



# Smart Flex Effector application package Kuka - Operating instructions



# Table of contents

1. About this documentation	4
1.1. Validity of the documentation	4
1.1.1. This documentation applies to the following products	4
1.2. Glossary	4
1.3. Structure of safety information	4
2. Safety	6
2.1. Intended use	6
2.1.1. The following applications are approved for the product:	6
2.2. Improper use	6
2.3. Residual risks	7
2.4. General safety instructions	8
2.5. Personnel qualifications	9
2.6. Personal protective equipment	9
3. Application package setup	10
3.1. Supported robots	10
3.2. Basic structure	10
3.3. Connection of the Anybus Communicator with the Kuka controller	12
3.4. Connection of Anybus Communicator with SFE	12
3.5. Preparation of the EtherCAT coupler	12
3.5.1. Firmware update	13
3.5.2. Installing the configuration	14
3.6. Preparation of the SFE	14
3.7. Scope of functions	15
3.8. Installation of the Kuka option package Smart Flex Effector with WorkVisual	16
3.9. Operation of the application package on the robot	18
3.9.1. Activation of concurrent task	18
3.9.2. Possibility of tool adaptation in relation to the Smart Flex Effector	19
Description of the SFE kinematic chain	19
3.9.3. Possibility to adjust the I/O range of communication	20
3.9.4. Possible change of the message output	21
3.9.5. Cyclic update of the deflection	21
3.9.6. Control of the used interface	21
3.10. Available inline forms (ILF)	22
3.10.1. SFE lock / unlock	22
3.11. Status query	23
3.11.1. Status of the execution of the inline form	23
3.11.2. Status of locking state	24
3.11.3. Status of deflection	25
3.11.4. Command status in general	25
3.11.5. Status of deflection	25
3.12. Status bar	26
3.13. Application examples	27
3.13.1. Examples in the package	27
3.13.2. Example for insert workpiece	27

4. Description of the EtherCAT protocol .....	29
4.1. Conversion from EtherCAT to Modbus RTU in Anybus Communicator .....	29
4.2. Modbus communication travel profile .....	33
5. Service and support .....	34

# 1. About this documentation

The data specified only serve to describe the product. Any references to possible uses are provided merely as a convenience and shall be understood as sample applications or suggestions. Catalog data may not be construed as guaranteed characteristics. The information given does not release the user from the obligation of own judgment and verification. It should be noted that our products are subject to a natural process of aging and wear.

The title page contains an illustration of a sample configuration. The product as delivered can differ from the illustration. The original instructions have been prepared in German.

## 1.1. Validity of the documentation

### 1.1.1. This documentation applies to the following products

## 1.2. Glossary

Abbreviation	Term
SFE	Smart Flex Effector
PWIS	Paint-wetting impairment substances

## 1.3. Structure of safety information



### Hazard classification

The hazards that can occur at the machine are divided into the following classes:

- Danger
- Warning
- Caution
- Attention

### Danger

This warning notice indicates a hazard with high risk. If the safety regulations are not observed, the danger will result in death or serious injury.

 <b>Danger</b>	
	<b>Type and source of danger</b>
	Hazard sequence
	✓ Remedy

### Warning



Type and source of danger

Hazard sequence

✓ Remedy

### Caution



Type and source of danger

Hazard sequence

✓ Remedy

### Attention



Type and source of danger

Hazard sequence

✓ Remedy

## 2. Safety

### 2.1. Intended use

The SFE serves as a sensor-based compensation element for robots and Cartesian systems to compensate for process-related misalignments in the translations X, Y, Z and the rotary motions Rx, Ry, Rz.

Intended use also includes compliance with the applicable legal and safety regulations and the operating, maintenance and servicing conditions specified by the manufacturer. Any deviating use is considered improper. The manufacturer shall not be liable for damage resulting therefrom. Any associated risk shall be borne by the operator.

The product (SFE) is exclusively intended for incorporation into a final machine or system or for assembling with other components to build a final machine or system. The product is intended exclusively for professional use and not for private use.

The SFE is not a safety component in the sense of intended use.

#### 2.1.1. The following applications are approved for the product:

- Mounting to a handling system between a flange plate and a tool, such as a gripper.
- Intended for installation in a machine/system or for mounting on a robot. The applicable guidelines must be observed and complied with.
- The SFE may be used for passive compensation movements and deflection data acquisition.
- The SFE may be unlocked for the duration of handling processes with effective contact of the tool and a workpiece or a device and during the necessary extend and retract processes. The SFE must be operated in a locked state during transfer travel or general movements with higher dynamics. During these process steps, the status query of the interlocking state must be used to ensure that the state is not changed in an unplanned manner.
- The SFE may only be used in compliance with the technical data, see the chapter Technical data. If necessary, this must be ensured by means of further measures.
- The SFE may only be positioned perpendicularly in the handling process (main axis "z" in the direction of gravity).

### 2.2. Improper use

Any use other than that described in the intended use is not in accordance with the intended use and is therefore not permitted.

Bosch Rexroth AG will not accept any liability for injury or damage caused by misuse of the product. The risks associated with any misuse of the product shall be borne by the user alone.

Improper use of the product includes in particular:

- A use of any kind for passenger transportation.
- Misuse of the SFE as a collision avoidance device.
- The movement of the SFE to the end position of the position compensation. The end positions are to be checked in the STATUS LED mode (see STATUS chapter).
- The SFE is a dynamically excitable oscillation system. Since reaching the end positions of the position compensation must be avoided, it is recommended to start with a travel speed of less than 100 mm/s

for setting up application processes in the unlocked state.

- Oscillating movement patterns must be avoided during operation in the unlocked state, as otherwise damage to the device may occur.

## 2.3. Residual risks

Risks for damage, failure or destruction are:

- Detaching of the SFE from the handling system
- Massive force due to incorrect application
- Collision
- Faulty power connection

### Warning



\*Exceeding of the mechanical limit values

Exceeding the mechanical limits can overload and destroy parts of the mechanical systems. Components that come loose in this way can lead to personal injury and damage to property.

- ✓ Only use the SFE within the permitted operating limits (see the chapter Technical data).
- ✓ Do not perform transfer travel or oscillating movements in the unlocked state

### Attention



\*Inactive protection measures

Transmission and processing of faulty sensor data can lead to touching and unexpected system behavior.

- ✓ When using the SFE, machine movement must be limited by means of suitable protective measures.
- ✓ The SFE may only be operated on systems and equipment with the protective measures provided for the system.
- ✓ Only operate the SFE with active protective measures.
- ✓ The system in which the SFE is installed must only be moved with the safety space active.
- ✓ Operate the SFE with inactive protective measures only in control state: "manual with safely reduced speed."

### Attention



#### Plastic bellows TPE – restriction when using with oil and grease

Prolonged exposure of the bellows to oil and grease may cause damage.

- ✓ Avoid wetting and residues and remove promptly.
- ✓ Avoid mechanical damage that causes cracks/holes in the bellows.

### Attention



#### Plastic bellows TPE - No absence of PWIS

Processes in which the absence of PWIS must be ensured could be affected.

- ✓ Do not use the SFE in applications with PWIS absence requirement.

### Attention



#### \*Warm-up behavior

Temperature-dependent behavior can negatively influence the sensor technology during operation due to environmental or application-specific influences.

- ✓ Ensure that the sensor system is working correctly with regular plausibility checks.
- ✓ Compensate for possible changes in sensor settings by locking and unlocking regularly.

## 2.4. General safety instructions

- The SFE has no protective measures against contact with other objects and no protective safeguard in case of defects. Any protective measures must be implemented by the higher-level machine/system.
- Highly dynamic processes can lead to overloading of the SFE and require appropriate safety measures.
- Observe the valid regulations for accident prevention and environmental protection.
- Observe the safety rules and regulations of the country in which the product is used.
- Use Bosch Rexroth products only if they are in perfect technical condition.
- Observe all instructions on the product.
- Persons who mount/dismount, operate, disassemble or maintain Bosch Rexroth products must not be under the influence of alcohol, other drugs or medications which might affect their judgment or slow down their reactions.
- Only use accessories and spare parts approved by the manufacturer in order to prevent endangering people and personal injury.
- Observe the technical data and ambient conditions specified in the product documentation.
- If unsuitable products are installed or used in safety-relevant applications, unintended operating states may occur in the application that can cause personal injury and/or damage to property. Therefore, only use a product in safety-relevant applications if this use is explicitly specified and permitted in the product's documentation.

- The product may not be put into operation before it has been established that the end product (for example, a machine or system) in which the Bosch Rexroth products are installed complies with the country-specific regulations, safety rules and standards for the application.
- Unless otherwise documented, Bosch Rexroth products are intended for operation in local, physically and logically secured networks, with access restricted to authorized persons and not classified according to IEC 62443-4-2.
- Only firmware or a GUI (SFE tool) provided by Bosch Rexroth may be used. It is provided by the Bosch Rexroth service or via the Bosch Rexroth website.

## **2.5. Personnel qualifications**

The activities described in this documentation require fundamental knowledge of mechanical and electrical engineering principles and familiarity with the associated technical terminology. In order to ensure safe use, these activities may therefore only be performed by appropriately trained specialists or instructed persons working under the supervision of a trained specialist. A trained specialist is a person whose professional training, knowledge, experience and familiarity with the relevant regulations enable him/her to assess the tasks assigned to him/her, identify potential hazards and take appropriate safety precautions. A trained specialist must adhere to the relevant technical rules and standards.

## **2.6. Personal protective equipment**

During mounting of the SFE, safety footwear must be worn for safety reasons. All items of personal protective equipment must be intact.

## 3. Application package setup

An application package contains software and hardware components and enables a robot-specific connection and fast commissioning of the Smart Flex Effector. Commissioning on a KUKA robot is described below.

The application package, the manual and the software for configuration (SFE-Tool) can be downloaded from the Rexroth store:

[https://store.boschrexroth.com/Linear-motion-technology/Smart-Flex-Effector?cclcl=de\\_DE](https://store.boschrexroth.com/Linear-motion-technology/Smart-Flex-Effector?cclcl=de_DE)

Sufficient hardware:

- Smart Flex Effector with flange subcircle 31,5 (material number R124300001 → adapter plate can be selected individually, see selector on the Bosch Rexroth website)
- Anybus-coupler (material number R913096488)
- Connection cable (Y-cable) (material number R124500077)
- Sub-D cable (material number R901581243)
- 24 V DC-cable (material number R901581244)
- 24 V DC power supply

Sufficient software:

- Techpackage SmartFlexEffector.kop (install on robot, see this documentation)
- Firmware for Anybus-coupler: Firmware for EtherCAT (also externally available)
- Configuration for Anybus-coupler: To establish a communication between coupler and robot

### 3.1. Supported robots

All robots with the following controller types from the manufacturer Kuka are supported by this application package:

- KR C4, KR C4 compact
- KR C5, KR C5 micro

The functionality (e.g. compensation of individual axes) may differ depending on the robot type.

The supported software versions are KRC 8.6 (KR C4, KR C4 compact) and KRC8.7 (KR C5, KR C5 micro).

### 3.2. Basic structure

The SFE can be connected to the coupler Anybus Communicator (material number R913096488) with the Y-cable from the Programming-Kit (material number R124500077) and the Sub-D cable (material number R901581243) for communication and the 24V DC cable (material number R901581244) for power supply.

The Anybus-coupler and the SFE have to be connected to a 24V DC power supply.

For configuration of the Anybus-coupler an Ethernet network cable is necessary.

For the connection of the EtherCAT field bus and the Anybus-coupler an EtherCAT capable network cable is

necessary.

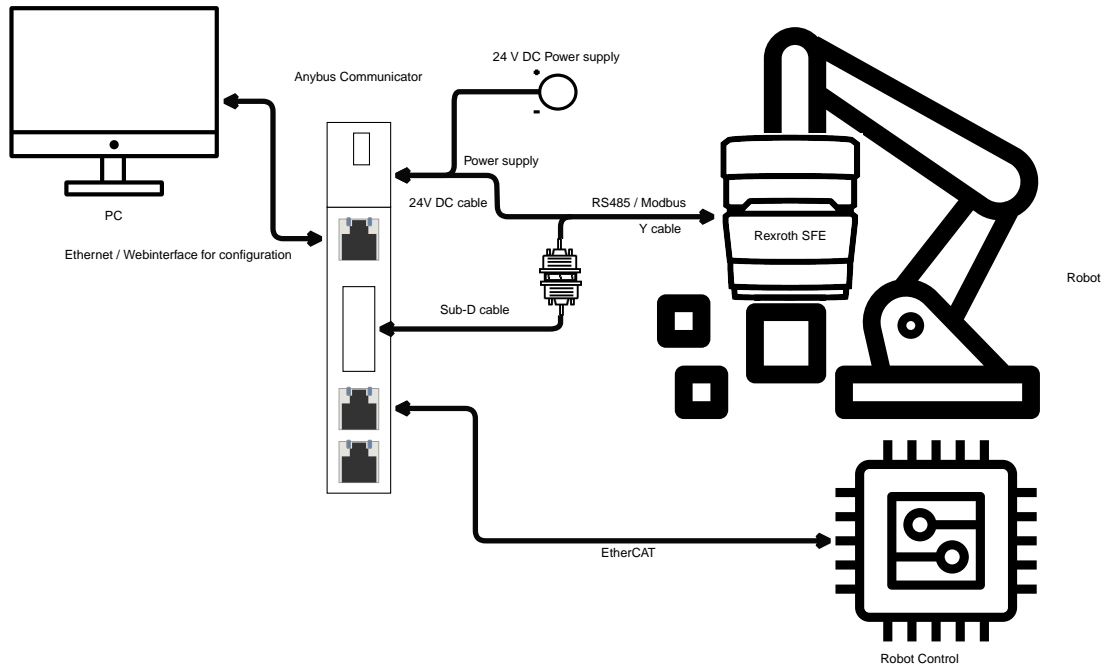


Figure 1. connection of the SFE to a Kuka robot

The SFE is equipped with an RS485 interface by default, which can be addressed with a text-based protocol or Modbus RTU.

The EtherCAT field bus is used as the interface to the robot. The Kuka controller offers native EtherCAT support without additional software options.

The connection between the SFE and Kuka controller is established via the Anybus Communicator from HMS coupler (type ABC3090, Common Ethernet, configuration EtherCAT). The Anybus Communicator controls the SFE via the RS485 interface using the Modbus RTU protocol.

An adapted configuration is available for the coupler, which provides the interface between the SFE and the EtherCAT field bus.

#### Note



The Kuka basic package with the EtherCAT coupler does not require any additional software except for the Smart Flex Effector option package on the robot side.

#### Attention



✓ Also observe the operating instructions for the Smart Flex Effector and the robot.

### 3.3. Connection of the Anybus Communicator with the Kuka controller

Connect the EtherCAT port of the robot (e.g. X44 for Kuka KRC4) with the upper Industrial Ethernet port of the Anybus coupler.

### 3.4. Connection of Anybus Communicator with SFE

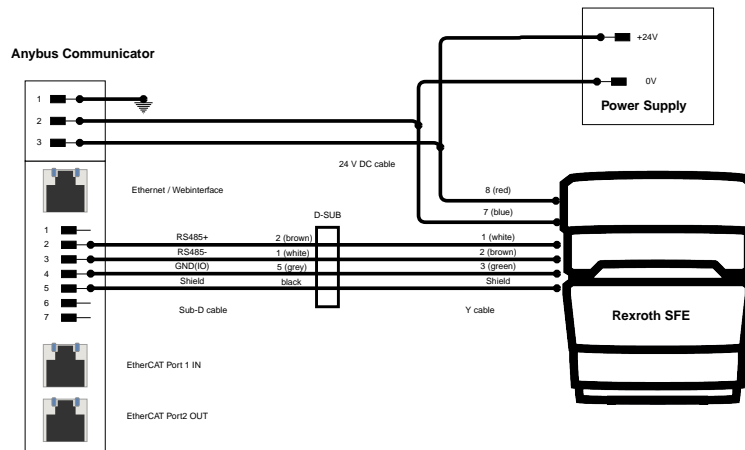


Figure 2. wiring Anybus Communicator / Smart Flex Effector

Table 1. pin assignment Anybus-SFE

Pin Anybus	Pin SFE (M8)	Wire color (M8)	Pin D-SUB	Wire color (D-SUB)	Signal	Description
2	1	White	2	Brown	RS485+	RS485 non-inverted signal
3	2	Brown	1	White	RS485-	RS485 inverted signal
4	3	Green	5	Gray	Ground(I/O)	Ground of IO interface
5	Shield	-	Shield	Black	Functional Earth	Shield
3 (Power Port)	8	red	-	-	24 V+	Supply voltage 24 V
2 (Power Port)	7	blue	-	-	0 V	Supply voltage ground/0 V
1 (Power Port)	-	-	-	-	Functional Earth	Functional Earth (external)

### 3.5. Preparation of the EtherCAT coupler

The Anybus Communicator from HMS (type ABC3090, Common Ethernet, configuration EtherCAT) is used.

The application-specific software must be installed on the Anybus coupler.

During installation and configuration, the EtherCAT connection must be disconnected, otherwise the configuration cannot be confirmed.

To install the software on the Anybus coupler, a network connection must be established between the Anybus coupler (configuration network connector above the serial interface connector) and the development PC.

## Note

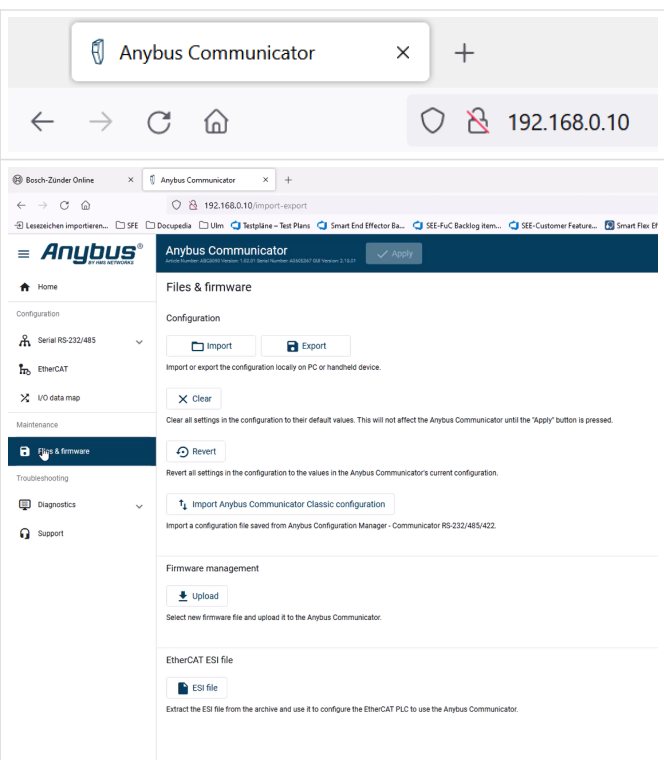


Please also refer to the operating instructions of the Anybus Communicator.

### 3.5.1. Firmware update

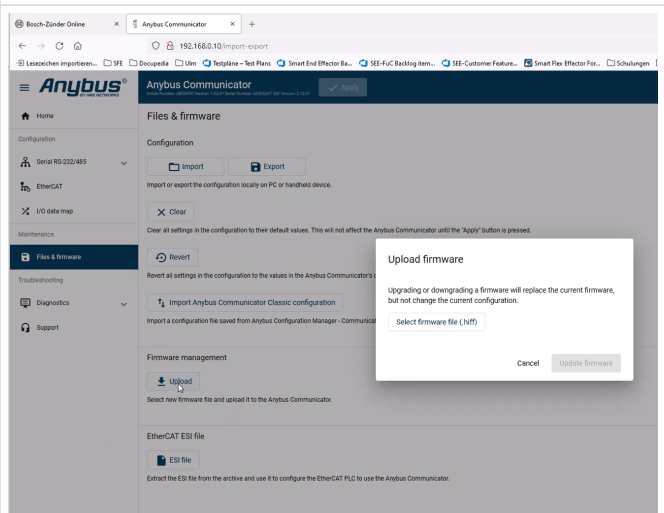
Connection to coupler via browser to the set IP address (default setting of the coupler: 192.168.0.10, subnet mask 255.255.255.0).

In the "Maintenance → Files & Firmware" menu, click on "Update" under "Firmware Management."



Download the actual firmware for EtherCAT from Anybus: <https://www.anybus.com/technical-support/resources/firmware?ordercode=ABC3090>

Select the .hiff file and click on "Update Firmware."

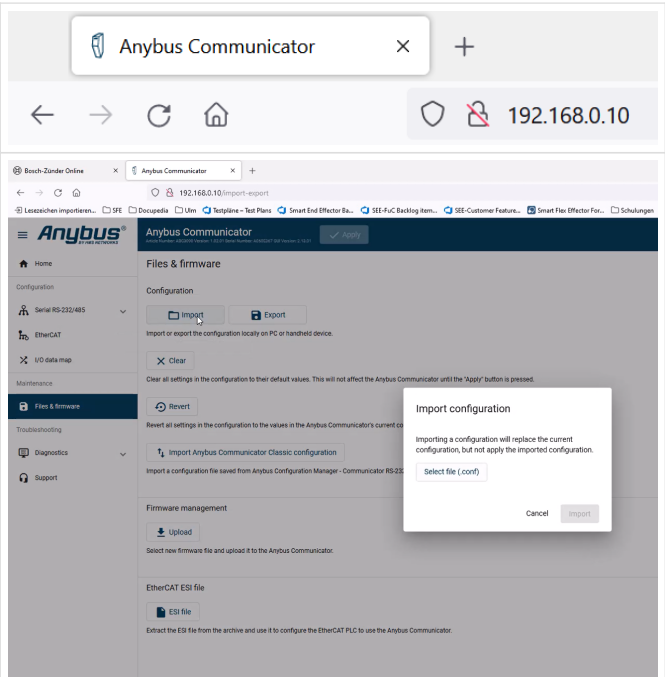


The update may take a few minutes.

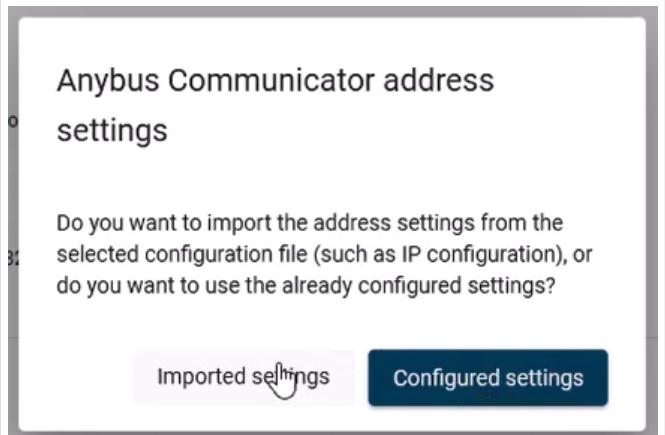
### 3.5.2. Installing the configuration

Connection to coupler via browser to the set IP address (default setting of the coupler: 192.168.0.10, subnet mask 255.255.255.0).

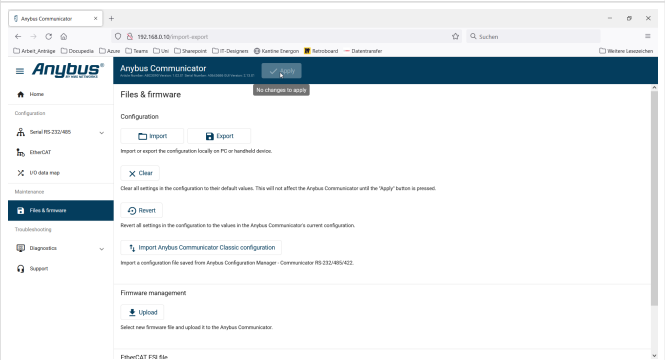
In the "Maintenance → Files & Firmware" menu, click on "Select File" under "Import."



Select the provided .conf file and click on "Import"



Press "Apply"



## 3.6. Preparation of the SFE

The SFE communicates via a text-based protocol in its default setting. To use it in the application package, it must be switched to Modbus RTU.

## Note



To connect the SFE to the PC, use the programming kit that can be ordered separately (order number EU version R124000030, US version R124000031).

Start the SFE tool and connect to the SFE (select the appropriate COM port assigned to the USB adapter of the programming kit)

Switch to the console tab and switch the baud rate with the command "SET;BAUD;115200"

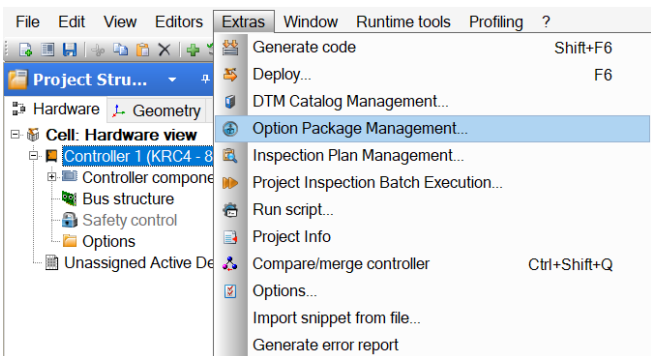
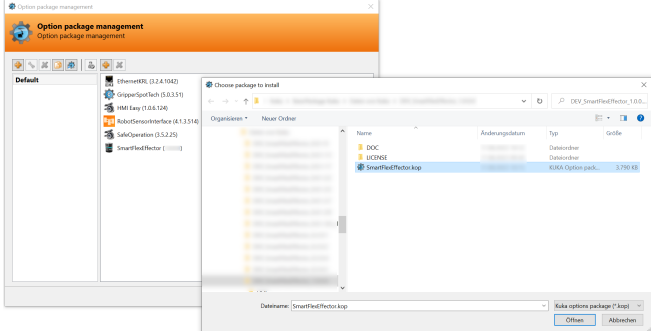
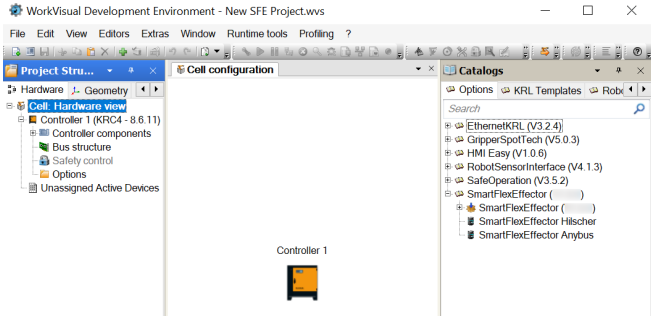
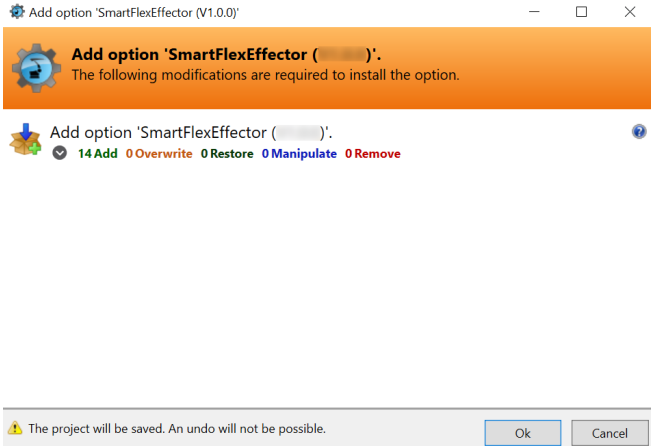
Switch back to the connection tab and press "Modbus Mode." Make sure that the Modbus device address is set to 1.

This is the case in the delivery state. If you have changed the address manually, it must be set to 1 again to use the application package.

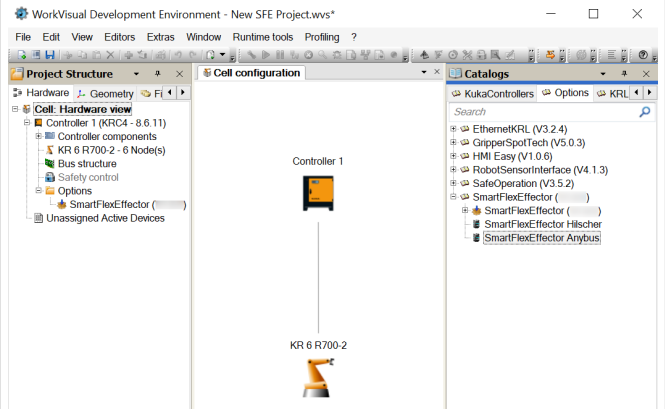
The SFE tool shows that the SFE is operated in Modbus mode. If you want to use the SFE tool, press "Text Mode" to switch back to the text-based protocol.

## 3.7. Scope of functions

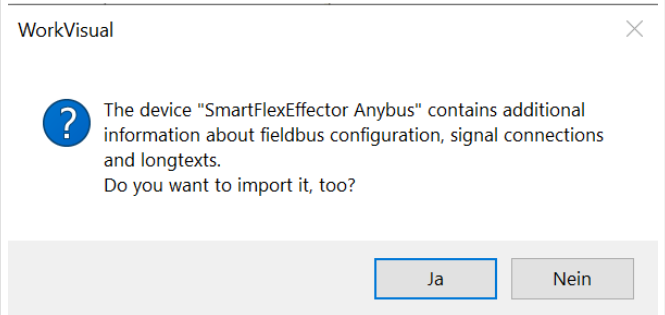
- SFE Lock
- SFE Unlock
- SFE Query interlock status
- SFE Pose
- SFE Pose related to TCP
- Correction of the current robot position with the deflected TCP to zero deflection (robot-specific, depending on technical possibilities)
- Error messages

<p>Menu "Extras → Option package management"</p>	
<p>Press the plus icon and select and install the SmartFlexEffector.kop file</p>	
<p>Pull in the option package in a WorkVisual project</p>	
<p>Confirm with Ok</p>	

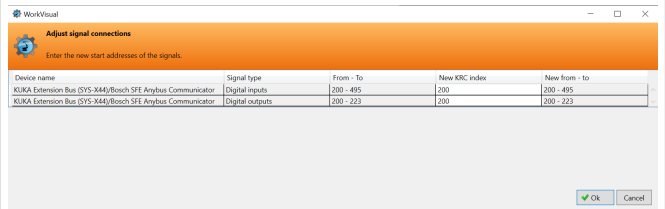
Drag the Smart Flex Effector Anybus element onto the controller (controller must not be switched to active). To be on the safe side, export the existing I/O mapping beforehand, as there is a bug in some WorkVisual versions that deletes the existing mapping when new hardware with mapping is pulled in.



Confirm dialog with "Yes"

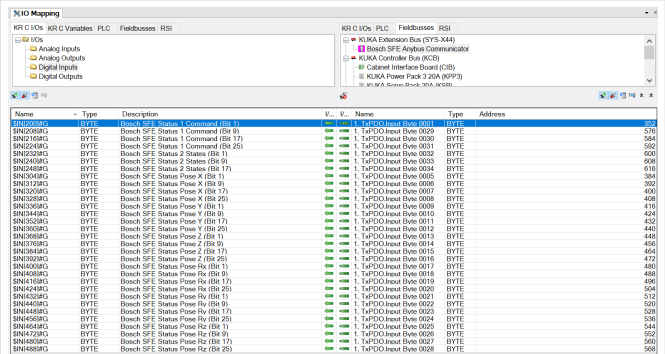


Confirm standard mapping with Ok



Double-click on the control to activate it, then you can see the mapping under "Editors-IO mapping"

Mapping of the robot inputs (ideally done automatically)



## 3.9. Operation of the application package on the robot

### 3.9.1. Activation of concurrent task

The Kuka controller supports concurrent tasks, the submit interpreters.

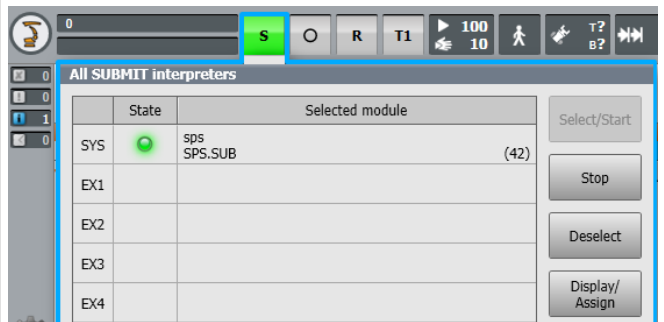
In order to use the deflection data of the Smart Flex Effector cyclically and for the status bar to work for commissioning, the program **SFE\_sps.sub** must be bound to a submit interpreter in the Kuka controller.

To do this, proceed as follows:

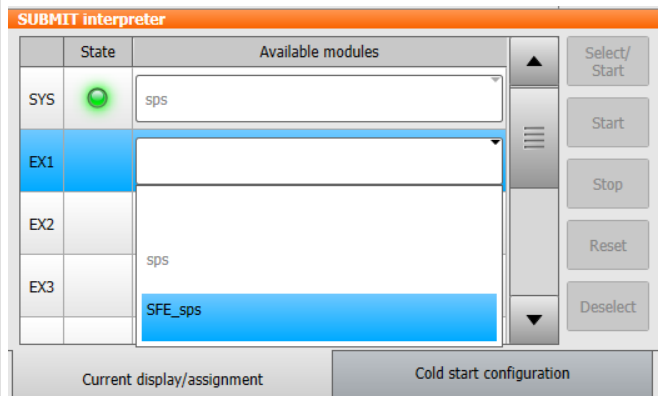
Log in as an expert in the Kuka control system

Press the "S" icon in the top bar.

Press "Display/Assign."

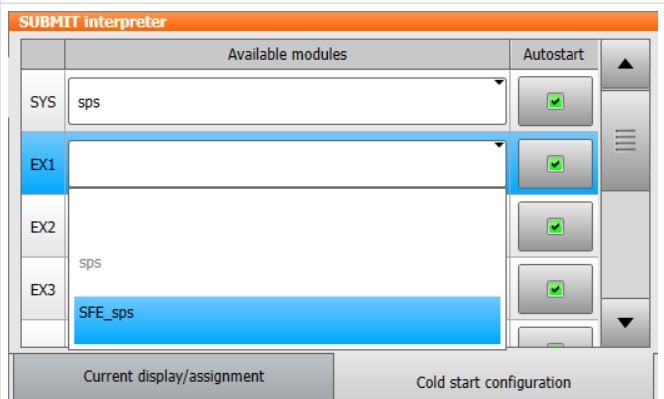
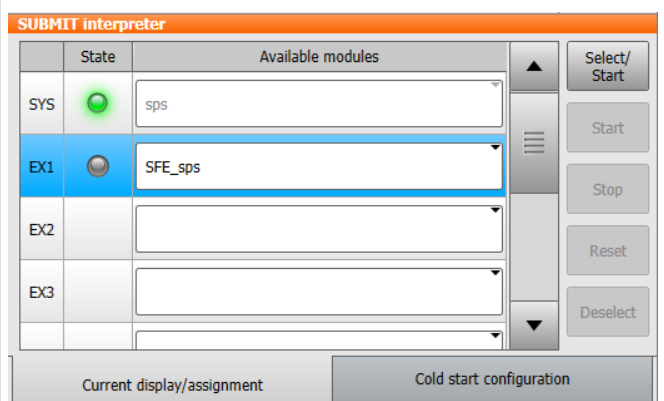


Select "SFE\_sps" with an unoccupied submit interpreter and press Select/Start.



Switch to Cold start configuration tab.

Check Auto Start for "SFE\_sps" (this will start the submit interpreter automatically on restart).



### 3.9.2. Possibility of tool adaptation in relation to the Smart Flex Effector

If you have calibrated or entered your tool, you can optionally enter the kinematics "ROB\_FL\_TO\_SFE-MOUNT" in the "SFE\_USER.dat" in addition to the fixed description of the SFE between the robot flange and the connection of the SFE to the robot, if a distance has been mechanically installed here (see also [Figure 3](#) ).

#### Description of the SFE kinematic chain

1. Shift of the robot flange for SFE attachment at the top (OPTIONAL changeable, default 0, means SFE mounted directly on the robot flange):

```
GLOBAL FRAME ROB_FL_TO_SFEMOUNT={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
```

2. Shift from the mounting plate of the SFE to the moving part of the SFE:

```
GLOBAL FRAME SFEMOUNT_TO_DEVPART={X 0.0,Y 0.0,Z 86.33,A 180.00,B 0.0,C 0.0}
```

3. **POSE**: Measured deflection between **SFEMOUNT\_TO\_DEVPART** and **DEVPART\_TO\_SFE\_FL**

4. Shift of the moving part of the SFE to the flange of the SFE:

```
GLOBAL FRAME DEVPART_TO_SFE_FL={X 0.0,Y 0.0,Z 17.20,A 0.0,B 0.0,C 0.0}
```

5. The "TOOL" is calculated by the currently set tool and stored in "SFE\_FL\_TO\_PART\_PIN\_CALC":

```
GLOBAL FRAME SFE_FL_TO_PART_PIN_CALC={X 0.0,Y 0.0,Z Werkzeuglänge,A -  
180.000,B 0.0,C 0.0}
```

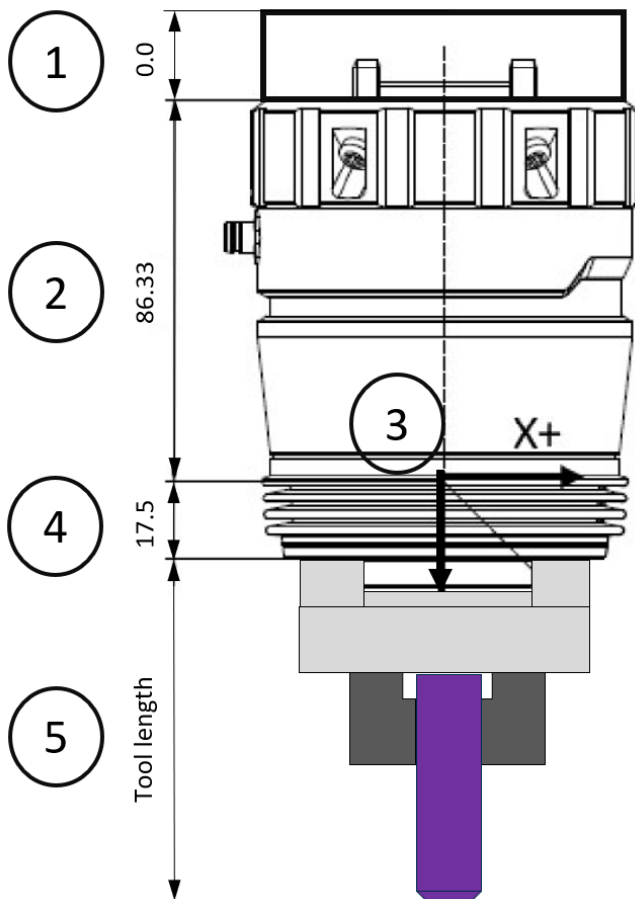


Figure 3. kinematic chain

### 3.9.3. Possibility to adjust the I/O range of communication

When pulling the Anybus coupler into the hardware circuit, there is an option to set the target input and output range to a user-defined range. This changes the circuit in the circuit editor.

Assignment in the KRL program of the controller is not yet automatic. This currently needs to be done manually in the "SFE\_USER.dat" file.

Here, the newly assigned I/O ranges must be adjusted manually. For example, if the start range is from input and output 300, the signals in the file must be increased by 100 accordingly:

Standard:

```
GLOBAL SIGNAL MOD_Comm_Activate $OUT[200] TO $OUT[207]
```

adapted circuit:

```
GLOBAL SIGNAL MOD_Comm_Activate $OUT[300] TO $OUT[307]
```

Standard interface assignment Anybus coupler to KRC controller in the "SFE\_USER.dat":

```
; FOLD ANYBUS declaration
;-----
; HMS Anybus communication
```

```

; Output signals
; Activate communication
GLOBAL SIGNAL MOD_Comm_Activate $OUT[200] TO $OUT[207]
GLOBAL SIGNAL MOD_Lock_Unlock $OUT[208] TO $OUT[223]

; Input signals
; Communication active
GLOBAL SIGNAL MOD_Comm_Active $IN[200] TO $IN[207]
GLOBAL SIGNAL MOD_Pose_State $IN[208] TO $IN[223]
GLOBAL SIGNAL MOD_Lock_State $IN[224] TO $IN[239]
GLOBAL SIGNAL MOD_Last_Command_State $IN[240] TO $IN[255]

; SFE positions

GLOBAL SIGNAL Pose_X_Signal $IN[304] TO $IN[335]
GLOBAL SIGNAL Pose_Y_Signal $IN[336] TO $IN[367]
GLOBAL SIGNAL Pose_Z_Signal $IN[368] TO $IN[399]
GLOBAL SIGNAL Pose_A_Signal $IN[400] TO $IN[431]
GLOBAL SIGNAL Pose_B_Signal $IN[432] TO $IN[463]
GLOBAL SIGNAL Pose_C_Signal $IN[464] TO $IN[495]

```

### 3.9.4. Possible change of the message output

The message output and logging behavior can be customized in the "SFE\_USER.dat":

- Boolean variable "SFE\_ERRMSG" controls the output of information and error messages
- Boolean variable "SFE\_POSE\_TO\_DB" controls the logging of the deflection values in the logbook. The deflection of the SFE, the deflection of the TCP at the current tool and, in case of movement, the correction are saved and stored.
- Boolean variable "SFE\_DEBMSG" controls the output of debug messages

### 3.9.5. Cyclic update of the deflection

The flag "SFE\_Activate\_Pose\_Cyc\_M" is used to control the cyclic update of the deflection of the SFE for the robot program.

By default, the cyclic update is active.

The cyclic deflection values are stored in the "SFE\_Pose\_Cyclic" variable.

The cyclic deflection values related to the current tool are stored in the "TCP\_Pose\_Cyclic" variable.

To explicitly do without it in the program flow, you can specifically switch off the flag "SFE\_Activate\_Pose\_Cyc\_M".

### 3.9.6. Control of the used interface

In the SFE\_USER.dat file, the used interface "SFE\_IFace" can be controlled.

For the Anybus coupler it must be defined as follows:

```
DECL GLOBAL SFE_Interface SFE_IFace=#SFE_Modbus
```

## 3.10. Available inline forms (ILF)

Inline forms are accessed via the soft key bar of the Kuka controller.

Command sequence: "Commands → SmartFlexEffector → Basic → SFE Command"

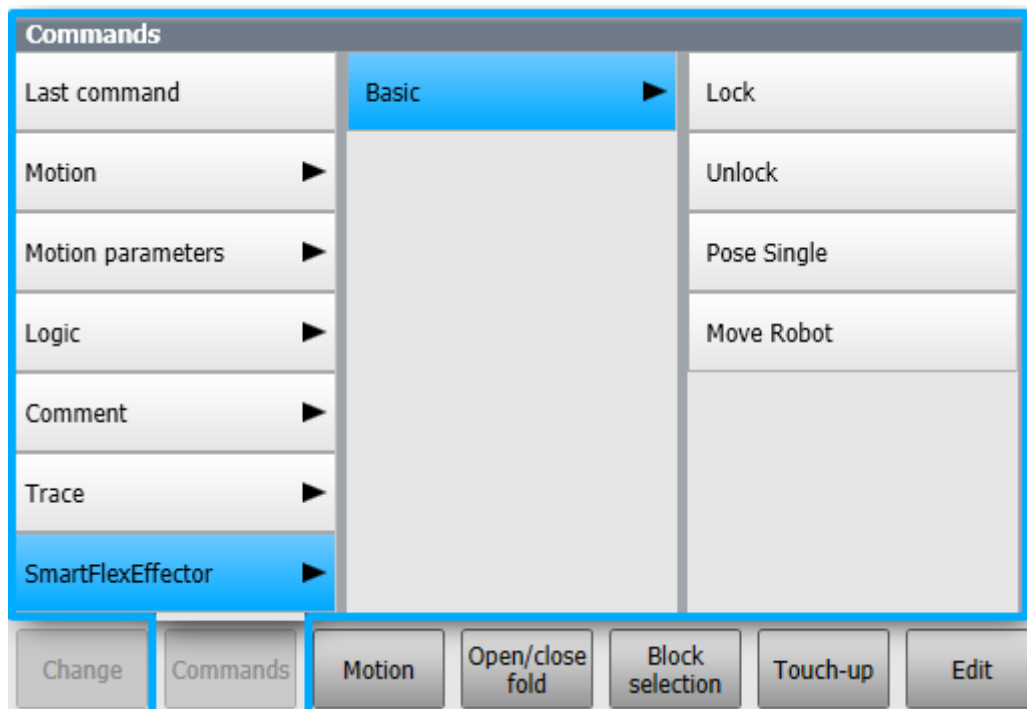


Figure 4. menu view

### 3.10.1. SFE lock / unlock

The SFE can be locked or unlocked with the respective ILF.

ILF Lock	SmartFlexEffector Basic <b>Lock</b>
ILF Unlock	SmartFlexEffector Basic <b>Unlock</b>
ILF Pose single  The ILF "Pose single" is used to read out the deflection values of the Smart Flex Effector.  The <b>measured values of the deflection</b> at the SFE are stored in the globally available frame "SFE_Po-se_Single."  The <b>measured values of the deflection with respect to the current tool (TCP)</b> are stored in the globally available frame "TCP_Pose_Single."	SmartFlexEffector Basic <b>Pose Single</b>

## ILF Move robot

The ILF "Move robot," allows to perform compensation travel with the robot.

Correction takes place with the currently set tool from a previous movement. At least one position with a defined tool must be programmed before using the compensation in order for the function to work correctly.

The ILF has an option window "Degrees of freedom," which is opened via the "Arrow with box" icon "Robot DOF" (degrees of freedom):

If all degrees of freedom are enabled, the robot's TCP remains in position and the robot compensates for the deflection of the SFE to zero by correcting its axes accordingly.

## ILF Robot move parameters

It is also possible to allow only the correction of individual degrees of freedom in the application. Releasing the X, Y and Z axes would perform a correction only in the translative axes. Any remaining deflections in the rotational axes are then only passively compensated by the kinematics of the SFE.



## 3.11. Status query

### 3.11.1. Status of the execution of the inline form

Each inline form returns a simple return via the "SFE\_RET" variable with the following states:

- Function executed successfully: FCTOK = 0
- Function executed incorrectly: FCTBAD = 1

In addition, when the inline form commands are executed, any error that occurs is saved in the "SFE\_Error\_Nbr" variable. This can also be queried in the KRL program if you want to react to errors accordingly.

### Attention



#### Error handling in the program flow

Error handling in the program sequence is to be programmed by the system programmer and is not part of the SFE basic package.

Table 2. error list

Value	Meaning
-------	---------

1	General error
2	Communication error
3	Unknown command number
4	Empty command
5	Unknown command
6	Invalid transfer parameter for command
7	Internal error
8	Calculation error: General
9	Calculation error: Abort tolerance Ftol exceeded
10	Calculation error: Abort tolerance Gtol exceeded
11	Calculation error: Abort tolerance Xtol exceeded
12	Calculation error: No valid result could be calculated within the maximum iterations
13	Calculation error: The calculation does not approach the solution
14	Calculation error: Calculation outside the plausible limits
15	Incorrect checksum
16	Transfer parameter out of valid range
17	Error in the command syntax or error of the number of passed parameters
18	No authorization for the command
19	Protection of the EEPROM due to too frequent writing
20	Error timeout communication
21	Unknown command type
22	Unknown result

### 3.11.2. Status of locking state

To explicitly query the interlock state of the SFE in the program, the following flags are available, which are updated by the cyclic task "SFE\_sps":

- **SFE\_LOCKED** ("TRUE" if the SFE is locked)
- **SFE\_UNLOCKED** ("TRUE" if the SFE is unlocked)
- "SFE\_LOCKED" and "SFE\_UNLOCKED" are "FALSE" in case of an undefined state, which may occur at restart or during a lock operation

In general, the status of the lock state can be queried according to the Modbus register with the variable "MOD\_Lock\_State."

Table 3. content variable "MOD\_Lock\_State"

Value	Meaning
0	Unlocked
1	Locked
2	Timeout
3	Running (lock/unlock active)
4	Error

### 3.11.3. Status of deflection

In general, the status of the deflection can be queried according to the Modbus register with the variable "MOD\_Pose\_State," refers to the calculation errors out of the SFE.

Table 4. content variable "MOD\_Pose\_State"

Value	Meaning
0	Measurement Ok
1	general error
2	Convergence error (trapped in local minimum)
3	Convergence error (maximum iterations)
4	Convergence error (F tolerance)
5	Convergence error (X tolerance)
6	Convergence error (G tolerance)
7	Plausibility error

### 3.11.4. Command status in general

In general, the status can be queried according to the Modbus register with the variable "MOD\_Last\_Command\_State," refers to the calculation errors out of the SFE.

Table 5. content variable "MOD\_Last\_Command\_State"

Value	Meaning
0	Command Ok
1	Invalid values
2	Incorrect mode
3	Write memory
4	Processing error

### 3.11.5. Status of deflection

All variables of the deflection at a glance, the variables are created as "FRAME", i.e. the variables each contain the deflection (in brackets the Kuka-specific designation) in X, Y Z, Rx "(C)", Ry "(B)" and Rz "(A)":

- "SFE\_Pose\_Single" (measured values of the deflection at the SFE. Updated by calling the ILF Pose read out)
- "TCP\_Pose\_Single" (measured values of the deflection with respect to the current tool (TCP). Updated by calling the ILF Pose read out)
- "SFE\_Pose\_Cyclic" (measured values of the deflection at the SFE. Cyclic update by SFE\_sps submit program if flag "SFE\_Activate\_Pose\_Cyc\_M" is active)
- "TCP\_Pose\_Cyclic" (measured values of the deflection with respect to the current tool (TCP). Cyclic update by SFE\_sps submit program if flag "SFE\_Activate\_Pose\_Cyc\_M" is active)

The variables can also be displayed on the robot controller via the menu "Display-Variable-Single."

The screenshot shows a software interface titled "Variable display - Single". It contains several input fields and buttons. On the left, there are three fields: "Name:" with a dropdown menu showing "sfe\_pose\_cyclic", "New value:" with an empty text box, and "Module:" with an empty dropdown menu. To the right of these fields is a large text area labeled "Current value:" with a refresh icon. This area displays the following text: "{FRAME: X 1, Y 0.5, Z -0.3, A 0.4, B 0.6, C 0.05 }". To the right of the text area is a vertical stack of five buttons: "Update", "Set value", and three unlabeled buttons.

Figure 5. TCP deflection cyclic

## 3.12. Status bar

The Kuka control has the possibility to define so-called status keys (4 keys on the control panel at the bottom left).

By clicking on the bar on the side at the bottom left, existing status bars can be switched.

Two functions for SFE locking and unlocking are available for the Smart Flex Effector to simplify commissioning.

Release for operation only available from operator level "User."

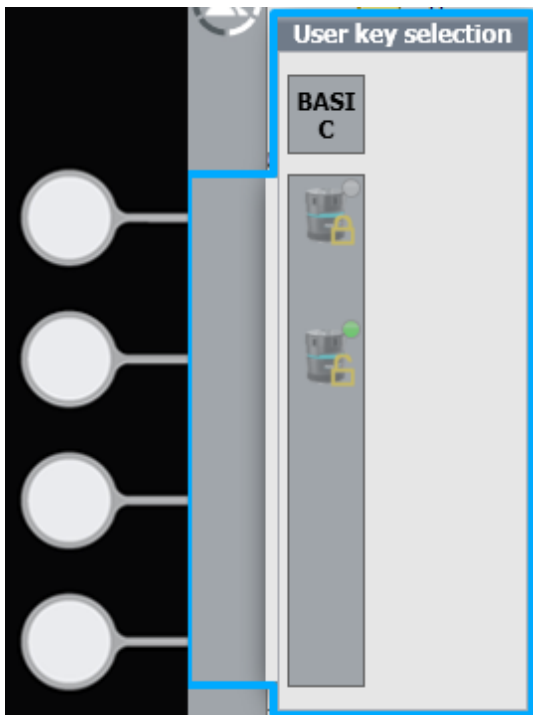



Figure 6. status bar

## 3.13. Application examples

### 3.13.1. Examples in the package

The package includes simple examples in the folder "KRC\R1\Program\KUKA\_SFE\_Inline."

Table 6. sample programs

Program name	Function
SFE_LOCK_IL	Locking the SFE
SFE_UNLOCK_IL	Unlocking the SFE
SFE_GET_POSE_IL	Read out pose of SFE
SFE_MOVEROBOT_IL	Compensation movement of the robot.    A movement to the home position is stored in this program. Check the position on the robot to see if it can be approached.

### 3.13.2. Example for insert workpiece

Example program with possible error evaluation and check if a part was inserted correctly after the correction movement:

```

1 →INI
2 SPTP HOME Vel=100 % DEFAULT
3 ; SFE lock
4 SmartFlexEffector.Basic SFE_LOCK
5 IF SFE_RET == FCTBAD THEN
6 ; Error handling
7 ENDIF
8 SLIN PrePlacePosition Vel=2 m/s CPDAT1 Tool[1] Base[0]
9 ; SFE unlock
10 SmartFlexEffector.Basic SFE_UNLOCK
11 IF SFE_RET == FCTBAD THEN
12 ; Error handling
13 ENDIF
14 SLIN PlacePosition Vel=0.8 m/s CPDAT2 Tool[1] Base[0]
15 ; correct deflection
16 SmartFlexEffector.Basic SFE_MOVE_ROB
17 IF SFE_RET == FCTBAD THEN
18 ; error handling
19 ENDIF
20 ; Read actual deflection
21 SmartFlexEffector.Basic SFE_GETPOSE_SINGLE
22 IF SFE_RET == FCTBAD THEN
23 ; Error handling
24 ENDIF
25 ; Check of Z-deflection
26 IF SFE_POSE_SINGLE.Z < -0.3 THEN
27 ; Error handling SFE still deflected
28 ELSE
29 ; Open gripper
30 ENDIF
31 ; move relative up in Z
32 LIN_REL {Z 50}
33 ; SFE lock
34 SmartFlexEffector.Basic SFE_LOCK
35 IF SFE_RET == FCTBAD THEN
36 ; Error handling
37 ENDIF
38 SPTP HOME Vel=100 % DEFAULT

```

SZ 1 | KRC:\R1\Program\KUKA\_SFE\_Inline\SFE\_EXAM | Ln 1, Col 2

Figure 7. example insert part and correct position

## 4. Description of the EtherCAT protocol

This chapter contains further information about the internal structure of the communication via the EtherCAT bus.

### 4.1. Conversion from EtherCAT to Modbus RTU in Anybus Communicator

The Anybus Communicator provides the part of the Modbus registers required for the application package as process data on the EtherCAT bus.

#### Note



Complete documentation of the SFE's Modbus registers is available online at <https://store.boschrexroth.com/Lineartechnik/Smart-Flex-Effector>.

Two process data objects are provided for input and output data respectively:

Table 7. process data input side coupler

Name	Byte number	Description	Values
Activate communication	1	Modbus communication activation byte 1 (slave 1..8)	0: Modbus communication off  1: Modbus communication subscriber 1 active (SFE)
	2	Modbus communication activation byte 2 (slave 9..16)	0
	3	Modbus communication activation byte 3 (slave 17..24)	0
	4	Modbus communication activation byte 4 (slave 25..32)	0
Set lock/unlock register	5	Lock and unlock low byte	0: SFE Unlock  1: SFE Lock
	6	Lock and unlock high byte	255: Undefined value for initialization → no locking or unlocking

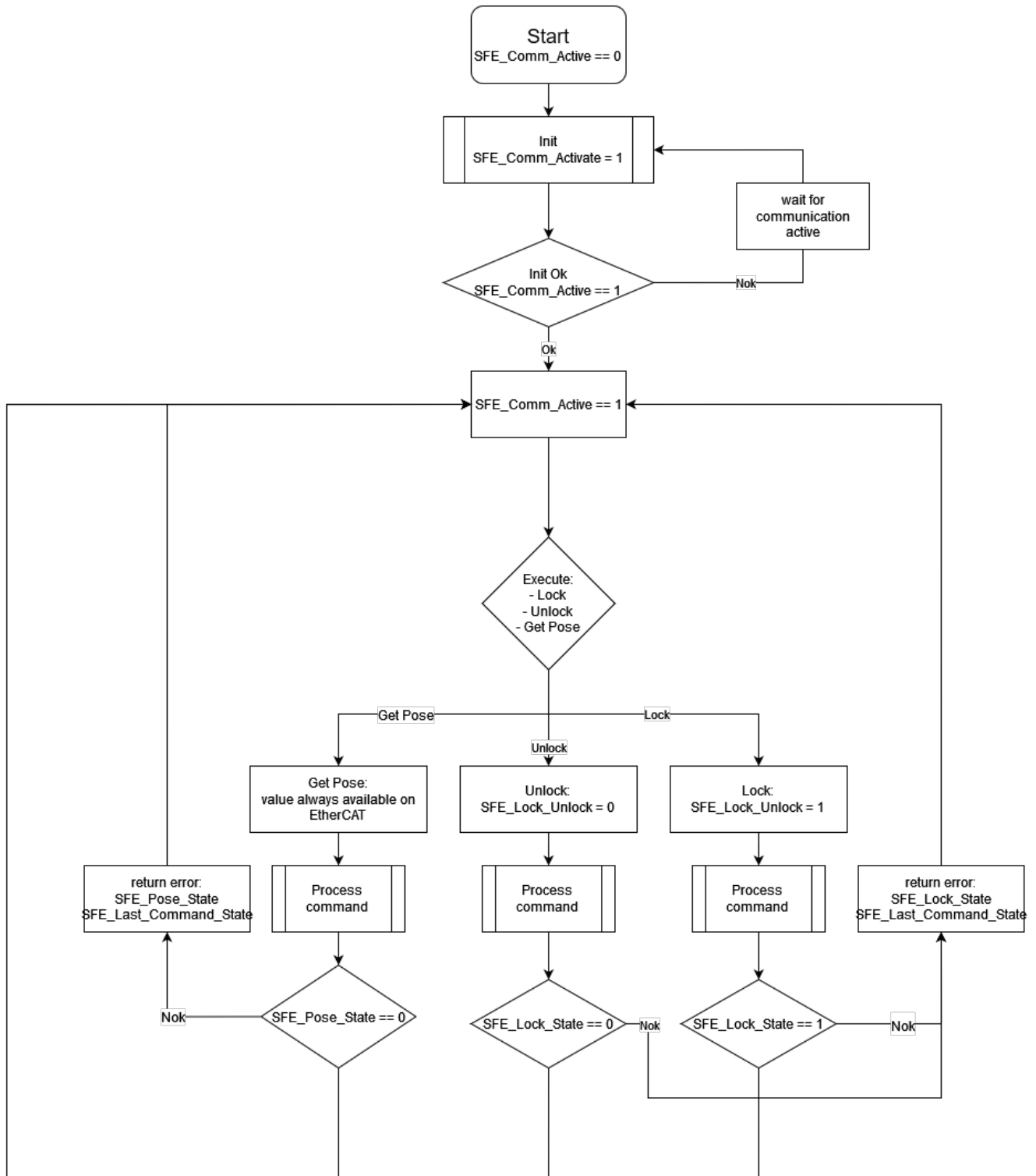
Table 8. process data output side coupler

Name	Byte number	Description	Values
Communication active	1	Modbus communication active byte 1 (station 1..8)	0: Modbus communication off 1: Modbus communication subscriber 1 active
	2	Modbus communication active byte 2 (station 9..16)	0
	3	Modbus communication active byte 3 (station 17..24)	0
	4	Modbus communication active byte 4 (station 25..32)	0
Status deflection X (4 bytes)	5	Deflection X byte 1	Value of deflection X REAL (float)
	6	Deflection X byte 2	
	7	Deflection X byte 3	
	8	Deflection X byte 4	
Status deflection Y (4 bytes)	9	Deflection Y byte 1	Value of deflection deflection Y REAL (float)
	10	Deflection Y byte 2	
	11	Deflection Y byte 3	
	12	Deflection Y byte 4	
Status deflection Z (4 bytes)	13	Deflection Z byte 1	Value of deflection deflection Z REAL (float)
	14	Deflection Z byte 2	
	15	Deflection Z byte 3	
	16	Deflection Z byte 4	
Status deflection Rx (4 bytes)	17	Deflection Rx byte 1	Value of deflection deflection Rx REAL (float)
	18	Deflection Rx byte 2	
	19	Deflection Rx byte 3	
	20	Deflection Rx byte 4	
Status deflection Ry (4 bytes)	21	Deflection Ry byte 1	Value of deflection deflection Ry REAL (float)
	22	Deflection Ry byte 2	
	23	Deflection Ry byte 3	
	24	Deflection Ry byte 4	

Name	Byte number	Description	Values
Status deflection Rz (4 bytes)	25	Deflection Rz byte 1	Value of deflection deflection Rz REAL (float)
	26	Deflection Rz byte 2	
	27	Deflection Rz byte 3	
	28	Deflection Rz byte 4	
Register deflection status	29	Deflection status low byte	0: Deflection status measurement Ok  1: Deflection status general error  2: Deflection status convergence error (trapped in local minimum)  3: Deflection status convergence error (maximum iterations)
	30	Deflection status high byte	4: Deflection status convergence error (F tolerance)  5: Deflection status convergence error (X tolerance)  6: Deflection status convergence error (G tolerance)  7: Deflection status plausibility error
Status register locked/unlocked	31	Locked/unlocked status low byte	0: Status unlocked  1: Status locked  2: Status timeout
	32	Locked/unlocked status high byte	3: Status in movement  4: Status error

Name	Byte number	Description	Values
Status register last command	33	Last command status low byte	0: Last command command Ok  1: Last command invalid values  2: Last command incorrect mode
	34	Last command status high byte	3: Last command write memory  4: Last command processing error

## 4.2. Modbus communication travel profile



At the Anybus coupler, the first 4 communication bytes are activated. These are used to specifically activate or deactivate communication with the Modbus slaves. The total of 4 bytes are assigned bitwise to the connected Modbus slaves. The SFE is assigned the slave number 1 by default. To start communication with the SFE, the first bit of the first byte must therefore be set on the field bus side so that communication with the SFE can start.

## 5. Service and support

We have a dense global service network for fast and optimal support. Our experts will be happy to assist you in any way they can. You can reach us 24/7 – even on weekends and holidays.

### Service Germany

You can reach our service hotline and our service helpdesk under:

Phone: +49 9352 40 5060

Fax: +49 9352 18 4941

E-mail: [service.svc@boschrexroth.de](mailto:service.svc@boschrexroth.de)

Internet: <http://www.boschrexroth.com>

Further notes on service, repair work (e.g. delivery addresses) and training can be found on our website.

### Service international

If you are located outside of Germany, please contact your contact person first. For phone hotline numbers, please refer to the sales addresses online.

### Information preparation

We will be able to help you quickly and efficiently if you have the following information ready:

- A detailed description of the malfunction and circumstances
- Information on the name plate of the affected product, particularly the type code and serial numbers
- Your contact information (phone and fax number and e-mail address)