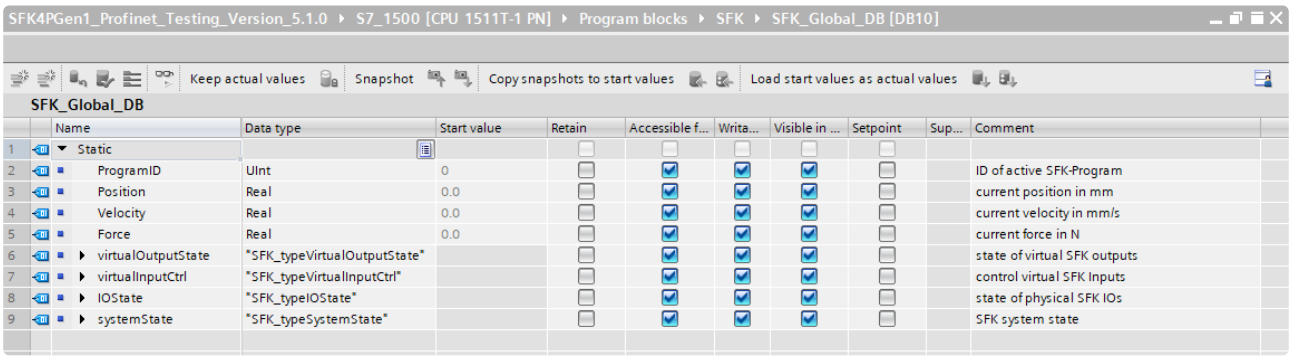
Offset	Name	Data type	Description
0	Command	SFK_typeCmdRequest	Variables for communication via the function block SFK_Command
32	VirtualInputCtrl	SFK_typeVirtualInputCtrl	Control of virtual inputs
34	FastVarRequest	SFK_typeFastVarRequest	Variables for communication via the FastVarResponse function block

Table 40 SFK input/master output

SFK_Global_DB

The Global-DB block in the example project is used to map the converted cyclic values (ProgramID, Position, Velocity, Force), virtual and physical inputs/outputs of the SFK and the system status.

The assignments are made in the Main-OB block. This block can be expanded as required depending on the application.



	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Sup...	Comment
1	Static									
2	ProgramID	UInt	0		☑	☑	☑			ID of active SFK-Program
3	Position	Real	0.0		☑	☑	☑			current position in mm
4	Velocity	Real	0.0		☑	☑	☑			current velocity in mm/s
5	Force	Real	0.0		☑	☑	☑			current force in N
6	virtualOutputState	"SFK_typeVirtualOutputState"			☑	☑	☑			state of virtual SFK outputs
7	virtualInputCtrl	"SFK_typeVirtualInputCtrl"			☑	☑	☑			control virtual SFK Inputs
8	IOState	"SFK_typeIOState"			☑	☑	☑			state of physical SFK IOs
9	systemState	"SFK_typeSystemState"			☑	☑	☑			SFK system state

Functions

SFK_ConvertCyclicData

This function converts various values from the cyclic fieldbus data into the corresponding data types and units.

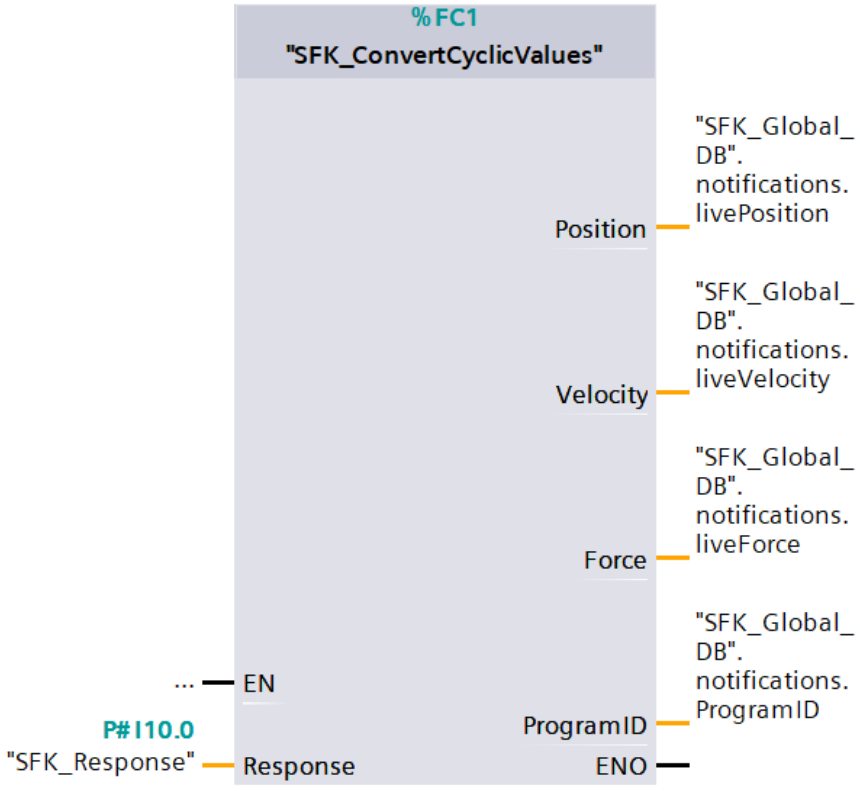


Figure 69 SFK_ConvertCyclicValues

Name	Data type	Description
Response	SFK_typeResponse	Fieldbus data from the SFK

Table 41 Inputs

Name	Data type	Description
ProgramID	UINT	ID of the loaded SFK program
Position	REAL	Position [in mm]
Velocity	REAL	Velocity [in mm/s]
Force	REAL	Force [in N]

Table 42 Outputs

SFK_SerialNumberConverter

This function breaks down a serial number of data type String into ASCII values and outputs them as an array of Real.

This conversion allows the serial number to be processed correctly by the SFK_Command block.

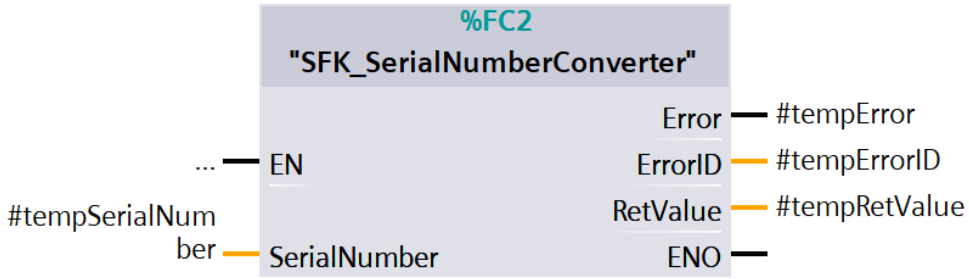


Figure 70 SFK_SerialNumberConverter

Inputs:

Name	Data type	Description
SerialNumber	STRING	Serial number

Table 43 Inputs

Outputs:

Name	Data type	Description
Error	BOOL	TRUE if an error occurred while converting the serial number
ErrorID	WORD	Error number (see chapter Error catalog)
RetValue	ARRAY[0..6] OF REAL	Converted serial number

Table 44 Outputs

Function blocks

SFK_Command

Der SFK_Command function block contains the abstraction layer for the Smart Function Kit.

The request and response scheme is implemented within this block to simplify operation.

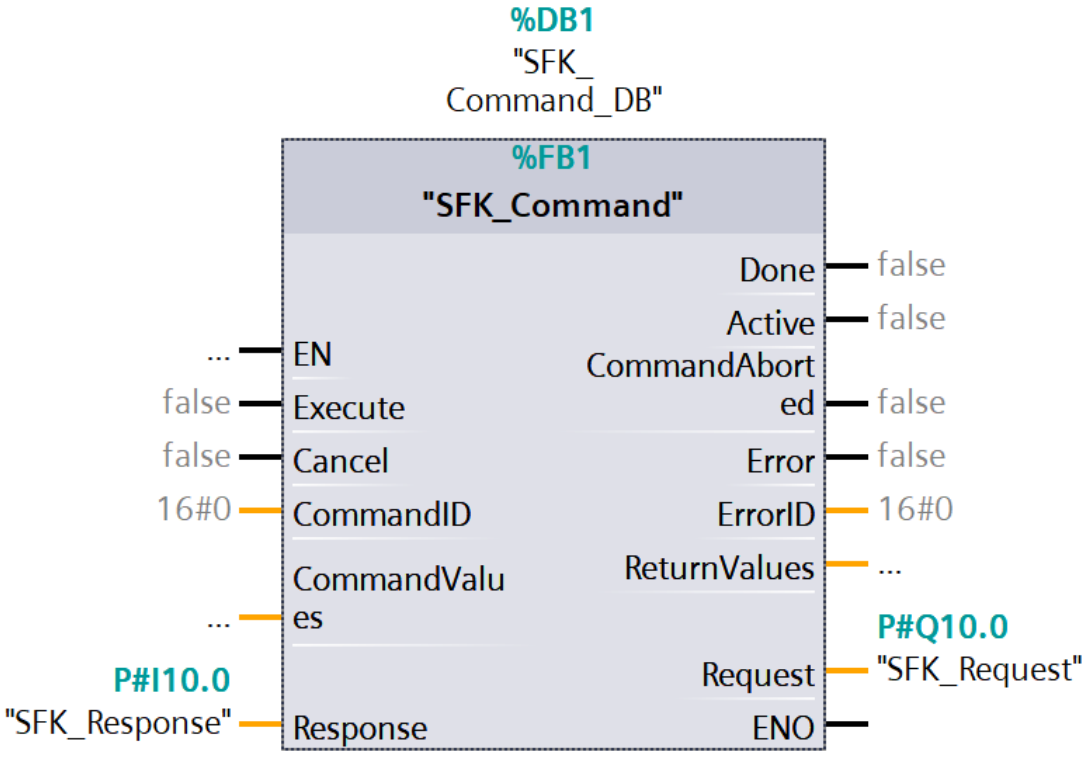


Figure 71 SFK_Command

Inputs:

Name	Data type	Description
Execute	BOOL	Execute the command on rising edge
Cancel	BOOL	Canceling a pending request sent via the function block
CommandID	WORD	ID of the fieldbus command to be executed (see chapter Requests and responses)
CommandValues	ARRAY[1..7] OF REAL	Command-specific input values
Response	SFK_typeResponse	Data exchange with the SFK via fieldbus

Table 45 Inputs

Outputs:

Name	Data type	Description
Done	BOOL	TRUE if the command was completely processed Is reset to FALSE as soon as input "Execute" is set to FALSE
Active	BOOL	TRUE as long as the command is active
CommandAborted	WORD	TRUE if the termination was successful Is reset to FALSE as soon as input "Execute" is set to FALSE
Error	BOOL	TRUE if the current command was acknowledged with an error Is reset to FALSE as soon as input "Execute" is set to FALSE
ErrorID	WORD	Error number (see chapter Error catalog) Is reset to 0 as soon as input "Execute" is set to FALSE
ReturnValues	ARRAY[1..3] OF REAL	Command-specific return values Will be reset to 0 as soon as "Execute" is set to FALSE
Request	SFK_typeRequest	Data exchange with the SFK via fieldbus

Table 46 Outputs

SFK_Fast_Variable_Access

The function block "SFK_Fast_Variable_Access" is used for fast reading and writing of selected variables (e.g. for reading process data for a press-in process).

The function block is a supplement to the "SFK_Command" block.

> **Notice**
> The use of the fast variable interface is only possible with the provided function block "SFK_Fast_Variable_Access".
> → Do not implement on your own.

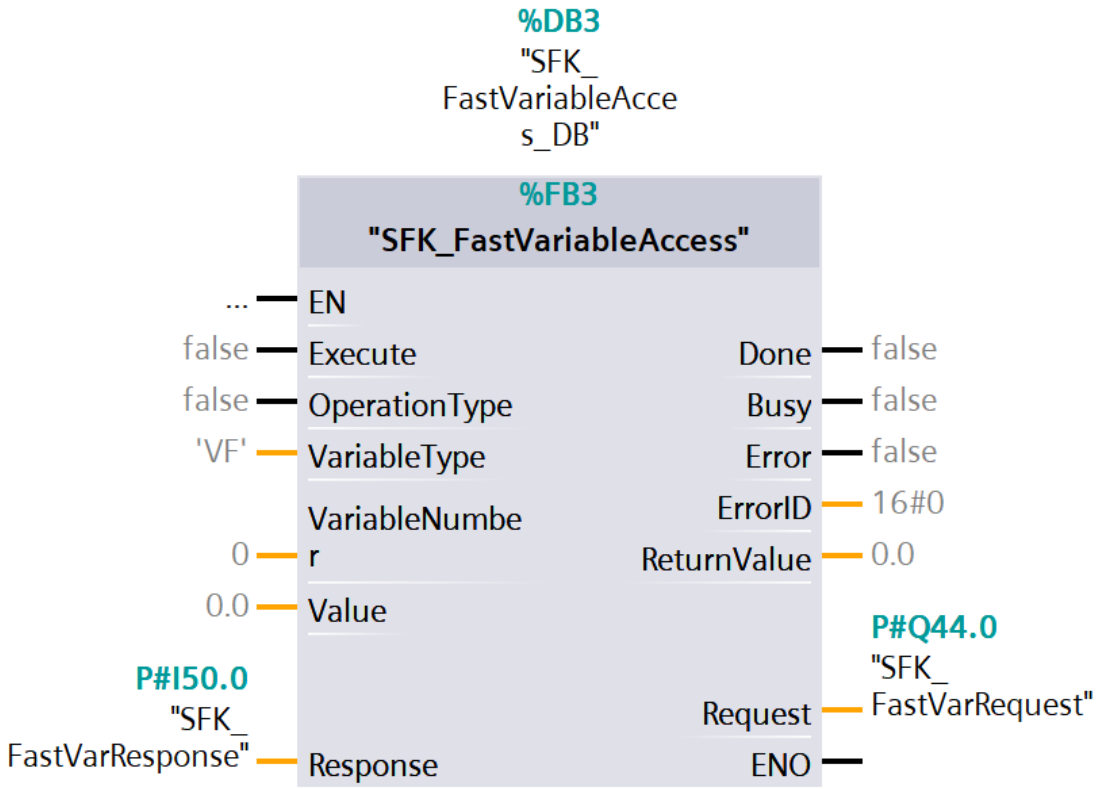


Figure 72 SFK_FastVariableAccess

Inputs:

Name	Type	Description
Execute	BOOL	Used to execute a command on a rising edge
OperationType	BOOL	Used to select the desired functionality: • FALSE = Read variable • TRUE = Write variable

Name	Type	Description
VariableType	STRING	Used to select the type of variable (VF, VFR, MF, MFR).
VariableNumber	UINT	Used to select the variable number: • Read valid values: 0 - 999 • Write valid values: 200 - 999
Value	REAL	value that is written to the variable
Response	SFK_typeFastVarResponse	Response data of the SFK via fieldbus

Table 47 Inputs

Outputs:

Name	Type	Description
Done	BOOL	TRUE if the command has been completely processed Used to reset to FALSE as soon as input "Execute" is set to FALSE.
Busy	BOOL	TRUE if a command is currently being processed
Error	BOOL	TRUE if the current command is acknowledged with an error Used to reset to FALSE as soon as input "Execute" is set to FALSE
ErrorID	WORD	Error number (see chapter Error catalog) Used to reset to 0 as soon as input "Execute" is set to FALSE
ReturnValue	REAL	Variable-specific response values Used to reset to 0 as soon as input "Execute" is set to FALSE
Request	SFK_typeFastVarRequest	Request data to the SFK via fieldbus

Table 48 Outputs

⚠ DANGER

Unwanted axis movement

The unauthorized writing of reserved variables can lead to unwanted axis movements.

→ Use the attached example function block.

→ Do not make any changes to the function block.

→ When using the interface manually, make sure that variables from 0 - 199 cannot be written.

ExampleOperation

This block implements a state machine in which the function block "SFK_Command" is called.

Within the state machine, a step chain is implemented in which first a program change and then a program start is triggered in the SFK.

The implementation via the state machine ensures that multiple fieldbus commands are processed correctly and in an orderly manner.

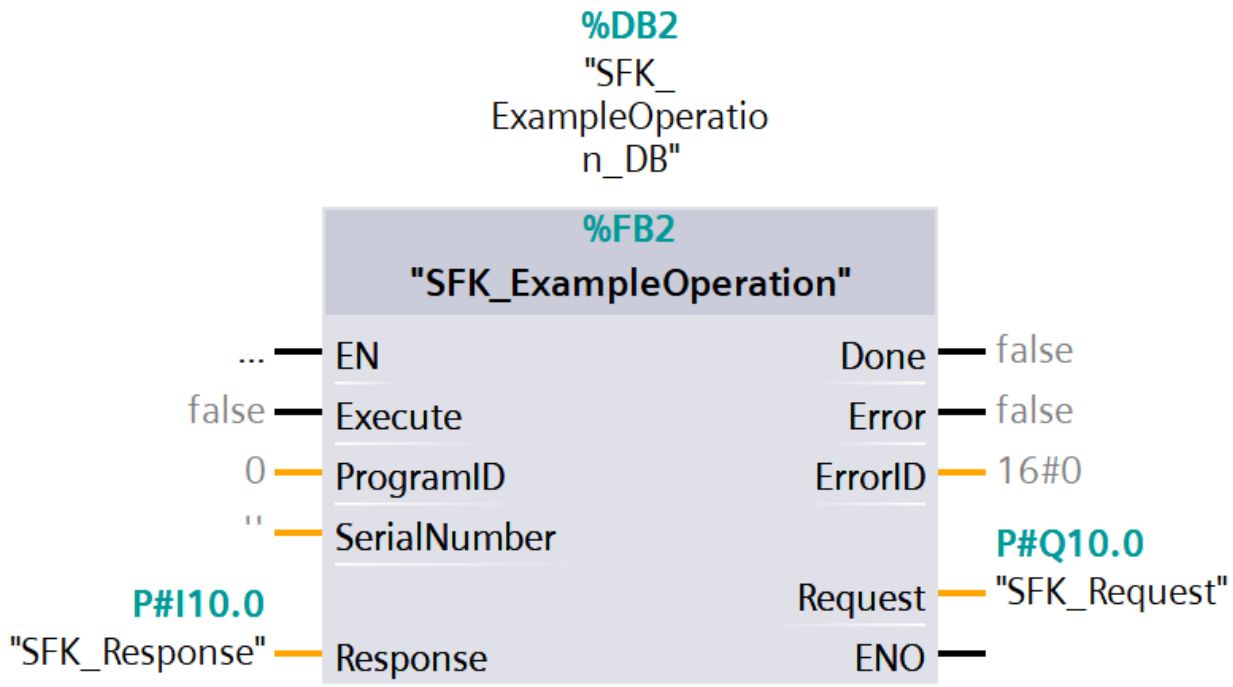


Figure 73 SFK_ExampleOperation

Inputs:

Name	Data type	Description
Execute	BOOL	Execute the command on rising edge
ProgramID	UINT	ID of the SFK program to be started

Name	Data type	Description
SerialNumber	STRING[28]	Serial number that should be associated with the program start
Response	SFK_typeResponse	Fieldbus data for communication via the function block "SFK_Command"

Table 49 Inputs

Outputs:

Name	Data type	Description
Done	BOOL	TRUE if the sequence chain has been completely processed Is reset to FALSE as soon as input "Execute" is set to FALSE
Error	BOOL	TRUE if the current command was acknowledged with an error Is reset to FALSE as soon as input "Execute" is set to FALSE
ErrorID	WORD	Error number (see chapter Error catalog) Is reset to 0 as soon as input "Execute" is set to FALSE
Request	SFK_typeRequest	Fieldbus data for communication via the function block "SFK_Command"

Table 50 Outputs

17 OPC UA + Rest-API

17.1 OPC-UA

17.1.1 Overview

OPC UA (Open Platform Communications United Architecture) is an open communication standard used in industrial environments. OPC UA is independent of platforms and allows data exchange regardless of the platform manufacturer.

For easy application or integration of the OPC UA interface, the SFK has a server implementation (data model). In this way, the data model can be requested and used via any OPC client.

17.1.2 Establishing a server connection

Communication with the SFK via OPC UA requires an OPC client.

Recommended OPC client: UA Expert, Unified Automation GmbH

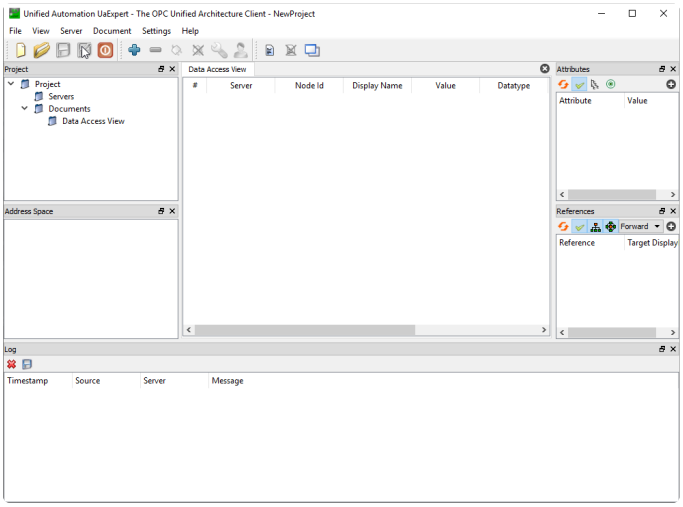


Figure 74 User interface UA Expert

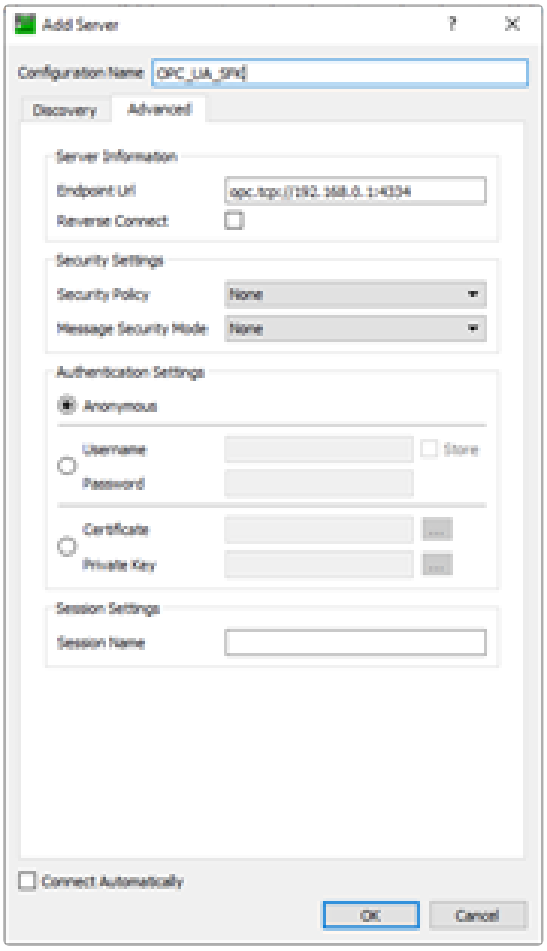
Screens/comments	Action steps
Notice The binary protocol is used to establish a connection to the SFK. The required connection configuration is: **URL** `"opc.tcp://192.168.0.1:4334"` Encryption type: No encryption  	Add the SFK as a server. → Using the IP address and the port number (port 4334), search for the server implementation in the network. If all parameters are correctly set, the OPC client calls up the data model of the SFK with the required interactions.

Table 51 Establishing a server connection

17.1.3 Data model/information model

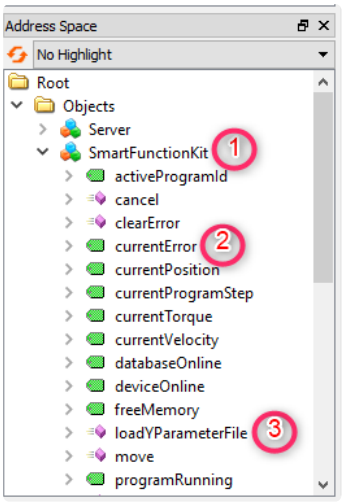


Figure 75 Data model/information model structure

The data of the SFK are provided in the form of an object (1).

Comparable to the object-oriented programming, the object can be assigned various properties and/or variables (2) and methods (3).

It is possible to execute a method (such as reading a parameter) or to access a variable (such as the drive controller switching frequency).

Methods

The integrated server provides the following methods for interaction with the SFK.

Method	Function	Input parameter	Type	Description
cancel()	• Cancels an active command and aborts it.			
clearError()	• Deletes all currently pending errors.			

<table>
<tr><th>Method</th><th>Function</th><th>Input parameter</th><th>Type</th><th>Description</th></tr>
<tr><td>loadYParameterFile()</td><td>
Parameter file type 1: Used to configure SMC.
Parameter file type 2: Used to parameterize an axis.
The distinction between type 1 and type 2 is made in the file name (e.g. if “Axis” is contained in the file name, the file is automatically assigned to type 2).

The file format of both parameter files is .SCD.

The parameter files type 1 and type 2 are used for commissioning and are stored in the directory/user on the SD card of the drive.

Example
Loading SMC configuration: Y_Param_System
Load axis parameterization: Y_Param_Axis1
Structure of a parameter (file content):
<table>
<tr><th>Element</th><th>Example</th></tr>
<tr><td>Number of Y parameter</td><td>Y1049</td></tr>
<tr><td>Name of Y parameter</td><td>Analog sensor</td></tr>
<tr><td>Unit of Y parameter</td><td>0/1</td></tr>
<tr><td>Minimum value of Y parameter</td><td>0</td></tr>
<tr><td>Maximum value of Y parameter</td><td>1</td></tr>
</table>
</td><td>fileName</td><td>String</td><td>File with the Y parameters WITHOUT“SCD” suffix</td></tr>
</table>

<table>
<tr><th>Method</th><th>Function</th><th>Input parameter</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td><table><tr><th>Element</th><th>Example</th></tr><tr><td>Actual value of Y parameter</td><td>0: no force sensor
1: force sensor in use</td></tr><tr><td>End of Y parameter</td><td>|</td></tr></table></td><td></td><td></td><td></td></tr>
<tr><td>move()</td><td>• Moves the SFK to the specified target position.
Example
move(50, 1, 30,30)</td><td>distanceOrPosition</td><td>UINT32</td><td>Target position or distance [mm]</td></tr>
<tr><td></td><td></td><td>relative</td><td>Boolean</td><td>1 → relative
0 → absolute</td></tr>
<tr><td></td><td></td><td>velocity</td><td>UINT32</td><td>Travel speed [mm/s]</td></tr>
<tr><td></td><td></td><td>acceleration</td><td>UINT32</td><td>Acceleration [mm/s^2]</td></tr>
<tr><td>readCommandHistory()</td><td>• Reads the command history of the system.
• Lists the last 20 commands/actions executed for diagnostic purposes when analyzing command sequences.</td><td></td><td></td><td></td></tr>
</table>

Method	Function	Input parameter	Type	Description
readIOs()	• Reads the properties of the set inputs and outputs of the system and returns the information as JSON string. The JSON object contains the following information: • System name • Alias name • Edge (Rising (R)/Falling (F)) • Type (program (PRG)/ system function (SYS)) • Sort (Input (IN)/Output (OUT)) • State (TRUE/FALSE) • Inactive (TRUE/FALSE)			
readNotificationHistory()	• Reads the notification history of the system. • Lists the last 20 notifications that have been executed for diagnostic purposes when analyzing command sequences.			
readParameter()	• Reads a specified S or P parameter and feeds back the parameter as a string. • Additionally, a list parameter can be read. **Example** Switching frequency of the power stage: readParameter(P-0-0001)	parameterName	String	Parameter ID in the form of S-0-xxx or P-0-xxx
readSMCVariable()	• Reads a system variable of the underlying technology package SMC and returns the contents of the variable as a string. **Example** Tolerance force sensor: readSMCVariable(VFR002)	VariableName	String	Parameter ID in the form of VFRxxx

Method	Function	Input parameter	Type	Description
readStatus()	• Reads the currently active command and its request context. ✅ The information is used to analyze which command or request was made to the system at which time from which source. • Returns a string containing the information for diagnostic purposes as output.			
readYParameter()	• Reads a system parameter of the underlying technology package SMC and returns the content as a string. **Example** Maximum acceleration: readYParameter(Y1006)	parameterName	String	Parameter ID of the form Yxxxx
reboot()	• Restarts the complete system.			
setPMOM()	• Allows manual switching between parameter and operating mode.	pm	Boolean	1 → Parameter mode active 0 → Operating mode active
setProgrammActive()	• If the program ID is transmitted, a created program can be loaded or set active. ✅ The program ID can be read from the selection list in the Programming area. **Example** setProgrammActive(5d08fb20c21695)	IdAsName	String	ProgramID
setReference()	• Sets the current reference point of the system to the value set for parameter S-0-0052 in the extended configuration.			

Method	Function	Input parameter	Type	Description
startProgram()	• Starts the currently active program.			**Notice** The method call does not check the status of the program execution. The response value of the method cannot be used to draw any conclusions as to whether the program was executed successfully.
startReference()	• Starts homing.			
tare()	• Allows the force sensor to be tarred to any value. The weight of a tool can be compensated. **Notice** For the active force sensor, the force in [N] can be specified as offset. If the motor current is used for determining the force, the torque percentage relative to the nominal torque of the motor must be specified.	offsetInPercent	UINT32	Value to be tarred to.

Method	Function	Input parameter	Type	Description
writeParameter()	• Writes a specified S or P parameter. **Notice** To describe the parameters: Before writing the parameters, switch to the parameter mode operating state. **Example** Time constant analog input filter: writeParameter(P-0-0217,4)	parameterName	String	Parameter ID in the form of S-0-xxx bzw. P-0-xxxx
writeSMCVariable()	• Writes a system variable of the underlying technology package SMC. **Example** Tolerance force sensor: writeSMCVariable(VFR002, 6.5)	VariableName	String	Name of the variable to be read.
		value	String	Value to be written.

Method	Function	Input parameter	Type	Description
writeSMCVariables()	• Writes several system variables of the underlying technology package SMC.	parameterName	string	Input in the form: 'Variable':'Value','Va riable':'Value' **Notice** When writing several variables of the same type, the single specification of the type is sufficient and, for example, 'VF500:10,501:20' **Notice** Variables of several types can be written at the same time, for example 'VF500:10, MF100:1'
writeYParameter()	• Writes a system parameter of the underlying technology package SMC. **Notice** To describe the parameters: Before writing the parameters, switch to the parameter mode operating state. **Example** maximum stroke: writeYParameter(Y1044, 150)	parameterName	String	Y-parameter ID
		value	String	Value to be written.

Table 52 Methods

Variables

The following variables can be read and monitored.

Variable	Function
activeProgramID <String>	Reads the program ID of the currently loaded program.
currentError <Double>	Displays the currently pending error. Example: E-Stop activated: *999476* (DEC) → *F4034* (HEX)
currentForce <Double>	Force applied at the force sensor.
currentPosition <Double>	Current position relative to the zero point of the system.
currentProgramStep <Double>	Current program step of the program being processed.
currentVelocity <Double>	Current travel speed.
databaseOnline <Boolean>	Checks whether the system database is online and whether data can be stored there.
deviceOnline <Boolean>	Indicates whether the system is available. It is checked whether a TCP connection can be established between PR21 and HCS01.
freeMemory <Double>	Indicates the available memory capacity of the PR21.
programmRunning <Boolean>	Variables value 1 indicates that the program is currently being executed.

Table 53 Variables

17.2 REST-API

This documentation provides an overview of the SFK REST paths.

◊ For more information, refer to the dashboard under Help → API Documentation.

17.2.1 Architecture and basics

The SFK for press applications contains an application programming interface (API) which allows for external access to various system components.

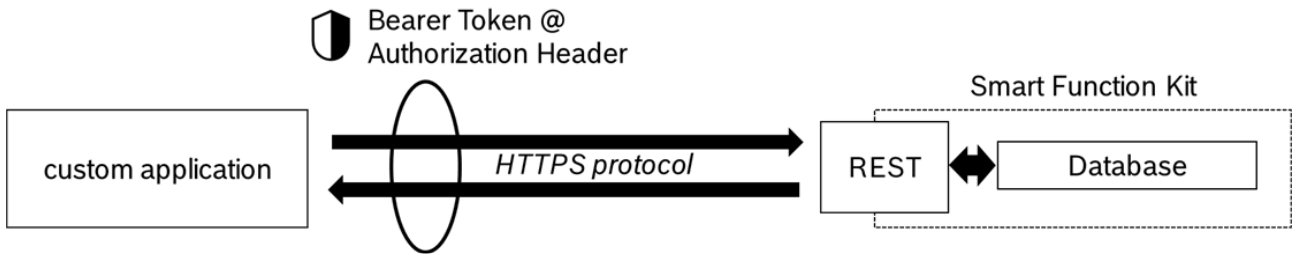


Figure 76 SFK API Architecture

The API specifies how a program must be written to query services of an operating system (e.g. Linux) or a software application (e.g. SFK operating software).

The REST technology defines the interaction method with the system. It is designed for browser-based cloud service interactions and is suitable for low bandwidths.

A REST-API uses the HTTP methods defined by the RFC-2616 protocol.

→ To request a resource: Use "GET".

→ To change the status of a resource (an object, a file, or a block) or to update it: use “PUT”.

→ Use “POST" to create a resource.

→ To remove the resource: Use “DELETE”.

The Smart Function Kit REST API provides the authentication service and access to the database, e.g. curve data or program configurations.

Notice
Most REST API methods access the database. This leads to increased database and processor utilization. Avoid executing REST API methods in short cyclic intervals or during program execution.

17.2.2 Overview of API access

The REST-API of the SFK provides access to the following data:

- Account
- Authentication
- Configuration
- Curves
- Programs
- Users
- Evaluation elements
- Activity
- Commissioning

For detailed descriptions on the implementation of specific requests, please see the end of this document.

17.2.3 Implementation

Various programming languages offer libraries for implementation of connections based on the http and WebSocket protocol (e.g. JavaScript/TypeScript, C#, Java). The examples in this documentation have been written in JavaScript.

Connection check

To check the connection, proceed as follows:

→ Make sure that the connection to the SFK is established before programming and testing individual applications.

→ Access the user interface via the web browser (Firefox, Chrome).

→ In the dashboard, enter the IP address of the SFK (for example, “https://192.168.0.1”) and verify the login data.

→ In case of problems, clear the cache of the web browser used and repeat the previous steps.

Authentication (POST)

The first step of data extraction is always a form of authentication: The user's login data is sent via an http request.

The response to this request contains a token (cryptic string) which must be used in the following request headers for authentication.

The authentication type is the provider to the application via HTTPS (SSL).

Authentication contains the POST command. This method always generates a new instance as feedback, in this case authentication by the identification token.

Example of a code

```
function login () {
                const Http = new XMLHttpRequest();
                let ipAddress = '192.168.0.1';                          // Enter your
SFK IPC IP here
                let password =
‘admin’;                              //Enter your SFK dashboard password here
                let user = {
                    username : "admin",
                        password : password
                };
                const url= 'https://' + ipAddress + '/api/authenticate';
                Http.open("POST", url);
                Http.setRequestHeader("Content-Type", "application/json");
                Http.setRequestHeader("Accept", "application/json");
                Http.send(JSON.stringify(user));
                Http.onreadystatechange = function() {
                 // Login successful, further REST API requests can be made with
bearer token
                       if(this.readyState == 4 && this.status == 200) {
                               const token = JSON.parse(Http.responseText).token;
                             // Do something (see following chapters)
                               return;
                       }
                   // Login error
                       else if(this.readyState == 4 & this.status != 200) {
                            //Handle error
                            return;
                       }
}
}
```

Individual data request (GET)

The GET method can be used to retrieve data from the database. The authentication token (see “Authentication” section (POST)) allows access to the data. The token is sent as part of the request header.

The data returned in response is natively structured in JSON format. To process in the customer program during the following step, the response string can be parsed from JSON into an object.

Example of a code

```
function GetAccount (token) {

            const Http = new XMLHttpRequest();
             let ipAddress = '192.168.0.1';                                    //
Enter your SFK IPC IP here
           const url='https://' + ipAddress + '/api/account';

            Http.open("GET", url, false);
            Http.setRequestHeader("Authorization", "Bearer " + token);
            Http.send();

                // Response in JSON format
             let response = JSON.parse(Http.response);
             let ID = response.id;
             let Login = response.login;
             let Language = response.langKey;
             let Created = response.createdDate;
             let LastModified = response.lastModifiedDateDate;

    return;
}
```

Removal of individual data (DELETE)

The DELETE method can be used to remove data from the database. The authentication token (see “Authentication” section (POST)) allows access to the data. The token is sent as part of the request header.

Notice
The ID required to delete a program does not match the program ID displayed on the dashboard. The ID for deleting a program is obtained via a REST API request (GET request).

Example of a code

```
function DeleteProgram(token, program) {
                let ipAddress = '192.168.0.1';              // Enter your SFK IPC IP
here

 // program as JSON object (see GET /api/programs/active)
let prog_id = program.id;
                const url = 'https://' + ipAddress + '/api/programs/' + prog_id;
                console.log('URL: ', url);
                const Http = new XMLHttpRequest();

                Http.open("DELETE", url, false);
                Http.setRequestHeader("Authorization", "Bearer " + token);
                Http.send();
}
```

Updating of individual data (PUT)

The PUT method can be used to update data in the SFK database. The authentication token (see “Authentication” section (POST)) allows access to the data. The token is sent as part of the request header.

Notice
For some PUT methods, further queries may be necessary before or after to ensure correct modification of the data. For more information, refer to the dashboard under Help → API Documentation.

To send update data via the body, the content type of the request must be defined.

Example of a code (simple)

```
function UpdateTime(token) {
                let ipAddress = '192.168.0.1';                                  /
/ Enter your SFK IPC IP here
                var date = new Date().toISOString().substr(0, 19);

                let time = {
                        backend : date,
                        drive : date
                };

                const url = 'https://' + ipAddress + '/api/configuration/time';
                const Http = new XMLHttpRequest();

                Http.open("PUT", url, false);
                Http.setRequestHeader("Authorization", "Bearer " + token);
                Http.setRequestHeader("Content-Type", "application/json");
                Http.setRequestHeader("Accept", "application/json");
                Http.send(JSON.stringify(time));
}
```

Example of a code (advanced)

```
function UpdateProgram(token, program) {
let ipAddress = '192.168.0.1';                                    // Enter your SFK IPC
IP here
let newStartPosition = 100;
let prog_id = program.id;

// Manipulate parameters of JS object
program.name = "Test_Start_" + newStartPosition + "mm";

// "Start position" as first step (array!) in program, position parameter 4 (array!)
program.functions[0].parameters[3].value = newStartPosition;

                    // Deactivate setting “keepLastModifiedDate”
const url1 = 'https://' + ipAddress + '/api/programs?keepLastModifiedDate=false';
const Http1 = new XMLHttpRequest();

Http1.open("PUT", url1, false);
Http1.setRequestHeader("Authorization", "Bearer " + token);
Http1.setRequestHeader("Content-Type", "application/json");
Http1.setRequestHeader("Accept", "application/json");
Http1.send(JSON.stringify(program));

                    // Send updated program
const url2 = 'https://' + ipAddress + '/api/programs/' + prog_id;
const Http2 = new XMLHttpRequest();

Http2.open("PUT", url2, false);
Http2.setRequestHeader("Authorization", "Bearer " + token);
Http2.setRequestHeader("Content-Type", "application/json");
Http2.setRequestHeader("Accept", "application/json");
Http2.send(JSON.stringify(program));

}
```

Notice
The ID required to change a program does not match the program ID displayed on the dashboard.
The ID for changing a program is obtained via a REST API query (GET request).
To be able to perform a change of parameters, the internal "keepLastModifiedDate" parameter must be set to “false”.

Notice
The program changes that have been made cannot be executed until the program has been reloaded into the PLC. Uploading can be done by calling the upload function or within the dashboard.

17.2.4 Export of process curves

Process curves can be requested via the REST API interface (e.g. for export to an external database). No authentication by means of bearer tokens is necessary for the export.

The structure of the ICurve scheme can be determined via export of a process curve via the web-HMI or the API-Swagger in the web-HMI.

Querying a curve

Query scheme: GET https:// IP-ADDRESS /api/curves/<CURVE-ID>

Response: ICurve with all parameters

Searching the curve data base (Query)

Query scheme: GET https:// IP-ADDRESS /api/curves?<PARAMETER1>&<PARAMETER2>& ....

Query parameters are linked with "&". Parameter scheme: <NAME>=<INPUT-VALUE>

Response: ICurve(s) with all or explicitly selected parameters. See the following parameter description:

Field name	Mandatory field?	Description	Data type	Valid range	Example
page	Yes	Internal page calculation is calculated with "size" (page*size, if page > 0)	number	>=0	page=0 Recommended for all external applications, no offset
size	Yes	Number of displayed data records (maximum)	number	>1	size=10 Maximum of ten curve sets as response
offset	No	For internal use only			
fields	No	Fields of the response data set	JSON-fields designation	All ICurve parameters	fields=id,createdDate,valid,program,maxPosition,maxForce,cycleTime,customId

Field name	Mandatory field?	Description	Data type	Valid range	Example
sort	No	Native Mongo-DB sort request is ignored if simultaneously used with “offset”. **Notice** Depending on the size of the data base, sorting requests may take longer with increasing number of data records. The stable long-term behavior of a connection can be optimized by the entry _id.	Mongo DB request	**Possible Fields:** _id, program, valid, status, customId, createdDate, lastModifiedDate, cycleTime, dataRecordingDisabled, validationTime asc/desc	sort={"customId":"desc"} Sorts the data set in descending order
q	No	Native Mongo-DB filter query	Mongo DB request	All ICurve parameters	q={"customId": {"$eq":"34764378691750"}} The query is filtered by curves with a parameter "customId" (serial number) with the value "34764378691750"

Table 54 Curve database parameter description

Examples

Confirm the Web browser certificate using the Web HMI to avoid rejecting requests that indicate a security error.

Exporting a curve on the basis of the curve ID:

Example of a code:

```
GET https://IP_ADDRESS/api/curves/6156a3807608d0799f93d88f
```

Exporting the last/newest curve:

Example of a code:

```
GET https://IP_ADDRESS/api/curves?page=0&size=1&sort={"_id":"desc"}
```

Exporting a curve on the basis of a specific serial number (customId):

Example of a code:

```
GET https://IP_ADDRESS/api/curves?page=0&size=1&q={"customId":
{"$eq":"34764378691750"}}
```

Exporting a data set (here, the last 5 curves) with specific parameters (here, "id", "valid", "maxPosition", "maxForce", "cycleTime" and "customId"):

Example of a code:

```
GET https://IP_ADDRESS/api/curves?page=0&size=5&sort={"_id":"desc"}
&fields=id,valid,maxPosition,maxForce,cycleTime,customId
```

Exporting a data set (here, the last 5 curves) without reference curves:

Example of a code:

```
GET https://IP_ADDRESS/api/curves?page=0&size=5&sort={"_id":"desc"}&q={"reference":
{"$eq":null}}
```

17.2.5 Export of press history

The press history can be queried via the REST API interface (e.g. for export to an external database). Authentication using bearer tokens is required for export.

The structure of the ICurve scheme can be determined via export of a process curve via the web-HMI or the API-Swagger in the web-HMI.

Export curve database (Query)

Query scheme: GET https:// IP-ADDRESS /api/curves/export?filter={FILTER1, FILTER2 ...}

Response: Array of ICurves with all parameters

Notice
The filter placeholder is any Mongo DB filter instruction. Several filters can be linked by a comma.
◊ For more information, see the online help.

The structure of the filter placeholder follows the same scheme as that of Deletion of process curves.

17.2.6 Window evaluation element

Evaluation elements of the type “Window” can be read via a simplified interface (GET) and changed (PUT).

The window ID appears in the list of evaluation elements (see chapter Evaluation).

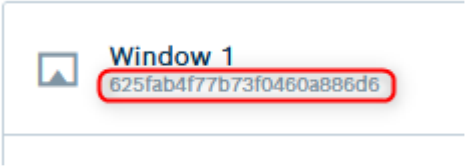


Figure 77 Window ID

Query scheme:

GET https:// IP-ADDRESS /api/validation-elements/validationWindow/<WINDOW-ID>

PUT https:// IP-ADDRESS /api/validation-elements/validationWindow/<WINDOW-ID> (body and authentication required)

Body: Evaluation window structure (scheme):

```
{
    "bottom": {
        "value": Number,
        "direction": DirectionType
    },
    "right": {
        "value": Number,
        "direction": DirectionType
    },
    "top": {
        "value": Number,
        "direction": DirectionType
    },
    "left": {
        "value": Number,
        "direction": DirectionType
    }
}
```

Direction type versions (see chapter Evaluation):

Description	Identifier
OFF	“OFF”
No crossing	“NOC”
Crossing with positive lead	“LTR”
Crossing with negative lead	“RTL”

Table 55 Direction type versions

Example:

Reading of window data via ID:

REST Call

```
GET http://IP_ADDRESS/api/validation-elements/validationWindow/
625fbb730751a704609990f6
```

Changing of window data via ID (body and authentication required):

Notice
Load and modify existing body via GET command.

REST Call

```
PUT http://IP_ADDRESS/api/validation-elements/validationWindow/
625fbb730751a704609990f6
```

Body: Evaluation window structure (scheme):

```
{
    "bottom": {
        "value": 300,
        "direction": "LTR"
    },
    "right": {
        "value": 100,
        "direction": "NOC"
    },
    "top": {
        "value": 500,
        "direction": "LTR"
    },
    "left": {
        "value": 80,
        "direction": "OFF"
    }
}
```

17.2.7 Deletion of process curves

Process curves can be deleted via the REST API interface for the following purposes:

- Freeing storage space on the industrial PC.
- Increasing the performance of the internal database to keep cycle times permanently constant.

No authentication by means of bearer tokens is necessary for this purpose. An instruction for authentication can be found in the examples below.

Notice
If process curves are deleted, they are permanently removed from the system.
To avoid data loss: Export the process curves before deleting them.

Deleting process curves

Query scheme: DELETE https:// IP-ADDRESS /api/curves?filter={FILTER1, FILTER2 ...}

Response: Number of deleted curves

Notice
The filter placeholder is any Mongo DB filter instruction. Several filters can be linked by a comma.
◊ For more information, see the online help.

Filter string for the creation date

The filter string for the creation date is as follows: {{"createdDate": { “Filter_abbreviation”:”Filter_date"}}

The filter abbreviations are:

- $gte : Greater than & equal (>=) → This date or newer
- $lte : Less than & equal (<=) → This date or older

The format of the filter date is: YYYY-MM-DDTHH:MinMin:SS.MsMsMsZ

Example: {"createdDate": { "$lte":"2022-05-31T11:41:03.234Z"}}

Digit	Description
Y	Year
M	Month
D	Day
H	Hour
Min	Minute
S	Second
Ms	Millisecond

Table 56 Filter date abbreviation

Filter string for production curves

If you do not want to delete any reference curves, apply the following filter: {"reference": {"$eq":null}}

Filter string for a specific program

If you only want to delete curves from a program, apply the following filter: {"program": { "_id": "PROGRAM-UUID"}

Notice
The 24-digit PROGRAM UUID can be found in the web interface under *Program management*.

Examples

> **Notice**
> To avoid rejecting requests that indicate a security error: In the web interface, confirm the certificate for the browser first.
> In case of direct implementation, acceptance of certificates must be supplemented.

Deletion of all production curves before 31/05/2022, 11:41:03.234 h:

REST Call

```
DELETE https://IP_ADDRESS/api/curves?filter={"reference":{"$eq":null},"createdDate":
{"$lte":"2022-05-31T11:41:03.234Z"}}
Request body:
    Authorization: Bearer BEARER TOKEN (From authentication call)
```

Deletion of all production curves of the program with UUID "asdfghjkl1234567":

REST Call

```
DELETE https://IP_ADDRESS/api/curves?filter={"reference":{"$eq":null},"program":
{ "_id": "asdfghjkl1234567"}
Request body:
    Authorization: Bearer BEARER TOKEN (From authentication call)
```

17.2.8 Using the API-Swagger

The API-Swagger is a graphical representation of the REST API and is available via the “Interfaces Documentation” help tile.

In order to perform calls that require registration, it is necessary to generate a token that must be stored with the corresponding request. To do so, proceed as follows:

→ Go to path `/api/authenticate`.

Comments on the transfer value `sysTime`:

- This is a perfect argument. If it is specified, the current time must be used. The returned token is valid for 7 days.
- The format corresponds to the Unix time in milliseconds. That means that a 13-digit figure must be available. In case of the default value (empty object), the system time of the industrial PC is used.
- The argument can be omitted. So the system time of the industrial PC will be used as basis for the token generation.

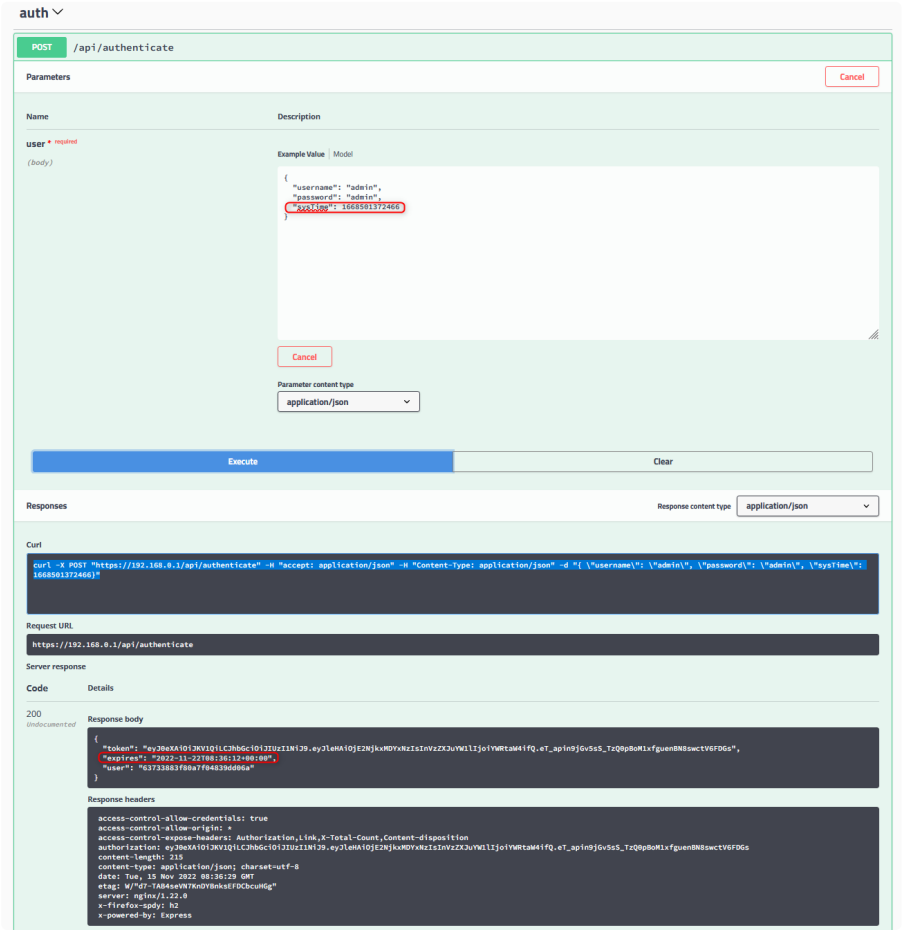


Figure 78 User interface api authentification

The registration information can be centrally stored for all other requests:

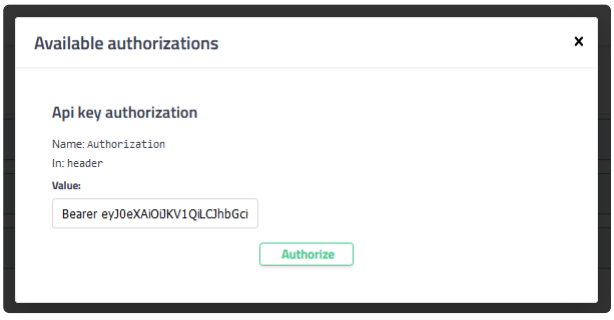
Screens/comments	Action steps
	→ Click *Authorize*.
	→ Copy the token from the registration response into the “Value” field; apart from that, it is necessary to prefix the designation “Bearer” and a space. → Click *Authorize*. ✓ The registration information is now centrally stored and call-ups requiring authorization can be used.

Table 57 Store registration information centrally

17.3 API documentation

17.3.1 REST

The following is an overview of the SFK REST paths.

◊ For more information, refer to the dashboard under Help → API Documentation.

/api/account

GET /api/account

POST /api/account

/api/account/change-password

POST /api/account/change-password

/api/authenticate

POST /api/authenticate

/api/configuration/functions

GET /api/configuration/functions

/api/configuration/ios

GET /api/configuration/ios

PUT /api/configuration/ios

/api/configuration/hardware

GET /api/configuration/hardware

PUT /api/configuration/hardware

DELETE /api/configuration/hardware

/api/configuration/system

GET /api/configuration/system

PUT /api/configuration/system

/api/curves/reference

POST /api/curves/reference

/api/curves

PUT /api/curves

GET /api/curves

/api/curves/{id}

DELETE /api/curves/{id}

GET /api/curves/{id}

/api/curves/{id}/validate

GET /api/curves/{id}/validate

/api/curves/validate/{validationElementId}

GET /api/curves/validate/{validationElementId}

/api/profile-info

GET /api/profile-info

/api/programs/active

GET /api/programs/active

/api/programs/fieldbus/nextId

GET /api/programs/fieldbus/nextId

/api/programs/fieldbus/{id}

GET /api/programs/fieldbus/{id}

/api/programs/{id}

GET /api/programs/{id}

DELETE /api/programs/{id}

/api/programs/{id}/statistics

GET /api/programs/{id}/statistics

/api/programs

GET /api/programs

POST /api/programs

PUT /api/programs

/api/programs/{id}/Smc

GET /api/programs/{id}/Smc

PUT /api/programs/{id}/Smc

/api/programs/transitiongroups

POST /api/programs/transitiongroups

/api/programs/initFunctionCall/{name}

POST /api/programs/initFunctionCall/{name}

/api/programs/{copyId}/copy

POST /api/programs/{copyId}/copy

/api/programs/{id}/export

GET /api/programs/{id}/export

/api/programs/import

POST /api/programs/import

/api/users/authorities

GET /api/users/authorities

/api/users/{login}

GET /api/users/{login}

DELETE /api/users/{login}

/api/users

GET /api/users

POST /api/users

PUT /api/users

/api/validation-elements

POST /api/validation-elements

PUT /api/validation-elements

GET /api/validation-elements

/api/validation-elements/{id}

DELETE /api/validation-elements/{id}

/api/validation-elements/validationWindow/{id}

PUT /api/validation-elements/validationWindow/{id}

GET /api/validation-elements/validationWindow/{id}

/api/activities

GET /api/activities

18 Software Update

The current software version can be downloaded from the homepage of Bosch Rexroth AG, in the SFK product area: https://www.boschrexroth.com/product/cls_eCommerce_lx_1555747.

The software is updated by means of an installation file and is provided by Bosch Rexroth AG.

NOTICE	
	**Production downtime during software update** The system cannot be operated during the software update. This results in production downtime. → Stop the running production before performing a software update.
	**Data loss in case of errors in the update process** Errors may occur during the software update. A loss of production data may result. → Create a backup of the production data before performing a software update.
	**Delays due to database activity in the update process** Database activities may delay the software update. This may result in the update process taking longer. → Check whether there are large amounts of data (high number of stored production curves). Before updating the software, delete any unnecessary production data from the Press history.

18.1 Installation

To perform a software update: Observe the following process.

1. The software update can only be completed successfully via the Commissioning the software . Some parameters are reset to their default values. Note down any modified parameters or create a backup of the respective parameter pages.

   The parameters are as follows:

Settings	Parameters involved	Rest value	Notice
Circular buffer	Status	deactivated	
Fieldbus	active fieldbus protocol	inactive, no fieldbus active	The safe fieldbus communication must be inactive for the update (see chapter Settings for safe fieldbus communication).
	Device name (optional)	SFK	

Settings	Parameters involved	Rest value	Notice
Extended configuration	all	respective default values	→ To save all parameters: Use the backup function. After the software update, they can be loaded with the import function (see chapter Hardware configurations).
I/O Configuration	all	IX31.3: Emergency Stop All other: unassigned	

Table 58 Installation

2. Download the new software version from the Bosch Rexroth AG homepage.
3. Save the file on a device with Windows 10 or Windows 11 operating system.
4. Connect your device with the industrial PC PR21. To do this, use the XF6 interface.
5. Please make sure that the network settings at the device match the settings of the SFK and that a connection can be established.
6. Unzip the file.
7. Start the executable installation file (.exe).
   ✅ After a brief loading period, the installation program is opened.
8. In the installation program, enter the IP address of the SFK (see IP for HMI display) and click *Connect*.
9. Click on *Installation/Update*.
10. To confirm: Click on *Update*.
11. The installation program automatically performs the update and subsequently checks the installation of the SFK.

> **Notice**
>
> → If you have problems during the installation process, use the direct connection to the device (see step 4).

12. Once the installation has been successfully completed, you can close the installation program.

18.2 Commissioning the software

After installing the new SFK software version, the commissioning wizard is automatically displayed upon the next connection with the system.

The software update can only be successfully completed by means of the recommissioning. System data is not lost.

The following settings must be set again.

1. Perform commissioning as described in the chapter Commissioning.
2. Make the following settings again (manually or by loading a backup):
   - Circular buffer
   - Fieldbus
   - User-defined changes to the extended configuration.
     Previously created backups can be loaded, see chapter Hardware configurations.
   - I/O Configuration
     Afterwards, check the programs used by the I/Os within the sequence and save the program again (click on *Save*).

NOTICE

Incorrect parameters after software update

The software update may generate incorrect parameters. This may result in system failure or damage.

→ After recommissioning, check all parameters for their validity.

→ Check your custom settings in the extended configuration. They may have been overwritten by the software update.

18.3 Settings for safe fieldbus communication

→ Before updating the software, deactivate the safe fieldbus communication (P-0-3345.0.0 = 0).

→ Start the recommissioning (see chapter Commissioning the software).

The software update is completing.

→ After the software update has been completed: Reconfigure the safe fieldbus communication (Safety).

Notice

→ To simplify the recommissioning of the safety: Create a parameter backup before updating.

19 Troubleshooting and fault clearance

19.1 Procedure for troubleshooting

Troubleshooting may only be carried out by authorized, trained and instructed specialists.

NOTICE	
	**Changes by external programs** Operating the system with external programs makes the system unusable. → Do not use remote programs (such as IndraWorks DS) to make settings at the PR21 or HCS01.

→ Proceed systematically and in a targeted manner, even under time pressure.

💡 Random, thoughtless disassembly and changing of individual values could result in the inability to determine the original cause of errors.

→ First, get a general idea of how your product works in conjunction with the overall system.

→ Check whether the product worked properly in conjunction with the overall system before the error occurred first.

→ Record any changes of the overall system in which the product is integrated:

- Were there any changes to the product's conditions of use or field of application?
- Have changes (e.g. refittings) or repair work been carried out on the overall system (machine/system, electrical systems, controller) or at the product? If yes: Which?
- Was the product or machine operated as intended?
- How does the fault become apparent?

→ Try to get a clear idea of the cause of error.

→ If necessary, ask the operator or the machine operator.

→ If you are unable to remedy the error, get in touch with one of the contact addresses listed at www.boschrexroth.com/contact. Submit a log file with the request as described in chapter Service tool – log file.

19.2 Display errors

If display errors occur in the web interface (e.g. values in the dashboard are not updated), they can usually be rectified by refreshing the page.

→ Use the Refresh page function (depending on the web browser used).

→ Alternatively, refresh the page by means of function button F5.

→ Reset website zoom to 100% or less.

→ If necessary, clear the web browser cache as described in chapter Web interface not available and reload the page.

19.3 Web interface not available

Loading spinner

If the web browser cannot load the web interface, even after a short wait, this may be due to a problem with the display of the content, e.g. due to a software update. Depending on the web browser settings, the contents are kept in the cache and can no longer be correctly displayed or loaded.

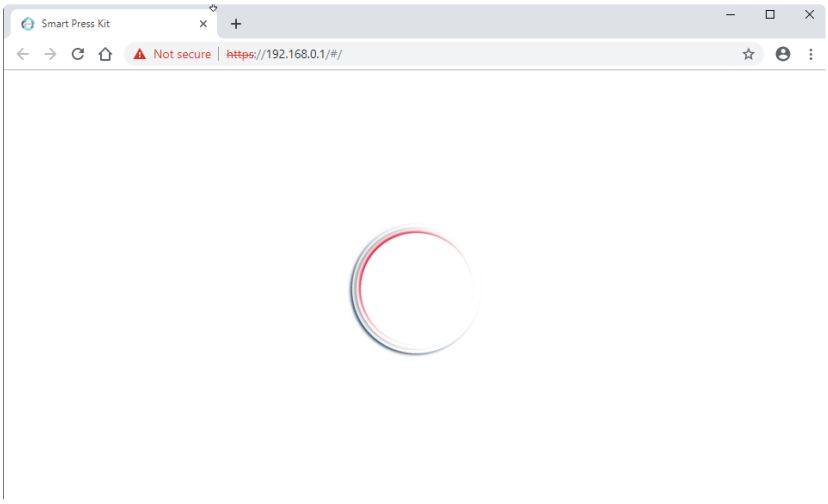


Figure 79 Web interface loading spinner

✓ The loading spinner shown above is permanently displayed and the web interface is not loaded.

→ To remedy the error: Clear the cache of the web browser. Use the shortcut Ctrl+Shift+Del.

✓ A dialog for clearing the cache is opened in your web browser.

◊ The procedure can be read on the Internet, for example, on the page of the web browser.

Web interface does not load

If the web interface does not load, the browser indicates a connection problem.

1. Follow the instructions and notes in chapter Initial commisioning.
2. Follow the instructions in chapter Procedure for troubleshooting.

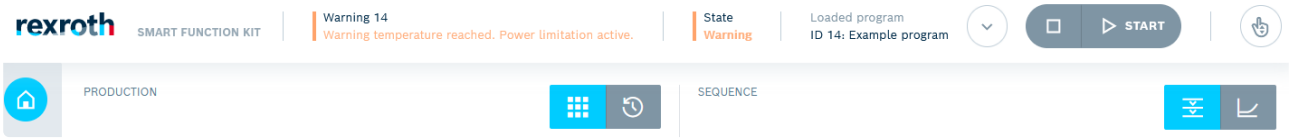
19.4 Temperature Warning

A temperature warning is issued as soon as the warning temperature of the motor and/or the drive controller is exceeded during operation.

The system can continue to operate. In order to avoid excessive temperatures and drive failure, sufficient load-free phases must be ensured.

Notice
Phases in which the system exhibits the maximum nominal force over a long period of time should be avoided.

Notice
If the warning is ignored, performance may be limited to protect the system.



19.5 Service tool – log file

The service tool log file contains information on the hardware and system configuration, settings of the industrial computer and system logs of the SFK.

How to generate the log file?

→ Download the latest software update installation file from the Bosch Rexroth AG website.

→ On our website www.boschrexroth.com, navigate to *Smart Function Kit* > *Downloads* or use the shortcut on the configuration page in the *Help* > *Product information menu*.

→ Execute the downloaded exe file.

→ To establish a connection: Enter the IP address of the SFK.

→ Select the *Service* option.

✅ The log file is stored in the directory from which the exe file was started.

→ Send the file to the Bosch Rexroth AG service.

19.6 Error catalog

19.6.1 Drive errors

All drive errors are listed in the error log.

◊ More information on the errors can be found in the IndraDrive MPx-20 (DE: R911345607 and EN: R911345608) documentation, see www.boschrexroth.com.

A selection of drive errors is listed below.

◊ For more information, see the IndraDrive MPx-20 standard documentation.

No.	Designation	Problem	Remedial measure
F2011 - F2017	PLC error	Internal error	→ Observe the detailed error text information in the web interface. → If the error message "Y1005 too large" appears, check the parameter Y1004 in the hardware configuration.

No.	Designation	Problem	Remedial measure
F2190	Incorrect Ethernet configuration	The configuration of the parameters for Ethernet communication (TCP/IP) is not permitted.	→ Make sure that the IP address and the gateway address are in the same IP network and that their IP addresses are not identical. → Make sure that the IP addresses of the Ethernet ports of the drive controller are different. Please note that port X26 must have the IP address 192.168.1.1.
F3134	Dynamization time interval incorrect	Error in the wiring	→ Check the wiring of the selection and correct the error. ◊ For more information on wiring of peripheral devices with the SFK, please see the assembly instructions (R320103194) and the exemplary circuit diagrams, see www.boschrexroth.com/product/cls_eCommerce_lx_1555747.
F4012	Incorrect I/O length	The master attempts to establish communication with an I/O length that does not match the parameterized I/O length.	→ Check the configuration according to the procedure in the Fieldbus configuration chapter, especially on the master page. → Primarily use the enclosed example projects on the master page (see Integrate/use example projects).
F6010	PLC runtime error	SD card not inserted correctly. Internal error	→ Create a log file (see chapter Service tool – log file) and inform the service. Then switch the drive back on. → Restart the drive controller. If the fault persists, perform another commissioning (see chapter Initial commisioning). → When using the fieldbus protocol "EtherCAT", make sure that the IO update time is not less than 4ms. → Create a log file (see chapter Service tool – log file) and inform the service.

No.	Designation	Problem	Remedial measure
F8027	STO/SBC while drive enabled	Special mode "Safe standstill" was selected with active control.	→ Check the correct control/wiring (selecting the STO input with a time delay after the E-Stop input), see also Connection to peripherals. ◊ For more information on wiring of peripheral devices with the SFK, please see the assembly instructions (R320103194) and the exemplary circuit diagram, see https://www.boschrexroth.com/product/cls_eCommerce_lx_1555747.
F9100	Hardware watchdog	The hardware-side runtime monitoring has responded. The drive was switched off.	→ Remove the display (programming module) for a short time. → Replace the drive controller.

Table 59 Drive errors

19.6.2 System errors

All system-specific errors are listed below. They are not included in the IndraDrive MPx-20 standard documentation.

Notice
→ For evaluation via fieldbus, please observe that the system errors must be interpreted as (unsigned) WORD.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
42 (only visible in Press history)	At least one evaluation element is NOK	One pressing was rated NOK because an evaluation element was violated.	No system error - a NOK evaluation can be an expected result of a joining process. → Check the sequence, associated evaluation elements and the associated curve.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
10001 / 0x2711	No function or invalid input	A command has been activated: • The parameters are missing. • The parameter values are invalid.	→ Check the transferred parameters for completeness and validity. → *Change program* command: Make sure that the ID set via fieldbus is available in the system. If necessary, save the program again. → *Change program* command: Make sure that the ID set via fieldbus is transmitted with correct scaling (e.g. in the request container: Input1 = 1000 for selecting the program with fieldbus ID = 1). → Use the fieldbus blocks provided, as these already carry out the necessary value scaling.
10201 / 0x27D9	No request possible	When sending a request/ command via the "SFK_command" block, the system is not ready to accept the request. The error message is sent directly via the "SFK_command block".	→ Wait until the system is set to "Request possible". "request ready" bit 11 receives the value 1.
10202 / 0x27DA	No fieldbus type stored	When controlling via the fieldbus blocks provided, the input value for assigning the fieldbus type is incorrect. The error message is sent directly via the fieldbus block.	→ Check whether the input "FieldbusType" is correctly assigned.
10203 / 0x27DB	Serial number not valid	When using the "SFK_SerialNumberConverter" block, too many digits were used for the serial number.	→ Shorten the serial number to a maximum of 28 characters.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
10205 / 0x27DD	Invalid character given within alphanumeric customID	When starting the program, a serial number was transmitted that contains invalid characters: The program will not start.	→ Check the transmitted customID for invalid characters. ◊ For more information about the settings, see chapter Fieldbus configuration and web interface. → Set the unused request values to 0.
10206 / 0x27DE	Invalid entry	The selected variable is read-only and cannot be written.	→ Select a non-read-only variable. Possible writable variables are: 200 - 999.
10207 / 0x27DF	Invalid data type	The data type used in the request is not supported.	→ Select a correct VariableType. Allowed values are: 'VF', 'VFR', 'MF', 'MFR'.
10208 / 0x27E0	Variable index exceeded	The index specified in the query exceeded the available number of entries.	→ Check the specified index. Allowed values are: 0 - 999.
10209 / 0x27E1	Value too small	When writing a variable via the "Fast_Variable_Access" block, a value that was too small was used.	→ Increase the value you want to write.
10210 / 0x27E2	Value too large	When writing a variable via the "Fast_Variable_Access" block, a value that was too large was used.	→ Decrease the value you want to write.
10211 / 0x27E3	Value invalid	When writing a variable via the "Fast_Variable_Access" block, an invalid value was used.	→ Select a permissible value.
20101 / 0x4E85	Command active, command transition not permitted	A command has been activated: The processing cannot be started due to an active program.	→ Cancel the currently active command via the stop command in the web interface, the OPC UA or the fieldbus. → Wait for the current command to be completed.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
20102 / 0x4E86	Access of participant locked	A command has been activated: Processing cannot be started as the requesting participant is locked.	→ Unlock the participant.
20103 / 0x4E87	Timeout during cancel	-	→ Inform the service.
20104 / 0x4E88	Maximum force of the system exceeded	The maximum force of the system has been exceeded.	→ Move the system out of the force by relative or absolute positioning (= retracting the cylinder). Using the VRF006 system variable, the maximum force (monitoring) of the system can be parameterized.
30001 - 31901 / 0x4E86 - 0x4E87	Internal error	-	→ Inform the service.
31006 / 0x791E	Faulty Oscilloscope evaluation	A program run was not successfully completed due to a faulty configuration of the drive-internal oscilloscope.	→ If you operate the oscilloscope with IndraWorks DS, first deactivate data recording using parameter MFR007 (see Hardware configurations). → Reset the changed oscilloscope configuration by clearing all configured parameters. → If you operate the oscilloscope with IndraWorks DS, terminate the connection to the oscilloscope. → Stop the oscilloscope, if you operate it with IndraWorks DS. → Restart the system after changing the oscilloscope configuration or when MFR007 has been changed.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
31007 / 0x791F	No available space left on IPC (PR21)	Less than 100 MB available space on the IPC.	→ Delete the production curves in the log file area. **Notice** To avoid data loss, no further pressing operations can be carried out. The jog functions remain active and the system can be moved manually.
31009 / 0x7921	No program selected	The start command was given without a program being selected.	→ Select a program. → Restart the program.
40101 / 0x9CA5	*Clear Error* Command not possible	The *Clear error* command was aborted with an error or the processing is not possible.	→ Check that the system is ready. → Wait for the current command to be completed. → Cancel the currently active command via the stop command in the web interface, the OPC UA or the fieldbus. → Restart the system and try executing the command sequence again.
40201 / 0x9D09	Reading of SMC variables not possible	In the system, an SMC variable from the underlying technology library is requested, which cannot be read.	→ Check the variable name entered. → Restart the system and read out the variable again.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
40301 / 0x9D6D	Positioning command cannot be started	A positioning command (e.g. absolute positioning, continuous *Jog* command, or a program with positioning command) was activated but cannot be processed.	→ Cancel the currently active command via the stop command in the web interface, the OPC UA or the fieldbus. → Check that the system is referenced. → Check that the power at the system is switched on. → Check if the system is ready for the procedure: There is no error, the safety technology is enabled and the parameter mode is not called.
40302 / 0x9D6E	Error in the processing of the stop command	To stop the SFK: The *Stop axis* command is executed in various system functions. An error has been identified during processing of the command.	→ Check the parameterization of the system. → If necessary, restore the default values of the extended configuration. → Restart the system and try executing the command sequence again.
40303 / 0x9D6F	Target position was not reached	A positioning command (e.g. absolute positioning, continuous *Jog* command, or a program with positioning command) was activated and terminated without having reached the target position.	→ Cancel the currently active command via the stop command in the web interface, the OPC UA or the fieldbus. → Check that the system is referenced. → Check that the power at the system is switched on. → Check if the system is ready for the procedure: There is no error, the parameter mode is not called.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
40304 / 0x9D70	Manual positioning command: Force limit exceeded	A positioning command (e.g. absolute positioning, continuous *Jog* commands) has been activated and during this, the defined maximum force has been reached or exceeded.	→ Check the value of the parameter VFR004. If necessary, adjust the value of the parameter (see chapter Hardware configuration). → Check that the system is correctly tared. If required, tare it to the value 0 N. → Check if the system is ready for the procedure: There is no error and the parameter mode is not called. → Check the plausibility of the displayed force value. If required, release any mechanical deadlocks.
40401 / 0x9DD1	Reading of motor feedback memory not possible	During commissioning, the system parameters are read from the motor feedback memory of the MS2N motor: An error has occurred during reading of the motor feedback memory.	→ Perform the commissioning step again. → Start commissioning from the beginning. → Contact customer service. An incorrectly parameterized system may result in severe damage to the plant.
40501 / 0x9E35	Reboot of drive not possible	The *Restart* command was started at the drive controller: The restart could not be processed.	→ Check if a drive command (e.g. *homing*) is active.
40601 / 0x9E99	Homing could not be started	The *Homing* command could not be started.	→ Check that the power is switched on. → Check if the system is ready for the procedure: There is no error and the parameter mode is not called.
40602 / 0x9E9A	Referencing under load not possible	The homing may only be started in unloaded condition.	→ Move the SFK to a load-free state. **Notice** Use the jog operation in the "Relative movement" mode.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
40701 / 0x9EFD	Could not activate program	An error has occurred during switching of a program: The program cannot be activated.	→ The currently active program is being processed. → Repeat the switching after processing of the program.
40702 / 0x9EFE	Timeout while program will be activated	A timeout error occurred while switching a program: The system could not activate the desired program within the expected period.	→ Repeat the switching. → As necessary, save the program again. → If the error occurs again: Contact the service.
40703 / 0x9EFF	Timeout after program is activated	An error has occurred during switching of a program: The program cannot be activated without errors.	→ Repeat the switching. → As necessary, save the program again. → If the error occurs again: Contact the service.
40801 / 0x9F61	Could not write SMC variable	An SMC variable of the underlying technology package SMC cannot be written: The write process was aborted with an error.	→ The SMC variable is read-only: Check for correct use of the variable in the SMC documentation.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
40901 / 0x9FC5	Error when switching the operating mode parameter mode <> operating mode	An error has occurred while trying to change the operating mode of the drive controller.	→ The system is in motion: Terminate the active motion command and try switching again. → Incorrect parameterization in the extended configuration: Enter valid parameter values for the parameters or reset the parameters to the default values. → Incorrect parameterization of the Safe Motion functionality: Make sure that the Safe Motion functionality does not prevent switching from Parameter Mode (PM) to Operating Mode (OM). → Check the configuration in IndraWorks DS. Safe bus communication (e.g. PROFIsafe): For (re-)commissioning, the fieldbus including the safe bus communication is deactivated (see chapter Settings for safe fieldbus communication). Among other things, this causes the safety networks to be reset. → Check the configuration in IndraWorks DS.
41001 / 0xA029	Could not start program	An error has occurred during start of the active program.	→ Check that the power is switched on. If required, switch on the power. → Check that the system is referenced. Reset the reference if necessary. → Check if the system is ready for the procedure: There is no error and the parameter mode is not called. → Restart the system. → Load the program again via the selection list in the status bar and repeat the Start command.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41002 / 0xA02A	Could not stop program	A program can be stopped via the stop command or by an identified program error: A malfunction has occurred during stopping.	→ Restart the system. → Load the program again via the selection list in the status bar and repeat the Stop command. → If the error occurs again: Contact the service.
41003 / 0xA02B	Program error: Max. position exceeded	The maximum position programmed in a module was exceeded during processing of the program: The active program step can be checked in the dashboard in the sequence display.	→ Check parameterization of the module and adjust the maximum position. → Check the parameterized dynamics of the overall process and adjust it as required.
41004 / 0xA02C	Program error: Max. force exceeded	The maximum force was exceeded in a module during processing of the program: • The active program step can be checked in the dashboard in the sequence display. • In the "Join to force" module: The system could not be stopped quickly enough to safely reach the specified target force value (with tolerance value VFR002). • In all other modules: The parameterized maximum force was exceeded. • The system has triggered an emergency stop.	→ Check the parameterization of the affected module and adjust the maximum force. → Check the parameterized dynamics of the overall process and adjust it as required. → In the module "Join to force": Reduce velocity and increase acceleration. Check whether a higher permissible target force overshoot can be specified using parameter VFR002. → Then restart the program.
41005 / 0xA02D	Program error: Cancelled manually	A program was canceled manually by the Stop command.	→ Check whether the stop command was carried out by the trigger sources web interface, OPC UA or fieldbus.
41006 / 0xA02E	Program error: Abort due to drive error	A drive error occurred during program execution which prevented further processing.	→ Eliminate the drive error and restart the program.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41007 / 0xA02F	Program error: Could not write Y-Parameter	When creating a program, Y parameters of the underlying SMC technology package are written in the program sequence, depending on the programmed sequence. One of these parameters is write-protected during processing of the program sequence or its value range has been exceeded.	→ Check the operating mode of the system. If the system is not ready (see information in the status bar), rectify the problem as suggested by the note. → Check the parameterization of the system, e.g. with regard to violation of limits.
41008 / 0xA030	Program error: Timeout exceeded	The programmed timeout was exceeded for a module during program execution: The triggering program step is marked in the dashboard.	→ Check parameterization of the module and adjust the timeout.
41009 / 0xA031	Program error: Safety active	The safety technology was activated during processing of the program.	→ Check the safety technology.
41010 / 0xA032	Program error: Out of target window, force too low	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum force was undershot. • The position was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.
41011 / 0xA033	Program error: Out of target window, force exceeded	During program execution, the following occurred in the module "Check target window force/displacement": • The specified maximum force was exceeded. • The position was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41012 / 0xA034	Program error: Out of target window, invalid force evaluation	During program execution, the following occurred in the module "Check target window force/displacement": • The specified minimum force was not reached and at the same time the specified maximum force was exceeded. • The position was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum force is lower or equal the maximum force.
41013 / 0xA035	Program error: Out of target window, position too low	During program execution, the following occurred in the module "Check target window force/displacement": • The specified minimum position was undershot. • The force was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.
41014 / 0xA036	Program error: Out of target window, force and position too low	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum force and minimum position were simultaneously undercut. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41015 / 0xA037	Program error: Out of target window, force exceeded, position too low	During program execution, the following occurred in the module "Check force/position target window": • The specified maximum force was exceeded and at the same time the specified minimum position was undercut. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.
41016 / 0xA038	Program error: Out of target window, invalid force evaluation, position too low	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum force was not reached and at the same time the specified maximum force was exceeded. • The specified minimum position was undershot. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum force is lower or equal the maximum force.
41017 / 0xA039	Program error: Out of target window, position exceeded	During program execution, the following occurred in the module "Check force/position target window": • The specified maximum position was exceeded. • The force was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41018 / 0xA03A	Program error: Out of target window, force too low, position exceeded	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum force was not reached and at the same time the specified maximum position was exceeded. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.
41019 / 0xA03B	Program error: Out of target window, force and position exceeded	During program execution, the following occurred in the module "Check force/position target window": • The specified maximum force and maximum position were exceeded simultaneously. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization.
41020 / 0xA03C	Program error: Out of target window, invalid force evaluation, position exceeded	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum force was not reached and at the same time the specified maximum force was exceeded. • The specified maximum position was exceeded. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum force is lower or equal the maximum force.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41021 / 0xA03D	Program error: Out of target window, invalid position evaluation	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum position was undercut and at the same time the specified maximum position was exceeded. • The force was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum position is lower or equal the maximum position.
41022 / 0xA03E	Program error: Out of target window, force too low, invalid position evaluation	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum position was undercut and at the same time the specified maximum position was exceeded. • The specified minimum force was undershot. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum position is lower or equal the maximum position.
41023 / 0xA03F	Program error: Out of target window, force exceeded, invalid position evaluation	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum position was undercut and at the same time the specified maximum position was exceeded. • The specified maximum force was exceeded. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum position is lower or equal the maximum position.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41024 / 0xA040	Program error: Out of target window, invalid force and position evaluation	During program execution, the following occurred in the module "Check force/position target window": • The specified minimum force was not reached and at the same time the specified maximum force was exceeded. • The specified minimum position was undercut and the specified maximum position was exceeded at the same time. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the minimum force is lower or equal the maximum force. → Make sure that the minimum position is lower or equal the maximum position.
41025 / 0xA041	Program error: Force control only available with force sensor	During program execution, the "force-controlled joining" module detected that no force sensor was in use. The triggering program step is marked in the dashboard.	→ To use the "force-controlled joining" functionality: Make sure the force sensor is configured correctly.
41026 / 0xA042	Program error: Error in the module "Force control": Control dynamics (P) is zero	During program execution, it was detected in the "force-controlled joining" module that the control dynamics are zero. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the control dynamics are not zero.
41027 / 0xA043	Program error: Error in the module "Force control": Time for reaching target force exceeded	During program execution, the programmed timeout in the "force-controlled joining" module has expired. The triggering program step is marked in the dashboard.	→ Check the module parameterization. → If necessary, adjust the module parameterization. → Make sure that the target force is achieved within the maximum time for reaching the target force, taking into account the force tolerance.
41028 / 0xA044	Program error: Error maximum force exceeded in module	During program execution, a maximum force defined in the module was exceeded.	→ Check the module parameterization. → If necessary, adjust the module parameterization.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41101 / 0xA08D	S- or P-parameter could not be read	An error has occurred while reading an S or P parameter internal to the drive.	→ Check the correct notation of the parameter IDN and repeat the process. → Restart the system and read out the parameter again. → Incorrect parameterization in the extended configuration: Enter valid parameter values for the parameters or reset the parameters to default values.
41201 / 0xA0F1	S- or P-parameter could not be written	An error has occurred while writing an S or P parameter internal to the drive.	→ Check the correct notation of the parameter IDN and repeat the process. → Check whether the parameter is temporarily write-protected. If necessary, activate the Parameter Mode operating state. → Check in the documentation of the drive controller whether the value to be written is compliant with the value range of the parameter. → Adjust the value and repeat the write process.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41301 / 0xA155	Could not load Y-Parameter file	During commissioning, the SMC technology package is automatically configured. The parameters required for configuration are read from a parameter file: One or several parameters from this file could not be loaded. When using OPC UA: The function loadYParameterFile() was requested via OPC UA: One or several parameters from this file could not be loaded.	→ Perform the commissioning step again. → Start commissioning from the beginning. → Contact customer service. An incorrectly parameterized system may result in severe damage to the plant. When using OPC UA: → Check the path specified for the parameter file. → Check the operating mode of the system. Some parameters can only be written in parameter mode. → Change to parameter mode and call up the function again.
41401 / 0xA1B9	Tare not possible	An error has occurred during processing of the *tare* command: The actual force value of the analog force sensor was not set to the set offset.	→ Restart the system and repeat the process. → If the error occurs again: Contact the service.
41501 / 0xA21D	Basic parameter loading not possible at drive commissioning	During commissioning, the drive parameters are reset to their default values and the initial configuration is performed.	→ Perform the commissioning step again. → Start commissioning from the beginning. → Stop the system. Loading the basic parameters requires that no movement (program start, positioning, etc.) be active. → Contact customer service. An incorrectly parameterized system may result in severe damage to the plant.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
41601 / 0xA281	Absolute dimension could not be set	The command *Set absolute reference* was aborted with error: The reference point was not set.	→ Check the parameterization of the system. → Check whether the reference point (S-0-0052, extended configuration) was set outside the system limits. → Check the operating mode of the system. 💡 Using the *Set absolute reference* command must be done in operating mode. → Stop the system. 💡 No movement (program start, positioning, etc.) must be active when using *Set absolute reference*.
41701 / 0xA2E5	Could not read Y-Parameter	An error occurred while reading a Y parameter of the underlying SMC technology package.	→ Check the correct notation of the parameter IDN and repeat the process. → Restart the system and read out the parameter again.
41801 / 0xA349	Could not write Y-Parameter	An error occurred while writing a Y parameter of the underlying SMC technology package.	→ Check the correct notation of the parameter IDN and repeat the process. → Check whether the parameter is temporarily write-protected. If necessary, activate the Parameter Mode operating state. → Check the documentation of the SMC technology package to see whether the value to be written corresponds to the value range of the parameter. → Adjust the value and repeat the write process.

No. (Dec. / Hex.)	Designation	Problem	Remedial measure
50001 / 0xC351	Character string too long, error while writing	An error has occurred in the communication between PR21 and HCS01: The data cannot be transferred completely.	→ Repeat the command sequence leading to the error. 💡 In exceptional situations, excessive data volume on the interface can lead to a transmission error. → Restart the system and repeat the command sequence. → Contact customer service.

Table 60 System errors

19.7 Warnings

Various monitoring functions are active in the Smart Function Kit. If a condition is detected that still allows proper operation but will result in an error or limited functionality if continued, a warning message is generated.

> **Notice**
>
> Warnings cannot be deleted. They remain pending until the condition for triggering the warning is no longer met.

Warnings appear in the title bar of the user interface. When the fieldbus is active, the warning bit is set in the status word.

> **Notice**
>
> Error messages have a higher priority than warnings and can therefore override pending warnings in the user interface.

No.	Designation	Cause	Remedial measure
10	SMO: General error	Drive type S4: Error configuration Safe Motion	→ Check the Safe Motion configuration.
11	SMES: Emergency off active	Drive type S4: Special operation "EMERGENCY STOP" active	→ Initiate the state change to NO.

No.	Designation	Cause	Remedial measure
12	SMST: Safe standstill active	Drive type S4: Special operation "Safe standstill" active	→ Initiate the state change to NO.
13	Stop and torque off active	Drive type L3: Stop and torque off function active	→ Release IX31.3, set the level to HIGH.
14	Temperature warning. Limitation of performance possible.	Warning temperature motor and/or drive exceeded	See chapter Temperature Warning.
15	Y0001 parameter value must be 4000 µs.	Cycle time of the integrated control was adjusted by external specification	→ Specify a fieldbus cycle time of 4000 µs or more on the control side.
31	Special Mode Safe Motion 1 active (SMM1)	Drive type S4: Special mode Safe Movement 1 active	
32	Special Mode Safe Motion 2 active (SMM2)	Drive type S4: Special mode Safe Movement 2 active	
33	Special Mode Safe Motion 3 active (SMM3)	Drive type S4: Special mode Safe Movement 3 active	
34	Special Mode Safe Motion 4 active (SMM4)	Drive type S4: Special mode Safe Movement 4 active	
35	Special Mode Safe Motion 5 active (SMM5)	Drive type S4: Special mode Safe Movement 5 active	
36	Special Mode Safe Motion 6 active (SMM6)	Drive type S4: Special mode Safe Movement 6 active	
37	Special Mode Safe Motion 7 active (SMM7)	Drive type S4: Special mode Safe Movement 7 active	
38	Special Mode Safe Motion 8 active (SMM8)	Drive type S4: Special mode Safe Movement 8 active	
39	Special Mode Safe Motion 9 active (SMM9)	Drive type S4: Special mode Safe Movement 9 active	
40	Special Mode Safe Motion 10 active (SMM10)	Drive type S4: Special mode Safe Movement 10 active	
41	Special Mode Safe Motion 11 active (SMM11)	Drive type S4: Special mode Safe Movement 11 active	

No.	Designation	Cause	Remedial measure
42	Special Mode Safe Motion 12 active (SMM12)	Drive type S4: Special mode Safe Movement 12 active	
43	Special Mode Safe Motion 13 active (SMM13)	Drive type S4: Special mode Safe Movement 13 active	
44	Special Mode Safe Motion 14 active (SMM14)	Drive type S4: Special mode Safe Movement 14 active	
45	Special Mode Safe Motion 15 active (SMM15)	Drive type S4: Special mode Safe Movement 15 active	
46	Special Mode Safe Motion 16 active (SMM16)	Drive type S4: Special mode Safe Movement 16 active	

Table 61 Warnings

20 Maintenance and component replacement

20.1 CMOS-Industry PC Battery PR21

The CMOS battery (Material number R911401323) has a service life of approximately 5 years, depending on the environmental conditions and usage scenario.

The service life can be significantly reduced if there are frequent or long interruptions in the power supply (for example when the system is switched off or when it is stored).

A preventive replacement after 2 years is recommended, although the actual lifespan may be significantly longer.

→ Replace the CMOS battery before it exceeds its expected lifespan.

→ Proceed according to the instructions in the manual of the Industrial PC PR21 (R911389662).

Notice
If the CMOS battery is empty, a warning message will be displayed in the BIOS when the system starts. In addition, the system can no longer be pinged successfully (XF6 does not blink). The characteristic beeping sound of the PR21 industrial PC during system startup is absent. The Smart Function Kit software does not start and the system is not immediately ready for use.

After replacing the CMOS battery, the warning message must be acknowledged once the next time the PR21 is restarted, following the steps below.
Alternatively, the message is automatically acknowledged by two restarts.

If the CMOS battery is empty, the system time may differ during operation. As a result, validation of an administrator login may fail. In this case, replace the CMOS battery.

To acknowledge the warning message, proceed as follows:

1. Press the SF2 button and wait about one minute.
2. Press the SF2 button again.

   ✅ The PR21 boot process continues.
   ✅ The SFK software starts automatically.

3. Acknowledge the warning message again the next time you restart the system.

Alternatively, you can acknowledge the warning message in the BIOS as follows:

1. Connect a monitor and keyboard to the PR21.
2. Acknowledge the warning message "Warning!! CMOS Checksum error" with the Key F1.

   ✅ The PR21 boot process continues.
   ✅ The SFK software starts automatically.

3. Acknowledge the warning message again the next time you restart the system.

After replacing the CMOS battery, the system time must be synchronized again:

1. Open the System configuration.
2. Press the SYNC button.

   ✅ The system time of the HMI operator panel is adopted.

20.2 Component replacement

When replacing components, certain parameters must be adjusted to ensure the correct functioning of the system.

> **Notice**
> Refer to the relevant component replacement instructions for detailed instructions.
> If adjustment is not carried out correctly, there is a risk of data loss and the system becoming unusable!

When changing, two cases must be distinguished:

- In an equivalent exchange, a component is exchanged for one with the same material number. The system configuration does not change.
- If the configuration differs, a component is exchanged for one with a different material number. The system configuration and material number of the plant change.

Carry out a process check after each component replacement:

→ Check the relevant programs for repeatability of previous process results.

→ Adjust the relevant process parameters to restore previous process results.

→ Check in the section Hardware configurations the displayed components and the system parameters for correctness.

20.2.1 HCS01 drive controller

Equivalent exchange

In an equivalent exchange, the controller is replaced with a controller with the same configuration (same material number).

1. Replace the drive controller. Make sure that the wiring is correct.
2. Perform a new *commissioning*.
   - For detailed information, see chapter Commissioning.
   - The component exchange can only be carried out successfully by commissioning the software. Some parameters are reset to their default values. Note down any modified parameters or create a backup of the respective parameter pages.
   - Check the displayed components and system parameters in chapter Hardware configurations to ensure they are correct.
3. Carry out *homing* and *taring*.

⚠️ Parameters that are affected by the controller replacement and must be checked/adjusted:

- (Motor), SafeMotion, Tare value, IOs, fieldbus, programs

→ Check if the parameters match the previous setting.

Different configuration

When replacing a controller with a different configuration, the configuration of the drive controller is changed, e.g. changing from L3 (STO) to S4 (SafeMotion).

1. Replace and install the controller with the different configuration. Make sure that the wiring is correct.

2. Perform a new *commissioning*.
    - For detailed information on commissioning, please refer to chapter Commissioning.
    - The component exchange can only be carried out successfully by commissioning the software. Some parameters are reset to their default values. Note down any modified parameters or create a backup of the respective parameter pages.
    - Check the displayed components and system parameters in chapter Hardware configurations to ensure they are correct.
3. Carry out *homing* and *taring*.

⚠ Parameters that are affected by the controller replacement and must be checked/adjusted:

- (Motor), velocity, acceleration, torque, force, SafeMotion, tare value, IOs, fieldbus, programs.

→ Perform a process control.

20.2.2 Force sensor

> **Notice**
> The position measurement takes place in the motor encoder. Maintenance or replacement of the force sensor influences the actual tool position.

Equivalent exchange

In an equivalent exchange, the force sensor is replaced with a force sensor with the same configuration (same material number).

1. Replace the old force sensor with an identical force sensor. Make sure the wiring is correct.
    - For further information on wiring the peripheral devices with the SFK, see the assembly instructions (R320103194) and the example circuit diagram, see www.boschrexroth.com/medienverzeichnis.
2. Since the length between the piston rod and the tool may have changed, an adjustment of the program or the reference dimension may be necessary.
    - Perform a process control with the relevant programs and adjust the target positions in all relevant steps based on the determined offset.
    - The reference dimension can be adjusted in chapter Hardware configurations → *S-0_0052 Reference dimension* .

→ Perform a process control.

Different configuration

Switch to a special force sensor or customer-specific force sensor.

1. Install the force sensor according to the manufacturer's instructions.
2. Tare the force sensor.
    - For more information on taring the force sensor, see section Tare force sensor.

⚠ Parameters that are affected by the force sensor replacement and must be checked/adjusted:

- Since the length between the piston rod and the tool may have changed, an adjustment of the program or the reference dimension may be necessary.
    - The reference dimension can be adjusted in chapter Hardware configurations → *S-0_0052*.

For parameters that need to be adjusted, see chapter Customized force sensor.

→ Perform a process control.

Maintenance

It is recommended that the force sensor be calibrated by an accredited laboratory at regular intervals of approximately 24 months.

1. Contact the manufacturer or an accredited laboratory.
2. Adjust the corresponding parameters after calibration:

- Since the length between the piston rod and the tool may have changed, an adjustment of the program or the reference dimension may be necessary.
  - The reference dimension can be adjusted in chapter Hardware configurations → *S-0_0052*.

→ Perform a process control.

Retrofit force sensor

Installation of a force sensor that was not included in the Smart Function Kit.

→ Install the force sensor according to the manufacturer's instructions.

⚠ Parameters that are influenced by the force sensor assembly and must be checked/adjusted:

- Tare value, Activation force sensor, Force sensor configuration

1. Tare the force sensor.
2. For parameters that need to be adjusted, see chapter Customized force sensor.

→ Perform a process control.

20.2.3 Motor cable

Equivalent exchange

Replacement of sensor or power cable.

→ Replace the old cable with the new cable.

✅ If electromagnetic compatibility is ensured, no adjustments are necessary.

20.2.4 PR21 industrial PC

The operating software and the SFK database are located on the PR21.

1. Make a backup if necessary.
   ◊ For more information on creating a backup, see chapter Installation.
2. Replace the old PR21 with a new PR21.
3. Perform a new *commissioning*.
   - For detailed information on commissioning, please refer to chapter Commissioning.
   - The component exchange can only be carried out successfully by commissioning the software. Some parameters are reset to their default values and can be adjusted to the previous value by manually changing them or restoring a backup.
3. Carry out *homing* and *taring*.

⚠ Parameters that are affected and need to be checked/adjusted:

- SmartFunctionKit software, variables, programs

→ Check if the parameters match the previous setting.

→ Perform a process control.

Maintenance

During maintenance, the old (empty) CMOS battery is replaced with a new (full) battery.

→ Replace the old battery with a new battery. Please follow the relevant instructions.

◊ For more information about the battery, see chapter CMOS-Industry PC Battery PR21.

→ Synchronize the system time. For detailed information, see chapter System configuration.

20.2.5 Electromechanical cylinder EMC

Notice
The position measurement takes place in the motor encoder. Maintenance or replacement of the electromechanical cylinder affects the actual tool position.

Equivalent exchange

In an equivalent exchange, the electromechanical cylinder is replaced with a cylinder with the same configuration (same material number).

1. Replace the cylinder with an equivalent cylinder. Please observe the relevant assembly instructions.

⚠ Manufacturing tolerances may cause deviations in the tool position.

2. Carry out *homing*.

→ Perform a process control.

Different configuration

Change to an electromechanical cylinder with a different configuration.

1. Replace and install the electromechanical cylinder with a different configuration. Please observe the relevant assembly instructions.
2. Carry out *homing*.

⚠ Parameters that are affected by the exchange and need to be checked/adjusted:

- Stroke, lead, velocity, acceleration, moment, force

3. Adjust the following parameters in the Extended Configuration based on the manufacturer information:
    - **S_0_0123** Ball screw lead
    - **P_0_1321** Force limit for homing
    - **Y1044** Maximum stroke positive
    - **Y1006** Maximum acceleration
    - **Y1007** Maximum drive torque
    - **VFR006** Maximum force

→ Perform a process control.

Maintenance

During maintenance, the electromechanical cylinder is lubricated.

→ After maintenance, carry out *homing*.

✓ Since no components are replaced, no parameters are directly influenced/changed.

20.2.6 Motor mounting axial/parallel

Notice
The position measurement takes place in the motor encoder. Maintenance or replacement of the motor attachment affects the actual tool position.

Equivalent exchange

In the case of an equivalent exchange, the motor attachment is replaced with a motor attachment with the same configuration (same material number).

1. Replace the motor attachment with an equivalent motor attachment. Please observe the relevant assembly instructions.
2. Carry out *homing*.

⚠ Manufacturing tolerances may cause deviations in the tool position.

→ Perform a process control.

Different configuration

Change to a motor attachment with a different configuration.

1. Replace and install the motor attachment with a different configuration.
2. Carry out *homing* .

⚠ Parameters that are affected by the exchange and need to be checked/adjusted:

- Gear ratio, driving direction, velocity, acceleration, torque, force

3. Adjust the following parameters in the Extended Configuration based on the manufacturer information:
    - **S_0_0121** Gear reduction input rate
    - **S_0_0122** Gear reduction output rate
    - **Y1004** Maximum velocity
    - **Y1006** Maximum acceleration
    - **Y1007** Maximum drive torque
    - **Y1011** Negation of position data
    - **VFR006** Maximum force

→ Perform a *process control*.

Maintenance

During maintenance, the timing belt is replaced (only for the "parallel" motor mounting variant).

→ After maintenance, carry out *homing*.

⚠ Manufacturing tolerances may cause deviations in the tool position.

→ Perform a process control.

20.2.7 Motor

Notice
The position measurement takes place in the motor encoder. Maintenance or replacement of the motor affects the actual tool position.

Equivalent exchange

In an equivalent exchange, the motor is replaced with a motor with the same configuration (same material number).

1. Replace the motor with an equivalent motor. Please observe the relevant assembly instructions.
2. Perform a new *commissioning*.
    - For detailed information on commissioning, please refer to chapter Commissioning.
    - The component exchange can only be carried out successfully by commissioning the software. Some parameters are reset to their default values. Note down any modified parameters or create a backup of the respective parameter pages.
3. Carry out *homing*.

→ Perform a process control.

Different configuration

Change to a motor with a different configuration.

1. Replace and install the motor with different configuration. Please observe the relevant assembly instructions.
2. Perform a new *commissioning*.
    - For detailed information on how to restart the system, please refer to chapter Commissioning.
    - The component exchange can only be carried out successfully by commissioning the software. Some parameters are reset to their default values. Note down any modified parameters or create a backup of the respective parameter pages.
3. Carry out *homing* .

⚠ Parameters that are affected by the exchange and need to be checked/adjusted:

- (drive), velocity, acceleration, torque, force

4. Adjust the following parameters in the Extended Configuration based on manufacturer information:
    - **P_0_1321** Force limit for homing
    - **Y1004** Maximum velocity
    - **Y1006** Maximum acceleration
    - **Y1007** Maximum drive torque
    - **VFR006** Maximum force

→ Perform a *process control*.