rexroth
A Bosch Company

Smart Function Kit SFK - Software

R320103208, 2022-01, 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.

R320103208, 2022-01, en

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 Continuous 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 Hinweis

Hinweis
Important information or notice on handling

1.1.6 Console input

testüberschrift

```
helloWorld();
```

1.1.7 Code snippet / Code files

testüberschrift

```
helloWorld();
```

1.1.8 Inline code

Alle `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
FE	**F**unctional **E**arth
HCS01	Drive controller
HMI	**H**uman **M**achine **I**nterface
HTTPS	**H**ypertext **T**ransfer **P**rotocol **S**ecure
ID	**Id**entification number
I/O	Digital input/output
IP	**I**nternet **P**rotocol
JSON	**J**ava**S**cript **O**bject **N**otation
NOK	Not okay rating
OK	Okay rating
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
PLC	**Programmable Logic Controller**
SYNC	**Sync**hronisation
XF5, XF6	Network interfaces at the industrial PC
X26	Network interface at the drive controller
X31	I/O connector strip at the drive controller

1 Abbreviations

2 Safety

2.1 Intended use

The Smart Function Kit is an assembly intended for integration into or combination with machines or incomplete machines or equipment.

Only use the product as intended.

Intended use also includes compliance with the operating, maintenance, and servicing conditions 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 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 power supply before performing 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.

→ Any defects identified at electrical components must be corrected immediately.

→ Always work in accordance with the wiring diagram.

→ Do not use defective parts.

⚠ DANGER

Uncontrolled movement due to software malfunction or due to incorrect wiring of I/Os
Impact, crushing, pinching, ejection/slinging away.

→ Never reach into moving parts.

→ Operation only with the safety door closed.

⚠ DANGER

Uncontrolled movement due to remote activation (HMI, field bus)
Impact, crushing, pinching, ejection/slinging away.

→ Never reach into moving parts.

→ Operation only with the safety door closed.

⚠ CAUTION

Indirect contact with live parts (due to fault, e.g. defective insulation)

Electric shock

3 Commissioning

Commissioning is performed in several steps and is guided by the software. During this process, basic configurations, such as language and password, as well as e.g. hardware-related settings can be performed. After successful commissioning, the system is immediately ready for operation.

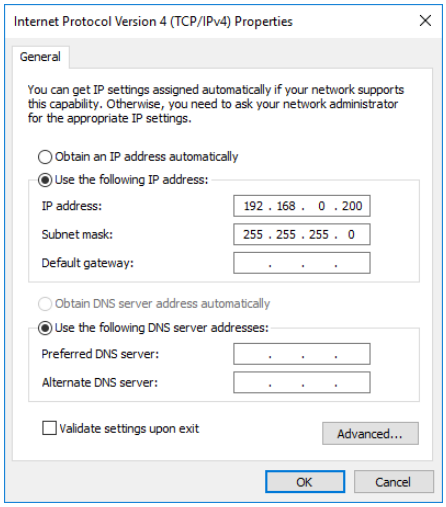
If the safety option S4 (Safe Motion) is used at the HCS01 drive controller, the safety technology must also be commissioned to ensure that the safety-specific functions of the drive controller can be used. Commissioning of the Smart Function Kit must be performed before commissioning of the safety technology.

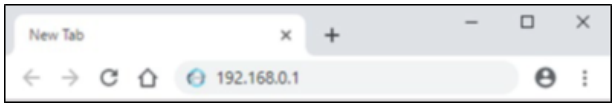
◊ Detailed information in the chapter Commissioning of safety technology with safety option S4

3.1 Initial commissioning

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

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

Screens /comments	Action steps
 Factory set network settings for the system: • IP address: 168.0.1 • Subnet mask: 255.255.255.0 • Gateway address: 0.0.0.0 Required network configuration of the commissioning PC: • Static IP address, not obtained automatically • IPv4 address: 168.0.x with x > 1 for example 192.168.0.200 • Subnet mask: 255.255.255.0	→ Ensure that your HMI terminal device or a commissioning PC is connected with the network of the Smart Function Kit.
	→ Check the network settings of your HMI terminal device.

Screens /comments	Action steps
	→ Open your web browser and enter the IP address of the Smart Function Kit (default: 192.168.0.1). ✅ The web browser will now automatically connect to the Smart Function Kit and display the commissioning dialog. ✅ Depending on the security settings of your web browser and on the browser, a security warning to be confirmed will appear during connection establishment. 💡 The reason for this is that the connection is established via HTTPS and the certificate used is not yet known to the web browser. **Notice** If the Smart Function Kit interface is not displayed, even after a short waiting time: → Check whether there is a network connection to the PR21. → Possibility: PING test

2 Initial commisioning

3.1.1 Login

→ Log in at the system.

→ Use the default password *admin*.

→ If the password has already been changed, use the new password.

1. Enter the password into the appropriate field.
2. Press *Enter* or click on *Login*.
3. After successful login, click on *Next* to continue to the next commissioning step.

3.1.2 Language selection and password

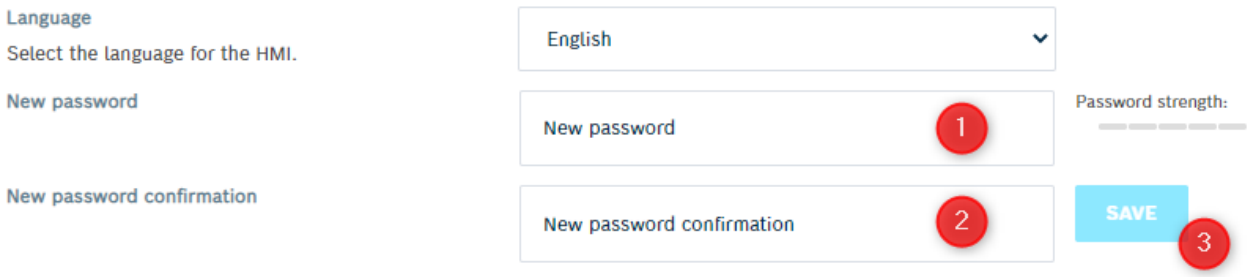
→ Select the language for the system.

✅ The languages German and English are available.

→ Select the desired language in the selection menu.

The Smart Function Kit has two user accounts. One administrator and one operator account.

- The administrator account is password-protected and used for system configuration, creation of programs, definition of quality assessment and export of data.
- The operator account is not password-protected and is the same as the logged-out condition. The operator account
  is used in actual operation and only allows restricted access to the system.



1 Change password

Possibility to change the password for the administrator account during commissioning:

1. Enter a new password into the field (1).
2. Confirm the new password in the field below (2).
3. Click on *Save* (3) to save the new password.

✅ Changing the password overwrites the default password.

✅ After that, 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 for the administrator account can also be changed any time in the system settings.

→ Click on *Next* to continue to the next commissioning step.

ATTENTION	
	**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.

ATTENTION	
	**Loss of system settings and system data** The administrator password cannot be reset. If the password is lost, access to the system can only be restored by new installation. → Note down the new password and keep it in a safe place.

3.1.3 System check

To check the system, the system performs a self-diagnosis and checks that all components are correctly connected and ready.

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

→ Click on *Next* to continue to the next commissioning step.

✅ If any faulty component is recognized during the system check, the *Next* button is disabled.

✅ Commissioning cannot be directly continued.

→ Check the following:

1. Correct connection of the drive controller HCS01 with the PR21.
2. Both devices are switched on.
3. Network settings at the drive controller HCS01.
4. Software is correctly installed on the PR21.

◊ For detailed information on the individual steps, please see the chapter Troubleshooting and fault clearance at the end of this instructions manual.

3.1.4 Installation of the software

In this step, the Smart Function Kit for is set up on the hardware and the system configured by means of its framework parameters.

ATTENTION	
	**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 nor the drive controller during the update. 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.

→ Click *Start* to start with the installation. The installation progress is then

✓ indicated by a progress bar.

→ As soon as the installation is completed, press *Next* to continue to the next commissioning step.

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 any additional modules like safety zone modules for correct connection and configuration.

→ Click on *Restart*.

✓ The entire installation is restarted.

◊ For detailed information, please see the chapter Troubleshooting and fault clearance at the end of this instruction manual in chapter 15.

3.1.5 Hardware configuration

The hardware used in the system is automatically recognized by the Smart Function Kit. In this step of commissioning, the recognized components and system parameters are listed.

ATTENTION	
	**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.

→ Click on *Next* to continue to the next commissioning step.

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

→ To do so, click on *Extended configuration*.

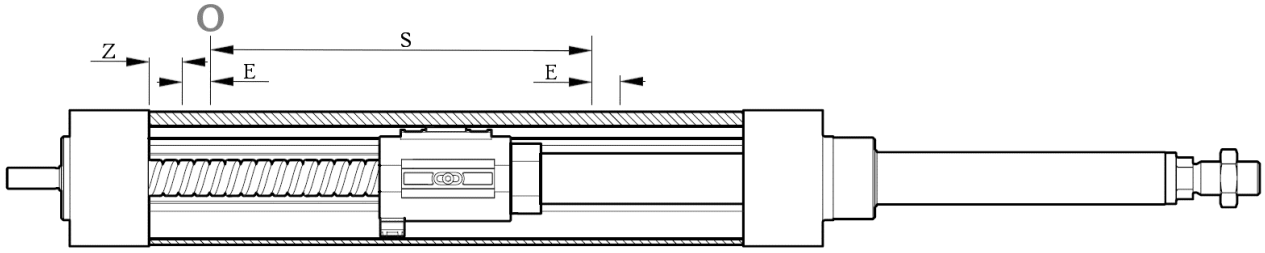
◊ For further information on *Extended configuration,* refer to the chapter System settings.

3.1.6 Homing Axis

→ In the last step of commissioning, you can optionally carry out a reference run to set the zero point of the system.

→ Alternatively, finalize the commissioning and carry out the reference run later in the system settings.

⚠ WARNING	
	**Risk due to moving parts!** Moving parts can cause severe lacerations, stabbing injuries and crushing. → Never reach into moving parts.
	**Risk due to moving axes!** Moving axes can cause damage to the product and the system. → Enable faultless retraction of the cylinder.



2 Positions of the Software Limits

- The fixed stop Z is at -3 mm from the zero point (0).
- The rear/top software limit switch is at -1 mm from the zero point (0).
- The maximum usable travel range of the system is S and corresponds to the dimension SMAX of the ordered configuration.

- The front/bottom software limit switch is at S+1 mm from the zero point (0).

→ Click on *Finish* to finish commissioning directly without any reference run.

✅ During the reference run, the cylinder moves in up to the motor-side fixed stop.

✅ The position of the fixed stop is used to determine the system zero point and the respective software limit switch.

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

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

→ Enable faultless retraction of the cylinder.

→ If system errors are displayed in the title bar, acknowledge them first.

◊ For detailed information, please see the chapter Troubleshooting and fault clearance at the end of this instruction manual.

→ Click on *Homing Axis* to start the reference run.

3.2 Recommissioning after standstill

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

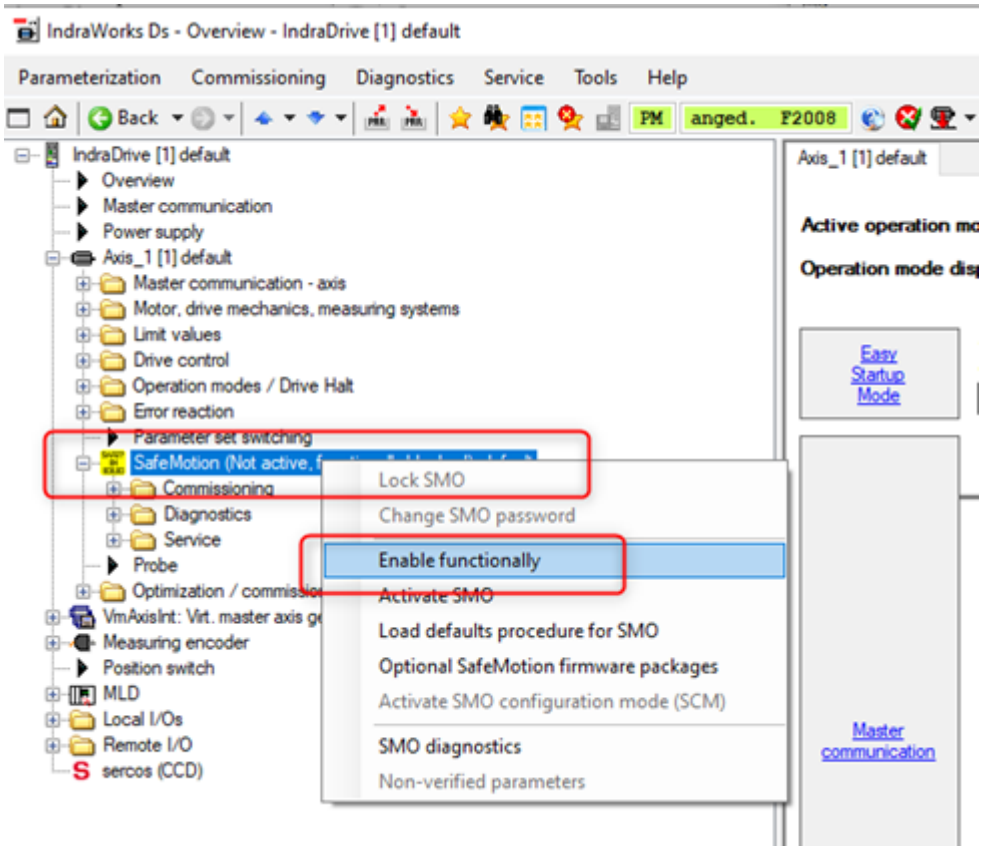
3.3 Commissioning of the safety technology with safety option S4

The following chapter shows you the relevant procedure for the commissioning of a Smart Function Kit with safety option S4. 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.

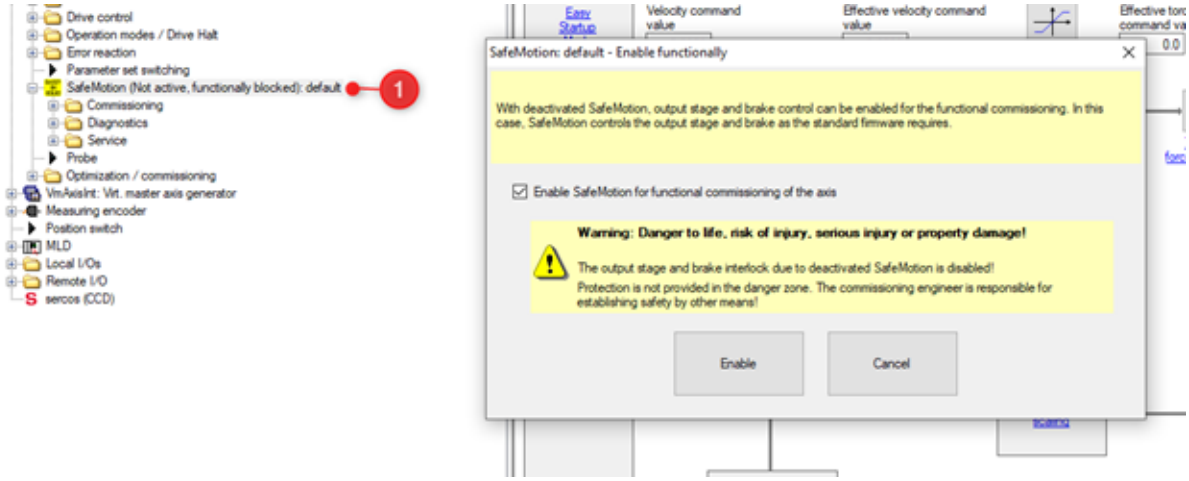
◊ For the commissioning of the safety technology in a drive controller HCS01 with safety option S4, you also need the IndraWorks DS software. It can be downloaded from the Bosch Rexroth homepage following this link: https://www.boschrexroth.com/de/de/produkte/produktgruppen/elektrische-antriebe-und-steuerungen/engineering/software-tools/indraworks-engineering .

Procedure for commissioning:

1. Open the IndraWorks DS software on your engineering device (e.g. laptop) first and establish a connection to the drive controller 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” functionality in IndraWorks and activate the functional commissioning. These steps are necessary to start the commissioning of the safety technology.



3 Activation of the functional commissioning (1)

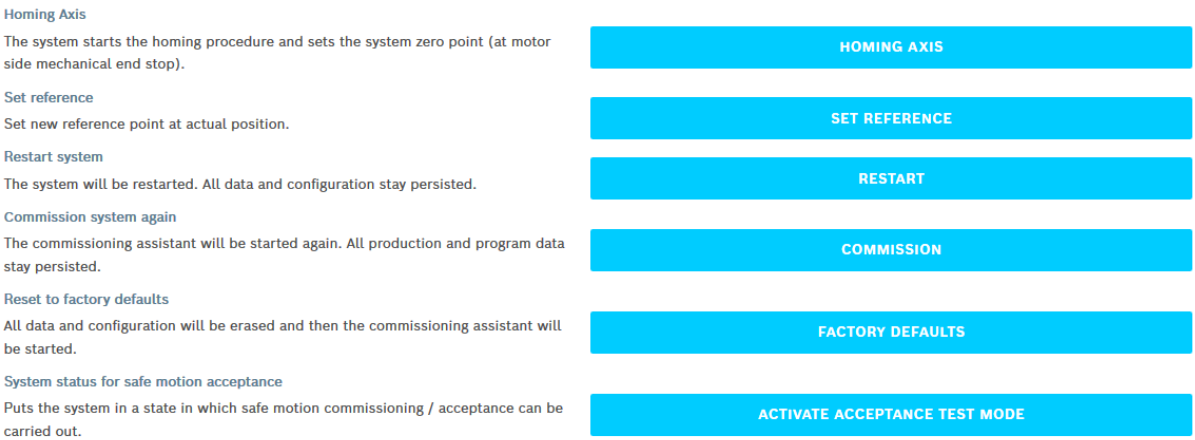


4 Activation of the functional commissioning (2)

3. As long as the “Safe Motion” functionality is deactivated, the connection between the drive controller HCS01 and the safety zone module HZS01 should be disconnected. If you have already pre-wired the drive controller HCS01 with the safety zone module HSZ01 via the safety bus (X42 and X43 at the safety zone module), separate this connection again. If necessary, screw off the motor encoder cable at the drive controller HCS01 (X4) to be able to operate the safety lug at the bus cable so that you can remove the bus cables. Afterwards, re-connect the motor encoder cable in the drive.

> **Notice**
> The zone bus (X42, X43) is only to be wired immediately before activation of “Safe Motion”. Otherwise, commissioning of the Smart Function Kit will not be possible, error “F7035 Zone bus error” will be permanently pending.

4. Release the intermediate circuit voltage at the drive controller HCS01.
5. Afterwards, carry out the commissioning of the Smart Function Kit according to chapter 3. At the end of the commissioning, you can carry out the reference run, which is required for the further steps. Otherwise, you can neither approach absolute positions nor start programs.
6. In the Smart Function Kit software, activate the acceptance condition. This is necessary in order to be able to use the drive-integrated oscilloscope for the machine acceptance. The setting necessary for this purpose can be found at Configuration → General → Activate acceptance condition.



5 Activation of the acceptance condition

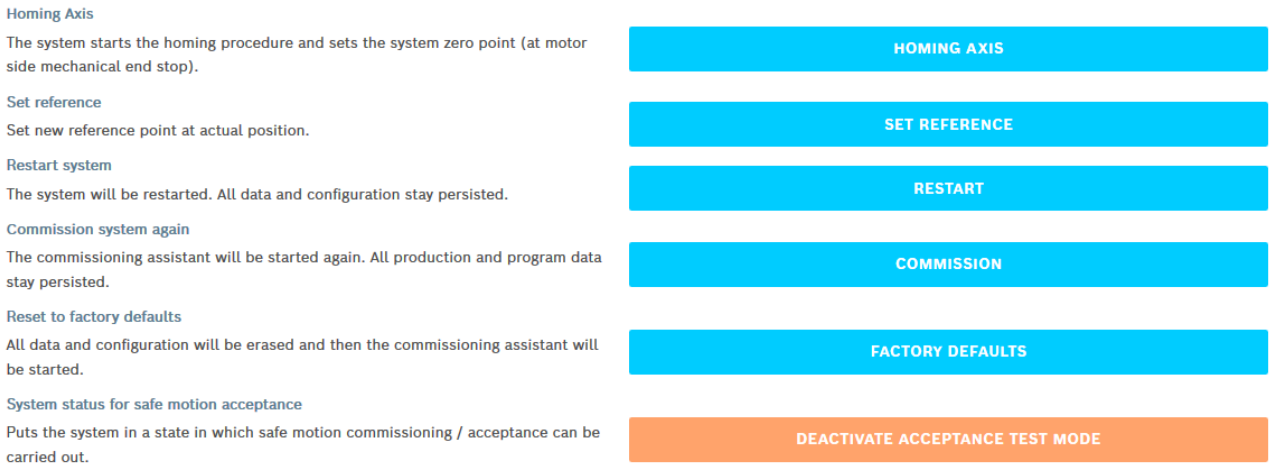
7. Switch off the drive controller HCS01 and use the bus cables to connect it to the safety zone module HSZ01 according to the assembly instructions (R320103194). Afterwards, start up the system again.
8. Now start up the “Safe Motion” safety technology with the IndraWorks DS software.

◊ For detailed information on commissioning the safety technology, please see the “IndraDrive Integrated safety technology “Safe Motion” (from MPx-18)” instructions (R911338919) in chapter 9. This can be found in the Media directory on the Bosch Rexroth homepage.

After successful commissioning of the safety technology, machine acceptance testing can be performed. For your acceptance tests, you can create programs in the Smart Function Kit software to carry out the desired axis movements with the required dynamics. At the same time, you can record the desired parameters, such as the actual speed, using the oscilloscope in the IndraWorks DS software. For more details 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 condition in the Smart Function Kit software must imperatively be activated.

As soon as you have commissioned the system and it is ready for productive operation, deactivate the acceptance condition that has been activated in step 6. This is necessary to activate the data recording during a program run.



6 Deactivation of the acceptance condition

> **Notice**
> If the error message "F8304 SMO: System Error" appears, please contact customer service.

3.4 Force sensor

For reliable force measurement and control, the Smart Function Kit requires a force sensor. Please carry out mounting and connection before commissioning.

◊ For detailed information on the wiring of peripheral devices with the Smart Function Kit, please see the assembly instructions (R320103194) and the exemplary circuit diagram. Both documents can be found in the Media directory on the Bosch Rexroth homepage.

3.4.1 Included force sensor

If the included force sensor is used, no further configuration is required in the system. All parameters are automatically configured in commissioning. Thanks to calibration at the factory, the force sensor does not require any individual adjustment of parameters.

For clear identification, every force sensor is marked with a serial number. For documentation, the sensor characteristics are also noted.

> **Notice**
> A 3.1 Acceptance test certificate can be requested for each force sensor for verification of calibration at the factory. Please contact your sales partner (see Help) and state the serial number of the force sensor.

> **Notice**
> It is recommended to have the tension/pressure sensor regularly recalibrated by the manufacturer at intervals of approx. 24 months (DAKKS calibration certificate). If necessary, the basic settings are corrected. Please contact your sales partner (see Help) and state the serial number of the force sensor.

3.4.2 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. Open the extended configuration (see the chapter Hardware configuration for a description of the process for changing parameters).
3. Activate the force sensor by setting of:
   Y1049 "Analog force sensor active", value: 1
4. Set the lead characteristics of the force sensor in [N/V]. These values should be specified in the data sheet of the force sensor or any used signal conditioners.
   💡 The lead characteristics in the tension and pressure range are typically identical.
   a. Tension
      Y1050 "Force sensor lead characteristic (tension)", value in [N/V]
   b. Pressure
      Y1053 "Force sensor lead characteristic (pressure)", value in [N/V]
5. Set the zero passage in [V]. This value should also be specified in the data sheet of the force sensor or any used signal conditioners.
   💡 The zero passage corresponds to the output voltage in non-loaded condition.
   Y1054 "Limit of analog value characteristic", value in [V].
6. Save the extended configuration. Changes are applied immediately.
7. Carry out taring via the control bar (see Taring the force sensor:).

Notice
Only force sensors with an output signal of 0-10 V can be used.

ATTENTION	
	**Incorrect configuration can lead to incorrect evaluation of the sensor.** Incorrect force measurement. Possible damage to the workpiece. → Check the entered parameters for plausibility.

3.4.3 Verification

The measured force can be verified by means of machine capability examination (MCE) or in other situations for determination of deviations for comparison measurement. This check can be carried out after commissioning.

For successful measurement, please ensure the following:

- External force sensors have received valid calibration and are operated in compliance with the applicable instructions.
- Suitable positioning of external force sensors without tilting to the cylinder.
- Force sensor is installed and wired according to assembly instructions (R320103194) and example circuit diagram.

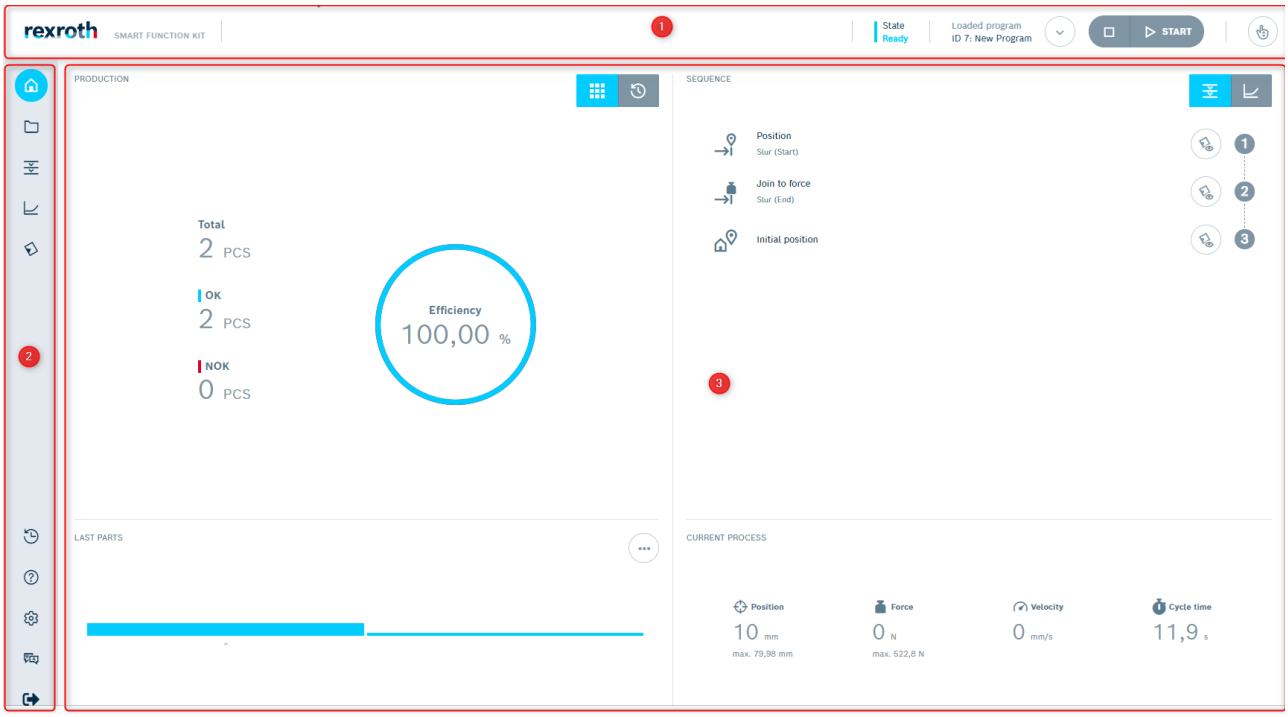
Proceed as follows:

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
If a fixed deviation is indicated over the entire measuring range by the external force sensors, this can be compensated by taring (see the chapter Taring the force sensor and the Library: "Tare" module).
If the deviation depends on the force, check the measurement setup. Linear deviations indicate an impact of the measurement setup.

4 Operation

The Smart Function Kit is operated via a modern and intuitive interface that can be displayed via a web browser. The interface consists of three main areas:

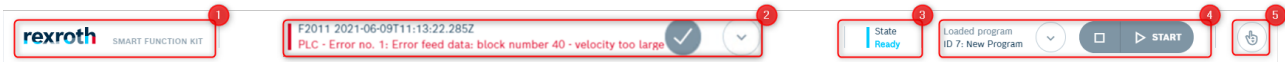


7 User interface overview of Smart Function Kit

1. Title bar: Display of status information
2. Navigation bar: Navigation through the Smart Function Kit interface
3. Content area: Display of the currently active navigation area

4.1 Title bar

The title bar displays the system status at a glance and provides the possibility of changing the program at any time and manually starting and stopping the program.



8 Title bar

Position	Designation	Function
1	Icon and title	Bosch Rexroth logo with "Smart Function Kit" wording. → Click the *logo* to open the full-screen mode.

Position	Designation	Function
2	Messages and errors	System messages and errors are displayed in the middle section of the title bar. If several errors are pending, the error with the highest priority is displayed. → Click the *Acknowledge* ( ) button to acknowledge (delete) the error messages. Doing so only acknowledges the last error. → Repeatedly click the *Acknowledge* ( ) button to acknowledge (delete) several pending error messages. → Click the *Info* ( ) button to change to the diagnostics area. In the diagnostics area, more information on the errors is displayed.
3	System status	Displays whether or not the system is ready for production. The following status are differentiated: **Ready**: The system is ready for production **Error:** An error is pending. Production is not possible **Parameter mode:** The system is in parameter mode, e.g. in order to write system parameters. Switching between parameter and operating mode is performed automatically. **No power**: The intermediate circuit voltage at the drive controller has not yet been switched on. **No Reference**: The reference point of the system has not been set.
4	Program selection	The active program can be selected in the program selection. → Click the arrow to open the selection control element. A list of all programs created and saved in the program management is displayed. → Select the desired program and click the *Start* button. The selected program is started and performed once. **Notice** The button can only be actuated if the system is not performing any other action. → Click the *Stop* button to stop all system movements. As a consequence, the currently running program is aborted and the produced part is rated NOK.
5	Control bar	→ Click the *Control bar* button to show or hide the control bar for manual operation of the system. For more information on the control bar, see the chapter Control bar.

3 Title bar

4.2 Content area

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

In the content areas Programming, Reference curves, and 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.

This selection element can also be used to quickly

4.3 Full-screen mode

The operator interface of the Smart Function Kit can be operated in full-screen mode. In this mode the navigation elements of the browser are hidden and the entire screen is used for display of the interface. Especially with small screens or mobile devices, this is very helpful to effectively use the entire screen size and make navigation easier.

→ Click the Rexroth logo in the title bar to activate the full-screen mode.

✔ Full-screen mode is activated.

→ Click the logo again to additionally adjust the width of the navigation bar.

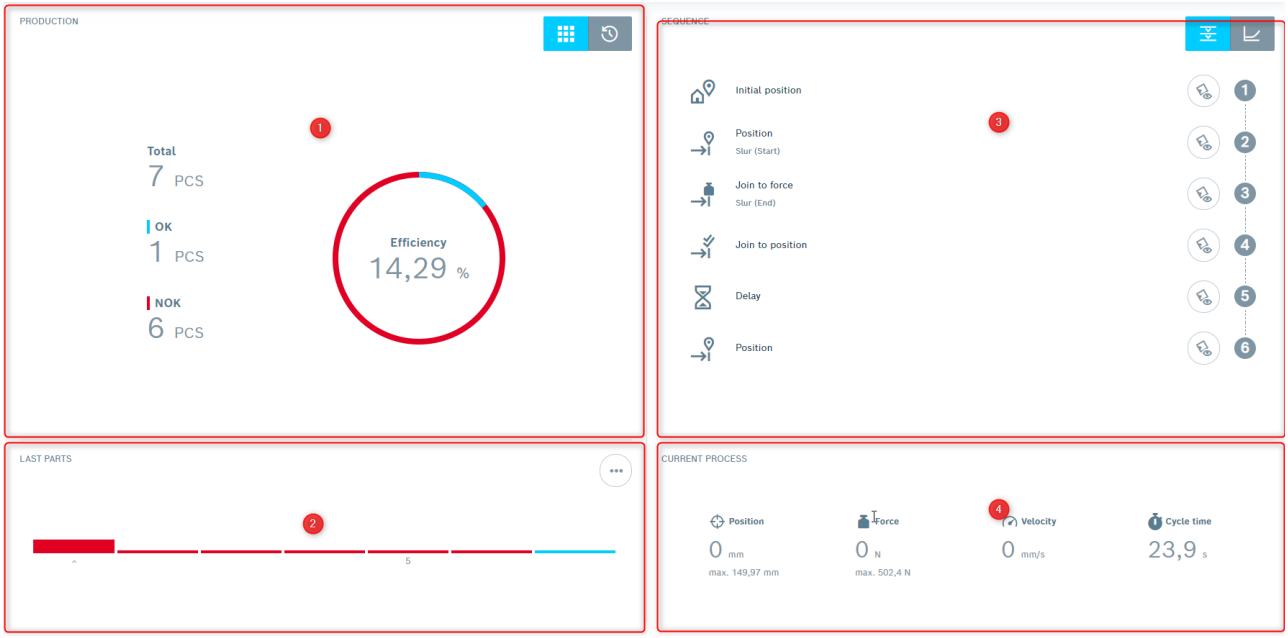
→ To deactivate, follow the instructions of the respective Internet browser and/or the operating instructions of the device.

5 Dashboard

The dashboard provides an overview of the system and production status. For this purpose it provides information on production statistics, quality evaluation of the parts produced last, on the current program progress and the current machine values, distributed to four display tiles.

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 status bar. As soon as programs are switched in the status bar, the contents of the dashboard are changed accordingly.



9 Dashboard

1 Production statistics

2 Last parts

3 Sequence

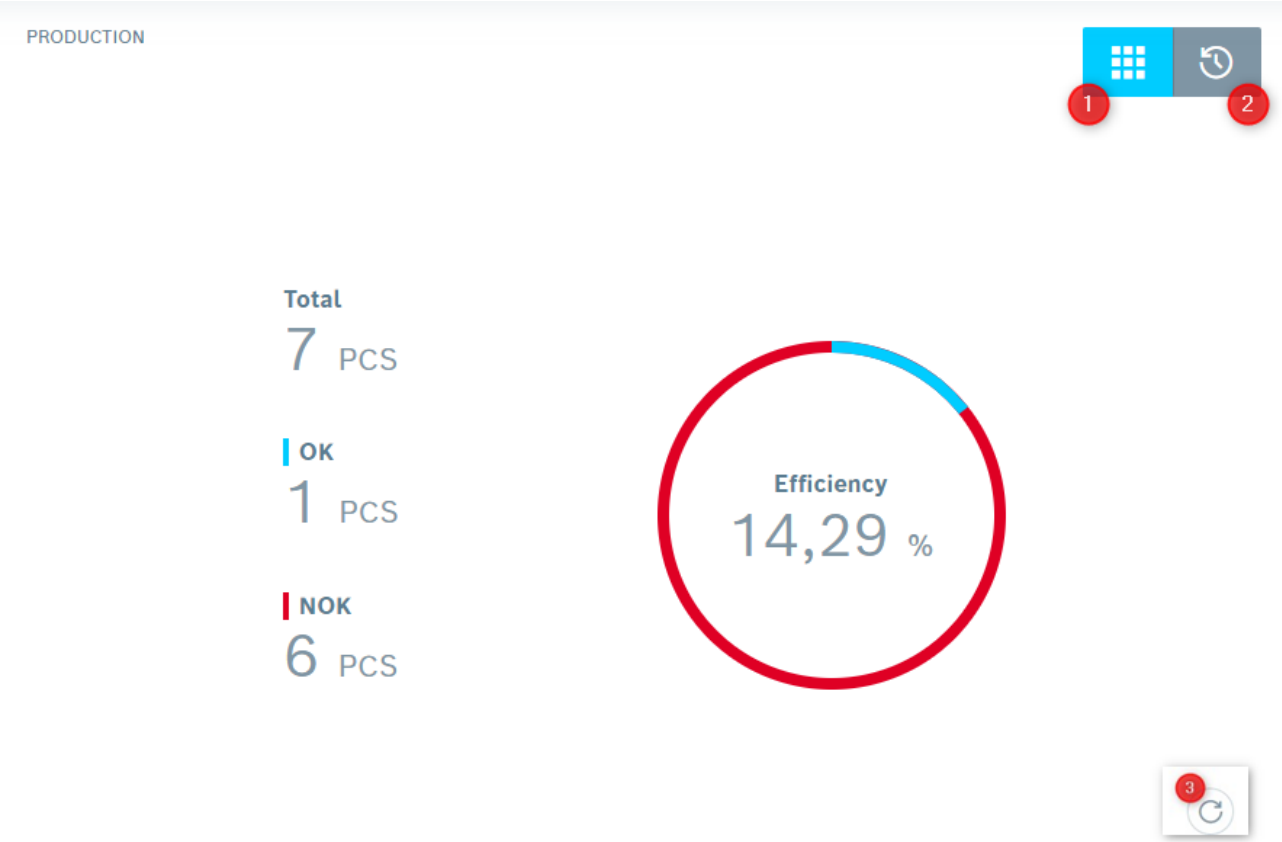
4 Current process

5.1 Production statistics

→ Click the clock icon (2) to display the number of parts produced during the current shift.

The production data recorded since the last reset are displayed, here “Since 06/11/2021 12:25”.

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



10 Production statistics

→ Click the statistics icon (1)

✓ The total number of parts produced with the currently selected program is displayed.

✓ In addition, it is also displayed how many of these parts have been rated OK and how many NOK.

The Efficiency element graphically shows the proportion of OK and NOK parts and thus

ensures a quick overview of the production quality.

The values of the production statistics are individually recorded for every program and not reset upon change of program.

> **Notice**
> A part is rated a produced part as soon as the production sequence has been started via the title bar or external actuation.
> Recording of reference curves does not count as produced parts.

→ Click the clock icon (2) to display the number of parts produced during the current shift.

✓ The production data recorded since the last reset are displayed, here “Since 06/11/2021 12:25”.

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

5.2 Last parts

11 Last parts

The Last parts tile shows the quality evaluation of the last 20 parts produced with the currently selected program in graphic form.

→ Click the ... (1) button to change to the *Last parts* area and view all production data of the current program.

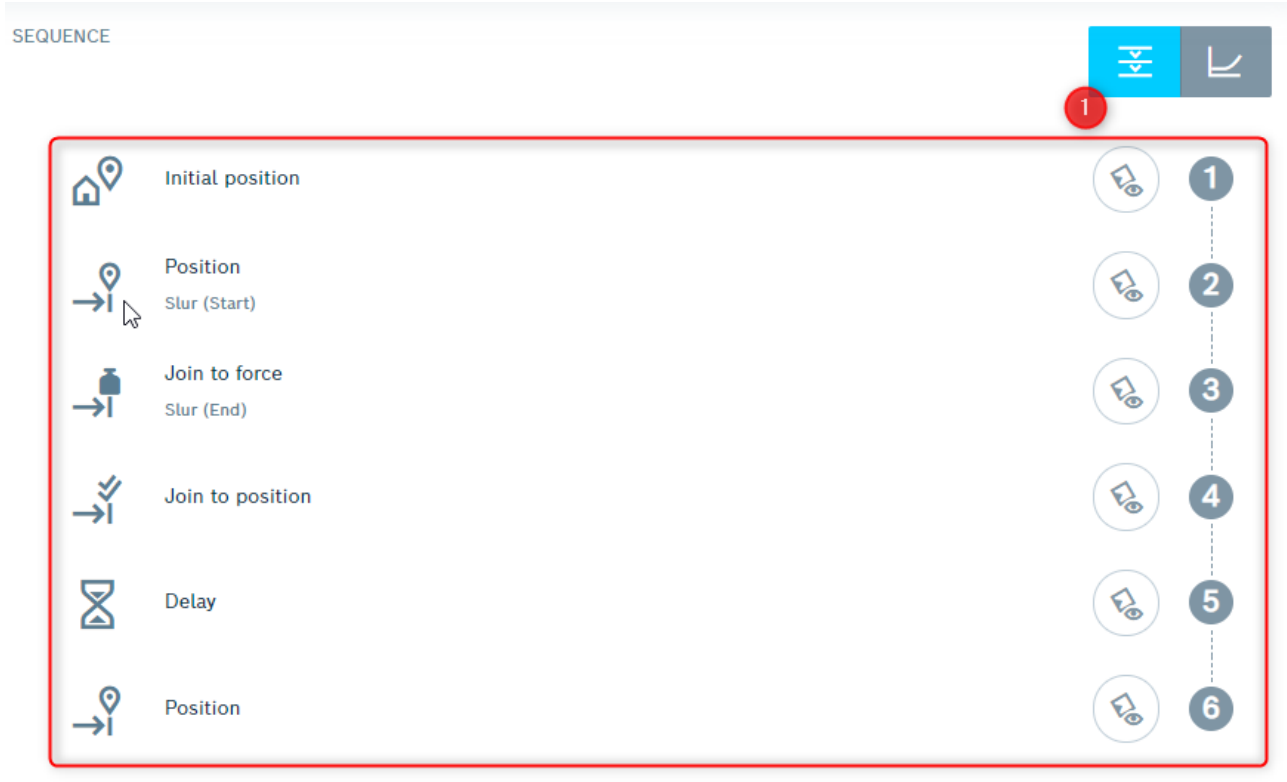
→ Click on one of the *Evaluations* (2) to show detailed information on the individual parts produced within the *Last parts* area

5.3 Sequence

As a standard, the sequence tile shows the various steps the currently selected program consists of. In this view the operator can check whether the loaded program is the required production program.

Sequence can be displayed either in the form of a force-displacement curve or as a sequence of steps.

Sequence of steps



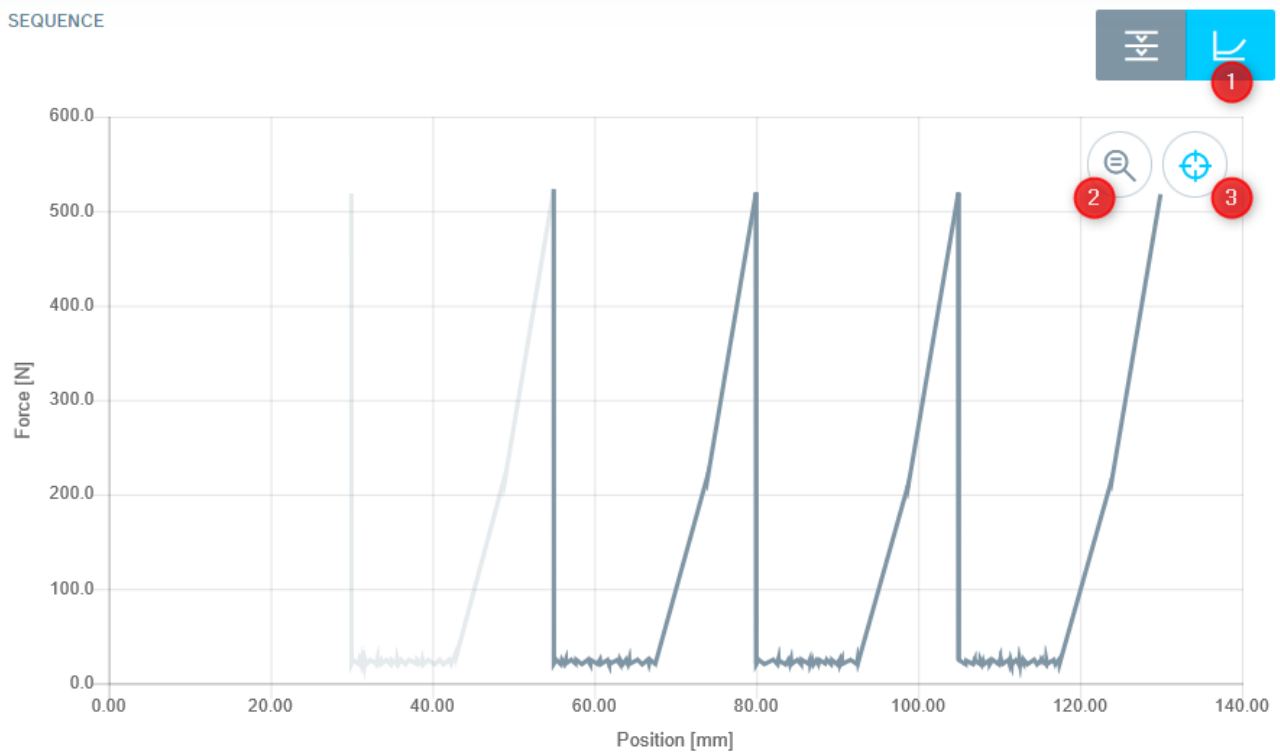
12 Sequence of steps

→ Click the *Program* (1) button to have the production program with its 5 steps displayed.

✅ If a part is currently being produced, the currently active step is highlighted.

✅ If an error occurs during the program and the system needs to stop, this makes it possible to determine in which step the deviation has occurred.

Force-displacement curve



13 Force-displacement curve

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

✅ If a program is currently running, the display shows the live build-up of the force-displacement curve, the status of the evaluation elements and the quality assessment resulting for the respective part.

> **Notice**
> Since data for this display are not processed in real time, the representation may slightly deviate from the exact actual values.
> If no program is currently running, the last recorded curve including quality assessment is displayed.

→ With your mouse, select the desired curve and scroll the mouse wheel to zoom in or out of the curve.

✅ The view thus provides the option of performing an initial analysis of the force-displacement curve and quality assessment.

→ Click the *Magnifying glass* (2) to reset the zoom.

→ The *Position symbol* (3) indicates that the position is selected as X-axis. Click on the switching element to switch to the time area. Click again to switch to the position view.

> **Notice**
> In the time area, creation and display of validation elements are not possible.

5.4 Current process

14 Process data tile

The *Process data* tile displays the current values for position, force, speed, and cycle time. While a program is running, the values are updated live.

💡 Display of the force is not possible in real time, and the values are filtered for better representation. Therefore, there may be deviations between the displayed and the real force value.

The maximum values during the process are displayed below the current actual values. These allow for a fast check of the maximum force or position reached during production. If a program is currently running, the maximum values are updated live according to the progress. If no program is currently running, the maximum values of the last part produced with this program are displayed.

6 Control bar

→ In the title bar click the *Index finger* button .

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

15 Control bar

The control bar has the following functions:

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

6.1 Taring the force sensor

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

16 Taring the force sensor

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

→ If required, adjust the value upwards or downwards.

→ Click the *Tare (2)* button.

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

✓ The value is intended as reference point for all future sensor values.

✓ The value is saved and is maintained until taring is performed again.

Notice
Taring applies globally to all force values and thus also for
e.g. the force values in the program steps, curve recording, and quality assessment.

6.2 Manual movement

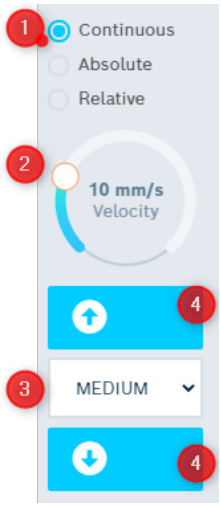
Three modes for manual movement are available via the control bar.

- Continuous
- Absolute
- Relative

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

For all modes, the travel speed can be set via a control element.

6.2.1 Continuous



17 Manual movement - Continuous

→ Select the *Continuous* (1) mode.

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

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

- *High:* Interruption of communication results in immediate system stop.
- *Low*: The system is only stopped if the interruption persists for a longer period of time.

→ Press one of the two *Arrow buttons (up/down) (4)* and keep it pressed.

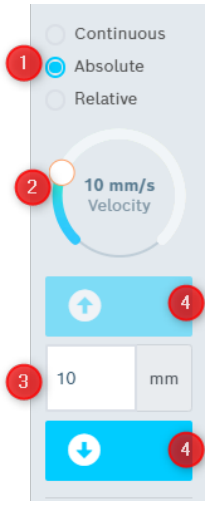
✓ The Smart Function Kit will move for as long as the arrow button is kept pressed.

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 communication is interrupted and the system is stopped for safety reasons, a movement command must be restarted via the arrow button.

ATTENTION	
	**High lag of the system if sensitivity set to low** High lag of the system with sensitivity set to low may result in damage to the system. → Only use the *low sensitivity* setting at low velocities.

6.2.2 Absolute

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



18 Manual movement - Absolute

→ Select the *Absolute* (1) mode.

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

→ Enter the desired 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.

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

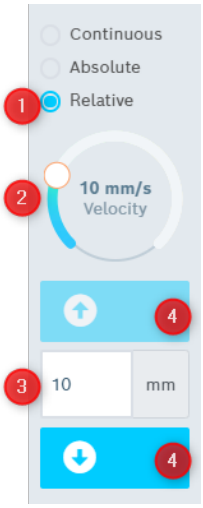
→ Adjust the value in the input field (3) according to the note.

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

✓ The Smart Function Kit 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 step width and speed.



19 Manual movement - Relative

→ Select the *Relative* (1) mode.

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

→ Enter a defined step width (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 step width (3) is traveled at the defined speed (2).

6.3 Display of current values

20 Display of current values

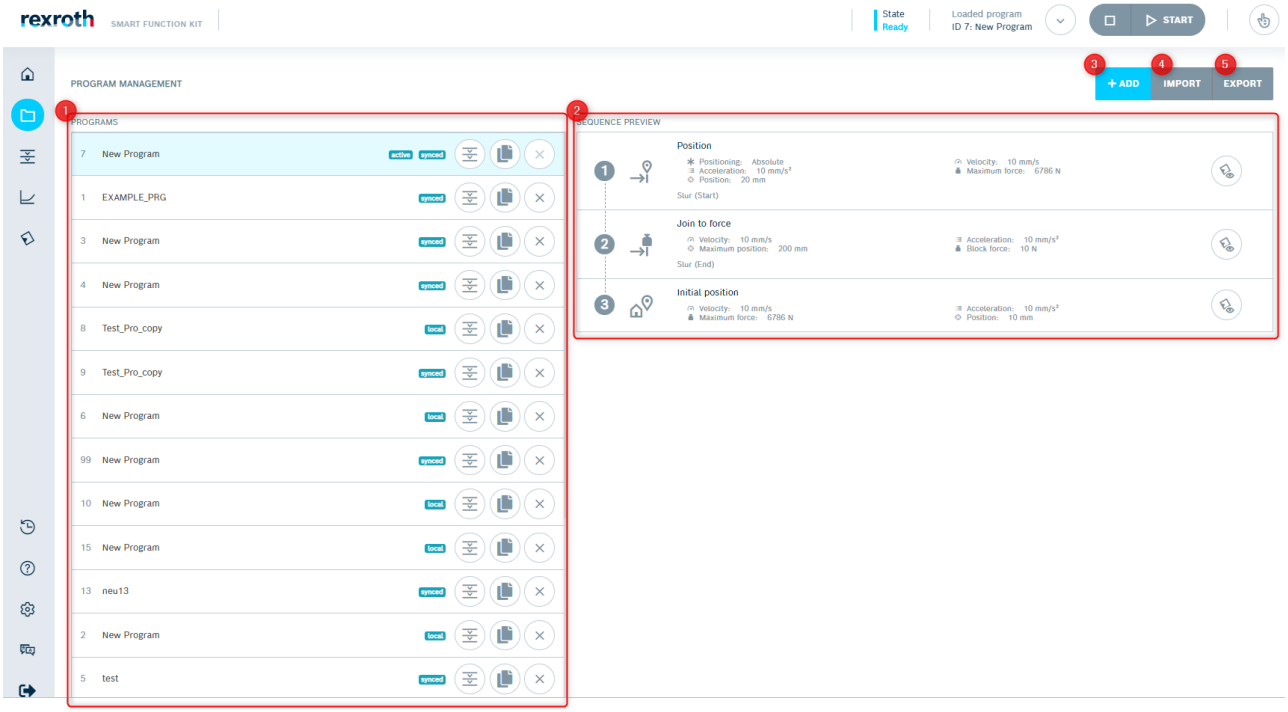
The current values are displayed in the lower area of the control bar:

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

Display of the force is not possible in real time, and the values are filtered for better representation. Therefore, there may be deviations between the displayed and the real force value.

7 Program management

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



21 Program management

Position	Designation	Function
1	Program list	List of all programs, including program name and program field bus ID. **Editing a program** → Click the *Edit* button. ✅ The view changes automatically to the *Edit program* area. 💡 Changes can be made to the program. The program sequence with all detailed parameters as well as the program name and the program field bus ID can be changed. **Copying a program** → Click the *Copy* button. ✅ The view changes automatically to the *Edit program* area. ✅ The program, including reference curves and 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 field bus 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. All entries in the history, including the appertaining production data, are maintained. Deleting a program from the program list therefore does not result in loss of production data.
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.
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 field bus ID is the next free field bus ID. → Manually change the two values in the *Edit program* area.

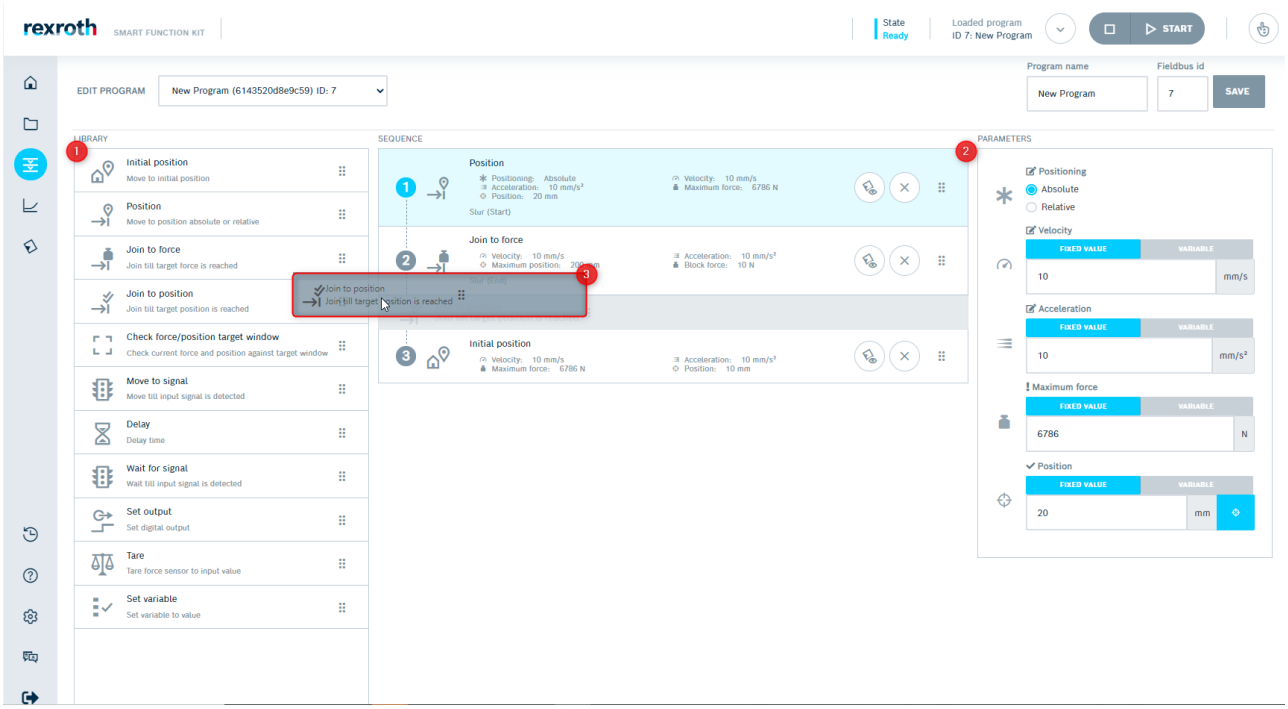
Position	Designation	Function
4	Importing a program	→ Click the *Import* button. → In the *File selection dialog* navigate to the storage location of previously exported programs (from this or any other system) and select the desired program. → Click *open*. ✅ The program is imported. ✅ During import, the system automatically changes 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 limits are complied with. If individual settings are not compatible, they can be manually changed before saving.
5	Exporting a program	→ Click the *Export* button. ✅ The selected program is exported to the local memory of the HMI device. 💡 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.

4 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.

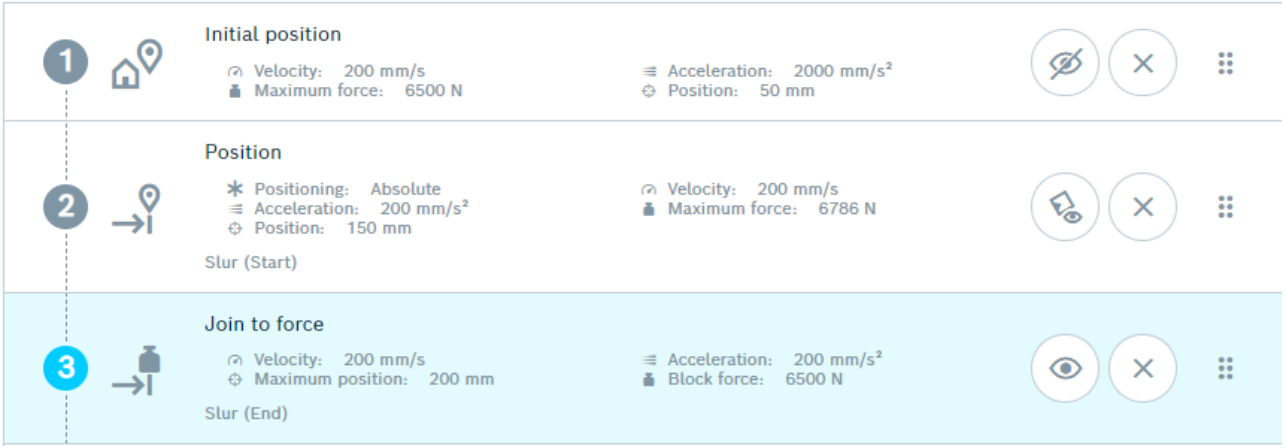


22 Programming

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

8.1 Module properties

The Smart Function Kit records the corresponding force-displacement curve for the program sequence. It can be set for every module whether or not it 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.



23 Module properties

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

1. **Invisible**: Curve section is not shown and not evaluated

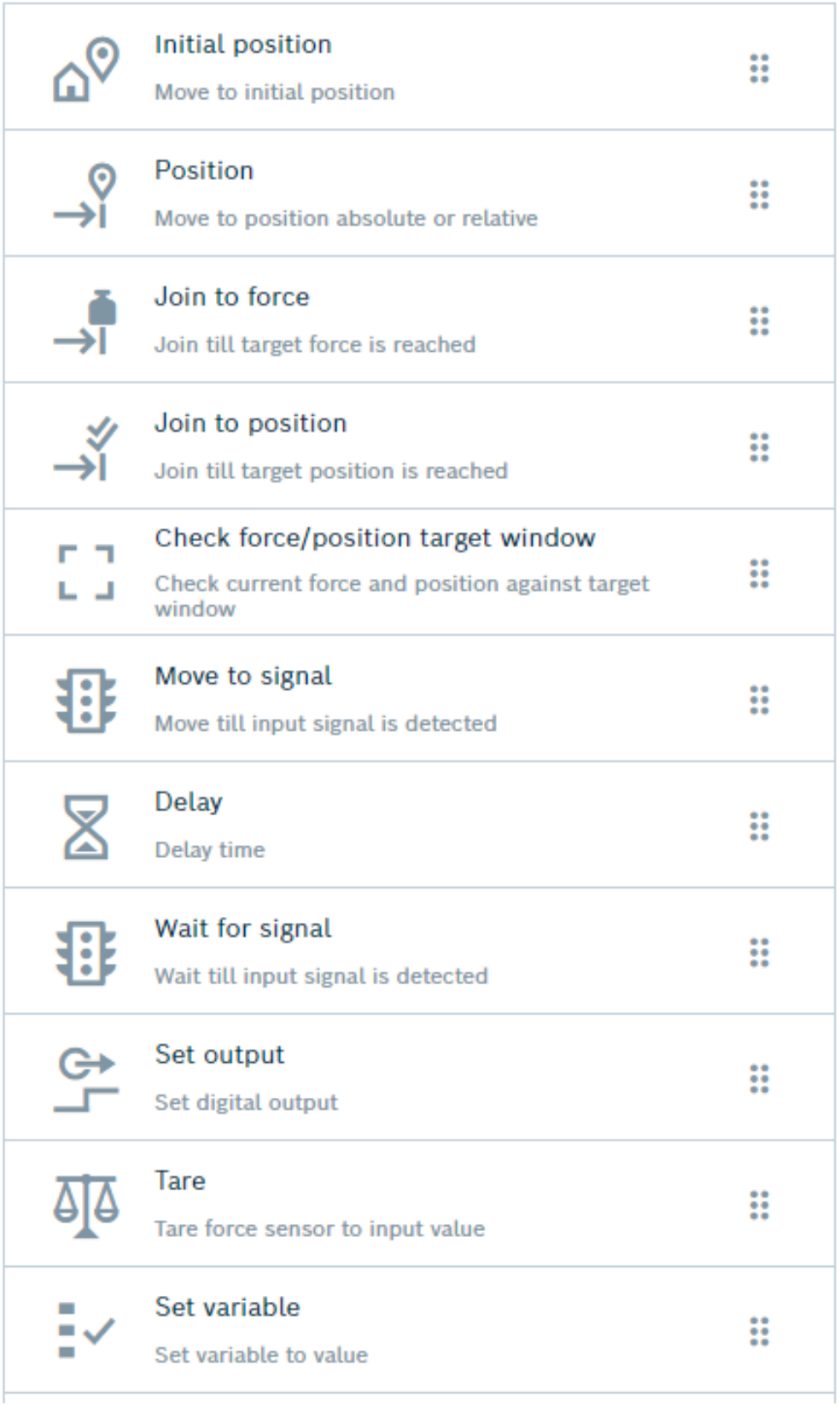
2. **Validation:** Curve section is shown and evaluated
3. **Visible but not validated:** Curve section is shown but not evaluated

> **Notice**
> 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 to allow for visual differentiation.

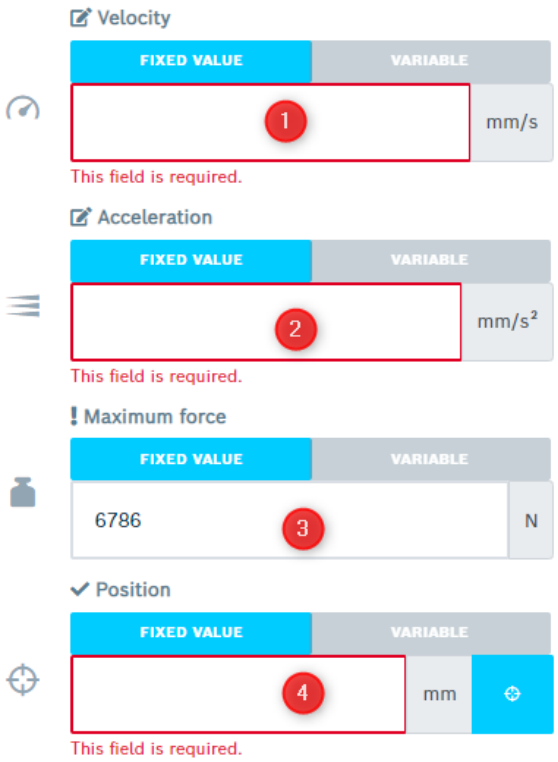
8.2 Library

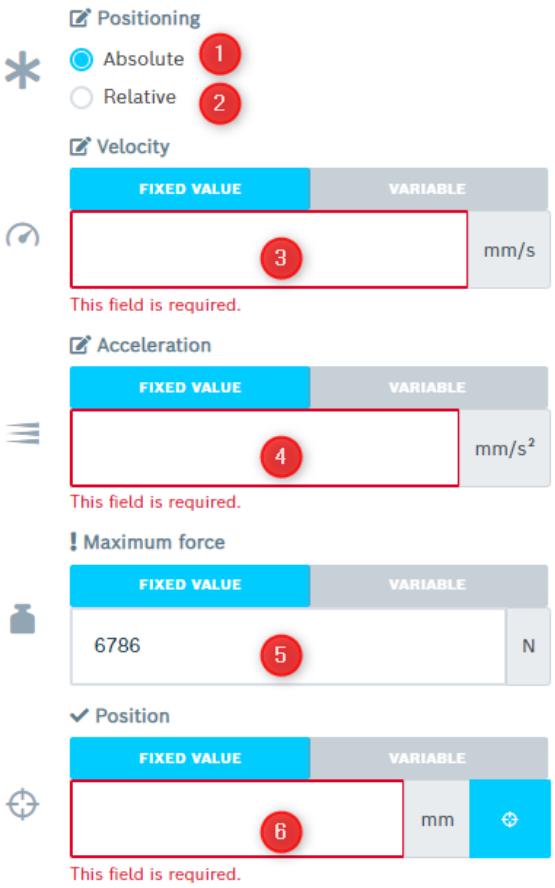
On the left-hand side the system provides a library with prepared modules. You can integrate the required modules from this library into the sequence by dragging and dropping or by clicking on them.

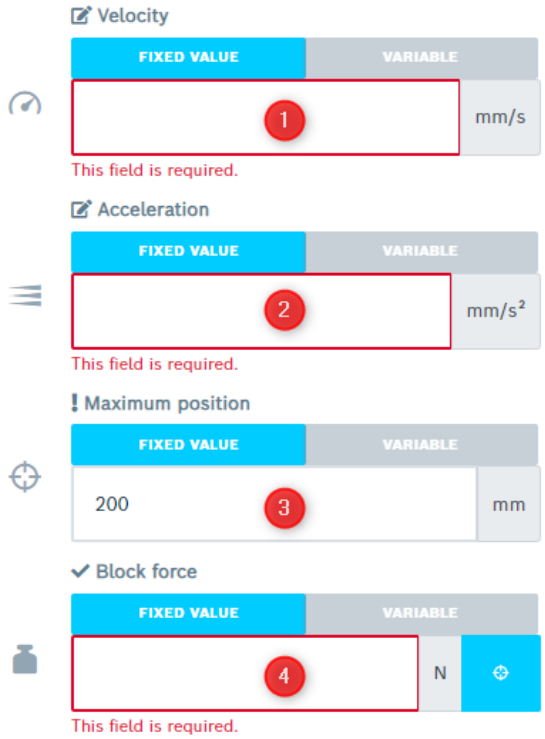
The predefined modules are designed in such a way that they can take on a specific task in the sequence.

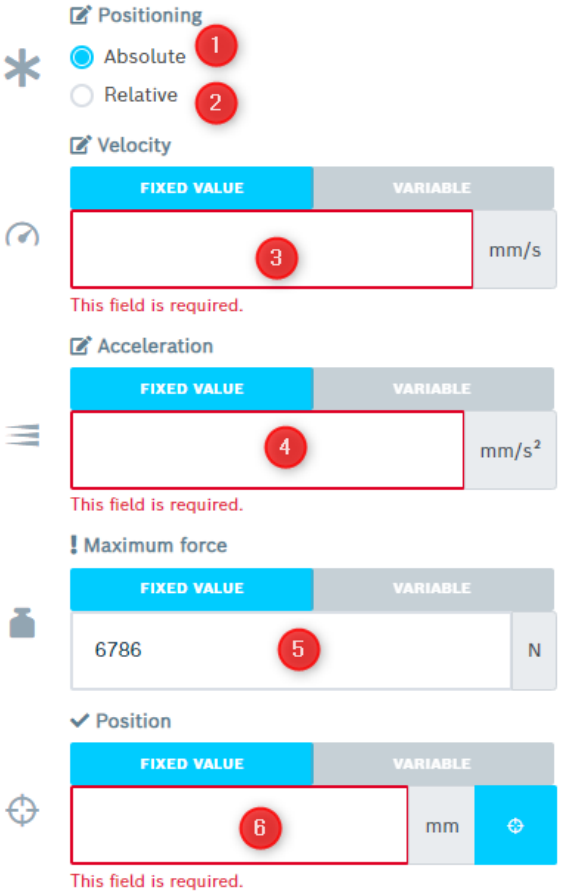

Initial position
Move to initial position
Position
Move to position absolute or relative
Join to force
Join till target force is reached
Join to position
Join till target position is reached
Check force/position target window
Check current force and position against target window
Move to signal
Move till input signal is detected
Delay
Delay time
Wait for signal
Wait till input signal is detected
Set output
Set digital output
Tare
Tare force sensor to input value
Set variable
Set variable to value

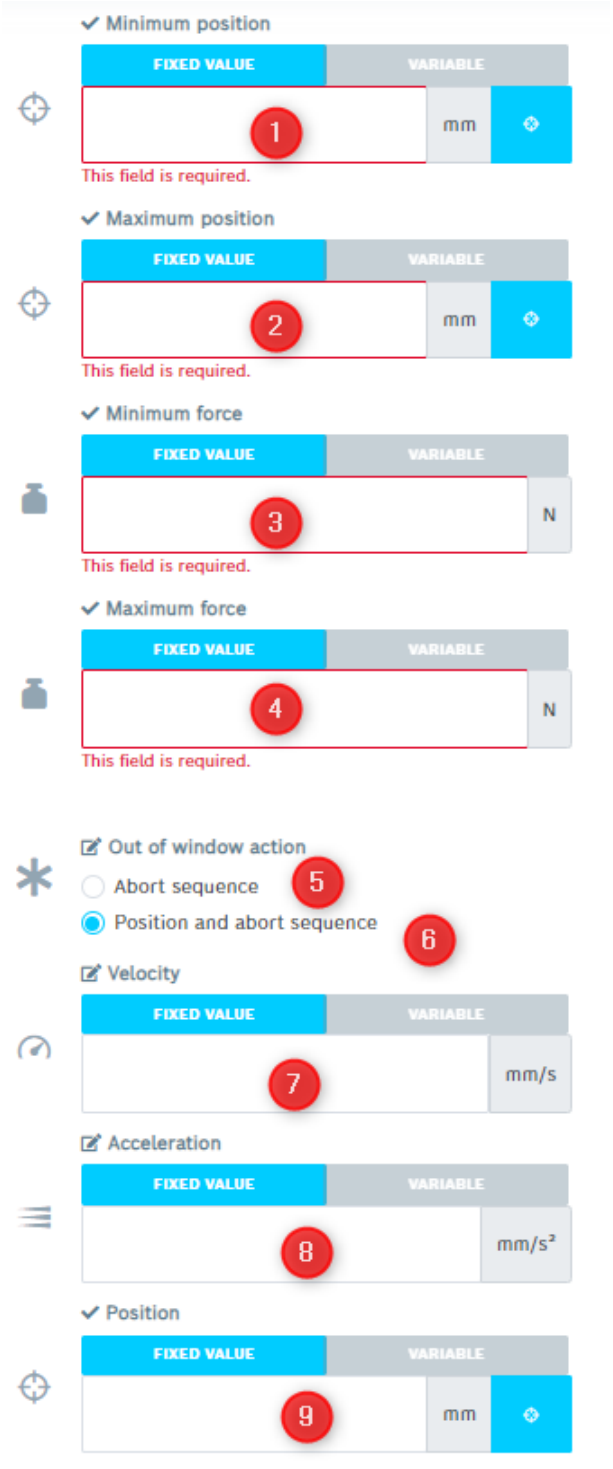
24 Library

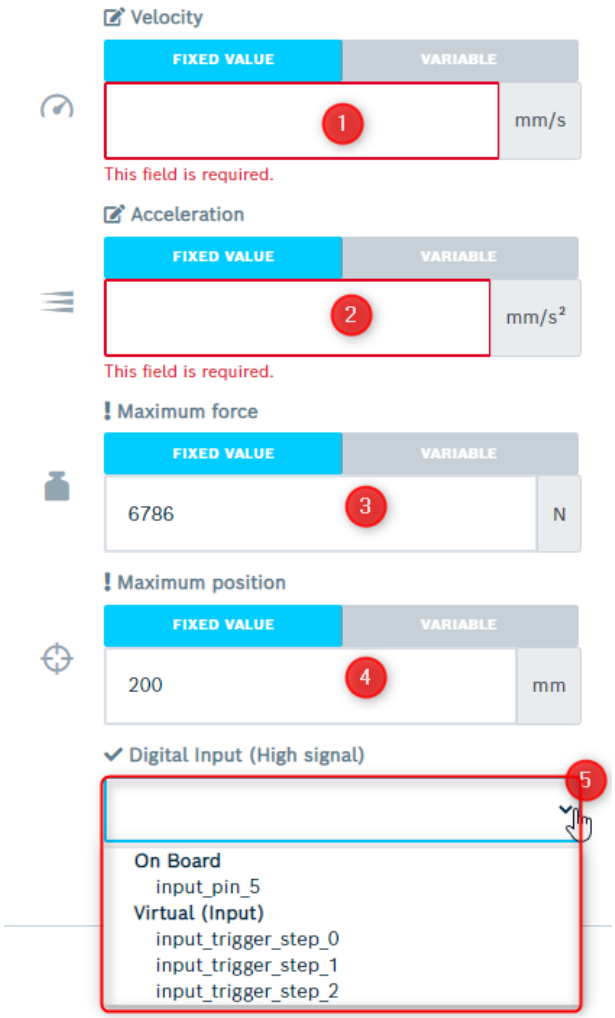
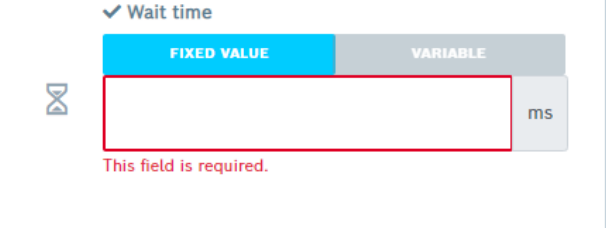
Screens / comments	Action steps
	**Initial position** The *Initial position* command is used to move the system to a defined initial position. This position is the starting point of the joining sequence. Once the position is reached, the system switches to the next module in the program sequence. → Individually specify the speed (1) and acceleration (2) with which the system is to move. ✓ The system does not check whether or not the set speed can be reached with the selected acceleration. → Specify the maximum force (3). ✓ If no force is specified, the maximum force of the system is automatically used. → If required, enter the position (5) in mm. ✓ While approaching the target position (4), the current force is monitored by the force sensor. ✓ If the actual force exceeds the specified maximum value, the system is stopped immediately to prevent damage to the system or the tools. ✓ The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. → In the respective input fields (1-4), alternatively click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the respective *fixed value*.

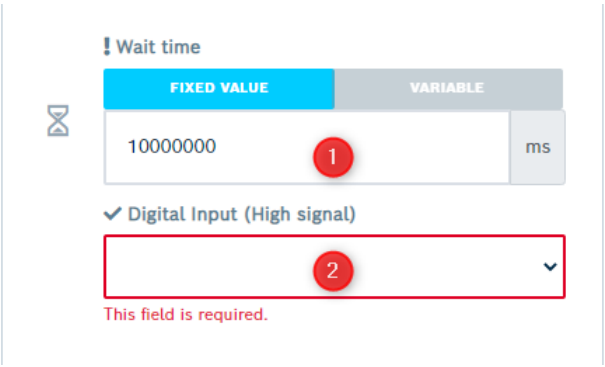
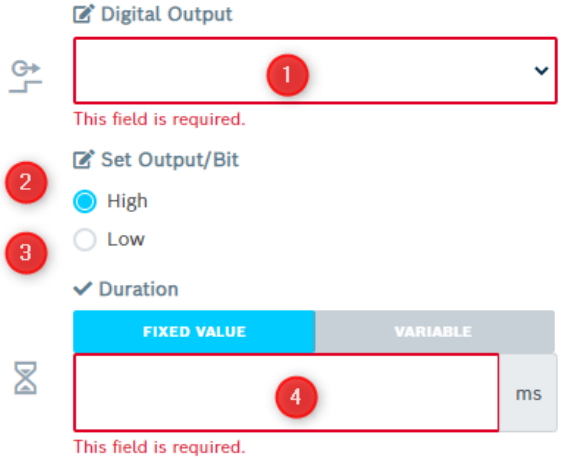
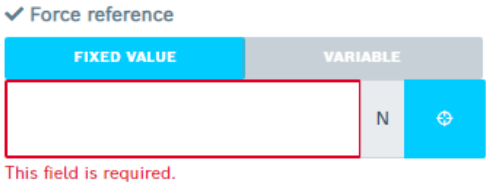
Screens / comments	Action steps
	**Positioning** The *Positioning* command causes the system to move to a defined position. → Select absolute (1) or relative (2) approach of the position. With absolute positioning, the target position is directly approached. With relative positioning, the specified distance is traveled, starting at the current position. Once the position is reached, the system switches to the next module in the program sequence. → Individually specify the speed (3) and acceleration (4) with which the system is to move. The system does not check whether or not the set speed can be reached with the selected acceleration. → Specify the maximum force (5). → 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. If the actual force exceeds the specified maximum value, the system is stopped immediately to prevent damage to the system or the tools. The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. → In the respective input fields (2-6), alternatively click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the respective *fixed value*.

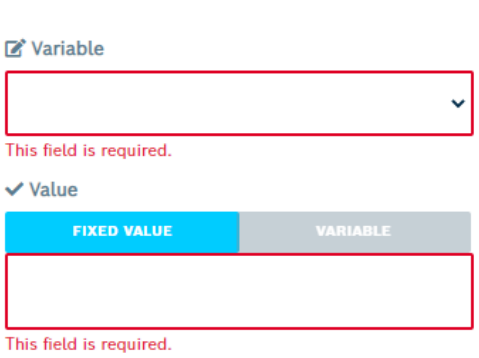
Screens / comments	Action steps
	**Join to force** The command *Join to force* causes the system to move until the defined target force is reached at the force sensor. The system will then switch to the next module in the program sequence. → Individually specify the speed (1) and acceleration (2) with which the system is to move. ✓ The system does not check whether or not the set speed can be reached with the selected acceleration. → If required, enter the maximum position (3) in mm. → Specify the maximum force (4). ✓ If no force is specified, the maximum force of the system is automatically used. ✓ While approaching the target position (4), the current force is monitored by the force sensor. ✓ If the actual force exceeds the specified maximum value, the system is stopped immediately to prevent damage to the system or the tools. ✓ The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. → In the respective input fields (1-4), alternatively click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the respective *fixed value*. 💡 In addition, it is possible to adjust the performance of the module via parameter MFR005 in the extended configuration. **ATTENTION** **Excessively high joining speed may cause the target force to be exceeded due to the high kinetic energy.** If the values are significantly exceeded, “Error 041004 – program error: max. force exceeded” will be displayed, see the chapter Error catalog. Possible damage to the workpiece. → Reduce the speed. → If necessary, increase the maximum overload with parameter VFR002.

Screens / comments	Action steps
	ATTENTION **If the system is operated without a force sensor, the applied force is incorrectly estimated on the basis of the motor torque.** Impaired performance in operation without force sensor. Possible damage to the workpiece. → Refit a force sensor, see chapter Customized force sensor.
	**Join to position** The *Join to position* command 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 distance is traveled, starting at the current position. Once the position is reached, the system switches to the next module in the program sequence. → Individually specify the speed (3) and acceleration (4) with which the system is to move. The system does not check whether or not the set speed can be reached with the selected acceleration. → Specify the maximum force (5). → 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. If the actual force exceeds the specified maximum value, the system is stopped immediately to prevent damage to the system or the tools. The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. → In the respective input fields (2-6), alternatively click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the respective *fixed value*.

Screens / comments	Action steps
	**Check force/position target window** The command *Check force/position target window* initiates a cross check of the actual force and displacement values against the specified target values. During this check, all 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 exceeded or undershot, the program sequence is aborted and a message is issued (numbers 041010 – 041024, see the 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 target values window is violated: • Abort sequence (5) • Position and abort sequence (6) With optimum positioning: → Individually specify the speed (7) and acceleration (8) with which the system is to move. ✓ The system does not check whether or not the set speed can be reached with the selected acceleration. ✓ The target position (9) is directly approached.

Screens / comments	Action steps
	**Move to signal** The command *Move to signal* causes the system to move 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 allows for integration of an external sensor and to use it to probe a workpiece surface, for example. As an alternative, the program module can be controlled via an overriding control with the help of the virtual inputs, e.g. to integrate additional process check options or measuring systems. → Individually specify the speed (1) and acceleration (2) with which the system is to move. ✓ The system does not check whether or not the set speed can be reached with the selected acceleration. → 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). ✓ 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. ✓ If the actual force exceeds the specified maximum value, the system is stopped immediately to prevent damage to the system or the tools. ✓ The overload causes the program sequence to be aborted and the currently produced part to be rated NOK. → In the respective input fields (1-4), alternatively click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the respective *fixed value*. → From the selection element (5) select an input that is to deliver the high signal.
	**Delay** The *Delay* command causes the system to wait for a defined period. → Enter the delay time in ms. → Alternatively, click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the *fixed value*. ✓ The system will then switch to the next module in the program sequence.

Screens / comments	Action steps
	**Wait for signal** The command *Wait for signal* causes the system to wait until a defined digital or virtual input provides a high signal. The system will then switch to the next module in the program sequence. With this command it is possible to synchronize the system with other parts of the machine, e.g. the clamping element. → Enter the maximum time (1) in ms. → Alternatively, click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the *fixed value*. ✓ Upon expiry of this period, the program sequence is aborted. ✓ The currently produced part is then rated NOK. → From the selection element (2), select an input that is to deliver the high signal.
	**Set output** The *Set output* command sets a defined digital or virtual output to a high signal 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 or a signal lamp. → From the selection element (1), select an output for which the high signal is to be set. → Select whether the bit is to be set *high* (2) or *low* (3). → Enter the duration in ms (4) for which the output is to be set to a high signal. → To set the high signal for an indefinite time, set the value to 0. → Alternatively, click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the *fixed value*.
	**Tare** The *Tare* command tares the force sensor. The force sensor can be tared to any desired value. This makes it possible to compensate for the weight of a tool, for example. Taring applies globally to all force values and thus also for e.g. the force values in the program steps, curve recording, and quality assessment. → Enter the force value in N to which the sensor is to be tared. → Alternatively, click the *Variable* tab and select the predefined variable from the selection element instead of manually entering the *fixed value*.

Screens / comments	Action steps
	**Set variable** The *Set variable* command 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. → Alternatively, click the *Variable* tab and select another predefined variable from the selection element with whose value the selected variable is to be overwritten. **Notice** When the program is started, all used variables are saved in the data base for documentation (variable table). Changes to variable values by means of the command *Set variable* are typically carried out afterwards, so that the new values are not listed in the variables table. See also the chapter Variables configuration.

5 Library

8.3 Transition groups

A transition group includes at least two subsequent components of a sequence. The elements of this group are then no longer considered individually but as a single unit. The movements of the respective modules are consolidated and automatically optimized in terms of their cycle times. This process is marked by adding the identifier “slur” to the respective modules.

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.

This results in reduced cycle times.

Conditions

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

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

- Positioning
- Join to position

The following two modules can be used as intermediate steps as often as required. Their respective target positions must always between the start and end positions of the slur group and must always be consecutive. They are each marked with “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
- Move to signal

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

💡 Position values must be specified either by means of fixed values or by variables.

Use of variables

If the position values are specified by variables, no check is performed whether the position values are consecutive or not. The respective modules are automatically joined to form a transition group.

If this is not desired, the transition group must be interrupted according to the following section.

Interrupting a transition group

A transition group typically improves the cycle time. However, in some applications, standstill of the axis is explicitly desired. In such cases, the groups can be broken up.

Please proceed as follows:

→ Use the “Start position” module instead of the modules “Join to position” or “Position”.

💡 This module is never part of a transition group.

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

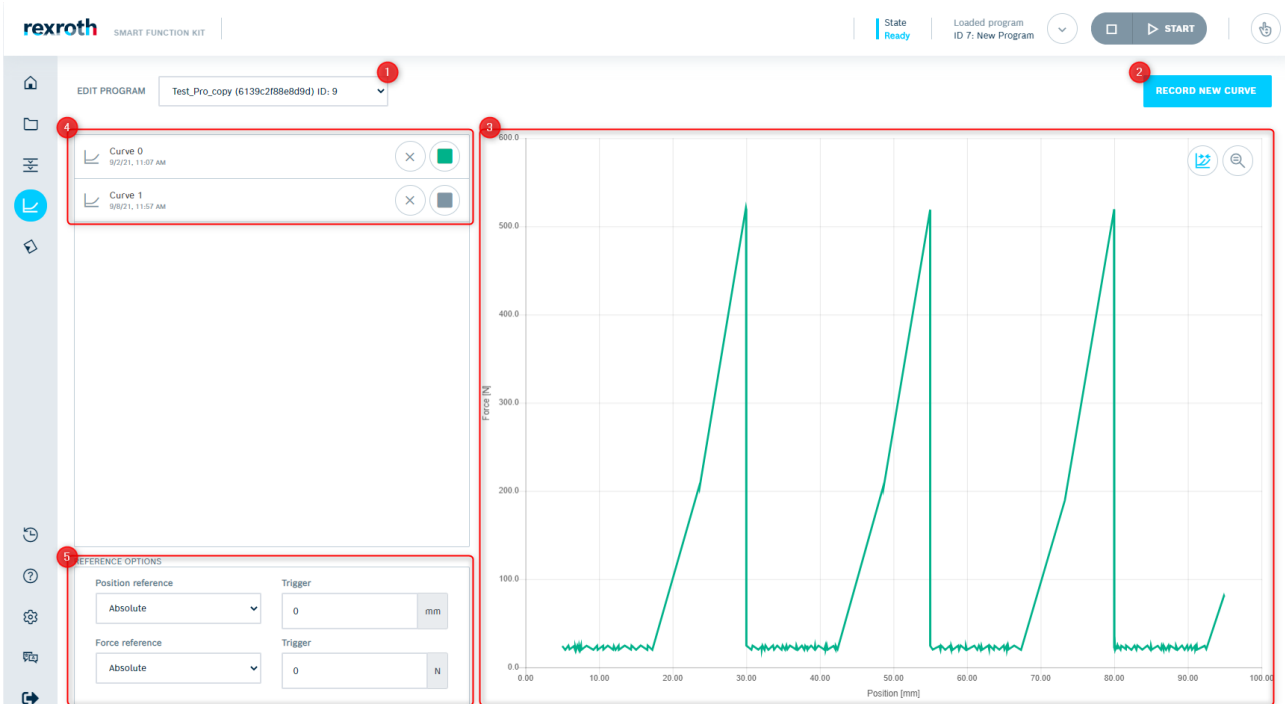
- Delay (waiting time 0 ms)
- Tare
- Set output

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.

However, at least two reference curves are required for defining evaluation elements and setting up automatic quality assessment.



25 Reference curves

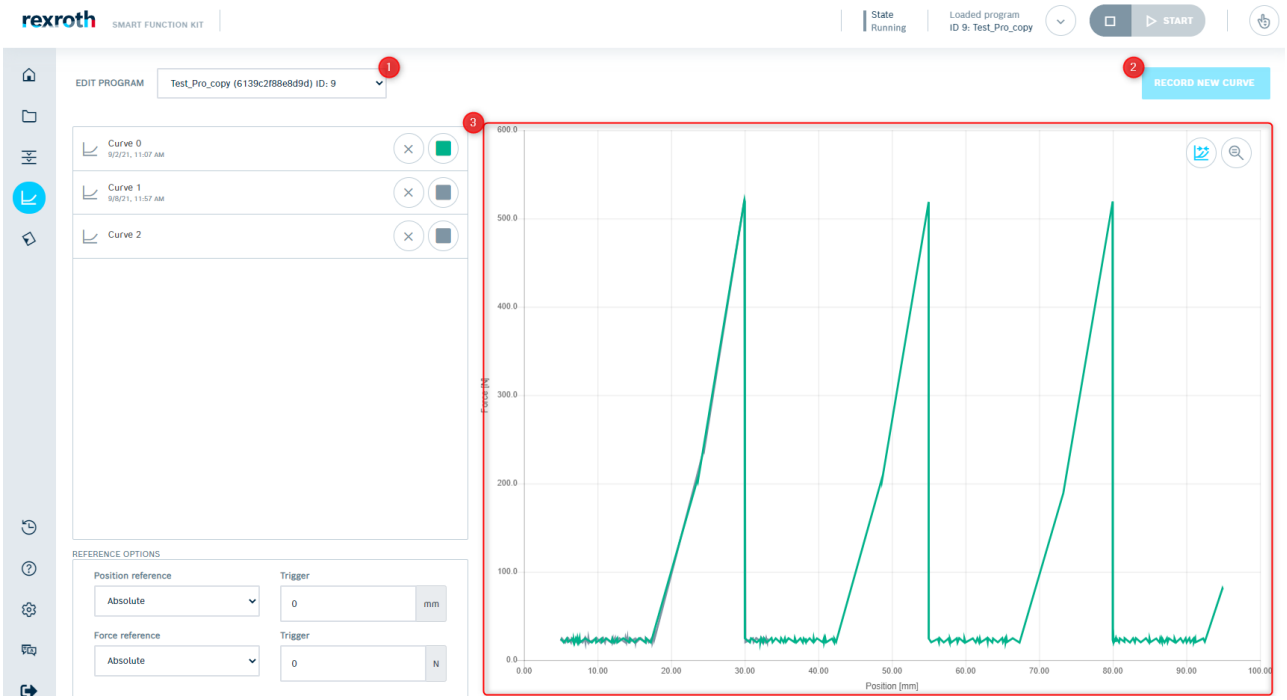
Item	Description	Function
1	Selection element program	→ From the selection element, select a program for recording a curve or viewing a previously recorded curve.
2	*Record new curve* button	→ Click the *Record new curve (2)* button. ✅ The program sequence is run once and the force-displacement curve is displayed live in the diagram.
3	Curve diagram	Recording of the curve is displayed live. If several curves are recorded at the same time, all these curves are displayed in the diagram as an array of curves.
4	Curves list	All recorded curves are included in the list (4). → You can evaluate or delete the listed curves, or rename them.

Item	Description	Function
5	Reference options	→ Set how and in which direction dynamic shift of the curve is to be performed.

6 Reference curves

9.1 Recording

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



26 Recording

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

→ Click the *Record new curve (2)* button.

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

✅ If several curves are recorded at the same time, all these curves are displayed in the diagram 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.

ATTENTION

Any reference curves that are already present are not automatically deleted when modifications are made to the program.

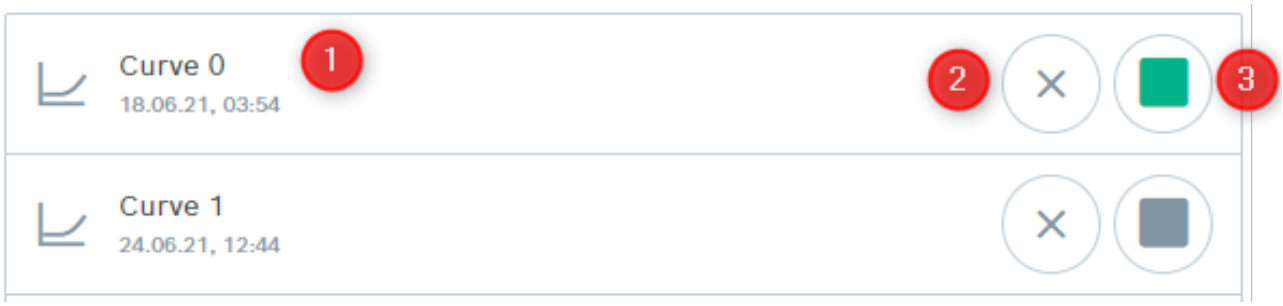
Misinterpretation when setting validation elements.

→ Manually check the sequence of programs and validation properties if they have been modified after reference curves have been generated.

→ 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.



27 Curves list

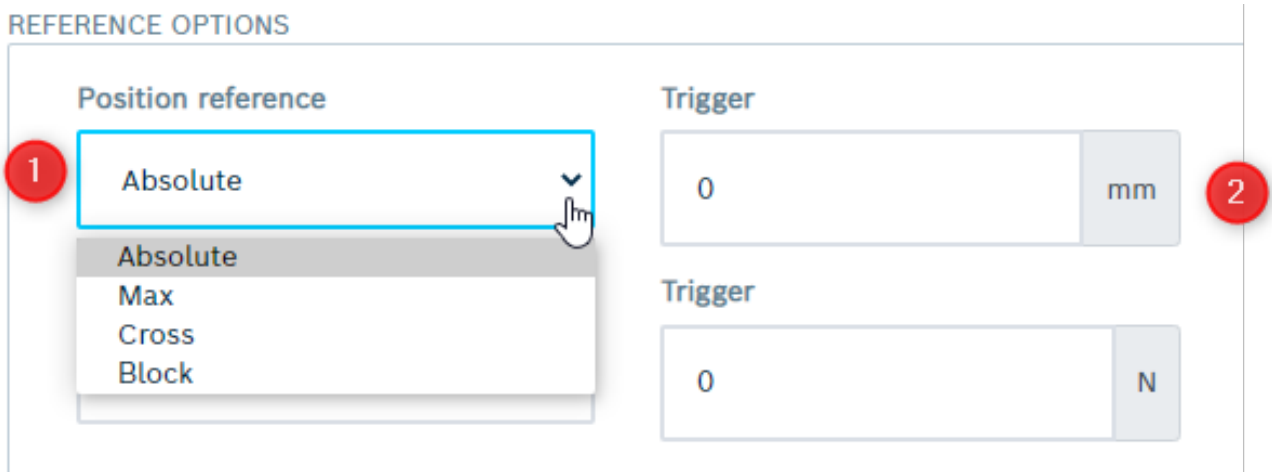
Item	Description	Function
1	Curve name	The reference curves are given an automatically generated name. → Click on the name to make it editable. → Enter the desired curve name and confirm your input by pressing *Enter* or by exiting the respective input field. 💡 Defining individual names can make assigning reference curves to the respective sample parts easier.
2	Delete	→ 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.
3	Evaluation	→ Click the *Evaluate* (3) button. ✅ A selection element is displayed. → Select one of the three marking options: • Inactive • Good reference • Bad reference ✅ In the diagram, curves rated *good* are displayed green, while curves rated *bad* are displayed red. ✅ Curves that are marked *not rated*, are hidden in the diagram and not considered in the evaluation. → Change the marking of an *inactive* curve to *good or bad* to have the curve displayed in the diagram.

7 Reference curves

9.3 Reference options

The Smart Function Kit supports dynamic shifting of force-displacement curves in order to ensure a reliable quality assessment even with varying component tolerances. In the reference options menu you can set how and in which direction dynamic shifting is going to take place.

9.3.1 Position reference (X-axis)



28 Position reference

Absolute

The *Absolute position* setting is the standard setting.

With *Absolute position*, 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* reference in the selection element (1).

✅ The distance is defined when creating the evaluation element.

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

Maximum

When creating the *Maximum position*, 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.

The position of the trigger point is individual for every produced part. The trigger point is the point at which the maximum position value is present. This position value is displayed in the *Trigger* input field and cannot be edited.

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

✅ This has the following effect:

1. The maximum position of **all** reference curves marked as *Good reference* (see section 2 Evaluation) is determined.
2. This determined maximum position is set as trigger position.
3. The determined trigger position is marked on the X-axis and represented in the plot by means of a vertical dashed line.
4. All subsequent curves are shifted along the X-axis so that their end position coincides with the determined trigger position. This shift is performed graphically – the shift offset is saved together with the data set.

Cross

When creating the *Cross position*, 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.

The position of the trigger point is individual for every produced part. The trigger point is the point at which the position value was reached for the first time.

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

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

✅ This has the following effect:

1. The **first** reference curve marked as *Good reference* (see section 2 Evaluation) is considered the reference curve.
2. 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 referred to as “reference force”.
3. 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

When creating the *Block position*, 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.

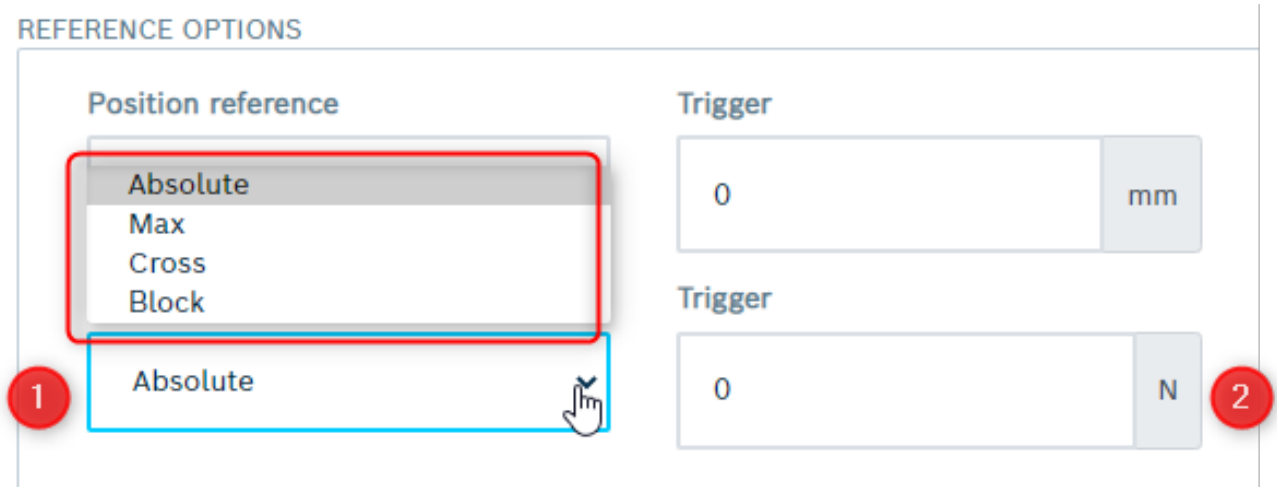
The position of the block point is individual for every produced part. The block point is located at the maximum position of the first reference curve. This position value is displayed in the Trigger input field.

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

✅ This has the following effect:

1. The **first** reference curve marked as *Good reference* (see section 2 Evaluation) is considered the reference curve.
2. 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 referred to as “reference force”.
3. 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)



29 Force reference

Absolute

The *Absolute force* setting is the standard setting.

With *Absolute force*, the X-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* reference in the selection element (1).

✅ The distance is defined when creating the evaluation element.

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

Maximum

When creating the *Maximum force*, 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.

The position of the trigger point is individual for every produced part. The trigger point is the point at which the maximum position value is present. This position value is displayed in the *Trigger* input field and cannot be edited.

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

✅ This has the following effect:

1. The maximum position of **all** reference curves marked as *Good reference* (see section 2 Evaluation) is determined.
2. This determined maximum position is set as trigger position.
3. The determined trigger position is marked on the Y-axis and represented in the plot by means of a vertical dashed line.
4. All subsequent curves are shifted along the Y-axis so that their end position coincides with the determined trigger position. This shift is performed graphically – the shift offset is saved together with the data set.

Cross

When creating the *Cross force*, 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.

The position of the trigger point is individual for every produced part. The trigger point is the point at which the position value was reached for the first time.

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

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

✅ This has the following effect:

1. The **first** reference curve marked as *Good reference* (see section 2 Evaluation) is considered the reference curve.
2. 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 referred to as “reference force”.
3. 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

When creating the *Block force*, 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.

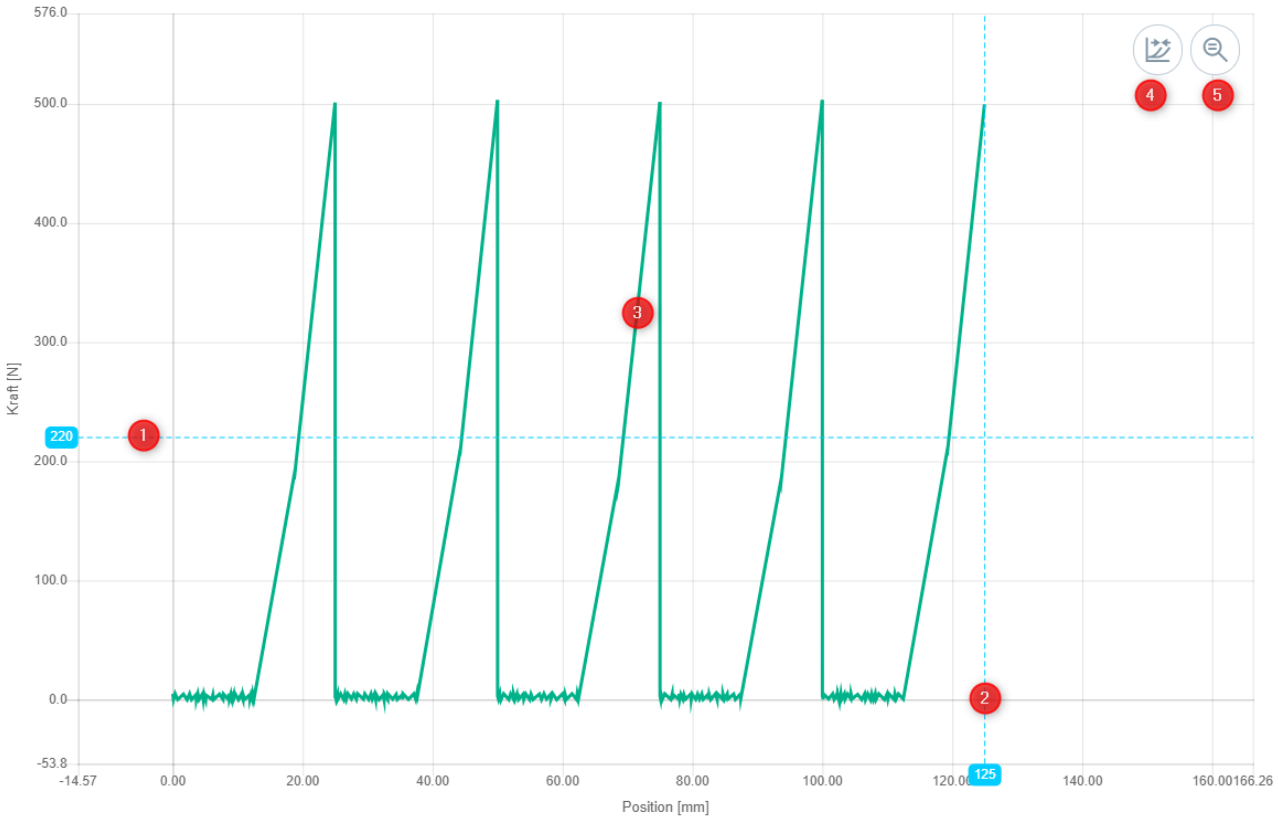
The position of the block point is individual for every produced part. The block point is located at the maximum position of the first reference curve. This position value is displayed in the Trigger input field.

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

✅ This has the following effect:

1. The **first** reference curve marked as *Good reference* (see section 2 Evaluation) is considered the reference curve.
2. The block position (maximum position) 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 referred to as “reference force”.
3. 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



30 Diagram representation

Item	Description	Function
1	Position reference	Depending on the selected position reference (cross or block), the block position (maximum position) 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	Force reference	Depending on the selected force reference (cross or block), the block position (maximum position) 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.
3	Curve representation	If the reference option is set to something other than *Absolute position* or *Absolute force*, 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.
4	Curve consolidation	→ Click the *Curve consolidation* button. ✅ The modified representation of the curve in the diagram is activated or deactivated.
5	Zoom	→ Click the *Zoom* button. ✅ The zoom of the curve diagram is reset to standard view.

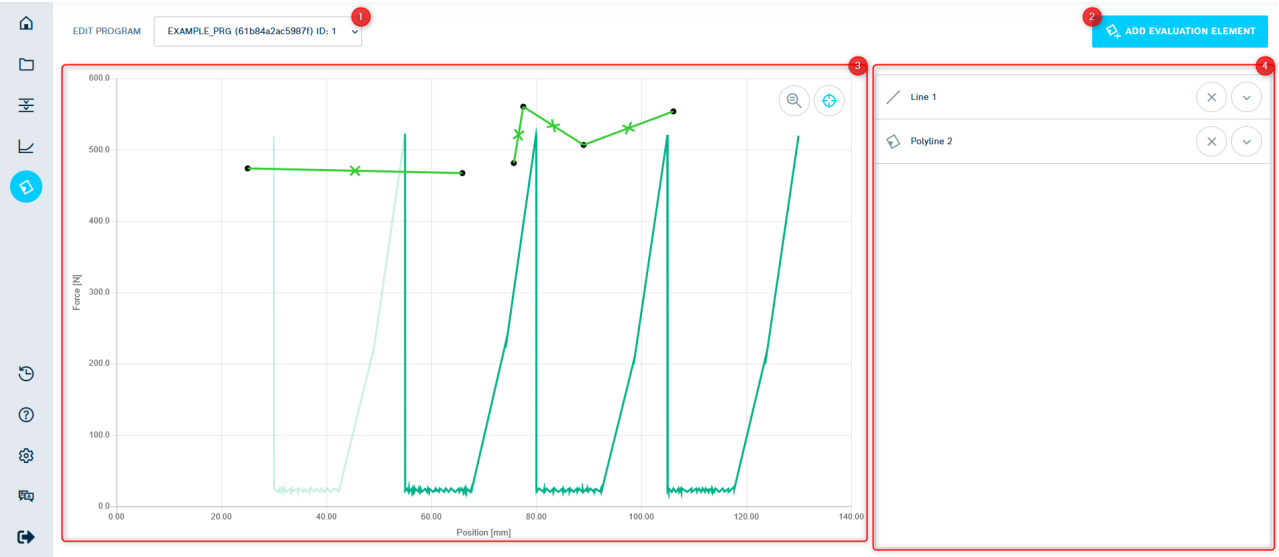
8 Diagram representation

10 Evaluation

In the evaluation elements area you can individually define evaluation elements for every program. The elements are used for the automatic quality assessment and by means of their evaluation the system determines whether a produced part is OK or NOK.

Notice
To be able to set the evaluation elements, at least one reference curve must have been recorded.

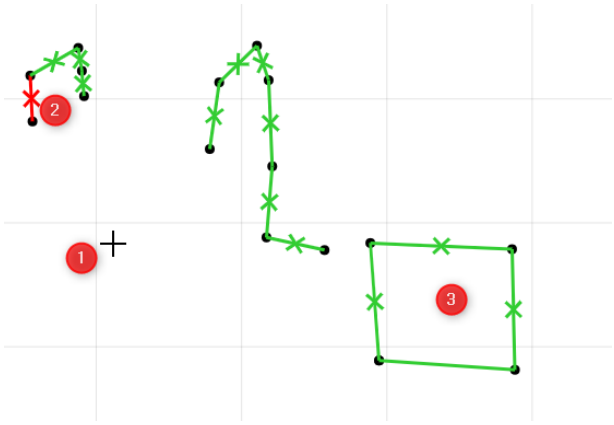
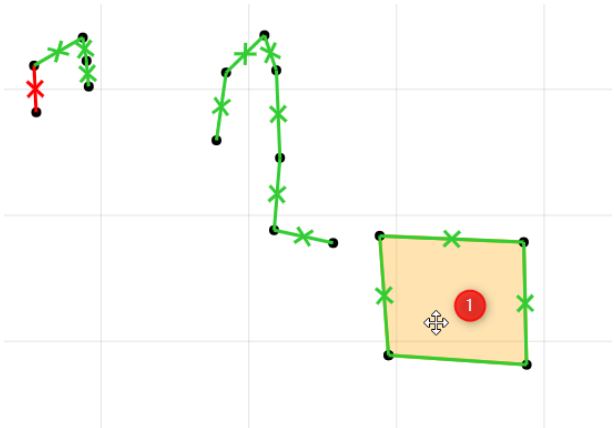
Notice
Evaluation elements can only be set in the displacement/force representation of the diagram.

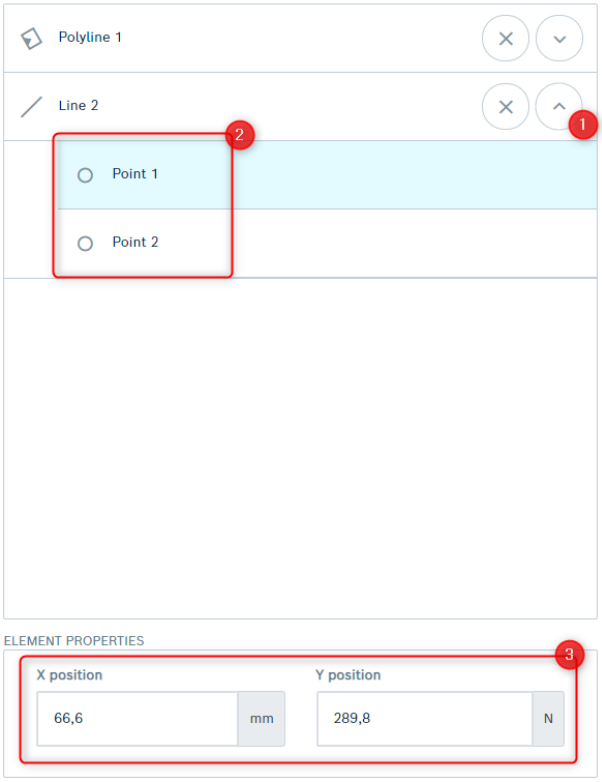
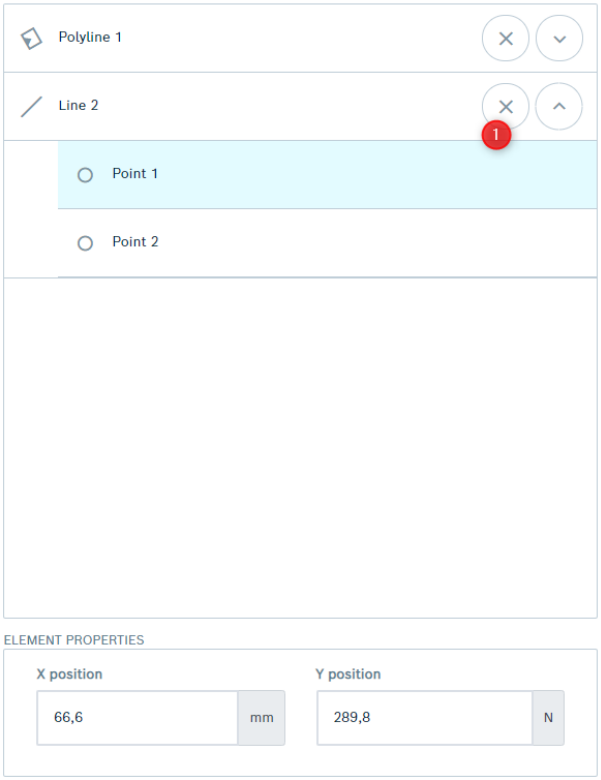


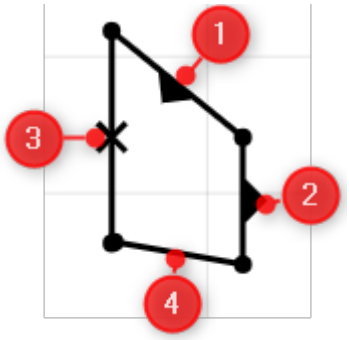
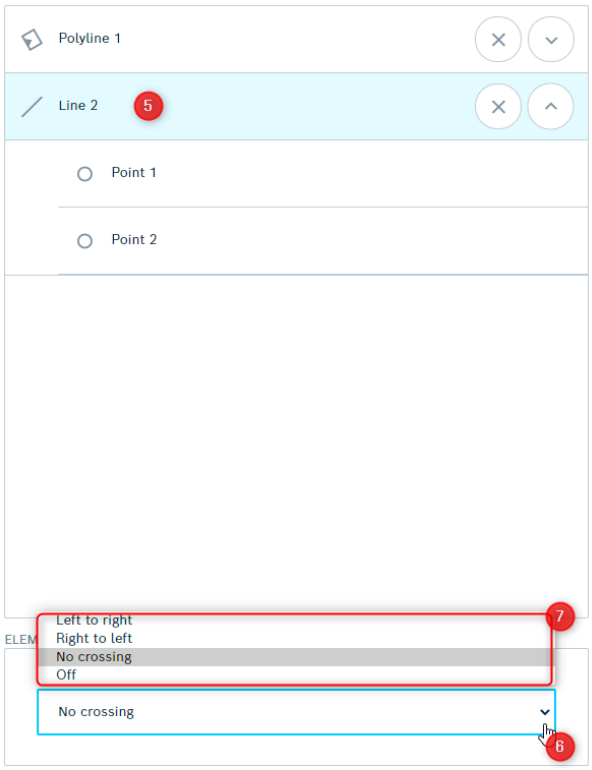
31 Evaluation

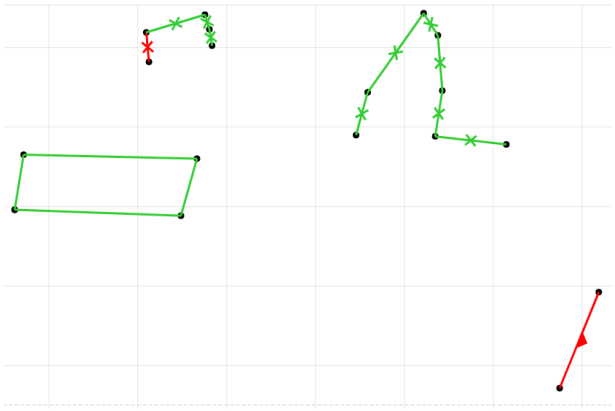
Item	Description	Function
1	Selection element program	→ From the selection element, select a program for evaluating a curve.
2	*Add evaluation element* button	→ Click the *Add evaluation element (2)* button. ✅ The drawing area of the diagram view (3) is activated.
3	Diagram view	Drawing area for evaluation elements. In time view, evaluation elements can no longer be displayed, created or edited.
4	List of evaluation elements	The position of the points of an evaluation element can be precisely defined in the list of evaluation elements.

9 Evaluation

Screens/comments	Action steps
	**Creating evaluation elements** Click the *Add evaluation element* button. → Left-click anywhere in the diagram. ✅ This creates a new point (1). → Click again on another position in the diagram to create another point. ✅ Points created one after the other are automatically connected by a line (2). 💡 Any number of points can be set when defining evaluation elements. → Place the end point on the start point to create a closed polygon (3). 💡 Polygons that are not closed can also be used as valid evaluation elements. → Click the *Complete evaluation element* button to complete the drawing process. → Alternatively, right-click on the drawing area to complete the drawing process. → Draw a closed polygon to complete the drawing process. → Click the *Add evaluation element* button to add a new evaluation element. ✅ The drawing mode is activated again.
	**Moving evaluation elements** → Click on the middle of the element (1). ✅ The element is marked and a cross-hair cursor is displayed. → Directly click on a point or a line to only move this item. ✅ 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 to move the element to the desired position.

Screens/comments	Action steps
	**Detailed positioning** The position of the points of an evaluation element can be precisely defined in the list of evaluation elements. → Open the menu of the corresponding evaluation element (1). → Mark the individual points (2). → In the lower section, enter the coordinates of the points in the input fields (3).
	**Deleting evaluation elements** → Click the *Delete* button (1) to remove an element from the list of evaluation elements. 💡 An element can only be deleted as a whole.

Screens/comments	Action steps
 	**Properties of evaluation elements** Evaluation elements are used for the automatic quality assessment of produced parts. For this purpose, the lines of an evaluation element are given the following properties: • Passage from right to left required (1) • Passage from left to right required (2) • No restriction (3) • No passage allowed (4) After a part has been produced, the force-displacement curve is automatically checked for compliance with these criteria. If one of the criteria is violated, the produced part is rated NOK. Only if all criteria are complied with, the part is rated OK. 💡 The drawing mode must not be active during quality assessment. → You can adjust the properties of an evaluation element line 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).

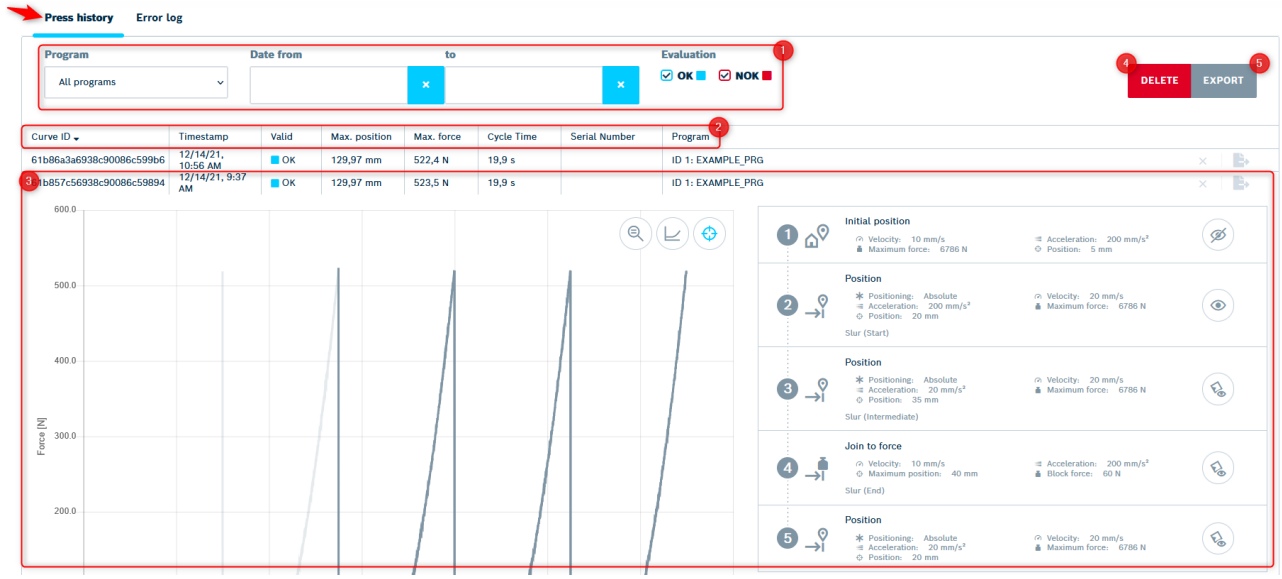
Screens/comments	Action steps
	The diagram displays a live preview of the evaluation. All curves rated good (marked green) are considered in the evaluation. The evaluation result is displayed by an appropriate color for every evaluation element line. A red line means that one of the curves rated good violates the criterion. Green lines mean that the criterion is valid for all curves rated good. 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 validation. Reference curves that do not correspond to this configuration, may cause unexpected behavior in the representation.

10 Evaluation 2

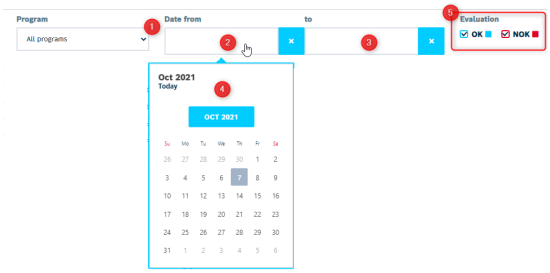
11 Log files

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.



32 Press history

Item	Description/image	Function
1	Filter options 	→ From the selection element (1), select a program to filter the table for the individual program. → Click the *Date from* (2) or *to* (3) input field. ✓ A calendar (4) is displayed. → Select a period using the calendar or manually enter the dates into the input fields. → Select or deselect the fields *OK* or *NOK* (5) to filter the table by the results of the quality assessment. ✓ The table will be filtered according to the selected criteria.

Item	Description/image	Function
2	Production data	The press history shows the production data in the form of a table. The individual columns of the table can be sorted in ascending or descending order. In addition, the production data can be filtered by the program name, the time stamp and/or the evaluation results. Production data include: • Unique part ID • Time stamp • Result of the quality assessment • Maximum position • Maximum force • Cycle time • Serial number (optional, see the chapter Smart Function Kit SFK field bus) • Program name
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. → In the diagram, click the *Zoom* button (4) to reset the focus of the diagram. → In the diagram, click the *Curve details* button (5) to show or hide the details in the diagram. → The *Position symbol* (6) indicates that the position is selected as X-axis. Click on the switching element to switch to the time area. Click again to switch back to the position view. → Click on the program row (1) again to hide the diagram and return to the table view. → Click on *Delete*(7) to delete the selected data set. → Click on *Export* (8) to export the selected data set. ✓ A dialog window for file selection is opened in the browser and the storage location in the HMI device can be selected.

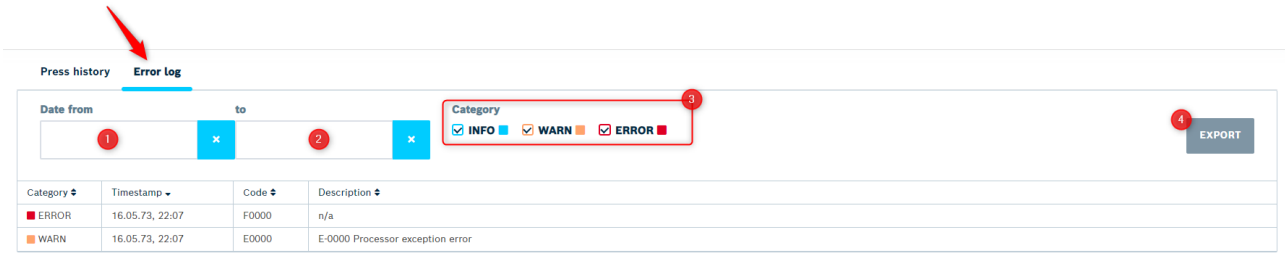
Item	Description/image	Function
4	Delete	→ Click the *Delete* button to delete production data and free memory capacity. 💡 If a filter is active, only the filtered results are deleted. 💡 Deleting data is only possible if the user is logged in as administrator. **Notice** If press history data are deleted,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

Item	Description/image	Function
5	Export	→ Click the *Export* button. ✅ A dialog window for file selection is opened in the browser and the storage location in the HMI device 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 for the selected data set. 💡 Exporting data is only possible if the user is logged in as administrator. **⚠ WARNING** **Data are streamed between the client and the system.** "During export (System → Client), data are written via a continuous data flow. If the network connection is instable, this may cause delays and/or the writing of the data may be aborted. → Ensure a stable network connection, preferably via LAN. → Ensure that the bandwidth is sufficient. → Perform the export as exclusive action, without any other activities, such as pressing, running in parallel. Not with other browsers, either. **Notice** **Reduced download speed during parallel production.** 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.

11 Press history

11.2 Error log

The Error log area presents information messages, warnings and errors of the Smart Function Kit in the form of a table.



33 Error log

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

✔ A calendar is displayed.

→ Select a period using the calendar or manually enter the dates into the input fields.

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

✔ The table will be filtered by the selected criteria.

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

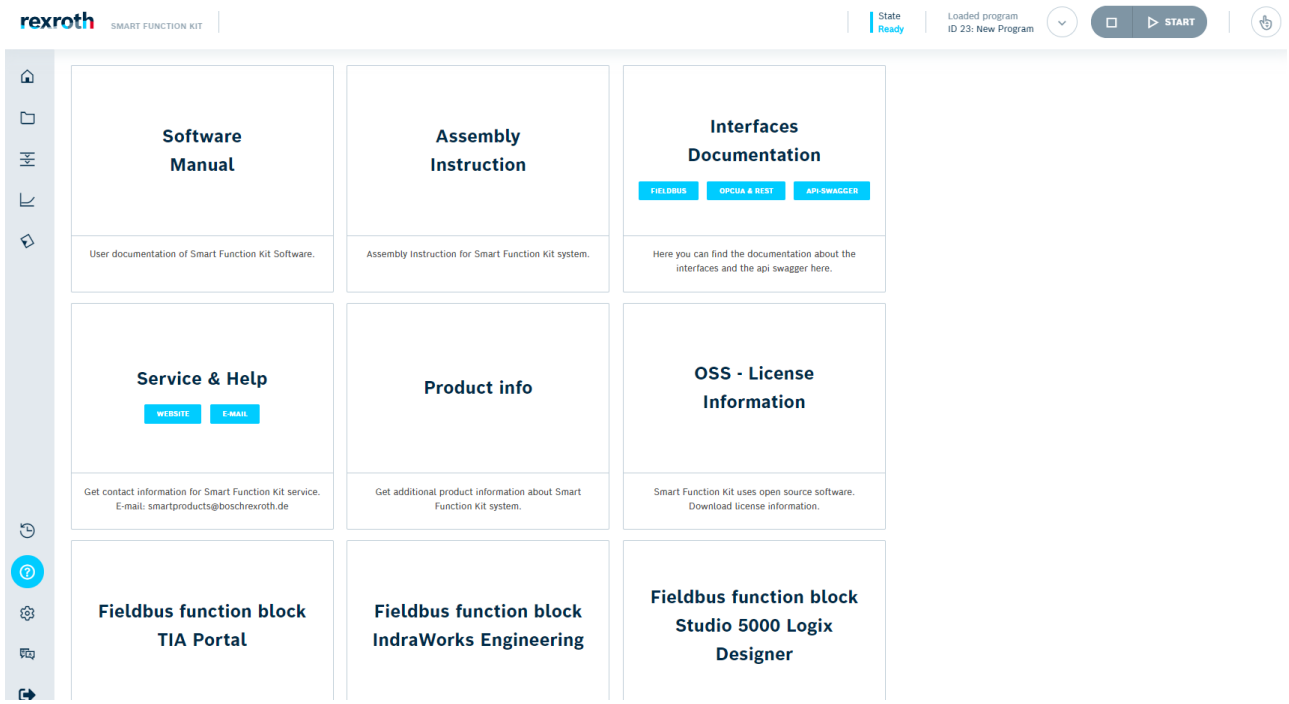
→ Click the *Export* button (4) to export the error log.

✔ A dialog window for data selection is opened in the browser and the storage location in the HMI device can be selected.

✔ Diagnostics are exported in JSON file format.

◊ For detailed information on the individual items, please see the chapter *Troubleshooting* at the end of this instructions manual. Please also observe the documentation of the individual products.

12 Help



34 Help

→ Click on a tile to open the respective document:

- Software instructions
- Assembly instructions
- Interface documentation
- Etc.

✅ The selected document will be displayed in PDF format.

→ Click on a tile to open the link behind it:

- E.g. Service or product information

✅ The corresponding page is opened (here: list of contact persons, etc.)

→ Click on a tile to download a file:

- E.g. Field bus function block ...

✅ The corresponding file is downloaded

13 Settings for Smart Function Kit

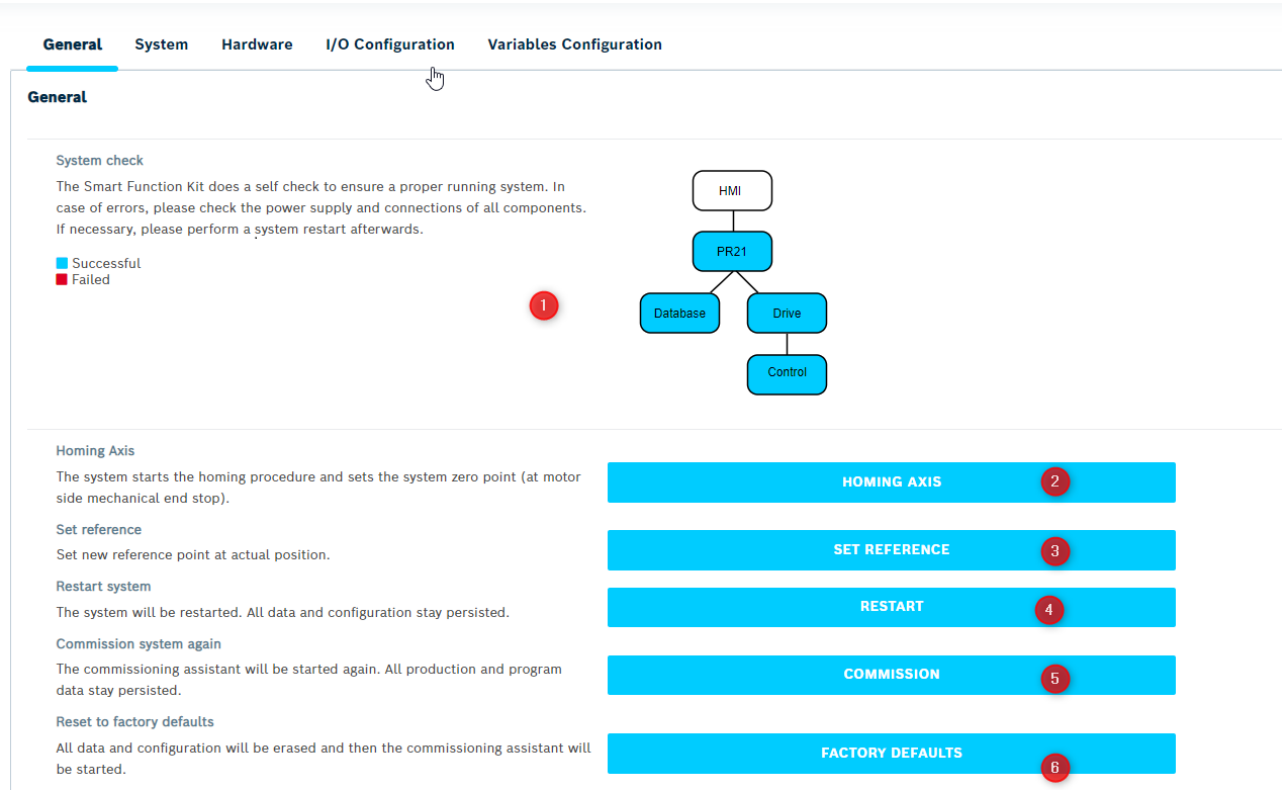
Über den Navigationsbereich "Einstellungen" könne verschiedene Systemeinstellungen am Smart Function Kit vorgenommen werden.

35 Settings for Smart Function Kit

You can switch between the following areas using the tabs:

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

13.1 General configuration



36 General configuration

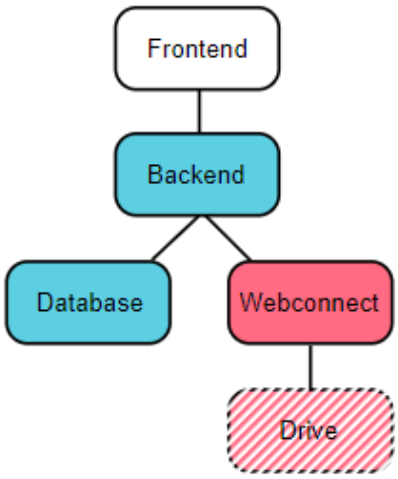
System status (1)

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.

The status is displayed as follows:

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

The system can only be used if all components are ready.



37 System status

→ If a faulty component is identified in the system status, please proceed as follows:

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

◊ For detailed information on the individual steps, please see the chapter Troubleshooting and fault clearance at the end of this instructions manual.

Homing Axis (2)

- For detailed information on the reference run, see the chapter Homing Axis.

Set reference (3).

ATTENTION	
	**Damage to the system and restriction of function** When using the *Set reference* function, the system limits and the software limit switches are not automatically adjusted to the new zero point of the system. Faulty software limit switches may result in damage to the system. The functions of the system may be restricted. → Only use the *Set reference* function if absolutely necessary. → Adjust the parameter *Y-1044 Maximum travel positive* in the *Advanced configuration* according to the new zero point.

→ Click the *Set reference* button to set the system zero at the current position of the cylinder.

Restart (4)

→ Click the *Restart* button to restart the entire system.

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

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

Commissioning (5)

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

All production data and programs will be maintained after recommissioning. All system settings are reset to standard values.

ATTENTION
If the *Commissioning* function is activated, all system settings will be lost. The functions of the system may be restricted. → Back up all settings before recommissioning.

Factory defaults (6)

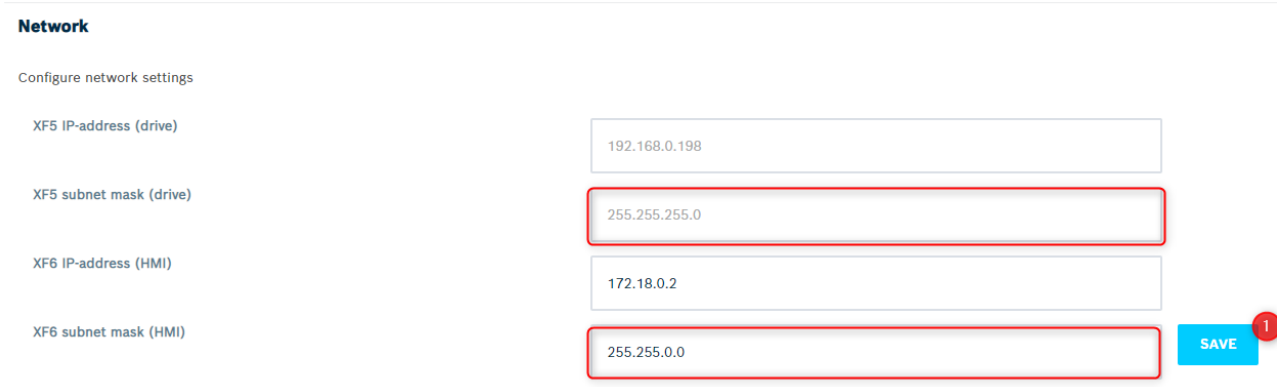
→ Click the *Factory defaults* button to completely reset the system to the factory settings.

All settings and data will be lost.

Before resetting to factory defaults, we recommend performing a data backup.

ATTENTION
If the *Factory defaults* function is activated, all data and settings will be lost. Loss of production data → Preform a data backup beforehand.

13.2 System configuration



38 Network settings
To be able to integrate the Smart Function Kit into a specific local network, the IP address and the subnet mask can be changed.

1. Enter a valid IP address and subnet mask in the input mask.

2. Click *Save*(1) to apply the settings.

The address range 192.168.1.x cannot be used for the Smart Function Kit.

→ Make sure that the HMI device is in the same subnet, as otherwise communication with the Smart Function Kit cannot take place.

The following network settings are factory set for the system:

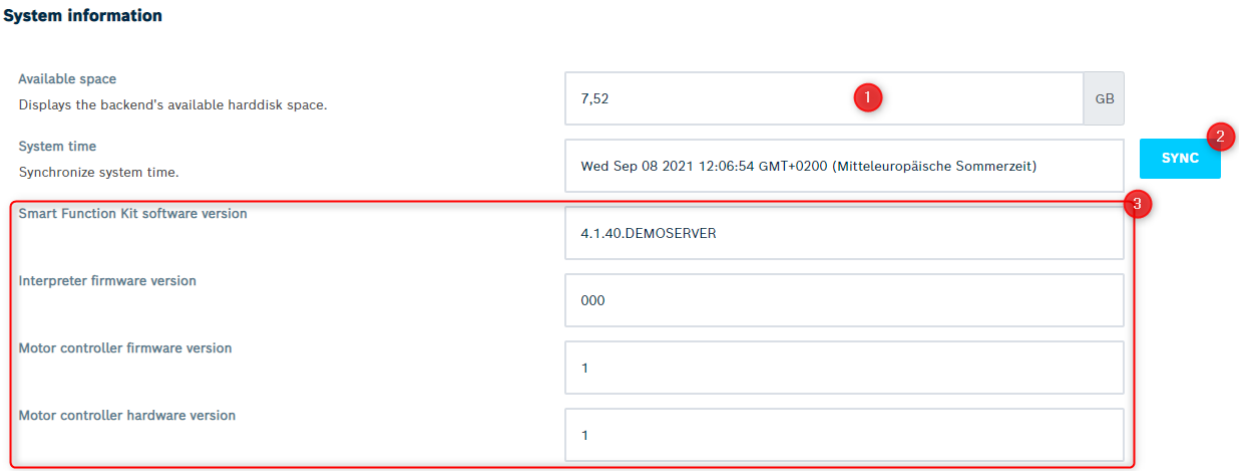
- IP address: 168.0.1
- Subnet mask: 255.255.255.0
- Gateway address: 0.0.0.0

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.

ATTENTION	
	**Loss of connection between drive controller and PR21** If the connection between the drive controller and the PR21 is lost, the Smart Function Kit 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



39 System information

The **memory capacity** display field (1) shows the available storage capacity of the PR21. If the memory is full, no more new production data can be saved.

In delivery condition, the device provides a storage capacity of approx. 25 gigabytes for production data. The size of a production data set strongly depends on the program and recording length and is typically between 10 and 100 KB.

→ If required, perform a backup of production data in the press history area.

→ Delete production data that are no longer needed in the press history menu to free up storage capacity.

ATTENTION

Data memory on the PR21 is full

If the data memory on the PR21 is full, no new production data can be saved. New production data will be lost.

→ Perform a backup of production data and free up storage capacity.

Synchronization of the system time is performed automatically during start-up. During synchronization, the system time of the HMI device is read and applied as the system time in the Smart Function Kit. All future production data will be time-stamped with the new system time.

→ Click the *SYNC (2)* button to synchronize the system time.

ATTENTION

Subsequent modification of the system time

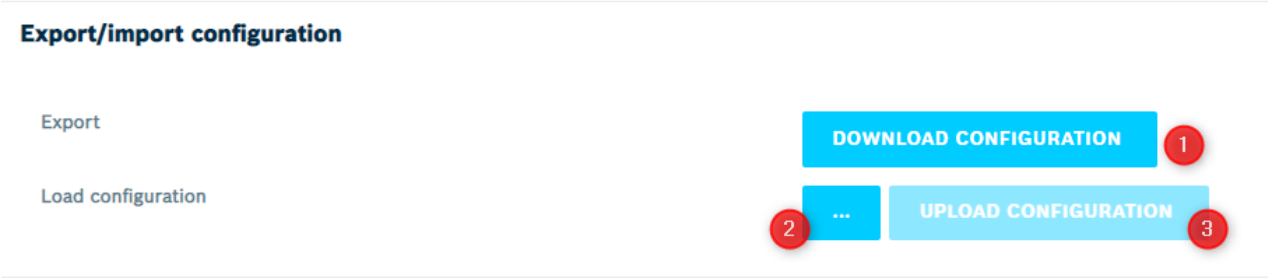
Subsequent modification of the system time will cause inconsistent time stamps in the production data. Previously saved data sets will not receive the new time stamp.

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

The **Component information** (3) area shows the version and/or hardware identification code of the software and hardware used.

- Smart Function Kit software version
- Interpreter version (SMC)
- Drive controller firmware version
- Drive controller hardware version

Exporting/importing configurations



40 Exporting/importing configurations

The Smart Function Kit 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.

→ Click the *Backup configuration* button (1) to perform a system backup.

→ Select a storage location on the HMI device for the backup.

Notice
It is possible to load a backup and thus restore the system settings as a whole e.g. after a “Reset to factory defaults”. It is also possible in this way to transfer settings from one system to another identical system.

→ Click the ... (2) button to select the storage location of the backup on the HMI device.

→ Click the *Restore configuration* button (3) to restore from a backup.

Notice
After loading a backup, a system commissioning must be performed to carry out all settings and ensure safe operation.

⚠ WARNING

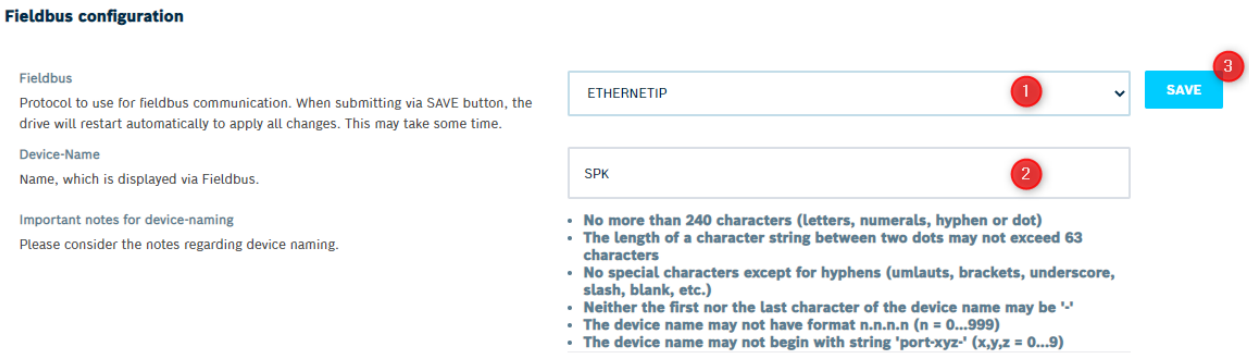
Delays or aborts due to unstable network connection.

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 instable, 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.

Selection of field bus



41 Selection of field bus

The Smart Function Kit can be operated as a subsystem with a field bus master.

→ Open the menu (1) and select one of the following field buses available for the Smart Function Kit:

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

→ Enter the corresponding device name to be used in the field bus for the SFK in the *input field* (2).

💡 Input is optional and depends on the field bus selected in field (1).

→ Click *Save* (3) to save the selection of the field bus. The drive is restarted.

◊ For detailed information on the topic of field bus, see the chapter Smart Function Kit SFK field bus.

13.3 Hardware configuration

The hardware configuration area provides an overview of the hardware of the Smart Function Kit.

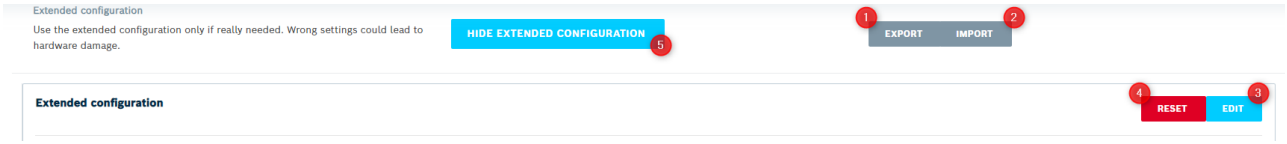
During commissioning, the hardware configuration is automatically read and all parameters and limit values are automatically set.

Extended configuration

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

→ Click *Show extended configuration* (5) to display the extended configuration. After clicking, the button changes to *Hide extended configuration*.

→ Click *Hide extended configuration* (5) to close the extended configuration. After clicking, the button changes to *Show extended configuration*.



42 Extended configuration

→ Click *Export* (1) to export the current values.

→ Click *Import* (2) to import previously set extended configurations.

→ Click *Edit* (3) to enable manual input in the input fields of the extended configuration. After clicking, the button changes to *Save*.

→ Click *Save* (3) to save the inputs made. After clicking, the button changes to *Edit*.

→ Click *Reset* (4) to reset any inputs made.

ATTENTION	
	**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 Smart Function Kit uses various components that have a configuration interface via parameters.

◊ For detailed information on the respective parameters, please see the corresponding documentation of the individual products.

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

Y parameters: Configuration of the SMC technology package.

VFR variables: Configuration of the interpreter firmware.

List of system parameters:

ID	Name
S-0-0052	Reference dimension [mm]
S-0-0121	Gear ratio input revolutions
S-0-0122	Gear ratio output revolutions

ID	Name
S-0-0123	Screw lead [mm]
P-0-0001	Change frequency of the power output stage [Hz]
P-0-0279	Time resolution for trending [ms]
P-0-0217	Time constant for analog input filter [ms]
P-0-1320	StrictMode function active
P-0-1321	Force limitation for reference run [%]
Y0015	E-stop function active
Y1004	Maximum speed [mm/min]
Y1006	Maximum acceleration rate [mm/s2]
Y1007	Maximum force [%]
Y1011	Negation of position data
Y1044	Maximum travel, positive [mm]
Y1045	Maximum travel, negative [mm]
Y1049	Analog force sensor active
Y1050	Force sensor lead characteristic (tension) [N/V]
Y1053	Force sensor lead characteristic (compression) [N/V]
Y1054	Limit of analog value characteristic [V]
VFR002	Tolerance of force sensor [N]
VFR004	Force limitation upon positioning command [n]
VFR005	Join to force: Relative minimum speed
MFR001	Continuous mode
MFR002	Actual data filter
MFR003	SafeMotion

ID	Name
MFR004	SMO: Pause program
MFR005	Cyclic variables documentation
MFR006	Program persistent after restart
MFR007	Data recording deactivated
MFR008	Stop and torque off function active

12 List of system parameters

S-0-0052 Reference dimension

In this parameter, the desired actual position value in relation to a specific axis position (reference position) is input. With the *Reference run* or *Set reference* commands, the value entered in parameter S-0-0052 becomes effective as the actual position value at the reference position.

S-0-0121 Gear ratio input revolutions

Gear input revolutions. Together with S-0-0122 it forms the gear ratio.

S-0-0122 Gear ratio output revolutions

Gear output revolutions. Together with S-0-0121 it forms the gear ratio.

S-0-0123 Screw lead

The screw lead is the distance traveled by the axis while the drive output shaft or the motor shaft performs one revolution.

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

The switching frequency of the power output stage can be adjusted via this parameter. A higher value reduces generation of perceivable noise of the drive controller, but results in lower mechanical efficiency. Only the following values can be set for the parameter: 4000, 8000, 12000, 16000.

P-0-0279 Time resolution for trending

This parameter is used to set the time interval during which measured values for force and position are recorded. By default, values are recorded every 4 ms. Only the following values can be set for the parameter: 2, 4, 8, 16, 32, ... , 2n.

P-0-0217 Time constant for analog input filter

The signal of the force sensor can be read with a digital low-pass filter. With this parameter, the time constant of the filter is set in milliseconds. Values from 0.25 ms to 60 ms can be set for the parameter.

At a value of 0.25 ms, filtering is deactivated.

P-0-1321 Force limitation for reference run

Limits the force generated at the fixed stop during reference run. If the specified force is reached, the fixed stop is identified and set as reference point. The value is given in percent of the nominal force of the motor.

Before delivery, this parameter is set to 50%.

Y0015 Activate E-Stop

This parameter can be used to define whether or not the signal for E-Stopis to be evaluated at the connection X31, pin 3. The parameter can be set to either *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 errorF4034 for E-Stop active is set.

Activation of parameter "MFR008 Stop and torque off", automatically sets the parameter Y0015 to the value 0. In this case, deactivating the parameter "MFR008 Stop and torque off" again does not lead to the parameter Y0015 automatically being set to the value 1.

Y1004 Maximum speed

The Y parameter Y1004 can be used to configure the maximum speed of the system in m/s. When creating the program, the input of this value is validated.

Y1006 Maximum acceleration

The Y parameter Y1006 is used to configure the maximum acceleration of the system in m/s². When creating the program, the input of this value is validated.

Y1007 Maximum force

The Y parameter Y1007 is used to set the maximum permissible force of the system. The value is specified as a percentage of the nominal motor force.

Y1011 Negation of position data

The Y parameter Y1011 is used to set the direction of the drive. This is required if the direction of rotation can be reversed by the gear or the belt side drive. The parameter can be set to either *TRUE* (negation activated) or *FALSE* (negation deactivated).

If a belt side drive is used in the system, the parameter must be set to *TRUE*.

Y1044 Maximum travel, positive

The Y parameter Y1044 is used to configure the position of the software end switch in positive direction in mm. The default value set for this parameter is SMax +1 mm. When creating the program, the input values are validated for ≤ (Y1044 – 1 mm).

This parameter can be used to limit the travel range in the case that interference contours require this or if a zero point deviating from the standard (fixed stop) is set for the system via the *Set reference* function.

Y1045 Maximum travel, negative

The Y parameter Y1045 is used to configure the position of the software end switch in negative direction in mm. The default value set for this parameter is mm. When creating the program, the input values are validated for ≥ (Y1045 – 1 mm).

Y1049 Analog force sensor active

Parameter Y1049 specifies whether a force sensor is present or not. The parameter can be set to the value *1* (sensor active) or *0* (sensor inactive). Option *99* is reserved for internal purposes.

If the force sensor is deactivated, all force values in the system are generated on the basis of the drive torque and the process data are recorded on the basis of the drive torque.

Y1050 Force sensor lead characteristic (tension)

This parameter is used to convert the analog force sensor value to a force in Newton. Itis only taken into account if parameter Y1049 is set to TRUE.

Input of the sensor characteristic is performed in N/V.

Y1053 Force sensor lead characteristic (compression)

This parameter is used to convert the analog force sensor value to a force in Newton. Itis only taken into account if parameter Y1049 is set to TRUE.

Input of the sensor characteristic is performed in N/V.

Y1054 Limit of analog value characteristic

The parameter specifies what signal the force sensor will deliver if a force of 0 N is applied.

This parameter is used to convert the analog force sensor value to a force in Newton. Itis only taken into account if parameter Y1049 is set to TRUE.

Input of the zero signal is performed in V.

VFR002 Tolerance of force sensor

This parameter can be used to set the tolerance for approaching the target force. The value describes the permissible deviation before an error message (typically 041004, cf. the chapter Error catalog) is issued.

The parameter is specified in Newton.

VFR004 Force limitation upon positioning command

This parameter can be used to set the maximum force of the positioning command. As soon as this limit is reached, the travel movement is stopped and error 040304 (see the chapter Error catalog) is displayed. As standard, this function is deactivated if the value is set to 0. If the force limit is exceeded, reverse movement (negative direction of travel) can be performed. To perform forward movement again, this value must either be set to a value higher than the current force value or the function must be deactivated by setting the value to 0 N. The limitation applies to positioning both via field bus and via the control bar.

VFR005 Join to force : Relative minimum speed

This parameter allows for setting a relative minimum speed for the “Join to force” module. The value gives the percentage with which the speed specified in the module is offset. That means that a value of e.g. 60 mm/s specified in the module at a relative minimum speed of 2% results in an absolute minimum speed of 1.2 mm/s.

The default value for this parameter is 1%. If the value is set to 0%, there is no minimum speed, which means that the speed progress is continuously proportional to the difference of time and actual force. Values above 100% result in deactivation of this functionality.

MFR001 Continuous mode

In continuous mode, a program is continuously repeated and run through as a continuous loop. In this mode, the system automatically jumps back to the beginning of the sequence to repeat the cycle after the last program module of one cycle has been completed.

If this parameter is set to 1, the mode is activated. The mode is deactivated if the value is set to 0. Continuous machining can be terminated by pressing the *Stop* button in the interface.

ATTENTION	
	**No data recording in continuous mode** No data is recorded and saved in continuous mode. Only the current values are displayed and updated in the dashboard. Upon manual termination of the continuous cycle, an entry is made in the data base without any corresponding curve data.

MFR002 Actual data filter

This parameter allows for activating and/or deactivating filtering of the current force, position, and speed. If the filter is active (value 1, default setting), the values are only updated while the program is being performed. Deactivation (value 0) will lead to continuous updating of the values on the dashboard and the activated field bus.

MFR003 SafeMotion

This parameter can be used to activate the SafeMotion functionality of an S4 drive controller. However, configuration of the safe input pairs must precisely comply with the specifications.

MFR004 SMO: Pause program

If the SafePause system function is activated (see Settings/Extended configuration), the system behavior will change. If a program is interrupted by selection of SMST, the program is only paused. If SMST is not deselected, the program is seamlessly resumed. The subsequent evaluation is performed by the evaluation elements as usual.

MFR005 Cyclic variables documentation

This parameter can be used to activate/deactivate cyclic variables documentation. By default, the parameter value is 1 (active), which means that the values of the variables used are read upon program start and saved in the data base (press history). If the parameter is set to 0, documentation of the actual variable values for the program sequence is not available.

MFR006 Program persistent after restart

This parameter enables deactivation/activation of whether the last active program is to be loaded or not after the system is restarted. The default parameter value is 1 (active). If the parameter 0 is set, a program must be activated manually after restart.

MFR007 Data recording deactivated

Dummy text (stop and torque off)

MFR008 Stop and torque off activated

This parameter can be used to activate the stop and torque off function. The function is deactivated by default (value 0). Activation (value 1) sets the parameter "Y0015 Activate E-Stop" to 0. If the signal edge is falling at input IX31.3, the fast stop functionality is called up.

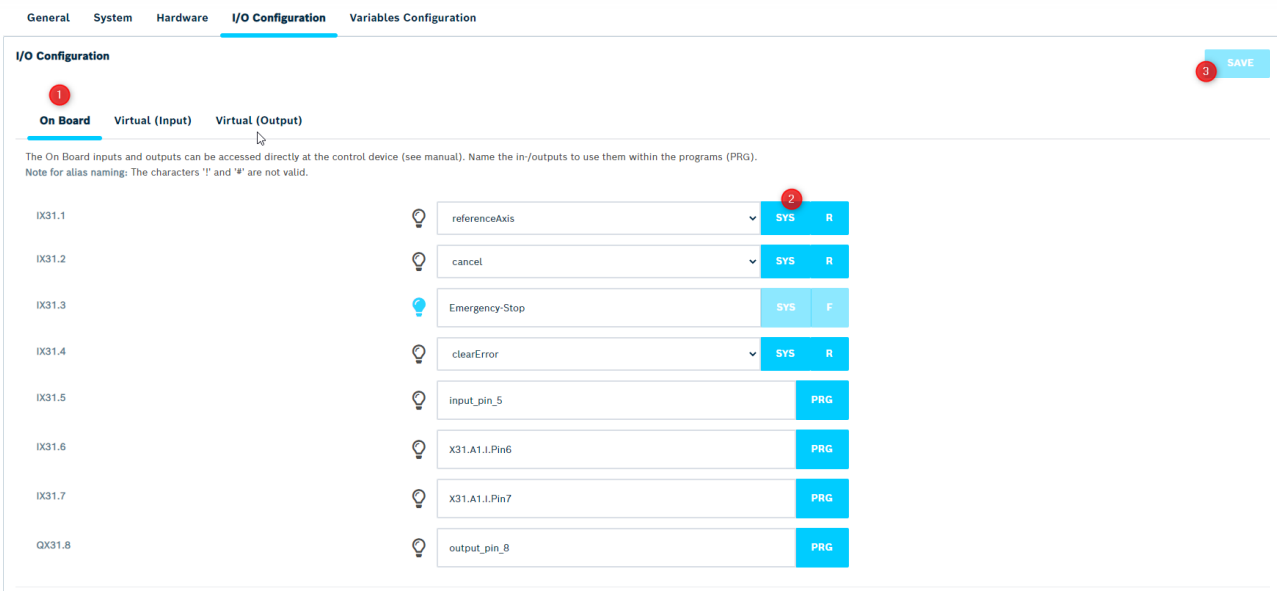
13.4 I/O configuration

The Smart Function Kit has different types of inputs and outputs. They are divided into

digital and virtual interfaces. The digital interface provides 7 digital inputs and 1 digital output. They are located at the drive controller at connection X31.

The inputs and outputs can be freely assigned to system and program functions. For the inputs, it can be set whether they are to react to a falling or a rising edge. The output can be used as system status or program function.

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



43 I/O configuration - on board

ATTENTION

Changing the settings

The settings for inputs and outputs are globally valid. A change of settings will result in a malfunction of the system.

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

System functions

→ Select the *Integrated* (1) tab.

→ Click the *SYS* (2) button for the corresponding input or output to be able to use a system function.

✅ This turns the text field to a selection list and you can select one of the following functions:

Inputs (IX31...):

- Start the current program (start)
- Stop the current program/action (cancel)
- Jog negative (jogNegative)
- Jog positive (jogPositive)
- Start the reference run (referenceAxis)
- Set reference (setReference)
- Tare the force sensor (tare)
- Clear error (clearError)

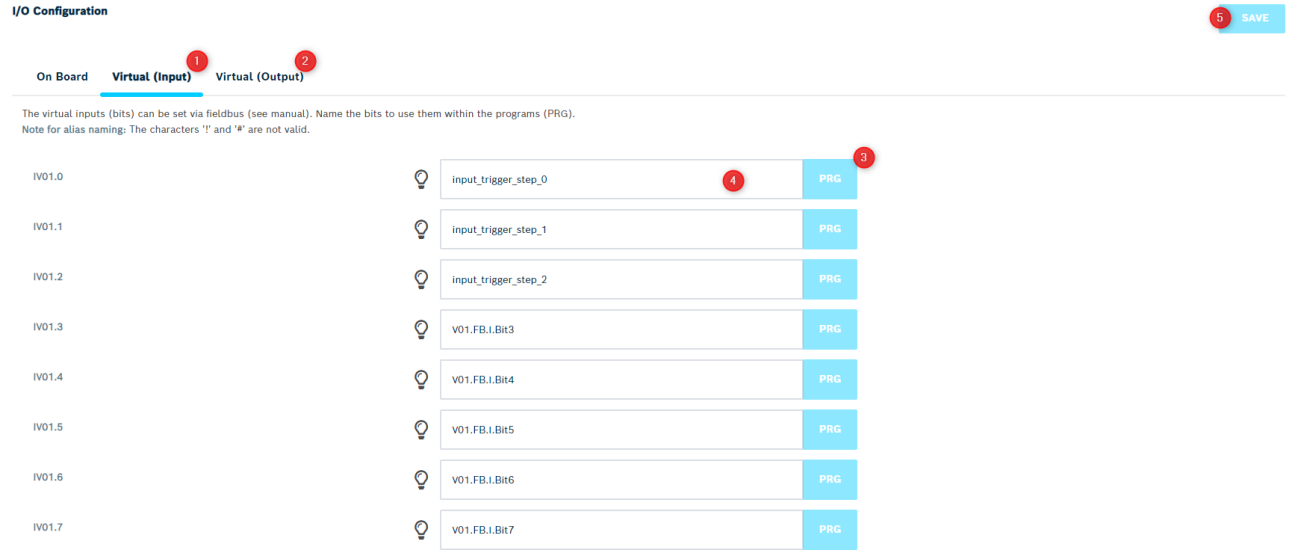
Output (QX31...):

- Program is running (programRunning)

- Parameterization mode active (parameterMode)
- Power is switched on (powerOn)
- Reference has been set (referenceSet)
- Validation active (validating)
- Validation result (validated, HIGH = OK / LOW = NOK)

→ Click *Save* (3) to activate all changes.

Program functions



44 I/O configuration - virtual inputs and outputs

→ Select the *Virtual (Input)* (1) or *Virtual (Output)* (2) tab to define an input or output for a program function.

→ With the selection button, select the *PRG* (3) function (fixed assignment for virtual inputs and outputs).

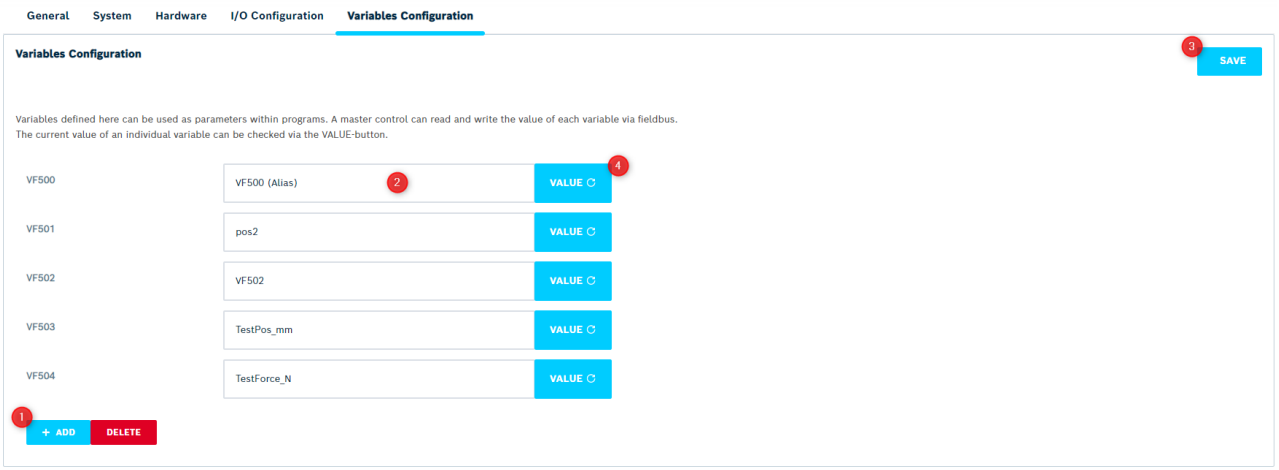
✓ You can then enter an alias name for the input or output in the opposite adjacent text field (4).

✓ This alias name can be used in the program modules for the creation of a program.

→ Click *Save* (5) to activate all changes.

13.5 Variables configuration

For variables to be used, they first need to be configured. The variables rangeautomatically starts at VF500 with a maximum of 100 usable variables.



45 Variables configuration

→ Click + Add (1) to add further variables.

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

Notice
You can freely choose an alias to be assigned to each 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).

→ Click *Save* (3) to save the configuration.

→ Click the *Value (4)* button at the respective variable to read the corresponding variable value.

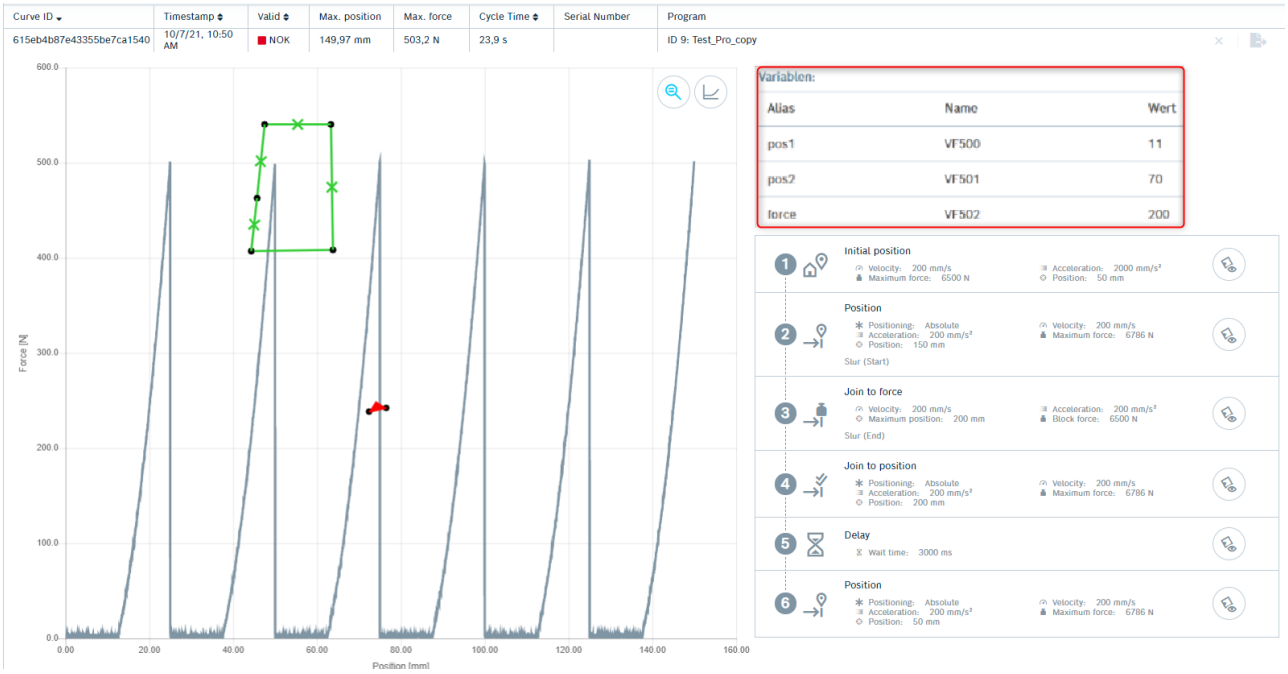
Notice
This function is helpful e.g. for checking the value assignment via field bus.

ATTENTION

Changing the settings

The settings for inputs and outputs are globally valid. A change of settings will result in a malfunction of the system.

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



46 Variables configuration 2

The variables used, including the respective variable 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.

In that case, the table of variables is no longer displayed.

ATTENTION

Invalid variables values

During the runtime, the variables values are subjected to a plausibility check. If invalid values are identified, the program is aborted. The fault message "PLC Error" is displayed.

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

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

→ 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
The variables can be set via the field bus, OPC-UA ,and via the SFK interface.
◊ For detailed information, see the chapter Library.

In case of a program change, the values of the variables are maintained and will *not* be reset to 0.

→ Please note that VF variables are not permanent and will be set to 0 after restart of the system.

14 Safety technology

The software of the Smart Function Kit supports a number of safety functions defined by EN 61800-5-2. As soon as the safety technology is activated, the Smart Function Kit will trigger appropriate reactions, depending on the configuration (e.g. immediate shut down of the drive).

→ When ordering, you can choose between safety option L3 or S4 or the drive controller.

→ Please see the type designation of the drive controller for the safety option of your Smart Function Kit.

Notice
The specified safety functions are only available in “Automatic” mode. Please also note the standard EN ISO 16092.

14.1 Safety option L3

If safety option L3 is selected, the drive controller HCS01 supports the safety function “safe torque off” (STO).

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

◊ For detailed information on the exact functionality of the safety function STO, please see the application description “IndraDrive on-board safety technology “safe torque off” (from MPx-16)“ (R911332633). This can be found in the Media directory on the Bosch Rexroth homepage.

14.1.1 Stop and torque off function

The stop and torque off functionality offers a different configuration of the 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. Afterwards, an error message appears and must be acknowledged before 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, a warning is displayed. In the web interface, the activated emergency off is displayed with a “Safety warning” status instead of “Error”. After a rising signal edge at input IX31.3 (e.g. emergency off is pulled), the drive is automatically switched back to “Drive stop” (AH) mode. The warning disappears and the system status is changed to “ready”. In this operating mode, the axis can be moved again.

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 also set to FALSE.
Please note that parameter Y0015 must be set to TRUE if switching back to emergency stop is intended.

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

14.2 Safety option S4

If safety option S4 is selected, the drive controller HCS01 supports the “Safe Motion” safety technology. In connection with the software of the Smart Function Kit, the following safety functions can be provided:

- Safe torque off (STO)
- Safe brake control (SBC)
- Safe operation stop (SOS)
- Safe stop 1 – time priorization (SS1-t)
- Safe stop 1 – standstill priorization (SS1-r)
- Safe stop 2 (SS2-r)
- Safe monitored deceleration (SMD)
- Safe maximum speed (SMS)

With the listed safety functions, a safety level up to category 4, PLe, SIL3 can be achieved.

◊ For detailed information on the exact functionality of the “Safe Motion” safety technology, please see the application description “IndraDrive on-board safety technology “Safe Motion” (from MPx-18)“ (R911338919). This can be found in the Media directory on the Bosch Rexroth homepage.

Notice
The Smart Function Kit is suitable for use of the S4 safety technology in “stand-alone operation”. Integration into a global safety system, e.g. using PROFIsafe, is currently not intended.

14.3 Connection to peripherals

Both with safety option S4 and safety option L3, it is possible to connect safety-related peripheral devices with the Smart Function Kit. These include e.g. light barriers, emergency-stop buttons or two-hand controllers.

With safety option L3, the peripheral devices are directly wired at the drive controller 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 first, before the energy supply of the motor is interrupted.

✓ This prevents coasting of the axis.

With safety option S4, the safety zone module HSZ01 is included in the scope of delivery. Peripheral devices are connected to the safety zone module. It is connected with the drive controller HCS01, so that signals from peripheral devices are transmitted to the drive controller.

With safety option S4, 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.

→ If the Safe Motion functionalities are used, make sure that parameter MFR003 is set to 1. (Standard setting, if a Smart Function Kit with drive controller option S4 is ordered.)

◊ For detailed information on wiring of peripheral devices with the Smart Function Kit, please see the assembly instructions (R320103194) and the exemplary circuit diagram. Both documents can be found in the Media directory on the Bosch Rexroth homepage.

14.4 Software behavior

The behavior is generally determined on the basis of the selected safety option of the drive controller HCS01. In addition, the desired behavior upon triggering of the safety technology can be configured in IndraWorks and the Smart Function Kit software.

Example:

→ It is possible to choose for a drive controller with safety option S4 whether the program is aborted or paused when safety technology is triggered (SMST actuation).

14.4.1 Behavior without pause function

To achieve the behavior described below, the parameter MFR004 must be set to 0 (standard setting) in the extended configuration:

✅ In case of triggering of the safety technology, the Smart Function Kit will abort any pending movement commands.

✅ A program that is run when the safety technology is triggered will be directly 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 Smart Function Kit will start at the first program step.

14.4.2 Behavior with pause function

To achieve the behavior described below, the parameter MFR004 (SMO: pause program) must be set to 1 in the extended configuration. The pause function is only suitable for drive controllers with safety option S4:

✅ If the safety technology is triggered during a program sequence, the Smart Function Kit will interrupt the program at the point where the safety technology is activated.

✅ The program remains paused until the safety technology intervention is completed.

✅ As soon as the safety technology is no longer active, the program is resumed from the point where it was interrupted.

💡 A new start command is not required for this.

💡 Thus, a previously paused program can still be rated OK.

15 Interfaces

15.1 Smart Function Kit SFK field bus

15.1.1 Introduction

The field bus interface allows for the *Smart Function Kit* to be used as a subsystem in an automation plant.

The following field bus protocols are available for the *Smart Function Kit:*

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

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 field bus protocol can be selected via the *Selection list* in the *System configuration* area. The exact procedure for this is explained in the following chapter.

15.1.2 Hardware

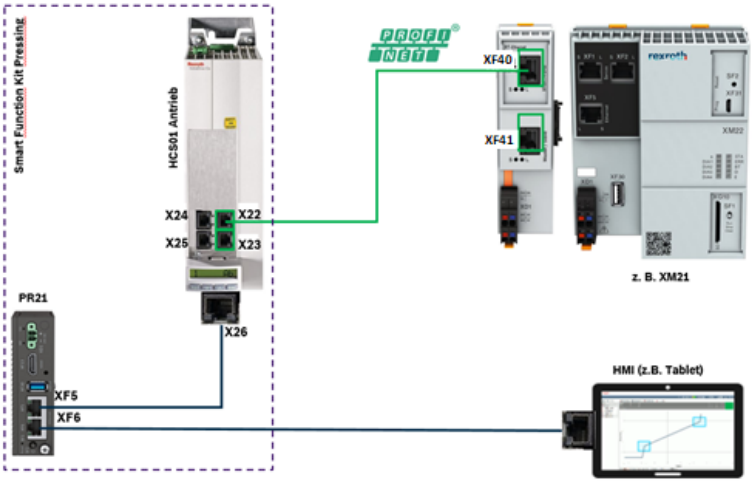
Hardware topology

The following figures show the physical integration of the *Smart Function Kit* (consisting of IndraDrive HCS01 and PR21) into a higher-level automation system. In doing so, care must be taken to ensure correct cabling and use of the appropriate device ports to ensure error-free communication between the devices involved.

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

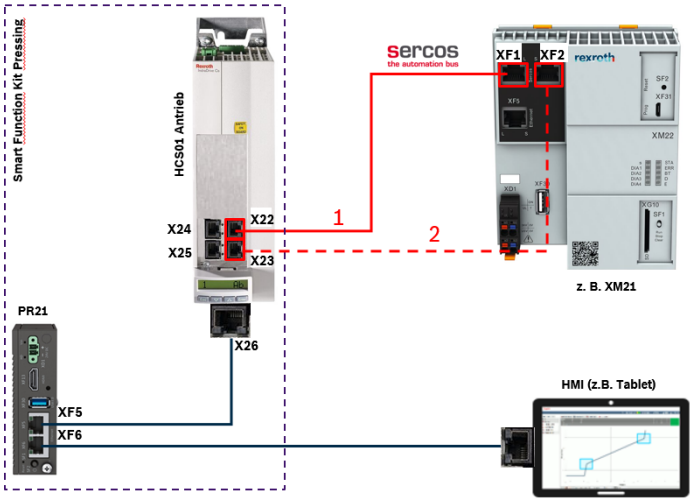
Notice
Third-party hardware can be used for the field bus master. In this case please refer to the additional documentation on the homepage.

1. **Structure of a Profinet field bus**



47 Physical structure of a Profinet field bus

2. Structure and/or cabling of a **SercosIII field bus.**



48 Physical structure of a SercosIII field bus

This field bus can be designed either as a line or ring topology. To increase the fault tolerance against EMC interference and cable defects, the SercosIII master expects a ring topology (cable 1 + 2). This ensures redundancy of the data.

Settings

General requirements

Communication with the *Smart Function Kit* web interface requires a faultless network connection between the HMI (e.g. tablet) and PR21/XF6.

The following requirements apply to the HMI (e.g. notebook, tablet) for connection to the *Smart Function Kit*:

- 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 120px 720px or higher

IP addresses configuration

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

The standard settings are listed below:

IndraDrive HCS01 – X22/X23 (MT):

- IP address: 172.31.254.1
- Subnet mask: 255.255.255.0

IndraDrive HCS01 – X26 (ET):

- IP address: 192.168.1.1
- Subnet mask: 255.255.255.0

Industrial PC PR21 – XF5 (cannot be changed):

- IP address: 192.168.1.100
- Subnet mask: 255.255.255.0

Industrial PC PR21 – XF6 (can be changed via the web interface):

- IP address: 192.168.0.1
- Subnet mask: 255.255.255.0

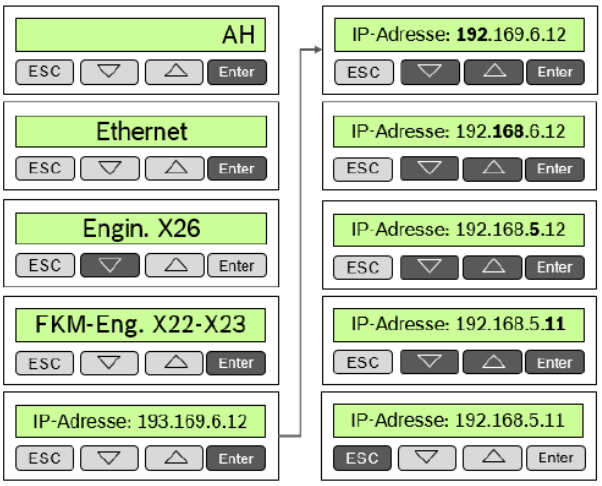
Notice
The IP addresses can be changed or adapted individually for some ports. The corresponding configuration procedure is explained below.
The IP addresses of the ports X22/23 (MT) can be adjusted and changed at the IndraDrive HCS01. For communication via ProfiNet, the desired IP address of the IndraDrive HCS01 **must always be** assigned via the TIA portal. Before assignment via the TIA portal, the IP address of IndraDrive HCS01 must not be identical with the desired IP address. After assignment, the addresses are automatically synchronized to the desired IP address.

ATTENTION

Changing the IP address of port X26

If the IP address of port X26 is changed, the system becomes inoperable.

→ Do not change the IP address of port X26.

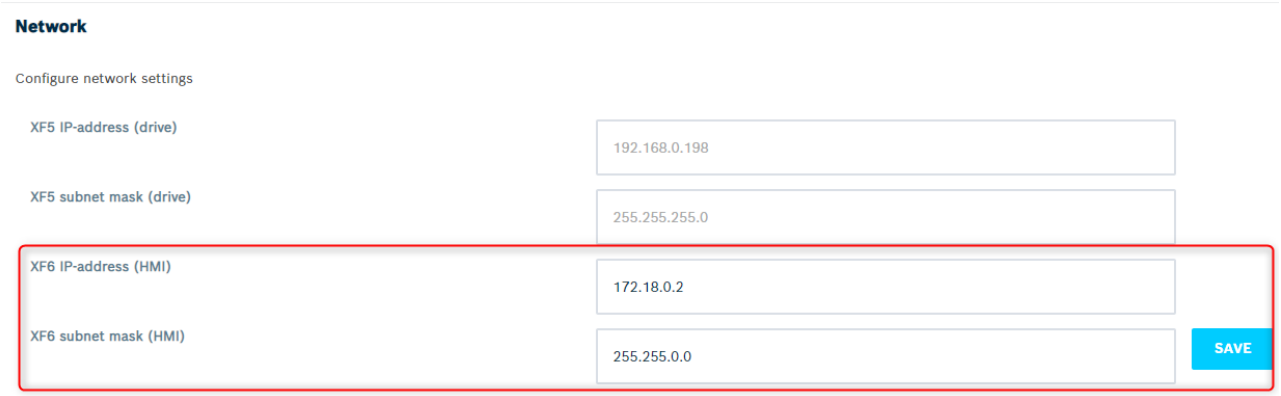


49 Adaptation of the IP address via the operating panel of the drive

The figure shows the procedure using the example of ports X22/23 (MT) via the control unit of the drive.

On the PR21 industrial PC, only the IP address for port XF6 can be changed, which is used to establish the connection between the HMI (e.g. tablet) and the web interface.

In order to adapt or change the IP address/subnet mask, please follow the steps below:



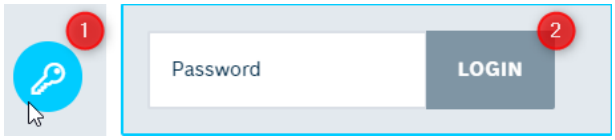
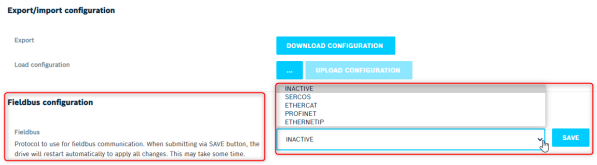
50 Network settings of PR21 - adjusting XF6

→ Open the ***Configuration***.

→ In the *System* tab, change the IP address and the subnet mask for port XF6 under *Network* .

Field bus activation

In order for the *Smart Function Kit* to communicate with or to enable control via a field bus, the corresponding field bus protocol must first be selected in the web interface. To do this, the following steps are required:

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** Web browsers usually have a private mode in which no data such as cookies are stored. If this is activated, there is no need to empty the cache or delete the history. → Enter the IP address in the address bar. *Default IP address: 192.168.0.1* The web interface of the *Smart Function Kit* is opened.
	→ Enter the password *admin* (if not previously changed) and log in.
 Specification of a station name is possible with Profinet and Ethernet/IP. By default, the name is set to to “SPK”. Other station names must comply with the following criteria: • No longer than 240 characters (letters, numbers, hyphen or dot) • 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.), except hyphens • Neither the first, nor the last character should be a hyphen • The designation should not have the following format: n.n.n.n (n = 0...999) • The designation should not start asfollows: port-xyz- (x,y,z = 0…9)	→ Scroll down under *Configuration -> System.* → Select the corresponding type under *Field bus configuration.* → If required, change the station name from “SPK” to the desired name. → Click *Save*.

13 Field bus activation

15.1.3 Field bus configuration

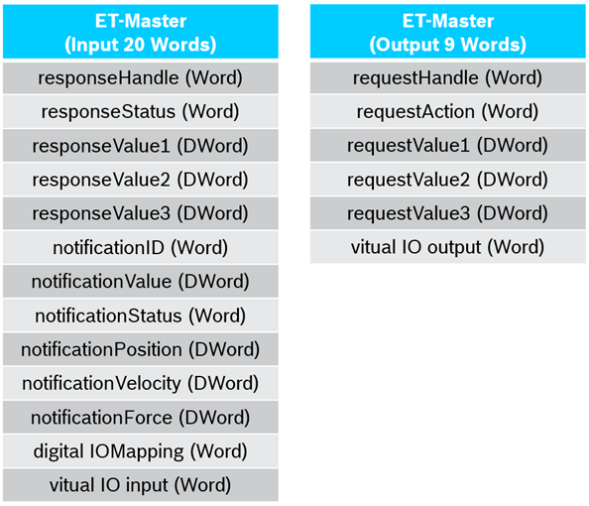
To establish the communication, the master control must be configured correctly. Depending on the field bus protocol, the configuration can look different.

As examples, the configurations for the **SercosIII** and **Profinet field bus** based on an **XM21 control** from Bosch Rexroth are described below. *IndraWorks Engineering* is used as the development environment.

Create variable list

For communication between the *Smart Function Kit* and the higher-level control, specific data is transmitted via the field bus in the form of **WORDS**. These are recorded by the field bus master as inputs or outputs.

The figure shows an overview of the field bus data:



51 Structure of the cyclic configuration

To enable the field bus master to access the corresponding field bus data via functions, it is useful to assign these data to global variables.

A practical implementation consists of creating one global array of the **WORD** data type each for the input and output data.

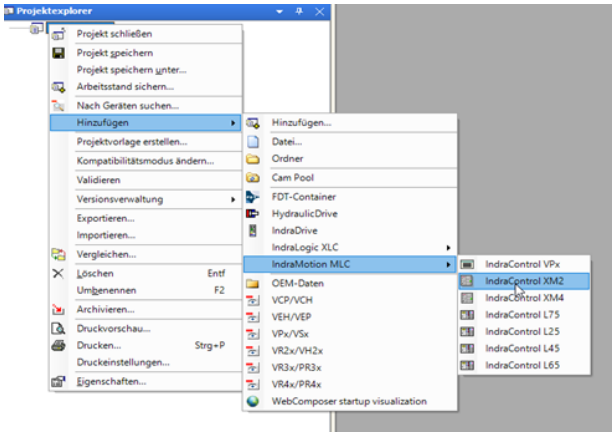
The corresponding assignment of the input and output data of the field bus to the corresponding array elements (mapping) is explained below for the respective field bus protocols.

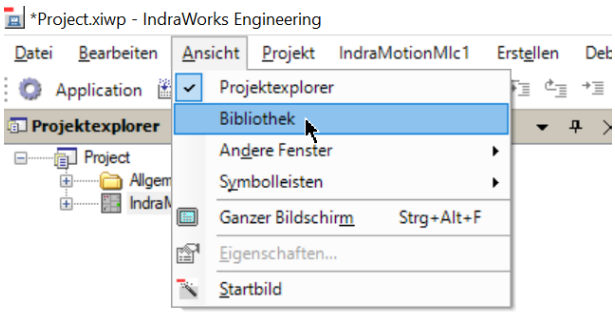
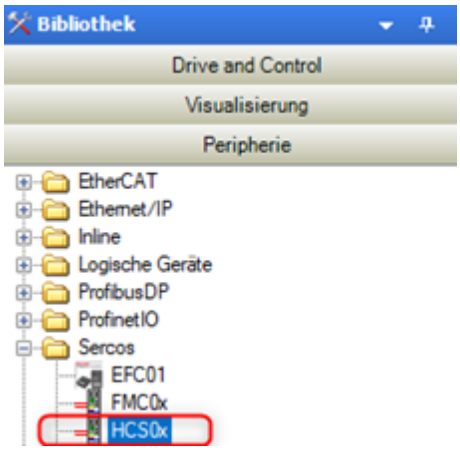
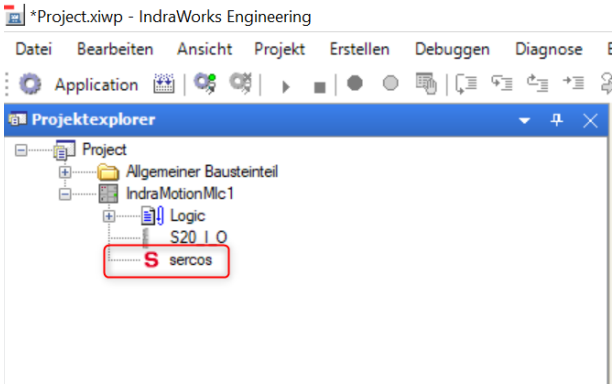
The following listing shows a printout from the global variable list *SFK_GlobalVariables* of the example projects provided on the homepage.

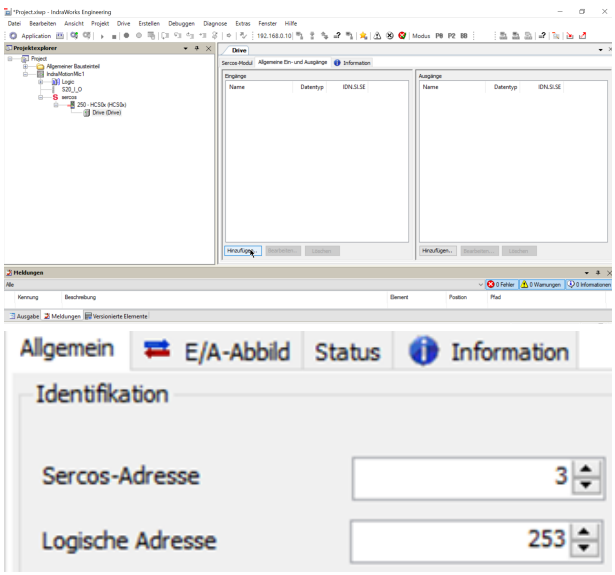
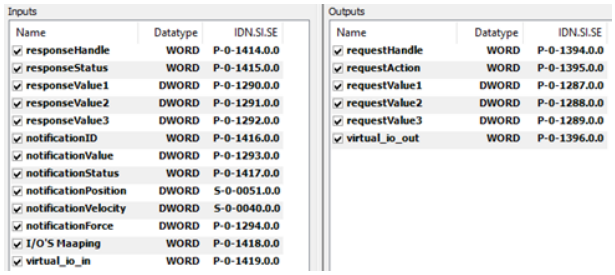
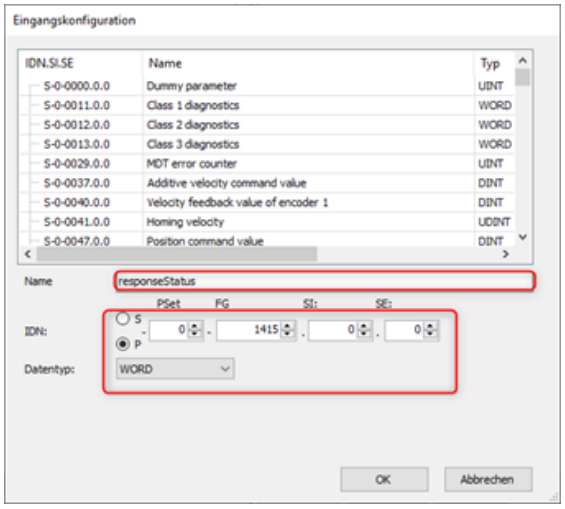
Listing: Excerpt from global variables list "SFK_GlobalVariables"

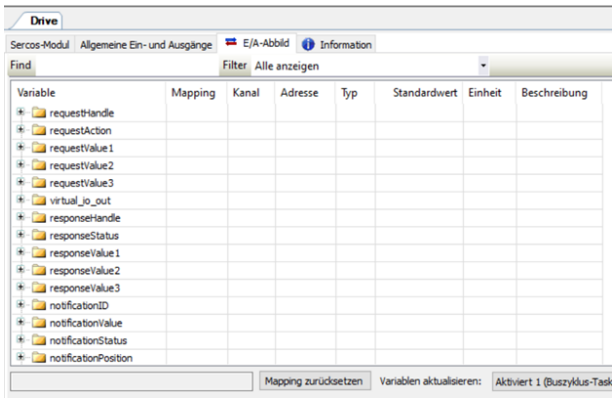
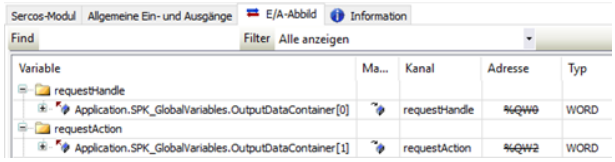
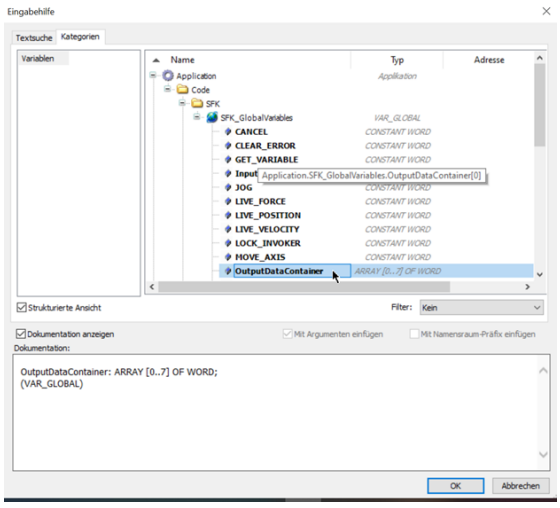
```
VAR_GLOBAL
                ///Cyclic mapping variables
    InputDataContainer: ARRAY[0..19] OF WORD;
    OutputDataContainer: ARRAY[0..8] OF WORD;

END_VAR
```

Sercos field bus configuration

Screens/comments	Action steps
	**First step:** Add the *Smart Function Kit* drive to the field bus master as a Sercos device: → 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 is opened for configuration of the MLC controller (XM2). The following steps of the configuration are performed chronologically. 1. General properties 2. Hardware/communication 3. Extended settings 4. Function packages 5. Interfaces → In every window enter the required inputs and click *Next >>*. → In the last Interfaces window click *Finish* when all inputs have been made. ✅ Configuration is completed and the controller is displayed in the project node.

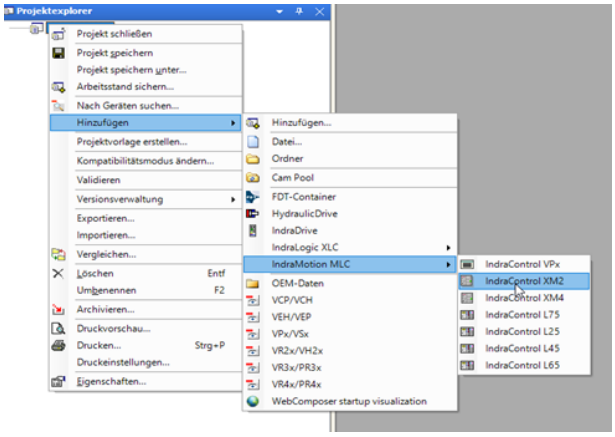
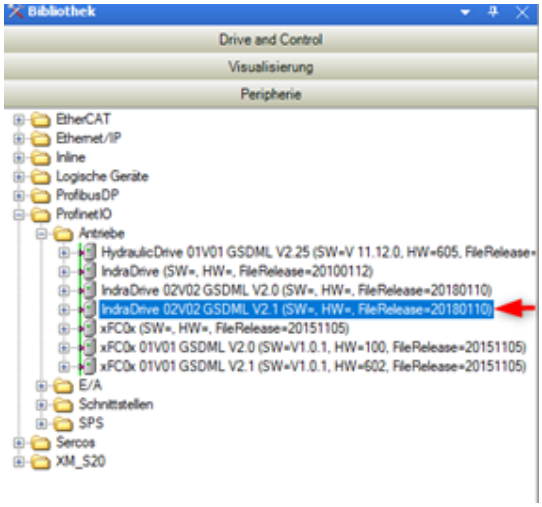
Screens/comments	Action steps
  	Adding a Sercos device (IndraDrive HCS01) → In the *View* tab activate the *Library*. → In the *Library* click the *Peripherals* tab. → Under *Sercos* select the device *HCS0x*. → Alternatively, you can drag the device under the *Sercos node* in the project explorer.

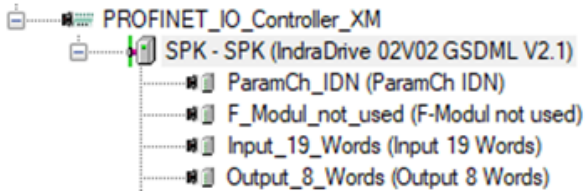
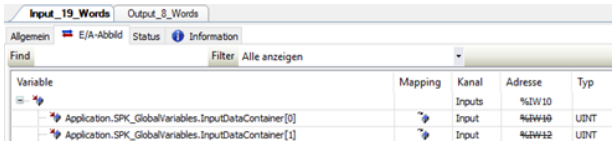
Screens/comments	Action steps
	**Second step:** Comparing the field bus addresses in IndraWorks with the device addresses and adapting them if necessary → In the project explorer navigate to *Sercos*. → *HCS0x.* → Select *Drive*. → In the General inputs and outputs tab, create the parameters.
	→ Change the cyclic inputs and outputs so that they are similar to the example opposite (left).
	→ Click *Add*. ✅ An input form is opened. → Enter the name of the input or output. → Select the data type and enter the corresponding parameters.

Screens/comments	Action steps
 	✅ After creating the cyclic inputs and outputs in the General inputs and outputs tab, they are displayed in the *I/O image*. → Via the “I/O image”, assign the inputs and outputs of the field bus to the internal variables using the + *symbol*. ✅ The field bus master will then access the field bus data via functions. **Notice** After configuration, a restart of the field bus may be required. This restart is carried out with the following changes of the control modes: • Operating mode (BB) • Download mode (P0) • Parameterization mode (P2) • Operating mode (BB)
	→ Select the correct array element from the input or output container.

14 Sercos field bus configuration

Profinet field bus configuration

Screens/comments	Action steps
	**Requirement:** • New project (empty project) has been created in IndraWorks. • Field bus master (here Bosch Rexroth XM21) has been added. **First step:** Add the *Profinet IO controller XM* device. → In the project tree navigate to the *RT_Ethernet...* tab. → Press the right mouse button. → In the menu navigate to *Set device*. → Select the *Profinet-IO-Controller XM* device.
	Second step: Add the IndraDrive HCS01 to the Profinet IO controller as a field bus slave. **First procedure:** → In the project tree navigate to the *Profinet- IO-Controller XM* device. → Press the right mouse button. → In the menu navigate to *Add → Device → Drives.* → Select the *IndraDrive 02V02 GSDML V2.1* device as field bus slave.
	**Second procedure:** → In the *Library* select the *Peripherals* tab. → In the peripherals tree navigate to *ProfinetIO → Drives*. → Drag the IndraDrive 02V02 GSDML V2.1 device into the project tree.
	→ In the corresponding *General* tab, enter *SPK* as the station name. **Notice** During activation of the field bus within the SFK software, another station name can be specified. No distinction is made here between lower and upper case.

Screens/comments	Action steps
	✅ In the project tree, the *IndraDrive HCS01* (Profinet slave) is displayed under the *Profinet IO controller*.
	✅ If Profinet is used, the IndraDrive obtains its IP address automatically from the field bus master (for this reason it is necessary to enter the correct station name). The following standard IP addresses are assigned by the field bus master: • PROFINET_IO_Cotroller_XM (Master): 192.168.2.1 / 255.255.255.0 • SPK (IndraDrive 02V02) (slave): 192.168.2.2 / 255.255.255.0 To enable access to the field bus data (inputs and outputs), the corresponding blocks of the field bus slave must be configured or added as follows: 1. ParamCH_IDN 2. Input_20_Words 3. Output_9_Words
	**Third step:** → Via the “I/O image”, finally assign the inputs and outputs of the field bus to the internal variables. ✅ The field bus master will then access the field bus data via functions.

15 Profinet field bus configuration

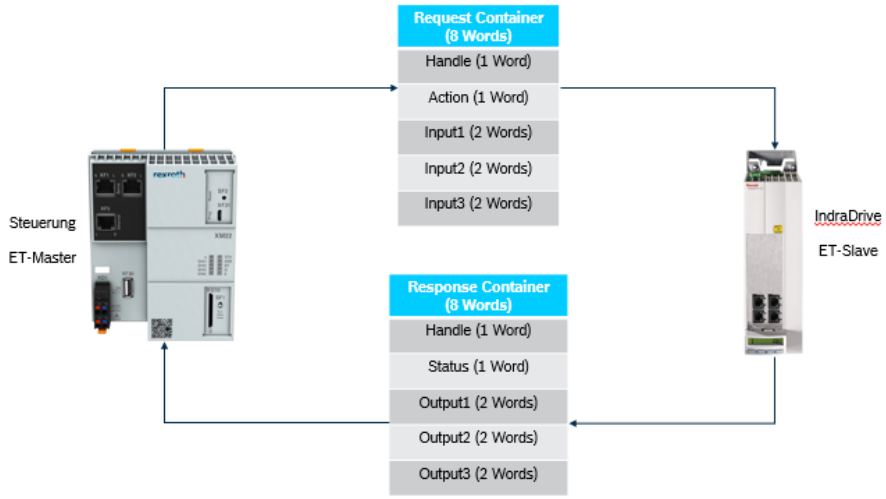
15.1.4 Communication

Control via the field bus is carried out via

- Requests,
- Responses
- Notifications.

For each **request**, the **response** corresponding to the request is sent. **Notifications** are divided into cyclic and acyclic information.

The following figure schematically shows the functional principle of the communication.



52 Schematic structure of field bus communication

This “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 query involving the fields “Action” and “Input 1 - 3”.

Requests

The response to a request is given after the entire command/request has been processed.

The following table contains all available commands that can be sent via the field bus.

ID	Command	Value 1	Value 2	Value 3
1	Start program	(Serial number, digits 1-7)	(Serial number, digits 8-14)	(Serial number, digits 15-21)
2	Set program active	Field bus ID	x	x
3	Positioning	Target position	Travel speed	Acceleration
4	Jog	Distance	Travel speed	Acceleration
5	Delete error	x	x	x
6	Stop movement	x	x	x
7	Restart drive	x	x	x
8	Reserved	-	-	-
9	Tare	Offset	x	x

ID	Command	Value 1	Value 2	Value 3
10	Start of reference run	x	x	x
11	Set system variable	Type	Number	Value
12	Block participant	Participant	Block (1) / Unblock (0)	x
13	Set reference	x	x	x
14	Read system variable	Type	Variable number	x

16 Requests

1 Start program:

With this command, the currently active program is started. The program is run once completely and after completion the corresponding response is provided. The program sequence is considered to be finished when the quality assessment has been completed.

There is also the option of linking a self-assigned serial number with up to 21 digits with the data set of a program sequence. For this purpose, the serial number must be divided to up to 3 REAL values, depending on the total length (see table “Requests”).

Example:

Serial number: '123451234567541'

Value 1 = 4567.541 | Value 2 = 2345.123 | Value 3 = 0.001

2 Set program active:

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

3 Positioning:

The command enables the movement to a defined target position. The desired travel speed profile can be realized via the parameters *travel speed* and *acceleration*. When the target position is reached, the corresponding response is provided.

4 Jog:

With the *Jog* command, a distance relative to the current position can be covered. The desired travel speed profile can be realized via the parameters *travel speed* and *acceleration*. When the target position is reached, the corresponding response is provided.

5 Delete error:

The command acknowledges all errors present on the *Smart Press Kit*. Once the command has been processed by the system, the corresponding response is provided.

6 Stop movement:

The command terminates any currently pending command. If a movement is active, it is aborted and the drive is stopped. When the drive is stopped, the corresponding response is provided.

7 Restart drive:

The command restarts the drive controller.

9 Tare:

Executes the *Tare* command and sets the actual force value to the specified offset. When the command has been completed, the corresponding response is provided.

10 Start reference run:

The command starts the reference run.

> **Notice**
> Please always observe the notes in the Reference run chapter!

11 Set system variable:

This command sets the values of system variables. All available variables are listed in the table below. The system does not enter parameterization mode when changing these values.

◊ For further information, see the chapter Hardware configuration.

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

17 Coding the variable types

Example: Writing the system variable VF500

Arg. 1: 10

Arg. 2: 500

Arg. 3: Value

Variable	Description	Type	Number	Value
VFR004	Set force limitation for positioning command	0	4	Force value [N] 0=inactive (default)

Variable	Description	Type	Number	Value
MFR002	Actual data filter	1	2	0 = Off | 1 = On
MFR005	Cyclic variables documentation	1	5	0 = Off | 1 = On

18 System variable

12 Block participant:

The command blocks or unblocks access for a participant. If the access of a participant is blocked, they cannot access the functions of the *Smart Function Kit*. Blocking of a participant is undone after every restart. Possible participants are:

- Web HMI (1)
- Field bus (2)
- Physical inputs and outputs (3)

Selection of the participant to be blocked is performed via the first input parameter. The corresponding number must be set for the parameter. The second input parameter is used for deciding whether a participant is to be blocked or unblocked. The following values apply:

0 = Unblock participant

1 = Block participant

13 Set reference:

The command *Set reference* sets a reference point at the position where the drive is located. The standard value for that position is zero. However, this value can be globally changed for the system via the extended hardware configuration.

14 Read system variable:

This command is used to read system variables. The variable type must be supplied as the first argument.

The second argument specifies the variable number.

◊ For more information, see the Coding the variable types table

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

19 Coding the variable types

Example:

Read system variable

VF500 Arg. 1: 10

Arg. 2: 500

✔ As response value, the Smart Function Kit supplies the value of the variable VF500 in ResponseValue [1].

Accordingly, the following values for the last press curve can be read via the field bus:

VF600	Max. position
VF601	Max. force
VF602	Cycle time

20 Press curve values

The values are only valid after a pressing process and must be read before a new pressing process begins. The period required for reading the values is not deterministic, since this is performed via the NRT channel.

Notice
A response value of -1 signals that the value is invalid. This may occur, e.g. for cycle time due to abortion of the program.

Status information

With the status information, the states of the *Smart Function Kit* can be queried. This enables targeted control of the *Smart Function Kit* via the field bus master. The following tables show the corresponding commands (ID) and assignments within the status word.

ID	Description	Functional description
301	Active program number	The program number active in the drive is read out.
401	Current position	Cyclically transmitted position is read out
402	Current speed	Cyclically transmitted travel speed is read out
403	Current force	Cyclically transmitted force is read out

21 Status information 1

Bit	Description	Value 1
0 - 2	Reserve	
3	Program active	1 – pressing active, program is being processed Data is recorded (force - travel curve) 0 - No program active No data is recorded
4	Last pressing OK	1 – Last pressing provides result Good 0 - No result

Bit	Description	Value 1
5	Last pressing NOK	1 – Last pressing provides result Bad 0 - No result
6	Sensor tared	1 – Force sensor is tared (offset active) 0 – Force sensor is not tared (no offset active)
7	Press ready	1 – Press is ready and can be moved 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 – Valid response available 0 - Response pending
13	Notification available	1 – Valid notification available 0 - Notification pending
14 - 15	Reserve	

22 Status information 2

Virtual inputs and outputs

The *Smart Function Kit* also provides virtual inputs and outputs to enable communication between the master controller and a program sequence. This enables various application options, such as dynamic stops and/or delays during the program sequence or waiting for external signals from measuring systems or grippers.

For this purpose, 16 bits are available for each direction, which can be written and/or read at any time. To be able to use them, the corresponding program variables must be assigned in the settings of the *Smart Function Kit*. They can then be used when creating the program.

15.1.5 Programming example

Positioning example

To move to a specific target position with the *Smart Function Kit*, the *Positioning* command can be sent via the *Request* channel. When the target position is reached, the result is displayed in the *Response* container.

Handle ID	Command	Value 1	Value 2	Value 3
151	3	50	150	150
Word 1	Word 2	Word 3 Word 4	Word 5 Word 6	Word 7 Word 8

23 Request scheme for container request

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.

When the target position is reached, the *Smart Function Kit* responds with the following response scheme:

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

24 Response scheme for container response

Status = 0 → no error

Status <> 0 → Error. The status value corresponds to the error number of the occurred error, see Error catalog.

Integration/use of example projects

The supplied example implementations can be used for **Indra Works Engineering** or the **TIA Portal** development environment.

The examples serve as an abstraction layer with the aim of making it easier for the user to familiarize themselves with the **request** and **response** scheme. Using function modules based on the PlcOpen Notation simplifies the interface and makes use more intuitive.

◊ Detailed description/integration and functionality in the respective documentation of the example projects

◊ The example projects can be directly downloaded from the user interface. To do this, navigate to the help menu and select the corresponding example project.

15.2 Smart Function Kit SFK - OPC UA + Rest-API

15.2.1 OPC-UA

Overview

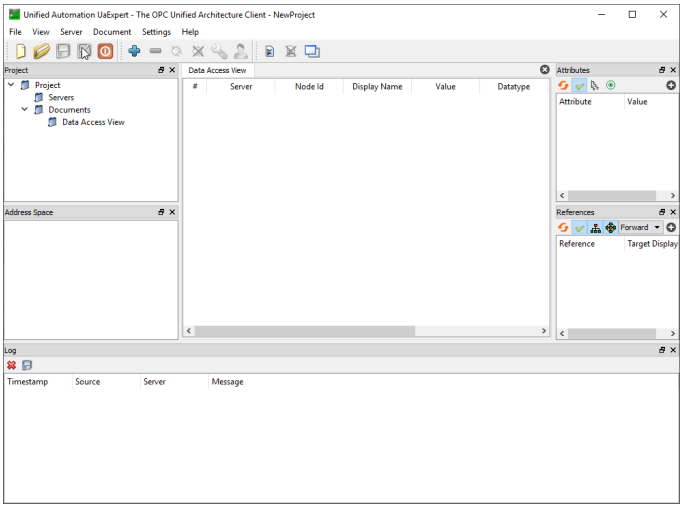
OPC UA (*Open Platform Communications United Architecture*) is an open communication standard used in industrial environments. The standard is independent of platforms and thus allows data exchange regardless of the platform manufacturer.

To ensure easy application and/or integration of this interface, the *Smart Function Kit* is equipped with a corresponding server implementation (data model). In this way the corresponding data model can be requested and used via any OPC client.

Establishing a server connection

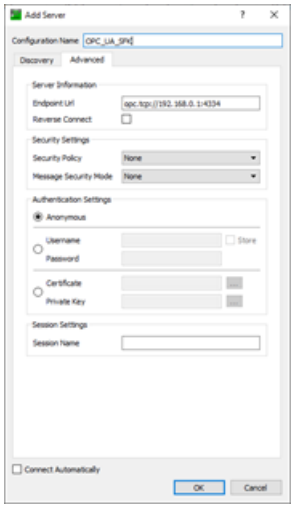
Communication with the Smart Function Kit via OPC UA requires an OPC client.

UA Expert, provided by *Unified Automation GmbH*, has shown to be a functional client.



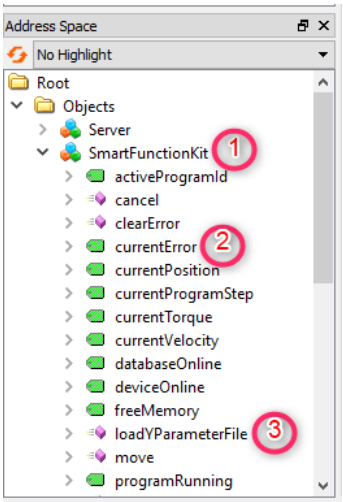
53 UAExpert user interface

Screens/comments	Action steps
Notice Establishing a connection with the Smart Function Kit is done using the binary protocol. There are two protocols for connecting to a server, which can be differentiated by the user by the URL of the server: Binary protocol • *tcp://<Server>* Web service • *https://<Server>*	**First step:** Adding the Smart Function Kit as a server

Screens/comments	Action steps
The required connection configuration is: `URL: "opc.tcp://192.168.0.1:4334"` *Verschlüsselungsart: Keine Verschlüsselung* 	→ 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 Smart Function Kit and provides the desired interactions.

25 Establishing a server connection

Data model/information model



54 Structure of the information model

The data of the *Smart Function Kit* 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).

✅ This enables access to a specific variable (e.g. switching frequency of the drive controller) and/or performance of a specific method (e.g. reading a parameter).

Methods

The integrated server provides the following methods for interaction with the Smart Function Kit.

Method	Function	Input parameter	Type	Description
cancel()	Cancels an active command and aborts it.			
clearError()	Deletes all currently pending errors.			

<table>
<tr>
<td>loadYParameterFile()</td>
<td>Here, there are two different types of parameter files. One is used for configuration of SMC and the other for parameterization of an axis. Differentiation is made via the file designation.

(If the name includes the word “axis”, this file is automatically interpreted for parameterization of the axis.)

The file format of both parameter files is “SCD”.

The two parameter file are used for commissioning and after that they are stored in the User directory on the SD card of the drive.

<u>Example:</u>

Loading SMC configuration: Y_Param_System; Loading axis parameterization: Y_Param_Axis1;

<u>Structure of a parameter:</u>
<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>
<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>fileName</td>
<td>String</td>
<td>File with the Y parameters WITHOUT “SCD” suffix</td>
</tr>
</table>

move()	Moves the Smart Function Kit to the specified target position. Example: *move(50, 1, 30,30);*	distanceOrPosition	UINT32	Target position or distance (mm)
		relative	BOOLEAN	1 → relative 0 → absolute
		velocity	UINT32	travel speed (mm/s)
		acceleration	UINT32	Acceleration (mm/s2)
reboot()	Restarts the complete system			
readHistorie()	Reads the command history from the system and lists the last 20 commands/actions that were issued. This is used for diagnostic purposes in the analysis of command processes.			
readIOs()	Reads the set inputs and outputs of the system with their corresponding properties and returns the information as a 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)			
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-0- xxx

readStatus()	Reads the currently active command and its request context. With this information it can be analyzed which command or request was issued to the system from which source. The output of the function is a string with the corresponding information. The function is used for diagnostic purposes.			
readSMCVariable()	Reads a system variable of the underlying SMC technology package and feeds back the content of the variable as a string. Example: Tolerance for force sensor *readSMCVariable(VFR002);*	variableName	String	Parameter ID in the form of VFRxxx
readYParameter()	The method reads a system parameter of the underlying SMC technology package and feeds back the content as a string. Example: max. acceleration *readYParameter(Y1006);*	parameterName	String	Parameter ID in the form of Yxxxx
setPMOM()	Allows for manual switching between parameterization and operating mode.	pm	Boolean	1 → Parameter mode active 0 → Operating mode active
setProgrammActive()	Created program can be loaded or activated by transmitting the program ID. 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.			
startProgram()	Starts the currently active program.			
startReference()	Starts a reference run.			

tare()	the force sensor is tared to any desired value. This allows for compensating for the weight of a tool. **Notice** If the force sensor is active, the force can be specified in [N] 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 tared to.
writeParameter()	Writes a specified S or P parameter. Many parameters can only be written in parameter mode. For this reason, before writing the parameter, we recommend changing to parameter mode. Time constant for analog input filter: *writeParameter(P-0-0217,4);*	parameterName	String	Parameter ID in the form of S-0-xxx or-P-0-xxx
		value	String	Value to be written.
writeSMCVariable()	Writes a system variable of the underlying SMC technology package. Example: Tolerance for force sensor *writeSMCVariable(VFR002, 6.5);*	variableName	String	Name of the variable to be read.
		value	String	Value to be written.
writeYParameter()	The method writes a system parameter of the underlying SMC technology package. Many parameters can only be written in parameter mode, so that we recommend changing to parameter mode before writing the parameter. Example: *max. travel range:* *writeYParameter(Y1044, 150);*	parameterName	String	Y parameter ID
		value	String	Value to be written

26 Methods

Variables

The following variables can be read and monitored.

Variable	Function
activeProgramID <String>	Reads the program ID of the currently loaded program.

Variable	Function
currentError <Double>	Displays the currently pending error. Example: E-Stop activated: *999476*
currentPosition <Double>	Current position relative to the zero point of the system.
currentProgramStep <Double>	Current program step of the program being processed
currentForce <Double>	Force applied at the force sensor.
currentVelocity <Double>	Current speed
databaseOnline <Boolean>	Checks whether the system data base is online and if data can be saved there.
deviceOnline <Boolean>	Indicates whether the system is available. It is checked whether a TCP connection can be established between PR21 and the drive controller.
freeMemory <String>	Indicates the available memory capacity of the PR21.
programmRunning <Boolean>	Variables value 1 indicates that the program is currently being executed.

27 Variables

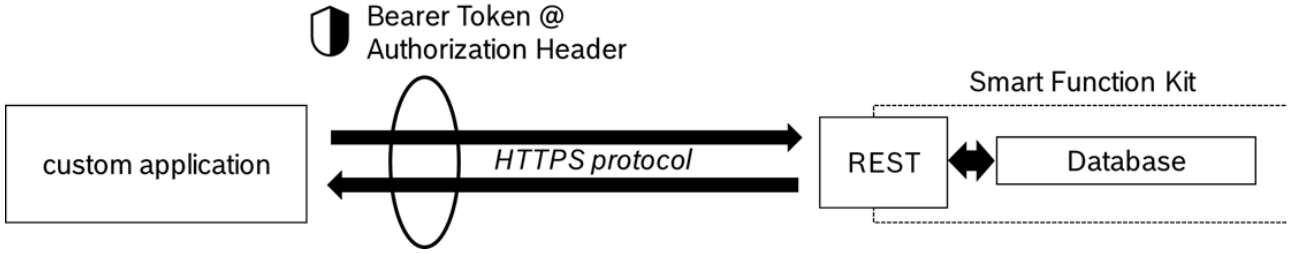
15.2.2 REST-API

This documentation provides an overview of the SFK REST paths.

◊ For detailed information, please see the dashboard in the help menu when clicking the link “API documentation”.

Architecture and basics

The Smart Function Kit (SFK) for press applications contains an application programming interface (API) which allows for external access to various system components. To give you a better understanding of the options this function provides, the functions of the REST technology (Representational State Transfer) used are described in this chapter.



55 SFK API-Architektur

Generally speaking, an API defines how a developer needs to write a program for requesting services from 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 specially intended for browser-based interactions with cloud services and therefore suitable for low bandwidths.

A REST-API uses the HTTP methods defined by the RFC-2616 protocol.

→ Use "GET” to request a resource;

→ Use "PUT” to change the status of a resource (an object, a file, or a block) or to update it;

→ Use "POST" to create a resource;

→ Use "DELETE” to remove a resource.

The REST-API of the SPK provides the authentication service and access to the data base, e.g. curve data or program configurations.

Overview of API access

The REST-API of the *Smart Function Kit* 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.

Implementation

There are various options available for 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

Prior to programming and testing of individual applications, it should be checked that the connection with the SFK has been established. This can be done by calling up the user interface via the browser (Firefox, Chrome). The dashboard should be called up by inputting the IP address (e.g. "https://192.168.0.1“) of the SFK. Additionally, the login data can be checked.

→ In case of problems, clear the cache of the browser used and load it again.

Authentication (POST)

The first step of data extraction is always a form of authentication. It is performed by sending the login data of the user 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 bearer, which should only be used 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 is used for requesting data from the data base. The authentication token (see the “Authentication”) is required to obtain access to the data. The authentication token is sent within the context of the request header.

The response/obtained data are 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.lastModifiedDate;

    return;
}
```

Removal of individual data (DELETE)

The DELETE method is used for removing data from the data base. The authentication token (see the “Authentication”) is required to obtain access to the data. The authentication token is sent within the context of the request header.

Important
The ID required for removing programs is not the same as the program ID shown in the dashboard. You can obtain the correct ID via a request through the REST-API (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 is used for updating data in the SFK data base. The authentication token (see the “Authentication”) is required to obtain access to the data. The authentication token is sent within the context of the request header.

Please note that for some PUT processes other requests are required beforehand or after in order to ensure correct modification of the data. For specific information on the program data model, please see the API documentation on the help page in the dashboard. 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));

}
```

Important
The ID required for changing programs is not the same as the program ID shown in the dashboard.
Access to the correct ID is only possible via a request through the REST-API (GET request).
To be able to perform a change of parameters, the internal "keepLastModifiedDate" parameter must be set to “false”.

Notice
The current program must be uploaded (again) to the PLC to be implemented with the changes performed (via function request or within the dashboard).

RPC/function request

Function requests, e.g. for starting or stopping the current SFK program , can be sent via the WebSocket technology. However, this requires authentication.

Message

```
{ "command": "<commandName>", "params": [], "handle": <Timestamp> }
```

Response

```
{ "handle": <Timestamp> }
```

RPC/function request

```
function start() {
                let ipAddress = '192.168.0.1';                          // Enter your
SFK IPC IP here
    const url='wss://' + ipAddress + '/websocket/socket/websocket';
                const webSocket = new WebSocket(url);
                webSocket.onerror = (event) => console.log(JSON.stringify(event));
                webSocket.onmessage = (event) => console.log(event.data);
                webSocket.onopen = () => {
                    webSocket.send(`{ "command": "startProgram",
"params": [false],
"handle": ${new Date().getTime()} }`);
                }
}

function stop() {
                let ipAddress = '192.168.0.1';                                  //
Enter your SFK IPC IP here
                const url='wss://' + ipAddress + '/websocket/socket/websocket';
                const webSocket = new WebSocket(url);
                webSocket.onerror = (event) => console.log(JSON.stringify(event));
                webSocket.onmessage = (event) => console.log(event.data);
                webSocket.onopen = () => {
                webSocket.send(`{ "command": "cancel",
"params": [],
"handle": ${new Date().getTime()} }`);
                }
}
function start() {
                let ipAddress = '192.168.0.1';                          // Enter your
SFK IPC IP here
    const url='wss://' + ipAddress + '/websocket/socket/websocket';
                const webSocket = new WebSocket(url);
                webSocket.onerror = (event) => console.log(JSON.stringify(event));
                webSocket.onmessage = (event) => console.log(event.data);
                webSocket.onopen = () => {
                    webSocket.send(`{ "command": "startProgram",
"params": [false],
"handle": ${new Date().getTime()} }`);
                }
}

function stop() {
                let ipAddress = '192.168.0.1';                                  //
Enter your SFK IPC IP here
                const url='wss://' + ipAddress + '/websocket/socket/websocket';
                const webSocket = new WebSocket(url);
                webSocket.onerror = (event) => console.log(JSON.stringify(event));
                webSocket.onmessage = (event) => console.log(event.data);
                webSocket.onopen = () => {
                webSocket.send(`{ "command": "cancel",
"params": [],
"handle": ${new Date().getTime()} }`);
                }
}
```

WebSocket notifications

Notifications are sent during the link connection (except “live” messages) or upon value change.

The object type of messages is JSON. A list of possible notifications is attached to this document. We recommend filtering messages by the desired topic.

Example:

JSON data model for the forced notification: { "subscription": { "force": number } }

WebSocket notifications

```
function notify () {
                let ipAddress = '192.168.0.1';                      // Enter your SFK
IPC IP here
                const url='wss://' + ipAddress + '/websocket/socket/websocket';
                const webSocket = new WebSocket(url);

                webSocket.onmessage = function(event) {
                    notification = JSON.parse(event.data);
                    if (notification.subscription.force != undefined)
                    {
                        // Do something with the data
console.log(notification.subscription.force + ' N') ;
                    }
}
```

Export of process curves

Process curves can be searched for via the REST-API interface, for example for export to an external data base. Exporting does not require authentication via the bearer token. 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

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 dataset	JSON-Fields designation	All ICurve parameters	fields=id,createdDate,valid,program,maxPosition,maxForce,cycleTime,customId
sort	No	Native Mongo-DB sort request is ignored if simultaneously used with “offset” **Notice** • Depending on the size of the data base, sorting queries may take longer with increasing number of data records. • This must be observed to ensure a stable long-term connection. • The following entries are optimized for long-term stability: _id	Mongo-DB Query	**Possible Fields:** _id, program, valid, status, customId, createdDate, lastModifiedDate, cycleTime, dataRecordingDisabled, validationTime asc/desc	sort={"customId":"desc"} Sorts the data set in a descending order
q	No	Native Mongo-DB filter query	Mongo-DB Query	All ICurve parameters	q={"customId":{"$eq":"34764378691750"}} The query is filtered by curves with a parameter "customId" (serial number) with the value "34764378691750"

Examples

Please try first to access the web-HMI of the SFK to confirm the certificate for the browser. Otherwise, the queries are rejected due to a security error.

Exporting a curve on the basis of the curve ID:

REST Call

```
GET https://IP_ADDRESS/api/curves/6156a3807608d0799f93d88f
```

Exporting the last/latest curve:

REST Call

```
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):

REST Call

```
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"):

REST Call

```
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:

REST Call

```
GET https://IP_ADDRESS/api/curves?page=0&size=5&sort={"_id":"desc"}&q={"reference":
{"$eq":null}}
```

15.2.3 API documentation

REST

This documentation provides an overview of the SFK REST paths. For detailed information, please see the dashboard in the help menu when clicking the link “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/activities
GET /api/activities

16 Software Update

A software update is always performed by means of an installation file provided by Bosch Rexroth. The current software version is available on the Bosch Rexroth website in the Smart Function Kit product area.

ATTENTION	
	**The system cannot be operated during the update process!** Failure of production. → Stop the running production before performing an update.
	**Data loss in case of errors in the update process!** Loss of production data → Create a backup of the production data before performing an update.

16.1 Installation

→ Please follow the procedure described below for performing an update:

1. Download the new software version from Bosch Rexroth.
2. Save the file on a device with a Windows 10 operating system.
3. Connect your device with the industrial PC PR21.
4. For this, use interface XF6 on PR21, which is normally used by the HMI device.
5. Please make sure that the network settings at the device match the settings of the Smart Function Kit and that you can establish a connection.
6. Unzip the file.
7. Start the executable installation file (.exe).
8. After a brief loading period, the installation program is opened.
9. Enter the IP address of the Smart Function Kit (see IP for HMI display) in the installation program and click *connect*.
10. The installation program then automatically performs the update and subsequently checks the installation of the Smart Function Kit.
11. Once the installation has been successfully completed, you can close the installation program.

16.2 Commissioning the software

After installing the new version of the Smart Function Kit software, the commissioning wizard is automatically displayed upon the next connection with the system. The update can only be successfully completed by recommissioning. No system data will be lost during this process.

1. Now, perform a regular commissioning process as described in the Commissioning chapter.
2. Check the user-defined settings in the extended configuration, since they may have been overwritten by the update.

ACHTUNG	
	**Faulty system parameters after software update!** Damage to the system. Malfunction of the system. → After recommissioning, check all parameters for their validity. → Check the user-defined settings in the extended configuration, since they may have been overwritten by the update.

17 Troubleshooting and fault clearance

17.1 Display errors

If display errors occur in the Web HMI, these can usually be rectified by refreshing the page.

→ Use the *Refresh page* function (depending on the browser used).

→ Alternatively, refresh the page by means of function button F5.

→ If necessary, clear the browser cache as described in chapter Web HMI not available and reload the page.

17.2 Error catalog

Drive errors

All drive errors are listed in the Smart Function Kit in the Error log area.

◊ For more information, please observe the chapter Error log.

◊ Detailed information on the errors can be found in the IndraDrive MPx-20 (DE: R911345607 and EN: R911345608) documentation, which is available for download under the URL www.boschrexroth.com/medienverzeichnis.

17.2.1 System errors

All system-specific errors are listed below. These are not included in the IndraDrive MPx-20 standard documentation and are therefore described in detail here.

For evaluation via field bus, please observe that the system errors must be interpreted as a(n) (unsigned) WORD.

No.	Designation	Problem	Remedial measure
010001	No function or invalid input	A command was triggered, but parameters are missing or their 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. As necessary, save the program again.
020101	Command active, command transition not permitted	A command was triggered, but processing cannot be started due to an active program.	→ Cancel the currently active command via the stop command in the interface, the OPC UA or the field bus. → Wait for the current command to be completed.
030001 - 031901	Internal error		→ Inform service.

No.	Designation	Problem	Remedial measure
040101	Delete error command not possible	The “Delete error” command was aborted with an error.	→ Check that the system is ready. → Wait for the current command to be completed. → Restart the system and try executing the command sequence again.
040101	Command active, command transition not permitted	The “Delete error” command has been triggered, but processing is not possible.	→ Cancel the currently active command via the stop command in the interface, the OPC UA or the field bus.
040201	Reading the SMC variables is not possible	The system requests an SMC variable from the underlying technology library, which cannot be read.	→ Check the variable name entered. → Restart the system and try reading the variable again.
040301	Positioning command cannot be started	A positioning command e.g. absolute positioning, continuous jogging, or a program with positioning command was triggered but cannot be processed.	→ Cancel the currently active command via the stop command in the interface, the OPC UA or the field bus. → Check that the system is referenced. → Check that the power at the system is switched on. → Check that the system is ready for moving (no error pending, not in parameter mode).
040302	Stop command cannot be started	The “Stop axis” command is subordinately performed in various system functions to put the Smart Function Kit in a standstill. An error has been identified during processing of the command.	→ Check the parameterization of the system. → If required, restore the standard values of the extended configuration. → Restart the system and try executing the command sequence again.
040303	Target position was not reached	A positioning command e.g. absolute positioning, continuous jogging, or a program with positioning command was triggered and completed, but the target position was not reached.	→ Cancel the currently active command via the stop command in the interface, the OPC UA or the field bus. → Check that the system is referenced. → Check that the power at the system is switched on. → Check that the system is ready for moving (no error pending, not in parameter mode).

No.	Designation	Problem	Remedial measure
040304	Manual positioning command: Maximum force exceeded	A positioning command e.g. absolute positioning, or continuous jogging was triggered and the defined maximum force was reached or exceeded during processing.	→ Check the value of the parameter VFR004. If required, adjust the value of the parameter (see chapter 0). → Check that the system is correctly tared. If required, tare it to the value 0 N. → Check that the system is ready for moving (no error pending, not in parameter mode). → Check the plausibility of the displayed force value. If required, release any mechanical deadlocks.
040401	Reading the 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.
040501	Restart of the drive not possible	The Restart command was started at the drive controller. However, restart could not be processed.	→ Check if a drive command, e.g. reference run, is active
040601	Reference run cannot be started.	The reference run system command cannot be started.	→ Check that the power is switched on. → Check that the system is ready for moving (no error pending, not in parameter mode).
040602	Referencing not possible with applied force	The reference run must only be started in unloaded condition.	→ Move the Smart Function Kit to a load-free state, if this is not the case already. **Notice** → Use the jog operation in the “Relative movement” mode.
040701	Program could not be activated	An error occurred during switching of a program so that the program cannot be activated.	→ The currently active program is being processed. Try switching to the next program as soon as the system has completed the process.

No.	Designation	Problem	Remedial measure
040702	Timeout while attempting to activate program	A timeout has occurred during switching of a program. The system could not activate the desired program within the expected period.	→ Try switching again. → Contact customer service if this error occurs again.
040703	Timeout after program is activated	An error has occurred during switching of a program so that the program cannot be activated correctly.	→ Try switching again. → Contact customer service if this error occurs again.
040801	Could not write SMC variable	Writing an SMC variable of the underlying SMC technology package is attempted. The write process was aborted with an error.	The SMC variable is write-protected. → Check for correct use of the variable in the SMC documentation.
040901	Error during 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 moving. Cancel the current movement command and try switching again. → Error parameterization in the extended configuration. Enter valid parameter values for the parameters or reset the parameters to default values.
041001	Program cannot be started.	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. Set a reference, if applicable. → Check that the system is ready for moving (no error pending, not in parameter mode). → Restart the system. → Then load the program again via the selection list in the status bar and repeat the Start command.
041002	Error while stopping the program	A program can be stopped via the Stopcommand 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. → Contact customer service if this error occurs again.

No.	Designation	Problem	Remedial measure
041003	Program error: Max. position reached	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.
041004	Program error: Max. force reached	The maximum force (results from the defined target force plus force sensor tolerance VFR002) 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 force. → Check the parameterized dynamics of the overall process and adjust it as required.
		The system could not be stopped in time to safely reach the defined target force value. The system has triggered an emergency stop.	→ Check the parameterized dynamics of the overall process and reduce the travel speed and increase the acceleration in the force module. → Then restart the program.
041005	Program error: Program manually aborted	A program was manually aborted by the Stop command.	→ Check whether the Stop command was initiated by the trigger sources interface, OPC UA or field bus.
041006	Program error: Canceled due to drive error	A drive error occurred during program execution which prevented further processing.	→ Eliminate the drive error and restart the program.
041007	Program error: Could not write Y parameter	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
041008	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.

No.	Designation	Problem	Remedial measure
041009	Program error: Safety technology active	The safety technology was activated during processing of the program.	→ Check the safety technology.
041010	Program error: Window violated, force undershot	The specified minimum force was undershot in the “Check force/ position target window” module during processing of the program. The position was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.
041011	Program error: Window violated, force exceeded	The specified maximum force was exceeded in the “Check force/ position target window” module during processing of the program. The position was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.
041012	Program error: Window violated, invalid force evaluation	The specified minimum force was undershot and the specified maximum force was simultaneously exceeded in the “Check force/ position target window” module during processing of the program. The position was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → Make sure that the minimum force is lower or equal the maximum force.
041013	Program error: Window violated, position undershot	The specified minimum position was undershot in the “Check force/ position target window” module during processing of the program. The force was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.
041014	Program error: Window violated, force and position undershot	The specified minimum force and minimum position were simultaneously undershot in the “Check force/position target window” module during processing of the program. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.

No.	Designation	Problem	Remedial measure
041015	Program error: Window violated, force exceeded, position undershot	The specified maximum force was exceeded and the specified minimum force was simultaneously undershot in the “Check force/ position target window” module during processing of the program. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.
041016	Program error: Window violated, invalid force evaluation, position undershot	The specified minimum force was undershot and the specified maximum force was simultaneously exceeded in the “Check force/ position target window” module during processing of the program. The specified minimum position was undershot. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → Make sure that the minimum force is lower or equal the maximum force.
041017	Program error: Window violated, position exceeded	The specified maximum position was exceeded in the “Check force/ position target window” module during processing of the program. The force was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.
041018	Program error: Window violated, force undershot, position exceeded	The specified minimum force was undershot and the specified maximum position was simultaneously exceeded in the “Check force/position target window” module during processing of the program. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.
041019	Program error: Window violated, force and position exceeded	The specified maximum force and maximum position were simultaneously exceeded in the “Check force/position target window” module during processing of the program. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary.

No.	Designation	Problem	Remedial measure
041020	Program error: Window violated, invalid force evaluation, position exceeded	The specified minimum force was undershot and the specified maximum force was simultaneously exceeded in the “Check force/ position target window” module during processing of the program. The specified maximum position was exceeded. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → Make sure that the minimum force is lower or equal the maximum force.
041021	Program error: Window violated, invalid position evaluation	The specified minimum position was undershot and the specified maximum position was simultaneously exceeded in the “Check force/position target window” module during processing of the program. The force was within the programmed tolerance range. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → Make sure that the minimum position is lower or equal the maximum position.
041022	Program error: Window violated, force undershot, invalid position evaluation	The specified minimum position was undershot and the specified maximum position was simultaneously exceeded in the “Check force/position target window” module during processing of the program. The specified minimum force was undershot. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → Make sure that the minimum position is lower or equal the maximum position.
041023	Program error: Window violated, force exceeded, invalid position evaluation	The specified minimum position was undershot and the specified maximum position was simultaneously exceeded in the “Check force/position target window” module during processing of the program. The specified maximum force was exceeded. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → Make sure that the minimum position is lower or equal the maximum position.
041024	Program error: Window violated, invalid force and position evaluation	The specified minimum force was undershot and the specified maximum force was simultaneously exceeded in the “Check force/ position target window” module during processing of the program. The specified minimum position was undershot and the specified maximum position was simultaneously exceeded. The triggering program step is marked in the dashboard.	→ Check the component against the module parameterization. → Check parameterization of the module and adjust it as necessary. → 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.

No.	Designation	Problem	Remedial measure
041101	Could not read S or P parameter	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 try reading the parameter again. Error parameterization in the extended configuration. → Enter valid parameter values for the parameters or reset the parameters to standard values.
041201	Could not write S or P parameter	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 required, activate the parameter mode. → 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.
041301	Could not load Y parameter file	The SMC technology package is configured automatically during commissioning. 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.
041401	Taring 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 defined offset.	→ Restart the system and repeat the process. → Contact customer service if the error occurs again.

No.	Designation	Problem	Remedial measure
041501	Loading basic parameters in the drive not possible	During commissioning, the drive parameters are reset to their standard settings and then 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.
041601	Could not set reference	The Set reference command was aborted with an 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 reference* command must be done in operating mode. → Stop the system. 💡 No movement (program start, positioning, etc.) may be active when using *Set reference*.
041701	Could not read Y parameter	A Y parameter from the underlying SMC technology package is to be read. An error has occurred that prevents reading.	→ Check the correct notation of the parameter IDN and repeat the process. → Restart the system and try reading the parameter again.
041801	Y parameter could not be written	When writing a Y parameter of the underlying SMC technology package, an error has occurred.	→ Check the correct notation of the parameter IDN and repeat the process. → Check whether the parameter is temporarily write-protected. If required, activate the parameter mode. → Check in the documentation of the SMC technology package whether the value to be written is compliant with the value range of the parameter. → Adjust the value and repeat the write process.

No.	Designation	Problem	Remedial measure
050001	Character string too long, error while writing	An error has occurred in the communication between PR21 and drive controller due to which data cannot be transmitted completely or correctly.	→ Repeat the command sequence leading to the error. 💡 In exceptional situations, it may be the case that the data volume is too high and causes transmission error at the interface. → Restart the system and repeat the command sequence. → Contact customer service.

28 System errors

17.3 Service tool – log file

In order for us to be able to quickly help you in case of errors, please use the service tool and send us the log file.

This file contains the hardware and system configuration, settings of the industrial computer and the system logs of your Smart Function Kit. That gives us a faster overview of your system.

How to generate the log file?

→ Download the latest software update file from our 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.

→ Then enter the IP address of the Smart Function Kit to establish a connection.

→ Select the *Service* option.

✅ The log file is subsequently saved on the desktop.

→ Then send us this file.

17.4 Procedure for troubleshooting

Troubleshooting may only be carried out by authorized, trained and instructed specialists.

ATTENTION	
	**Changes with external programs** System will become unserviceable → Do not attempt to make unauthorized settings on the industrial PC or drive controller using corresponding remote programs, e.g. IndraWorks DS. This may result in irreparable damage to the system!

→ Proceed in a systematic and 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 error.

→ First, get a general idea of how the product works in conjunction with the overall system.

→ Try to find out whether the product worked properly in conjunction with the overall system before the error first occurred.

→ Try to 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 repairs 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?

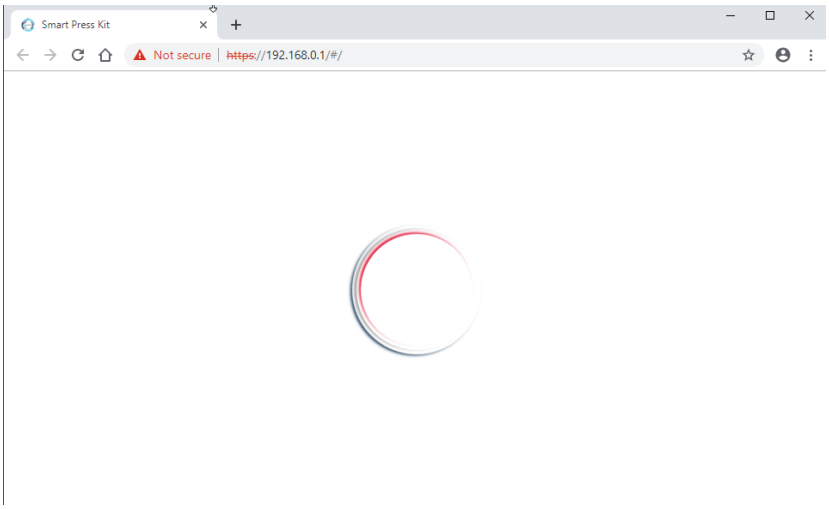
→ Try to get a clear idea of the cause of the error.

→ If necessary, ask the direct operator.

→ If you are unable to remedy the error, please get in touch with one of the contact addresses listed at the url: www.boschrexroth.com/adressen.

17.5 Web HMI not available

If the web browser cannot load the web HMI (even after a short waiting time), this may be caused by a contents display problem. The underlying cause may be software updates, for example. Depending on the browser settings, the contents are kept in the cache and can no longer be correctly displayed or loaded.



56 Web HMI loading spinner

The spinner shown above is permanently displayed and the interface is not loaded.

→ Clear the cache of the web browser to remedy this error.

→ Typically, the keyboard shortcut Ctrl+Shift+Del can be used to open a dialog for clearing the cache in your browser.

◊ Consult the Internet on how to proceed in this case.

Table of figures:

Table of tables: