Sun shading actuators KNX SA SMI ...

Software manual

Der SonnenLichtManager

Valid from 1st April 2019
2024839_0•en

General information

The publication of this document supersedes all previous corresponding documentation. We reserve the right to make changes in the interest of technical progress. Particular care was taken in producing the text and graphics in this document. In spite of this, no liability is accepted for any errors which may exist in this documentation, nor for the consequences thereof.

Legal information

Contact

Smart Building Solutions Customer Centre
Sales, order processing and applied engineering

Tel. +49 9391 20-3750 • Fax -3759
info.steuerungssysteme@warema.de

International
Tel. +49 9391 20-3740 • Fax -3749
steuerungssysteme.international@warema.de

Control Systems Helpline

Tel. +49 9391 20-9317 • Fax -6769
service@warema.de

Smart Building Solutions Sales

Dillberg 33, 97828 Marktheidenfeld
Tel. +49 9391 20-3720 • Fax -3719

Table of contents

Contents

1 Overview

This manual describes the functions of all KNX SA SMI sun shading actuators. Observe the corresponding notes at the start of the chapter that state which functions are available for your device model.

The designations and the number of objects, as shown in the illustrations, may vary depending on the device and software version.

CAUTION
The KNX SA SMI sun shading actuators position the sun shading drives with a high degree of accuracy. After the devices have been operated for an extended period, however, the positioning may begin to stray. To ensure that the devices continue to function properly, the sun shading drive should be calibrated once a week.

1.1 General information on the KNX SA SMI sun shading actuators

The KNX SA SMI sun shading actuators are used for directly positioning mutually independent drives for internal and external venetian blinds, awnings and other sun shading systems.
Up to 16 SMI drives can be assigned to the 16 available SMI channels. Each drive can be controlled individually if necessary.

There are various options for commissioning the device. Please see also *Chapter 5.1 on page 21*.

SMI (STANDARD MOTOR INTERFACE)

The STANDARD MOTOR INTERFACE is abbreviated to SMI and is a unique interface for electric drives. SMI has been developed for connecting drives with integrated, electrical circuits for applications in roller shutters and sun shading systems. It makes it is possible to exchange telegrams via the single interface, from the controller to the drive and vice versa.
Using SMI, drives and controls of various manufacturers are compatible with each other. It is now possible for roller shutter and sun shading system manufacturers, as well as control manufacturers and planners, to combine products from different manufacturers with one another. The SMI interface provides high-grade solutions. The applications for roller shutters and sun shading systems have high requirements for robustness and cost-effectiveness.
The SMI interface has been developed for these requirements.

(Excerpt from the SMI manual, further information via the interface on www.smi-group.com)

1.2 Device models

WAREMA offers the KNX SA SMI sun shading actuators as DIN rail-mounted devices (REG) and in surface-mounted housing (AP).

All devices have the following features:

- Buttons for emergency operation and commissioning
- Bluetooth module for emergency operation and commissioning via a smartphone app (iOS or Android)
- LEDs for the SMI communication display
- **Only AP:** Inputs for external operating elements (push buttons)
  Motor connections fuse-protected via integrated miniature fuses

Actuator	Voltage	SMI outputs	Push button inputs	Housing	Housing width	Art. no.
KNX SA SMI 16M230 SMI REG	230 V AC	16	–	DIN rail-mounted device	3 MW	2022211
KNX SA 16M230.32 SMI AP	230 V AC	16	32	Surface-mounted housing	12 MW	2022210
KNX SA 16MDC SMI LoVo REG	24 V DC	16	–	DIN rail-mounted device	3 MW	2022489
KNX SA 16MDC.32 SMI LoVo AP	24 V DC	16	32	Surface-mounted housing	12 MW	2022212

The detailed dimensions are provided in the devices' respective installation instructions.

1.3 Additional documentation

Further information on the installation and commissioning of the KNX SA SMI sun shading actuators can be found in the associated installation instructions.

General information on the SMI is available at www.smi-group.com.

In this document, group objects will be abbreviated to ***GO***.

2 Safety instructions

We have developed and tested the KNX SA SMI sun shading actuators in compliance with basic safety requirements.
Residual risks nevertheless remain.

- For this reason, please read this manual before commissioning and operating the control.
- **It is very important that you adhere to the safety instructions listed in this section and the warning information contained in this manual. Failure to do so will void any warranty claims against the manufacturer.**
- Keep this manual for future use.

2.1 Meanings of symbols and pictograms

The safety instructions contained in these instructions are marked with warning symbols. They are categorised into different warning types depending on the level of potential danger:

DANGER
warns of an **imminently dangerous situation.**
Possible consequences **may include serious injuries and even death (personal injury), property damage or environmental** harm.

WARNING
warns of a **potentially dangerous situation.**
Possible consequences **may include mild or serious injuries and even death (personal injury), property damage or environmental** harm.

CAUTION
Reminder to **exercise care**.
Failure to comply may result in **property damage**.

The following pictograms and symbols may be affixed to the control unit itself or to the connected devices, alerting you to potential danger:

WARNING
Warning against dangerous electrical voltage.

The i symbol designates important **information** and helpful **tips**.

Example The term **Example** marks an **example**.

- The **square** indicates an **instruction** or a **prompt for action**. Perform this action.
- ▶ The **triangle** denotes an **event** or the **result** of a preceding action.
- ▶ The **black triangle** is a **bullet point** for lists or selections.

2.2 Intended use

The KNX SA SMI sun shading actuators are used for directly positioning mutually independent drives for internal and external venetian blinds, awnings and other sun shading systems.

WARNING
Please obtain the approval of the manufacturer if you have questions regarding the connection of devices not listed in these instructions.

All control devices are intended to be installed **indoors** unless otherwise specified.

WARNING
The approval of the manufacturer must be obtained for uses outside of those listed here. The consequences of unintended use may include personal injury to the operator or third parties as well as property damage to the control unit itself, to connected devices or to moveable mechanical parts of the entire unit.

- Therefore, use our product only as intended.

2.3 Target group

These instructions are intended for persons who are commissioning a sun shading system in KNX technology as well as for qualified technicians. Knowledge of KNX technology is essential.

WARNING
Commissioning and operation by persons who are not sufficiently qualified and informed can cause severe damage to the unit or may even cause personal injury.

- Commissioning may therefore only be performed by properly trained and qualified technicians. These technicians must be able to recognise sources of danger that may be caused by the mechanical, electrical or electronic equipment.
- Persons commissioning the unit must know and understand the content of these instructions.

2.4 General safety instructions

The control system controls your sun shading system automatically. You must therefore observe the following safety instructions:

WARNING
An automatically controlled mechanism may begin to move unexpectedly.

- Therefore, never place any objects in the area of an automatically controlled mechanism. Make sure that no persons are located in the movement range of automatically controlled sun shading products during commissioning.
- If measuring or test work must be carried out on the active unit, make sure that applicable accident prevention regulations are observed under all circumstances.

CAUTION
The entire unit becomes non-functional if power fails. Therefore, move your sun shading system to a safe position ahead of time if a storm is pending. Changing individual parameters may impair the safety of the unit or reduce its effectiveness. It is better to consult a qualified specialist if you are not sure about the effects of a change.

3 General information

3.1 Technical data

Technical data, wiring diagrams and specifications for electrical lines and connectable devices can be found in the installation instructions for the respective actuators.

3.2 Outputs

The device has an SMI interface (there are several SMI I+ and SMI I- connection terminals on the devices).

Up to 16 SMI motors can be allocated to the 16 available outputs, making it possible to control each drive individually, if necessary.

Example The drives with the addresses 1, 2, 3, 4, 13, 14, 15 and 16 are allocated to Output 1 and the drives with the addresses 5, 6, 7, 9, 10, 11 and 12 are allocated to Output 2. The drive with address 8 is controlled via Output 3.

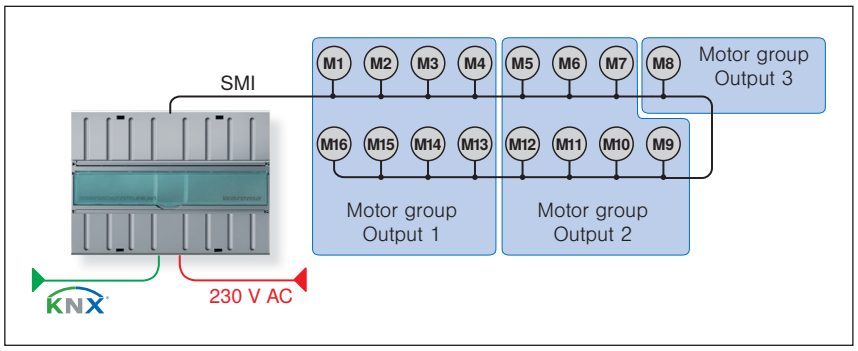


Fig. 1 Allocation of the drives to the outputs (e.g. KNX SA 16M230.32 SMI AP)

3.3 Master reset

The master reset returns the KNX SA SMI sun shading actuator to its delivery condition. All group addresses in the device are deleted, all parameters are set to the default values and the physical address is set to 15.15.255.

A master reset is performed as follows:

1. Switch off the operating voltage
2. Press and hold the programming button
3. Switch on the operating voltage
4. Wait for the programming LED to begin flashing and release the button after approx. 3 seconds
5. Wait for the programming LED to go out
6. Switch off the supply voltage
7. The master reset is finished

After a master reset, the actuator must be recommissioned.

4 Commissioning

The KNX SA SMI actuators are commissioned using the Engineering Tool Software ETS (**at least ETS 5**).

Before initial operation of the KNX SA SMI sun shading actuator, move all connected sun shading products to a safe position, e.g. move external venetian blinds to their upper limit position.

4.1 Electrical connections

Technical data, wiring diagrams and specifications for electrical lines and connectable devices can be found in the installation instructions for the respective actuators.

CAUTION
Only connect sun shading products with correctly adjusted limit switches in order to prevent damage when commissioning.

4.2 Commissioning sequence

Commissioning is performed as follows:

1. Switch on the operating voltage
2. Switch on the bus voltage
3. Press programming button on the device (programming LED lights up)
4. Load the physical address and application into the device from the ETS
5. Wait for the programming LED to go out
6. Check function of the device

After commissioning or after voltage recovery, the sun shading actuator does not recognise the position of the connected sun shading products. For this reason when a move command is executed for the first time, the connected sun shading products initially perform a calibration in some circumstances.

4.3 Manual override operation

The KNX SA SMI sun shading actuators can be operated manually for commissioning. The device can be operated using the two buttons on the device or via the smartphone app.

When the physical addresses have been loaded into the actuator, it can also be operated via the ETS DCA app.

4.3.1 Buttons on actuator

The KNX SA SMI sun shading actuators **REG** are equipped with a **keypad**.

The KNX SA SMI sun shading actuators **AP** are equipped with a **keypad in the terminal area**, which can be operated with a small screwdriver.

- The UP/DOWN buttons of the keypad function as follows:
  Stop when pressed briefly,
  move to limit position when pressed for longer.
- The buttons directly affect all connected SMI motors (via SMI broadcast telegrams).
- Button operation has the highest priority. A currently active safety function is overridden by the push button operation.
- After commissioning, the buttons continue to affect all motors connected to the SMI interface.

The buttons on the actuator ensure that the connected devices can be operated during the commissioning phase and in fault situations such as if the bus voltage should fail. They are not intended to replace the external buttons or other operating elements.

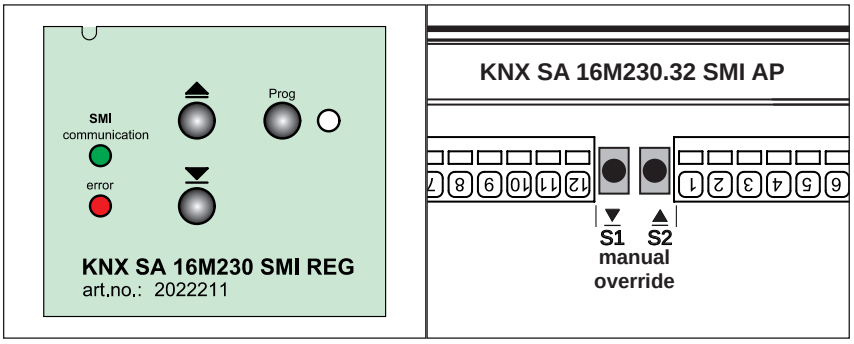


Fig. 2 REG and AP buttons

4.3.2 SMI communication display

Two LEDs are located on the cover plate to indicate communication via the SMI interface.

Communication	error	Description
Flashes green		Normal SMI communication
	Flashes red	Fault in the SMI communication detected
Lights up green	Lights up red	Actuator not yet parameterised or or device motor list empty

The red LED is disabled during the motor search and the SMI bus initialisation.

4.3.3 Smartphone app

The KNX SA SMI sun shading actuators are equipped with a Bluetooth module. This allows for operation via a smartphone app. The communication between the smartphone and the KNX devices is established via Bluetooth LE (Low Energy).

The app ensures that the connected devices can be operated during the commissioning phase and in fault situations such as if the bus voltage should fail. It is not intended as a substitute for push buttons.

In order to protect against operation by unauthorised persons, access via the app is protected by a password (Bluetooth Login Key). When loading with ETS for the first time, the actuator's password in delivery condition is overwritten with the preset password in the ETS (*see Fig. 3*). This is then required to operate the actuator via the app.

If necessary you can change the password to anything in the range from 0 to 9999 in the ETS. Do not forget to document the change, in case operation via the app is needed once again later.

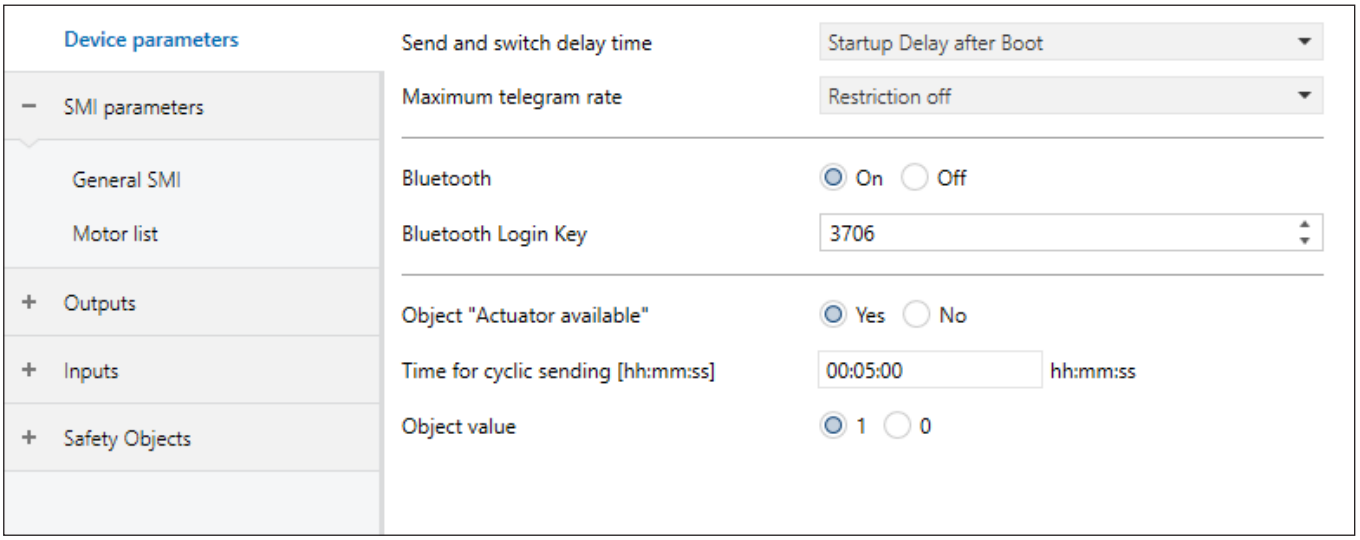


Fig. 3 Parameter dialogue: Device parameters

Alternatively the Bluetooth function of the actuator can also be completely switched off in the parameterisation via the ETS (the function is always preset at the factory to “On”).

4.3.3.1 Loading and starting app

- Download the WAREMA app for operating the KNX SA SMI sun shading actuators from the appropriate app store for your smartphone.

Android: Google Play Store http://www.warema.de/KNX-SA-Android	**iOS: App Store** http://www.warema.de/KNX-SA-iOS
Requirements: Android 4.3 or higher for Android 6 or newer GPS must be active	Requirement: iOS 9 or higher

- Start the app.
- ▶ The surrounding area is automatically scanned for one minute for WAREMA sun shading actuators.
- ▶ All actuators found are shown in the display.

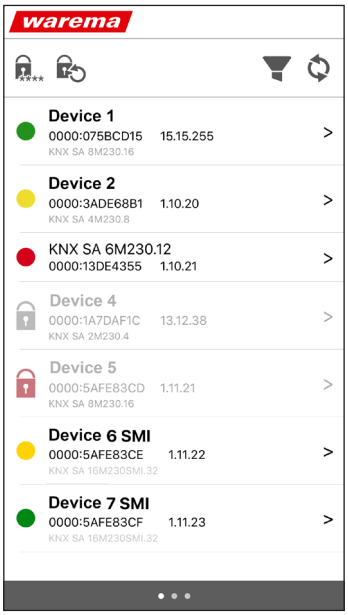
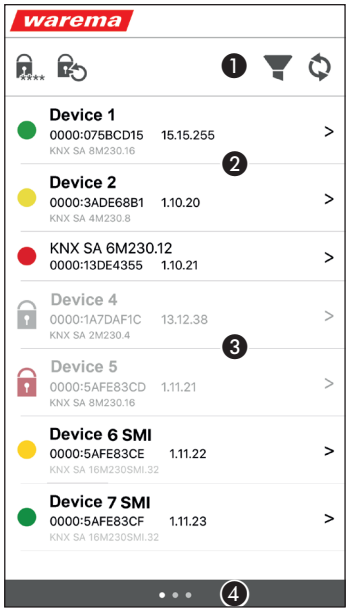
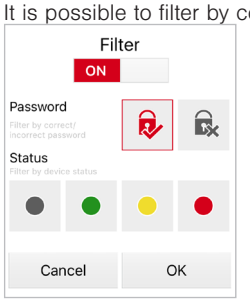


Fig. 4 Bluetooth app

4.3.3.2 Device list



1 Menu bar	Change password. Reset password to factory setting. Filter displayed devices in the device list. When the filter is active, this symbol is red. It is possible to filter by correct password or status.  Scan for devices again (scan duration one minute). An ongoing scanning procedure can be cancelled with the symbol then shown at this point. (The scanning procedure can also be started by dragging down the device list.)
2 Device list Password identical	All devices found are displayed in the device list. All devices, whose password matches the password currently set in the app, are shown in black. The colour of the point displays the status of the device. The device type, the KNX serial number and the physical address are displayed. (If equipment labelling has been assigned, this is displayed first. The device type then appears in grey beneath the other information.) Briefly touch a device to switch to the operating window.
3 Device list Password different	All devices, whose password does not match the password currently set in the app, are shown in grey. A lock in the device's status colour is shown instead of a point. If you briefly touch the device, you must first enter the device's password to be able to switch to the operating window.
4 Page indicator	Displays the window in which you are currently located. You can switch between the device list, the cache (total list of all scanned devices) and Help by swiping sideways on the screen.

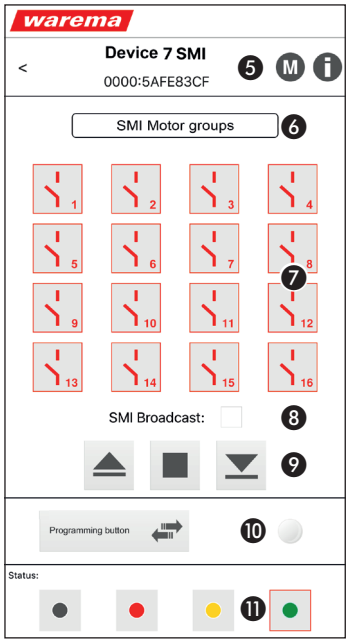
The KNX SA SMI sun shading actuator can only ever establish one Bluetooth connection. As soon as you select an actuator in the device list (operating window opens), it stays connected to the smartphone until you select another actuator. If you scan again or completely exit the app, any existing connection is lost.

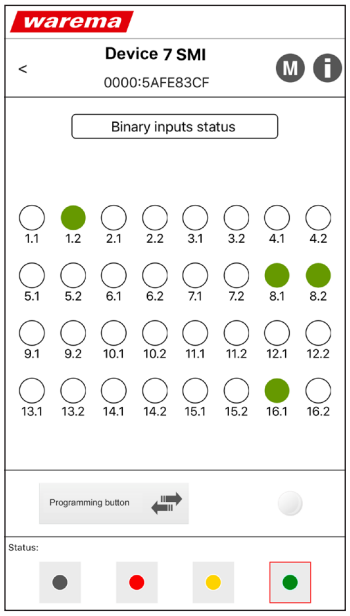
As long as a sun shading actuator is still connected to a smartphone, it is not found in scans by other smartphones operating at the same time.

4.3.3.3 Operating window

The **operation of motor groups** via the app has the same priority as manual operation via group objects. A currently active safety function can prevent operation via the app.
The **SMI broadcast** takes effect when operating on all connected motors; active safety objects are ignored.

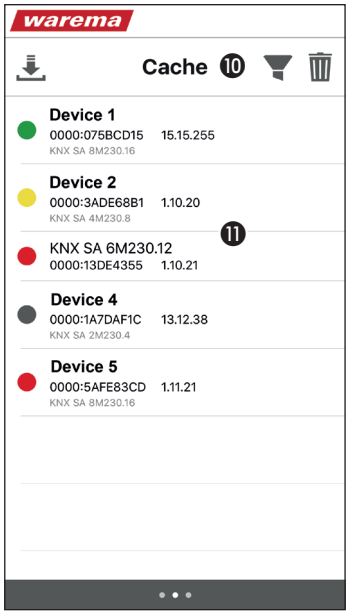
WARNING
Never randomly press the buttons on the app without having a line of sight to the sun shading system.





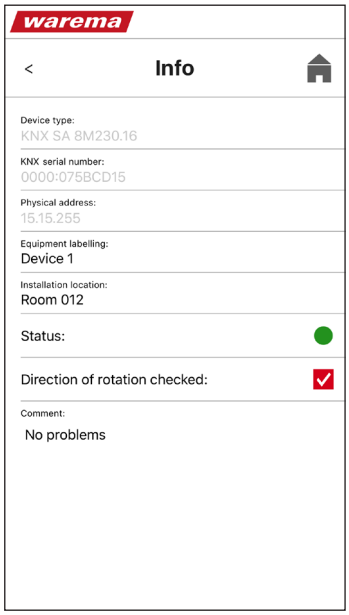
5 Header	Equipment labelling (or device type) and KNX serial number are displayed in the header. M Open motor list (only for SMI actuators) i Open actuator info window
6 Toggle view	For actuators with inputs, it is possible to switch between the following: Outputs Operating 230 V / 24 V actuators or SMI Motor groups Operate SMI actuators Binary inputs status Display state of inputs (only display, see figure below) The button for toggling the view is displayed depending on the device. It only appears for actuators with inputs.
7 Device outputs	In this area you can choose whichever outputs or SMI motor groups you want to operate. The operating elements affect all selected outputs. RED: selected GREY: not selected Motors that have not been allocated to an output can only be operated via SMI Broadcast. The number of displayed outputs depends on the device type.
8 SMI broadcast	Only displayed for SMI actuators (switching on/off with the checkbox). The SMI broadcast takes effect when operating on all connected motors. **Active safety objects are ignored.**
9 UP/STOP/DOWN operating elements	All selected (red) outputs receive the corresponding up or down move command when the button is pressed. The operating behaviour is as follows: Brief push of button = Step / Stop (only Stop for SMI) Long push of button = Move. The actuator switches the associated outputs on or off or sends telegrams to the SMI interface.
10 Programming button and LED	The programming button and the LED have the same function as on the device. Programming the physical address, *see chapter 5.3 on page 35*.
11 Status	Here you can specify a status for the device. It is displayed in the device list before the device. The status is purely informative and is used to give a better overview of many devices.

4.3.3.4 Cache



10 Menu bar	Export the cache list as a csv file. The list is always exported in full, filter settings are ignored. Android: Select a delivery option in the dialogue box. You can send the csv file by e-mail, save it on Google Drive or transfer it via Android Beam. Alternatively, you can access the file through any file manager. iOS: A draft e-mail opens with the csv file attached, so that you can send it to any e-mail address. Alternatively, you can access the file via iTunes (under “Release” select the app “KNX SA”, the file is then displayed in the documents window.). Filter displayed devices in the device list. When the filter is active, this symbol is red. It is possible to filter by status. Delete the entire cache
11 Cache list	All devices previously found while scanning are displayed in the cache list. This also allows you to see the devices that were no longer registered in the repeated scan (e.g. in another part of the building). No operation is possible from this list, as there is no communication with the listed devices.

4.3.3.5 Actuator info window



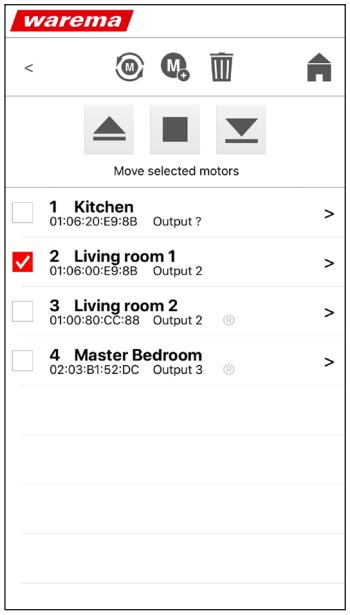
The device information is displayed in the actuator info window.

The device type, KNX serial number and physical address can not be modified and are therefore shown in grey.

The fields shown in black can be modified. Here you can enter the appropriate information as needed. It is saved in the actuator and is available for continued commissioning or future access via the app.
Equipment labelling: Here you can enter a name, an allocation number or a similar label for the actuator.
Installation location: The location of the actuator is entered here.
Status: The status of the device selected in the operating window is displayed here.
Direction of rotation tested: Here you can set a checkmark if you have checked the direction of rotation of all connected drives.
Comment: Field for additional information (info about actuator, special features, notes for colleagues, ...)

Back to the homepage

4.3.3.6 Motor list



The actuator's motor list is displayed with all entered motors. Motors can be selected for operation or deletion via the check boxes ☐ / ☑.

Motor search:
All motors in the displayed list are deleted and a search is performed for new motors. These are automatically entered into the list.

Add motor:
A search is performed for new motors, the motor list is retained. The newly found motors are added to the motor list. New motors are highlighted in green.

Delete the selected motors from the motor list.

Operate the selected motors.

Back to the homepage

4.3.3.7 Motor info window

The motor information is displayed in the motor info window.

The motor manufacturer, motor ID (HEX) and address can not be modified and are therefore shown in grey.

The fields shown in black can be modified. Here you can enter the appropriate information as needed. It is saved in the actuator and is available for continued commissioning or future access via the app.
Output allocation: Here, you can allocate the motor to one of the 16 outputs (SMI motor groups).
Reference?: Check here if the motor is intended to serve as a reference motor for the motor group.
Alias name: Here you can enter a name, an allocation number or a similar label for the actuator.
Installation location: The location of the actuator is entered here.
Status: The status of the device selected in the operating window is displayed here.
Direction of rotation tested: Here you can set a checkmark if you have checked the direction of rotation of all connected drives. If the motor rotation direction is reversed, the motor limit positions are switched.
Comment: Field for additional information (info about actuator, special features, notes for colleagues, ...)

Back to the homepage

5 Project setup

The KNX SA SMI actuators are commissioned using the Engineering Tool Software ETS (**at least ETS 5**).
The product database required for this (.knxprod) can be found in the online catalogue of the ETS or on the internet at http://www.warema.de/knx.

5.1 Parameters

The SMI motors can be parameterised in three ways:

1. Parameterisation only via the ETS parameter dialogue
   The manufacturer code and the key ID of the motors must be known. The information is entered into the parameter dialogue of ETS. The motors are allocated to the outputs.
2. Parameterisation via the ETS DCA app.
   If the manufacturer code and the key ID of the motors are known, these can be entered into the DCA app. It is also possible to search the motors via the DCA app. The motors can then be allocated to the outputs.
3. Parameterisation via the smartphone app and the DCA app.
   In the first step, a search for motors can be performed via the smartphone app. The motors can be allocated to outputs in the smartphone app.
   In the second step, the allocations made via the smartphone app are read out from the device in the DCA app.

Details on all the parameters are provided in *Chapter 1 Overview on page 5*.

5.1.1 Parameterisation via the ETS parameter dialogue

The actuators are parameterised using the parameter dialogue of the ETS. For the sake of clarity, the parameters there are presented in parameter groups.

The parameter settings can be created in the following order:

1. Select the operating mode of outputs 1 – 16 (the same for all or separately)
2. Activate/deactivate the safety objects and set the parameters
3. Parameterise outputs
4. Parameterise SMI motor list (the manufacturer ID and the key IDs of the motors must be known in decimals, e.g. observe the barcode label on the motors) and allocate the motors to the outputs.

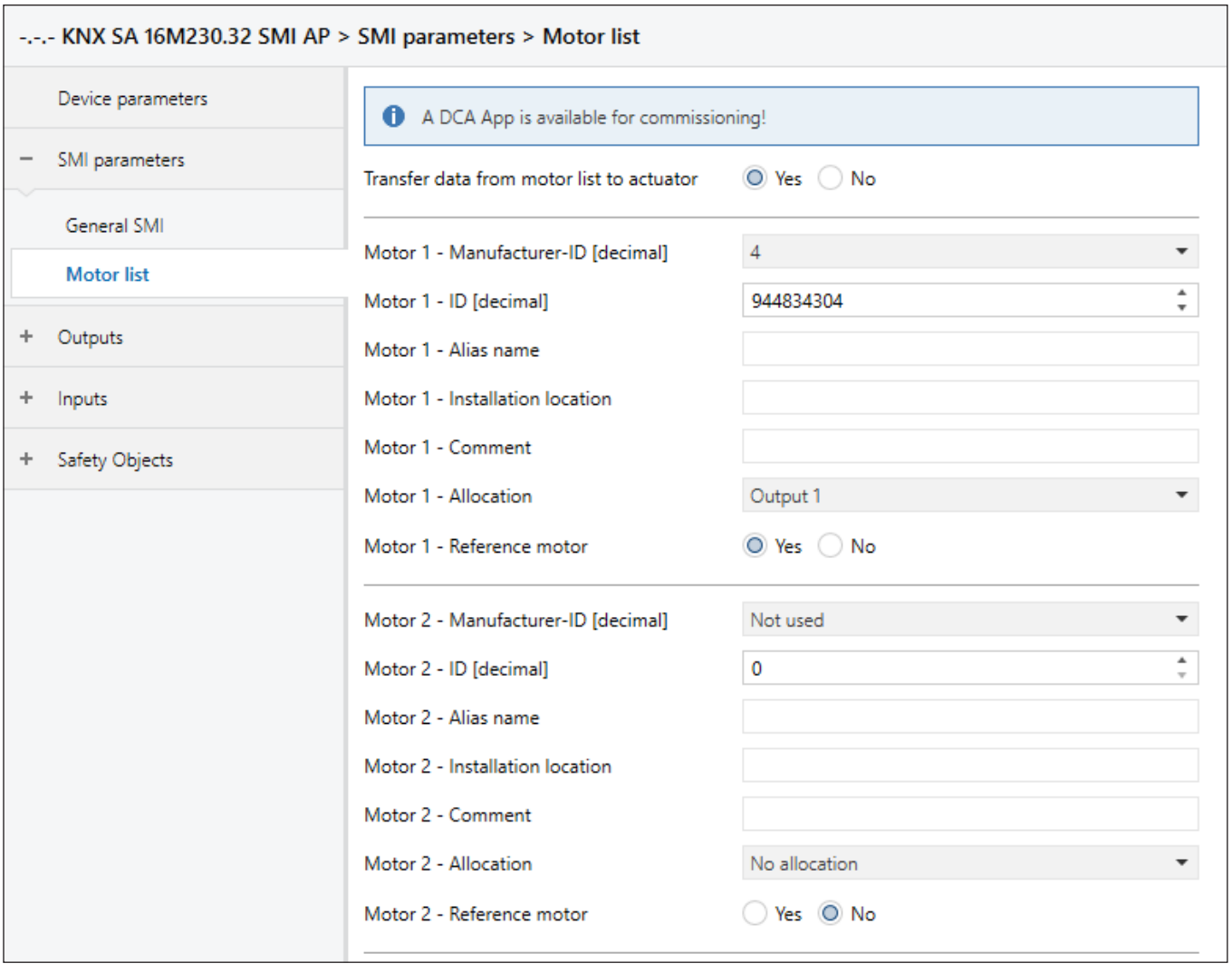


Fig. 5 SMI parameters - Motor list

5. Set "Transfer data from motor list to actuator" to ***Yes***

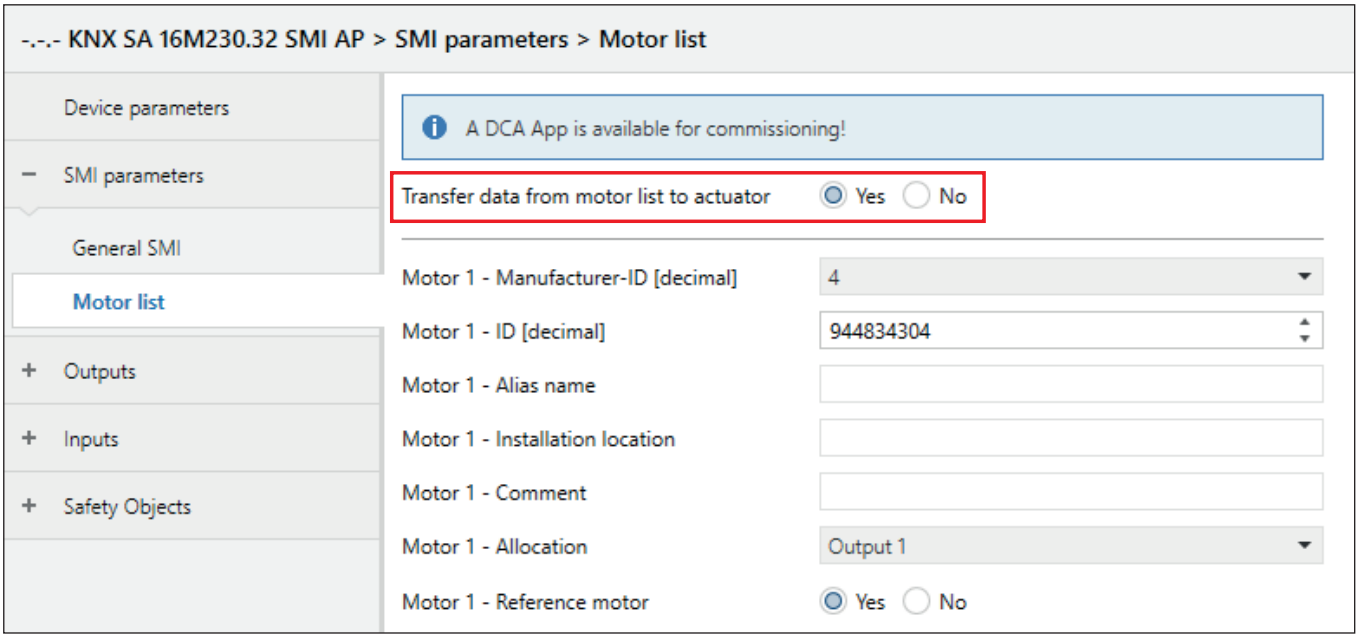


Fig. 6 SMI parameters - Motor list

6. Load application program with the ETS in the actuator

Details on all the parameters are provided in *Chapter 7 Parameter dialogue on page 41*.

5.1.2 Parameterisation in the ETS via the DCA app

In order to be able to use the full functionality of the DCA app, the actuator must be able to be reached via the KNX bus.

In **ETS parameter dialogue:**

1. Select the operating mode of outputs 1 – 16 (the same for all or separately)
2. Activate/deactivate the safety objects and set the parameters
3. Parameterise outputs

Continue with the **DCA app:**

A detailed description of the DCA app can be found in *Chapter 5.1.4 DCA App on page 26*.

4. Device motor list → Search all motors
   All motors are searched and listed after a successful motor search.
5. Device motor list → Identify the motors by moving them with the arrow keys
   Assign an alias name to the motor to aid later allocation.
6. Comparison ETS <> Device
   Apply the motor data from the device to the ETS parameters
7. ETS motor allocation
   Allocate the motors to the outputs by dragging and dropping. The alias name is displayed in the name of the columns.

Continue in **ETS parameter dialogue**

8. Load application program with the ETS in the actuator

5.1.3 Parameterisation with smartphone app and DCA app

The actuator must be able to be reached via the KNX bus.

In **ETS parameter dialogue:**

1. Select the operating mode of outputs 1 – 16 (the same for all or separately)
2. Activate/deactivate the safety objects and set the parameters
3. Parameterise outputs

Continue with the **smartphone app:**

A detailed description of the smartphone app can be found in *Chapter 4.3.3 Smartphone app on page 15*.

4. Search motors via the smartphone app Then allocate the motors in the smartphone app to the outputs.

Continue with the **DCA app** (optional):

A detailed description of the DCA app can be found in *Chapter 5.1.4 DCA App on page 26*.

5. Device motor list → Load motors from device
   Read out and display the motor list from the device
6. Comparison ETS <> Device → Apply all device data
   The motor list from the actuator is transferred to the motor list of the ETS. In order to save the parameters in the ETS parameters, the "Save configuration in ETS" button must be pressed.

Continue in **ETS parameter dialogue**

7. Load application program with ETS in the actuator
   If the allocation of the smartphone app has not been read out, set the *Transfer data from motor list to actuator* parameter to "No" (*cf. Fig. 6)*

5.1.4 DCA App

The DCA app is available in the KNX Online Shop as a free download and can be installed in ETS5 and later. Following installation, the app is available under the menu item DCA.

In order to be able to use the full functionality of the DCA app, the actuator must be able to be reached via the KNX bus.

5.1.4.1 Device motor list

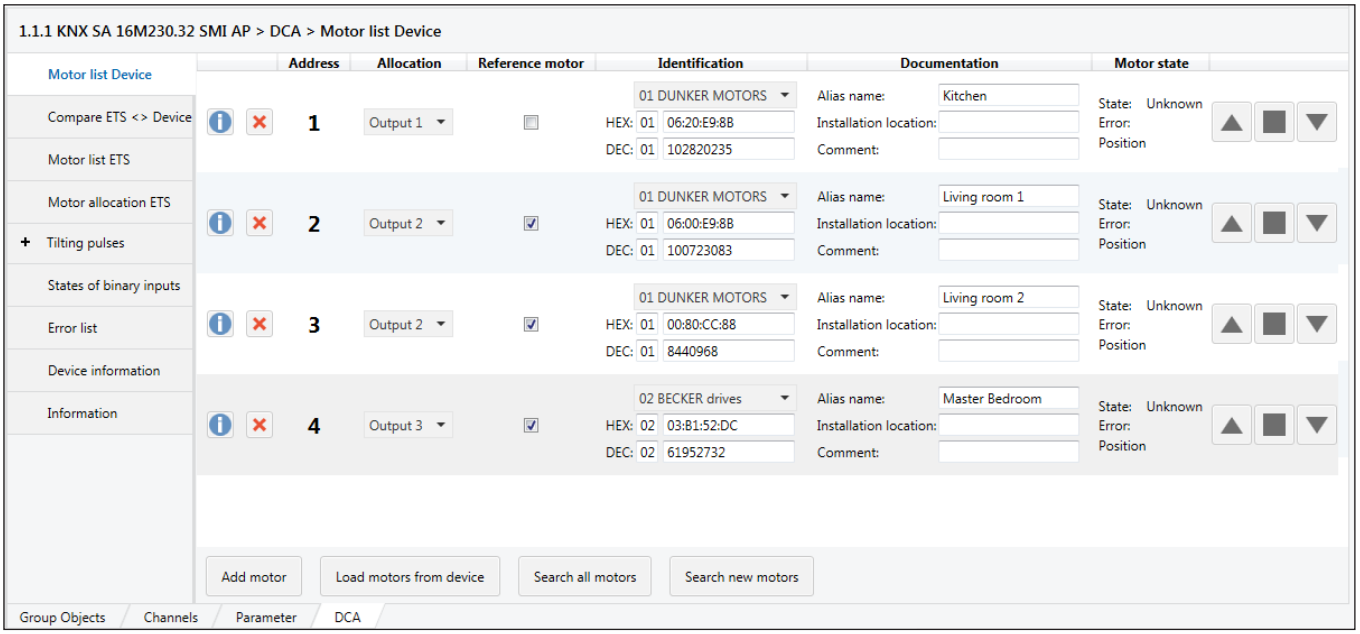


Fig. 7 DCA app: Device motor list

Function	Description
Add motor	Enter the motor and motor ID into the motor list
Load motor from device	Load and display the motor list from the device
Search all motors	All motors in the displayed list are deleted and a search is performed for new motors. These are automatically entered into the list.
Search new motors	A search is performed for new motors, the motor list is retained. The newly found motors are added to the motor list.
▲ ■ ▼	The motors can be individually moved using the three buttons.
ⓘ	Read the motor status.
✕	Delete the motor from the motor list.

5.1.4.2 Comparison ETS < > Device

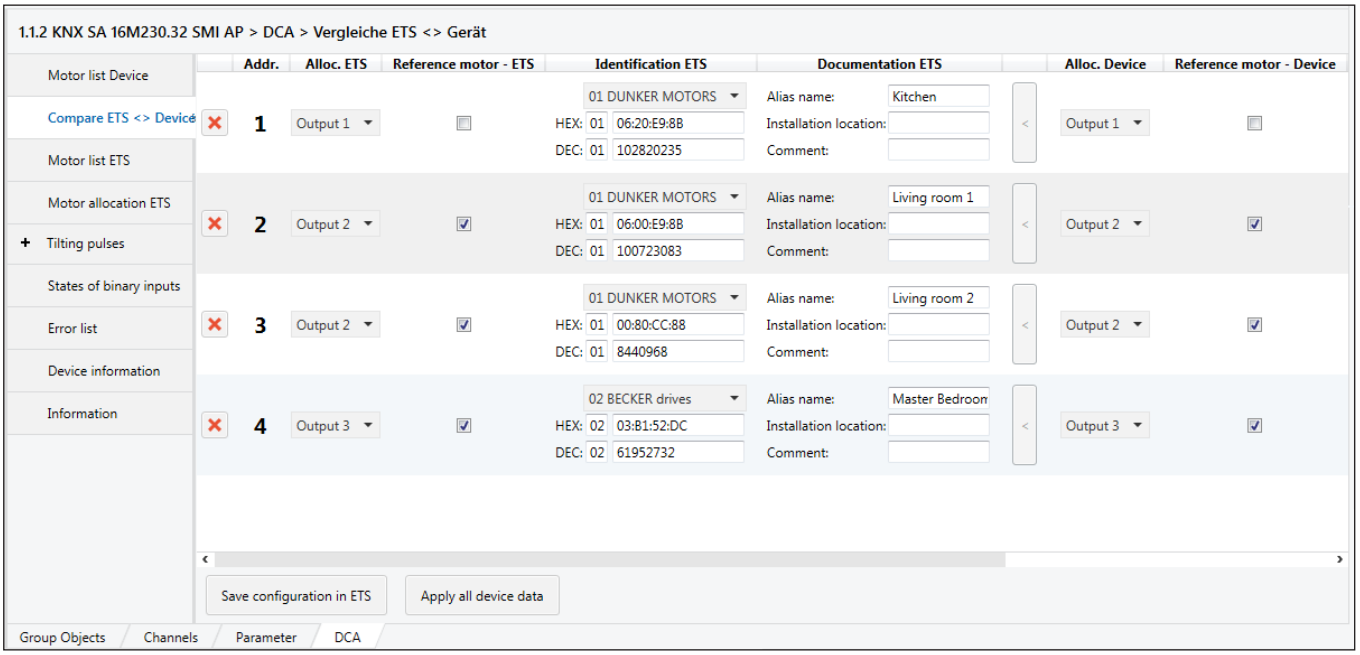


Fig. 8 DCA app: Comparison ETS < > Device

The parameters in the ETS are compared with the parameters loaded from the actuator. All disparities are highlighted in orange.

Using the < button, the parameters for a single motor can be transferred to ETS. ***Apply all device data*** immediately applies the parameters for all motors.

Function	Description
Save configuration in ETS	The motor list is transferred to the ETS parameters.
Apply all device data	The motor list from the device is transferred to the motor list of the ETS. In order to save the parameters in the ETS parameters, the "Save configuration in ETS" button must be pressed.

5.1.4.3 ETS motor list

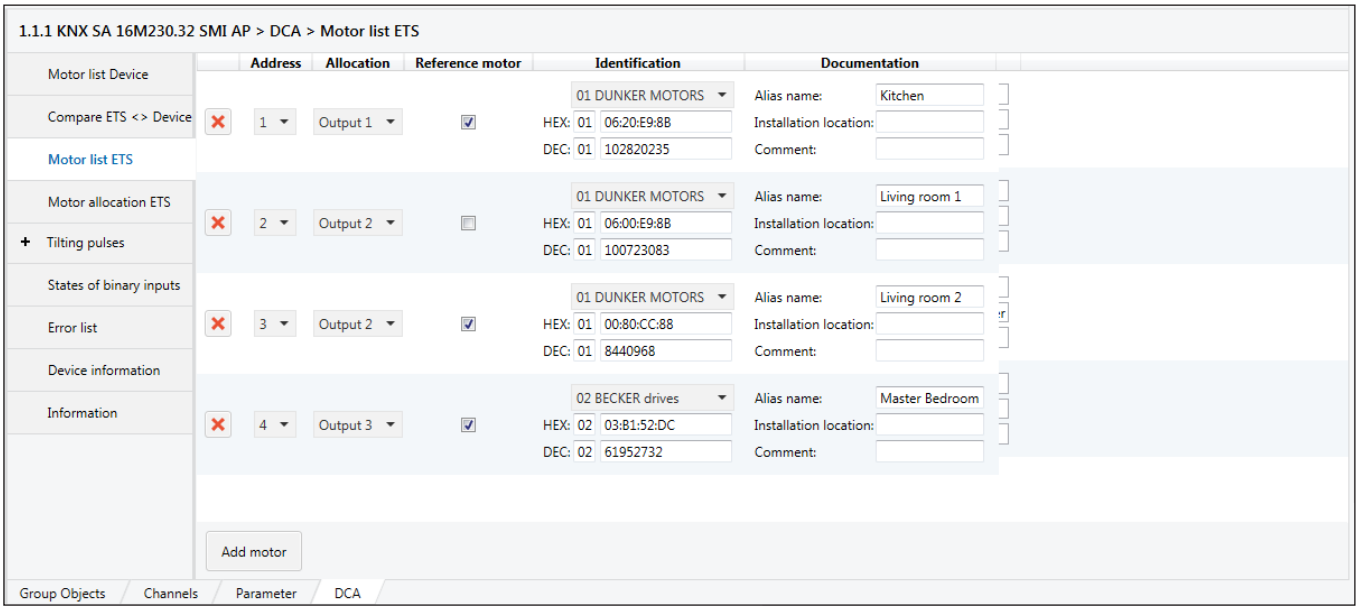


Fig. 9 DCA app: ETS motor list

Function	Description
Add motor	Enter the motor and motor ID into the motor list

The specifications from the motor manufacturer on the motors are not uniform.

The motor IDs are sometimes given as a decimal but then sometimes as a hexadecimal. When written as a hexidecimal, the manufacturer code is sometimes placed first and sometimes not.

If the motor ID is given as a hexidecimal and if this ID only consists of 4 bytes (12:67:14:05), the manufacturer code is not included and the manufacturer must be additionally parameterised.

If the motor ID is given as a hexidecimal and if this ID only consists of 5 bytes (01:0D:08:94:88), the manufacturer code is not included and the manufacturer is correctly detected automatically.

If the motor ID is given as a decimal, the DCA app checks whether the manufacturer code is included. In this case, the manufacturer is automatically detected. In other cases, the manufacturer must be additionally parameterised.

5.1.4.4 ETS motor allocation

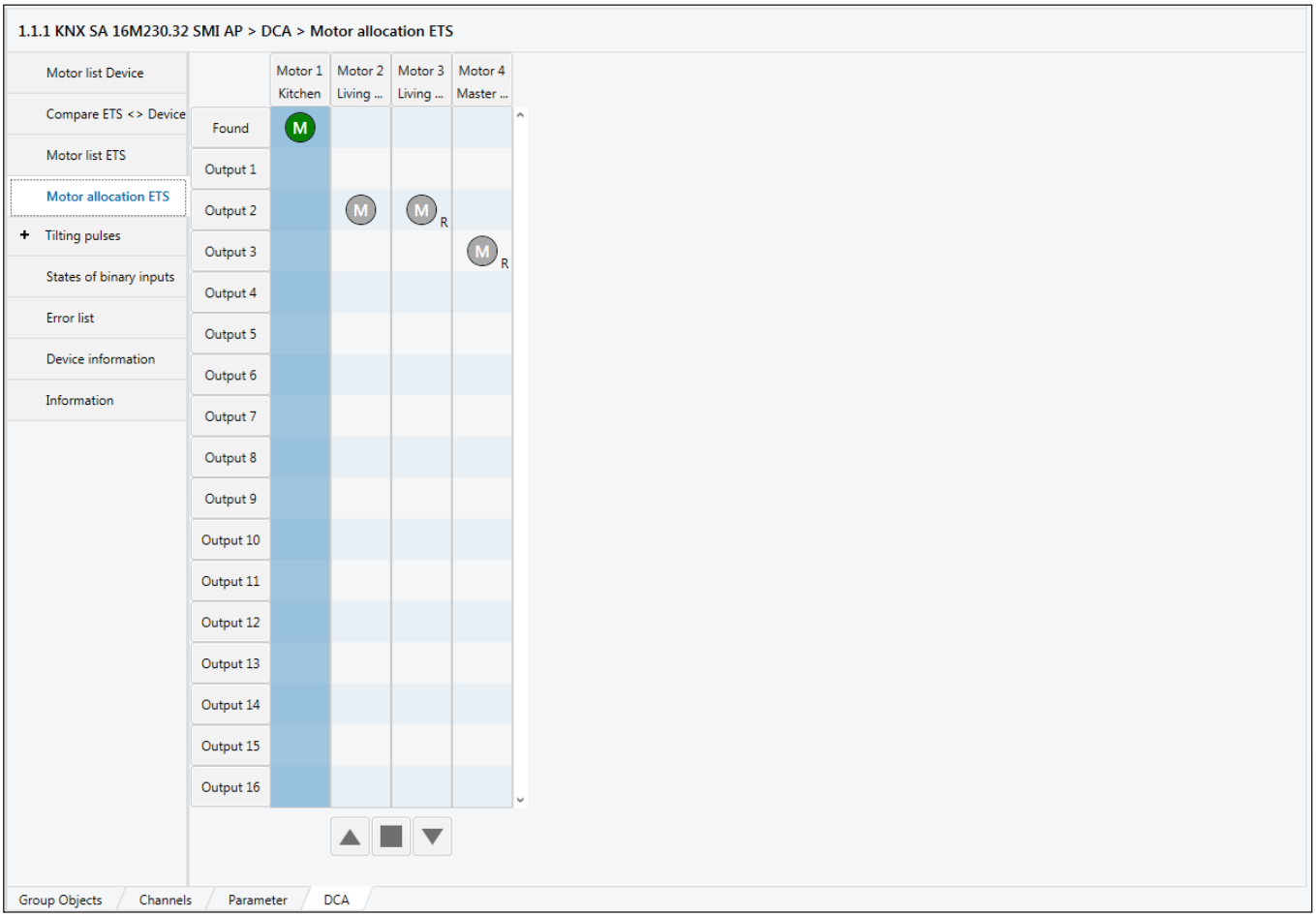


Fig. 10 DCA app: ETS motor allocation

If the order of addresses is changed in the ***ETS motor list*** view, the order of motors also changes in the ***ETS motor allocation*** view.

Allocating motors to an output
All found motors are displayed in the upper line of the table in green.
Allocate the motors to the desired outputs by dragging and dropping.
The motor which is first allocated to an output is automatically set as a reference motor for this output.
If a motor which has already been set as a reference motor elsewhere is allocated, this becomes the new reference motor for the group.
By right-clicking on any motor, this can be set as the reference motor.

Move motors
Select a line or column from the table. Move the motors with the three push buttons under the table.

Selected column: the motor in this column can be moved individually.

Selected line: all motors allocated to this output are moved together.

5.1.4.5 Tilt pulses

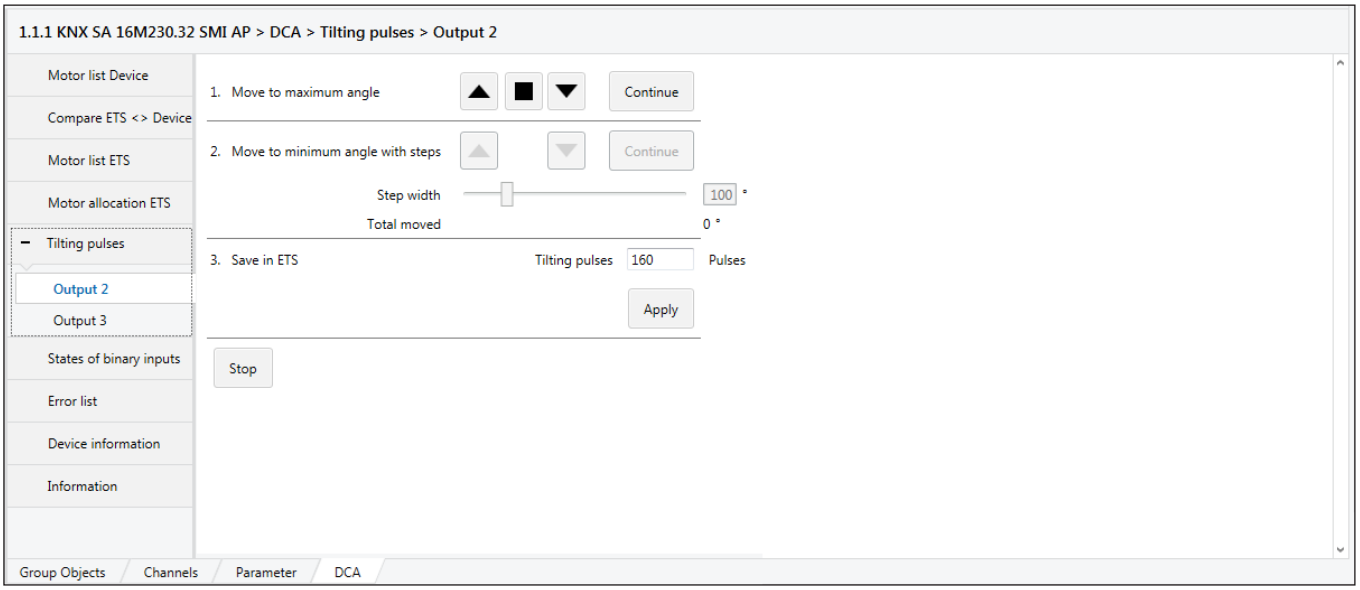


Fig. 11 DCA app: tilt pulses

Determining tilt pulses

Select an output. The window shown above appears.

1. Move slat products with the push button to the maximum angle (open).
   Continue to the next step by pressing ***Continue***.

2. Move slat products with the push button to the minimum angle (closed). The slats are tilted by the value set by the lower ***Step size*** with each push of the button.
   Continue to the next step by pressing ***Continue***.

3. The number of tilt pulses for a complete tilting is displayed.
   This value is automatically transferred into the ETS parameter for this output by pressing ***Apply***.

   The procedure can be interrupted at any time by pressing the ***Stop*** button.
   The procedure must then be restarted at Step 1.

5.1.4.6 Binary inputs status

This menu can only be viewed for actuators with inputs.

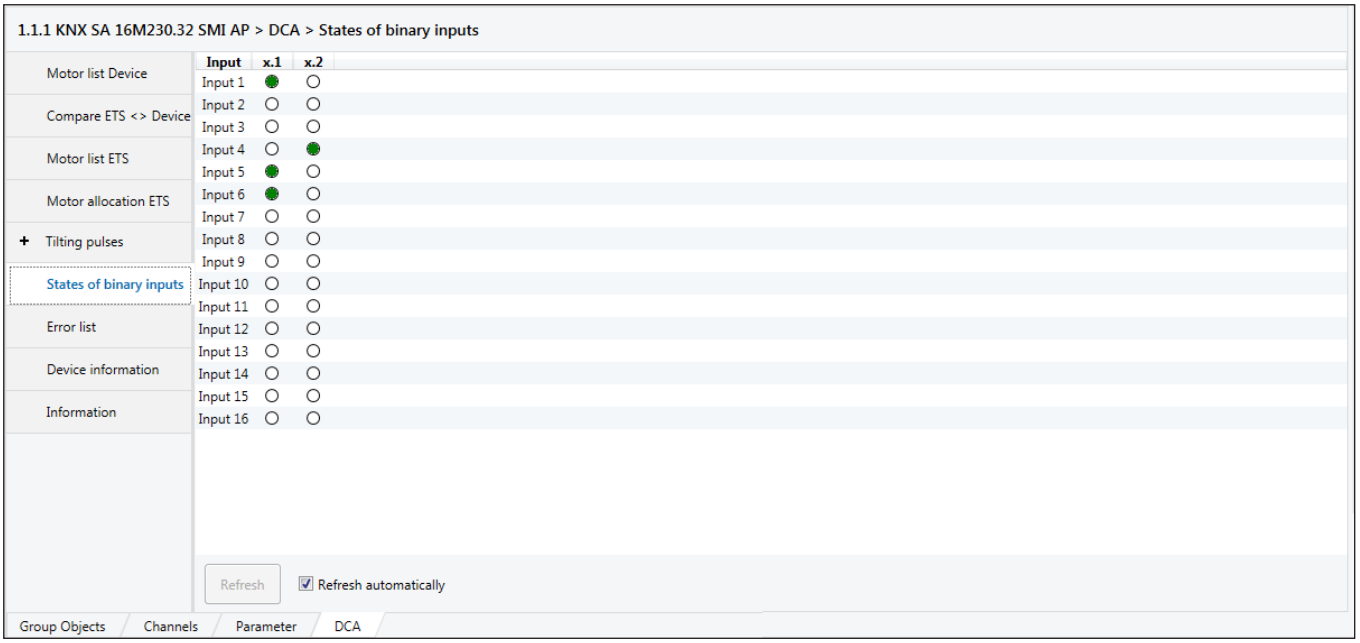


Fig. 12 DCA app: Binary inputs status

The current status of the inputs is displayed. When the contact is closed, the input is highlighted in green.

Function	Description
Updating	It updates the input view once
Update automatically	It updates the input view cyclically

5.1.4.7 Error list

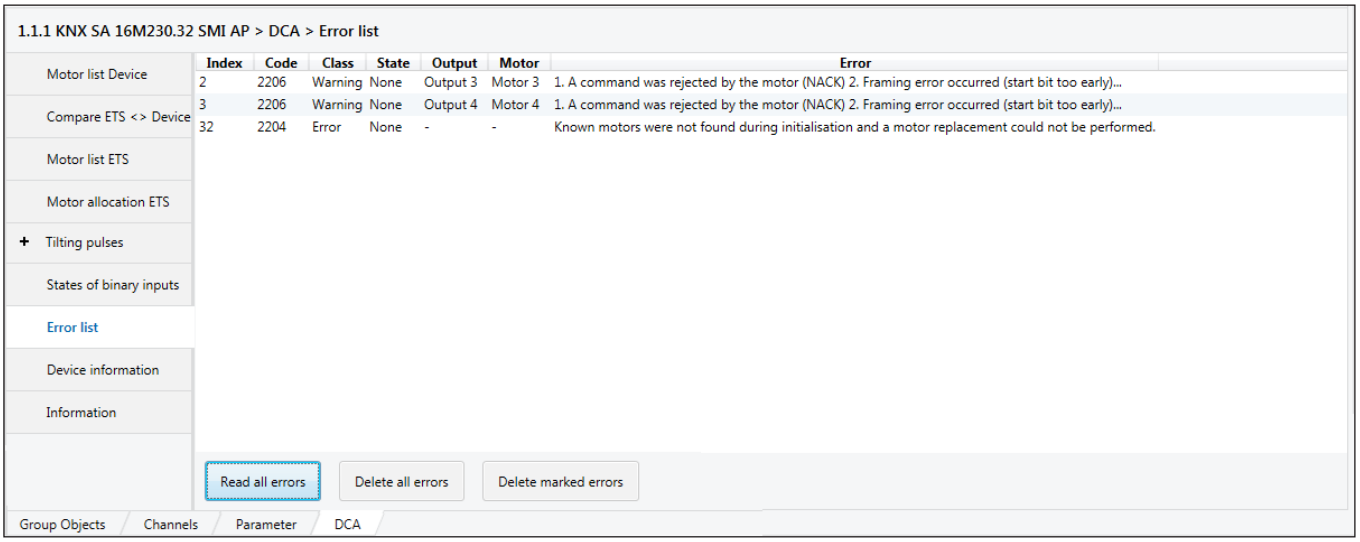


Fig. 13 DCA app: Error list

Function	Description
Read all errors	All errors are read from the error list.
Delete all errors	All errors are deleted from the error list.
Delete selected errors	Only selected errors in the error list are deleted.

5.1.4.8 Device information

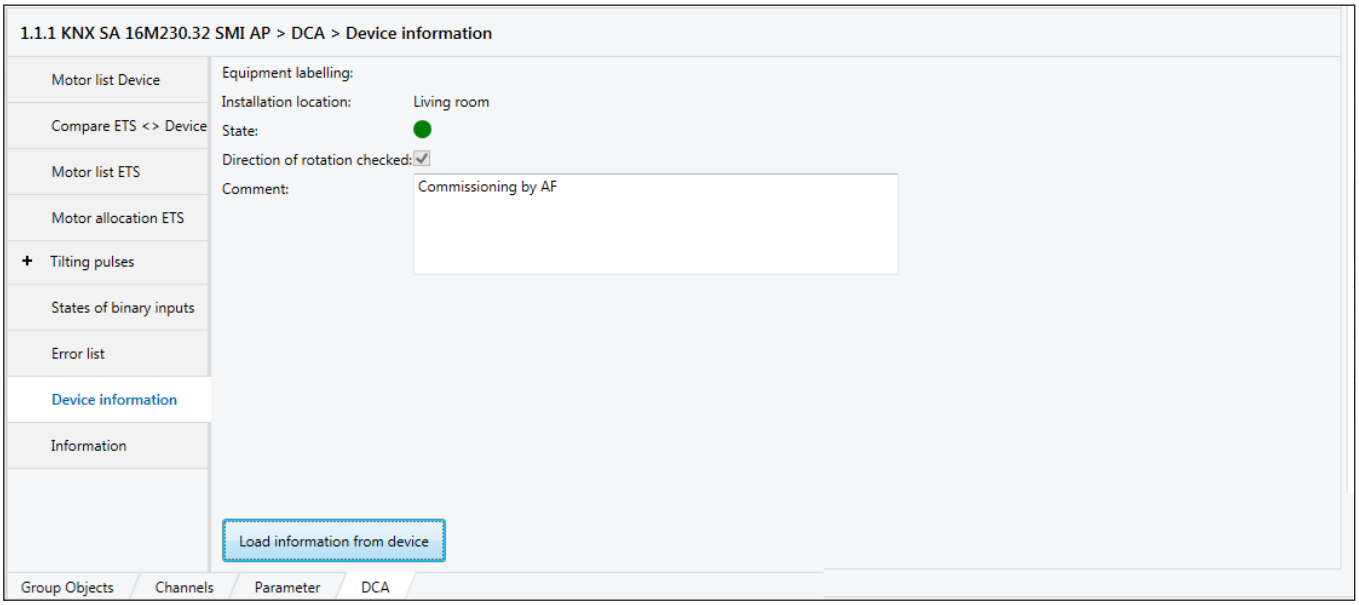


Fig. 14 DCA app: Device information

Here, the information saved in the actuator, which has been entered with the smartphone app, is displayed.

The data for the existing KNX bus connection is read out from the actuator using the ***Load information from device*** button.

Changes to the entries can be made in the ***Device motor list*** tab.

5.1.4.9 Information

Version information display for the DCA app.

5.2 Group addresses/linking

The operating modes of the outputs are set in the parameter settings. For each selected operating mode, only a specific set of group objects (GO) is required in the ETS. Group objects that are not required are automatically hidden by the ETS. If necessary, when the operating mode is changed, links that already exist may be deleted from the ETS project.

5.3 Physical address

The physical address is used for the exact identification of a device.

5.3.1 Program addresses via the programming button or smartphone app

You can perform programming either in the app or directly on the actuator. There is a Prog button for programming and a display LED both in the app and on the actuator.

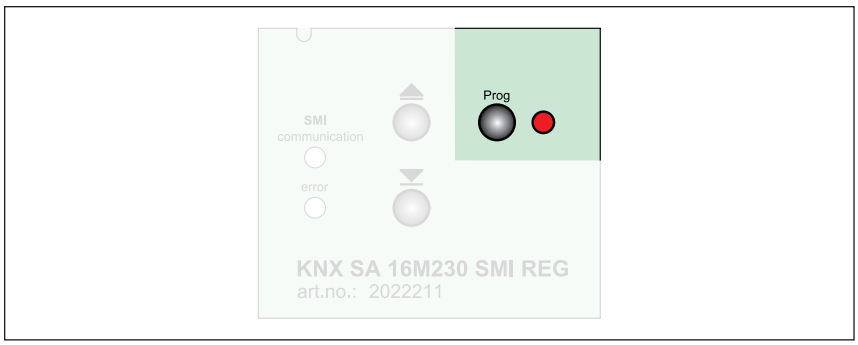


Fig. 15 **REG:** Programming button on the keypad

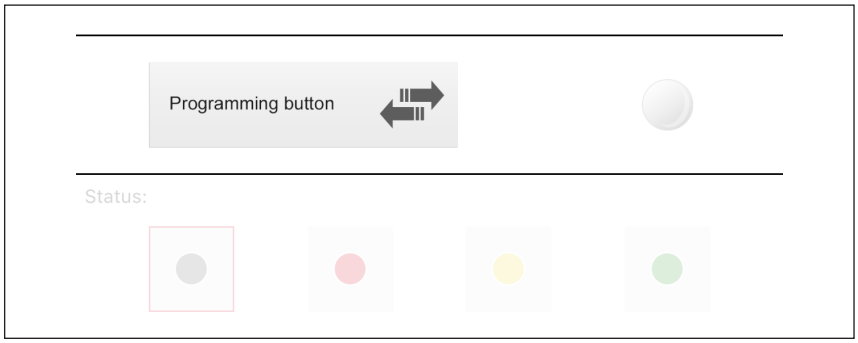


Fig. 16 **AP:** Programming button in the smartphone app.
The actuator is also equipped with a programming button.

The procedure here is basically the same:

- Start the programming in the ETS with [Program physical address].
- Press the programming button in the app or on the actuator to put the actuator into programming mode.
- ▶ The red LED lights up when programming mode is active. Programming is started using the ETS. Programming mode is automatically ended and the red LED goes out.

If the programming mode is to be ended earlier, press the programming button again. The red LED goes out.

After the physical address is programmed, the KNX SA SMI remains operable via the keypad or smartphone app.

The device is delivered with the physical address 15.15.255.

5.3.2 Program addresses via the ETS App

To enable the commissioning of the devices with the KNX serial number via ETS, a two-part label is applied to the device. The KNX serial number of the device appears on both parts of the label as a barcode and as plain text. One part of the label can be removed by the installer and applied to the layout plan of the building.

Siemens are then able, thanks to the free ETS App **SIEMENS Address by ID**, to commission the devices without needing to press the programming button.

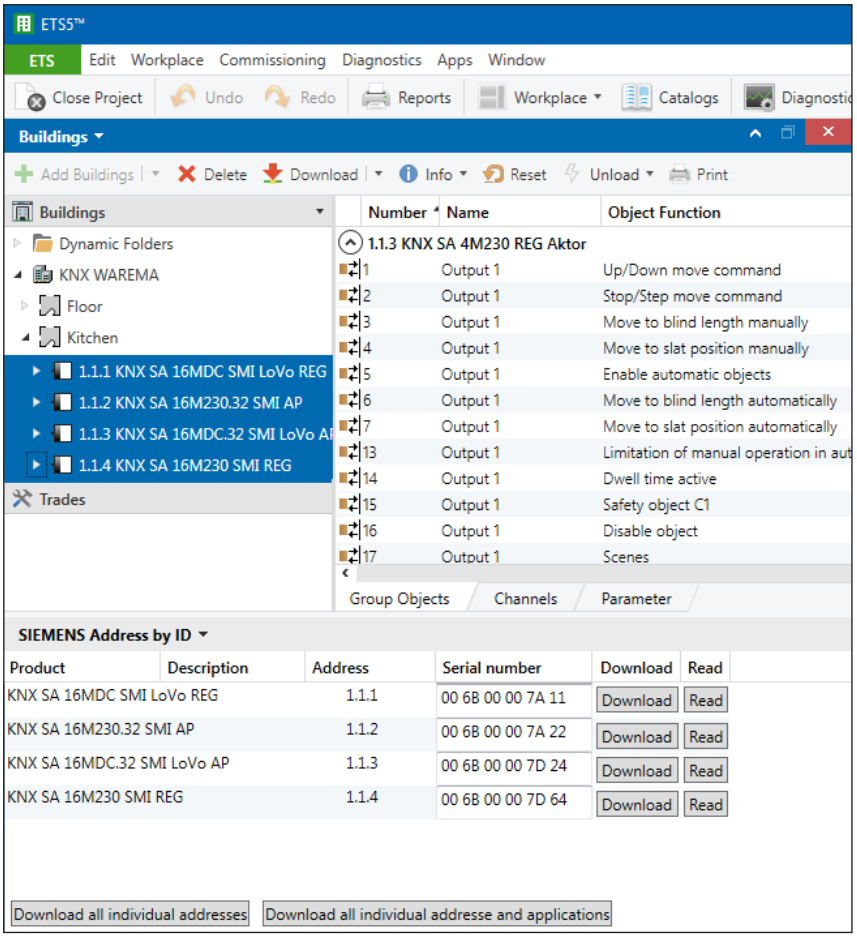


Fig. 17 SIEMENS ETS app for commissioning using the KNX serial number

5.4 Application program

During initial operation of the KNX SA SMI, the physical address, group objects, parameters and group addresses must be programmed. If a project is changed later on, only the group addresses and parameters need to be programmed.

Group objects are loaded, for example, by selecting the following in the ETS: [Programming...] > [Application program].

5.5 Automatic replacement of a motor

If a motor is replaced, a voltage reset must be performed on the actuator (or "Reset device" in the ETS). The actuator restarts the motors after every reset.

If a motor is not found and a new motor is detected during a reset, the missing motor is automatically replaced by a new motor.

After the motor replacement, the data in the ETS does not match the motor list in the actuator. The replaced motor is now entered in the actuator's motor list.

The DCA app can be used to align the data.
See *Chapter 5.6 Changes to an existing unit on page 38.*

5.6 Changes to an existing unit

The actuator must be able to be reached via the KNX bus.

in the **DCA app:**

A detailed description of the DCA app can be found in *Chapter 5.1.4 DCA App on page 26*.

1. Device motor list → Load motors from device
   Read out and display the motor list from the device

2. Comparison ETS <> Device → Disparities are highlighted in colour
   Compare the motor list in the device with the motor list of ETS. If a motor has, for example, been automatically replaced by the control, apply this to the ETS list from the motor list.

3. Comparison ETS <> Device → Save configuration in ETS
   Save the modified data in the ETS parameters.

Continue in **ETS parameter dialogue**

4. Load application program with the ETS in the actuator
   The ETS and the actuator now have identical parameterisation again.

6 The operating modes of the KNX SA SMI actuators

Two different operating modes can be set for each output:

- ▶ Venetian blind/external venetian blind
- ▶ Roller shutter/textile sun shading system

WARNING
The KNX SA SMIs do not have equipment, algorithms or similar features to switch off connected drives based on load. The danger of pinching and crushing must be prevented using on-site measures.

6.1 Venetian blind/external venetian blind

Internal and external venetian blinds are sun shading or dim-out elements with slats. They are controlled by movements and tilting of the slats. Internal and external venetian blinds differ in their purpose and physical dimensions.

In the `Output for venetian blind/external venetian blind` operating mode, the KNX SA SMI executes the **Up, Down and Tilt slats** move commands. Each output can be used for controlling an internal or external venetian blind.

Each output is equipped with group objects for move commands and status messages.

When moving to a particular slat position, the product may first move to the minimum or maximum slat position and then to the target slat position.

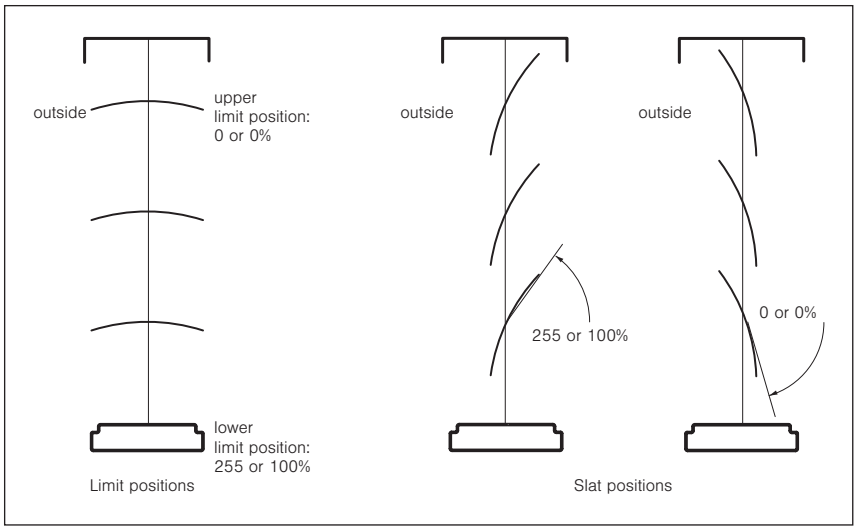


Fig. 18 Motor limit positions, slat positions

6.2 Roller shutter/fabric sun shading system

A roller shutter is a rolling closure for the additional closure of window and door openings, for example. Among other things, it provides visual, sun, intrusion and insect protection.

Fabric sun shading products consist of a movable mechanism with a fabric cover. Depending on the model, they provide visual privacy or sun shading.

In the `Output for roller shutter/Textile sun shading` operating mode, the KNX SA SMI executes the **Up and Down** move commands.

Each output can be used to control a roller shutter or a textile sun shading system.

Each output is equipped with group objects for move commands and status messages.

7 Parameter dialogue

For the KNX SA SMI actuators, the parameter dialogue is divided in the ETS into five groups:

Parameter group	Functions	Description
Device parameters	General actuator settings as well as the Bluetooth functions for AP devices	Section 7.1 on page 42
SMI parameters	Parameterisation of the SMI functionality. Power saving mode, motor list and allocation to the outputs.	Section 7.2 on page 43
Outputs	In the Outputs area, in addition to the operating mode and all parameters for movement behaviour, scenes and the reaction to the safety objects and automatic objects are also set for each output.	Section 7.3 on page 46
Inputs	This group is only visible for AP devices. The operating modes and the telegrams that are sent to the bus are parameterised here for the inputs on the device.	Section 7.4 on page 65
Safety objects	The general behaviour of the device is parameterised here to the safety objects of all four priority levels. The reaction of the individual outputs to safety objects is specified in the *Outputs* area.	Section 7.5 on page 71

The default values are shown in **bold** in the following parameter tables.

7.1 Device parameters

In this window, you will find the settings specific to the devices as well as the settings for the Bluetooth functions.

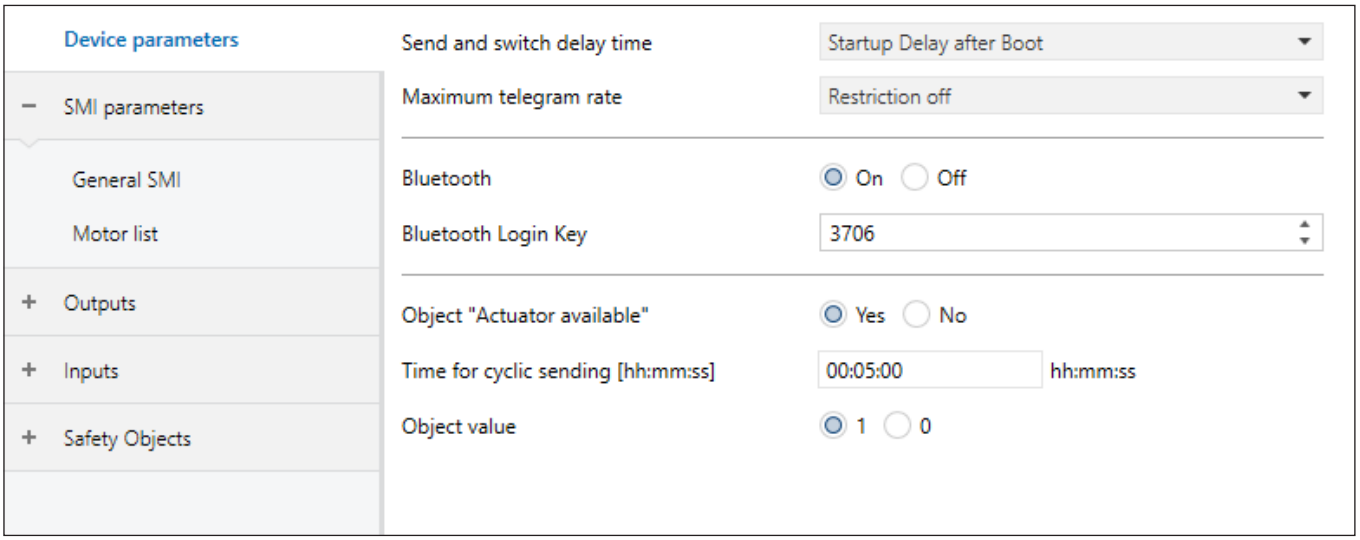


Fig. 19 Parameter dialogue: Device parameters

Parameter	Function	Values
Send and switch delay time	It can be specified here whether the device starts up with a delay after being switched on.	**Boot time**
		Boot time + 1 second
		Boot time + 3 second
		Boot time + 10 second
Maximum telegram rate	Restriction of the maximum number of telegrams that the device sends per second. The load of the KNX bus from the device can be reduced as required in this way.	**Restriction off**
		20 telegrams per second
		10 telegrams per second
		3 telegrams per second
		1 telegram per second
Bluetooth	The Bluetooth module can be switched off here. Operation via Bluetooth is then no longer possible.	**On**
		Off
Bluetooth Login Key	The login key for the Bluetooth operation can be set here. The key is requested when operating via Bluetooth.	0 : **3706** : 9999
Object "Actuator available"	Switches on the remaining parameters of this function and the GO "Actuator available". The actuator sends a status bit cyclically. If the actuator fails, this object remains off and can be evaluated in a KNX unit.	Yes
		No
Time for cyclical sending [hh:mm:ss]	Telegrams for the GO "Actuator available" can be sent repeatedly. The interval between two consecutive repeats can be parameterised here. Setting the value to 0 causes the telegram to be sent only once. This parameter is only visible if the *object "Actuator available"* is set to "Yes".	00:00:00 : **00:05:00** : 23:59:59
Object value	Specifies which value is sent on the GO "Actuator available".	**1**
		0

7.2 SMI parameters

7.2.1 SMI general

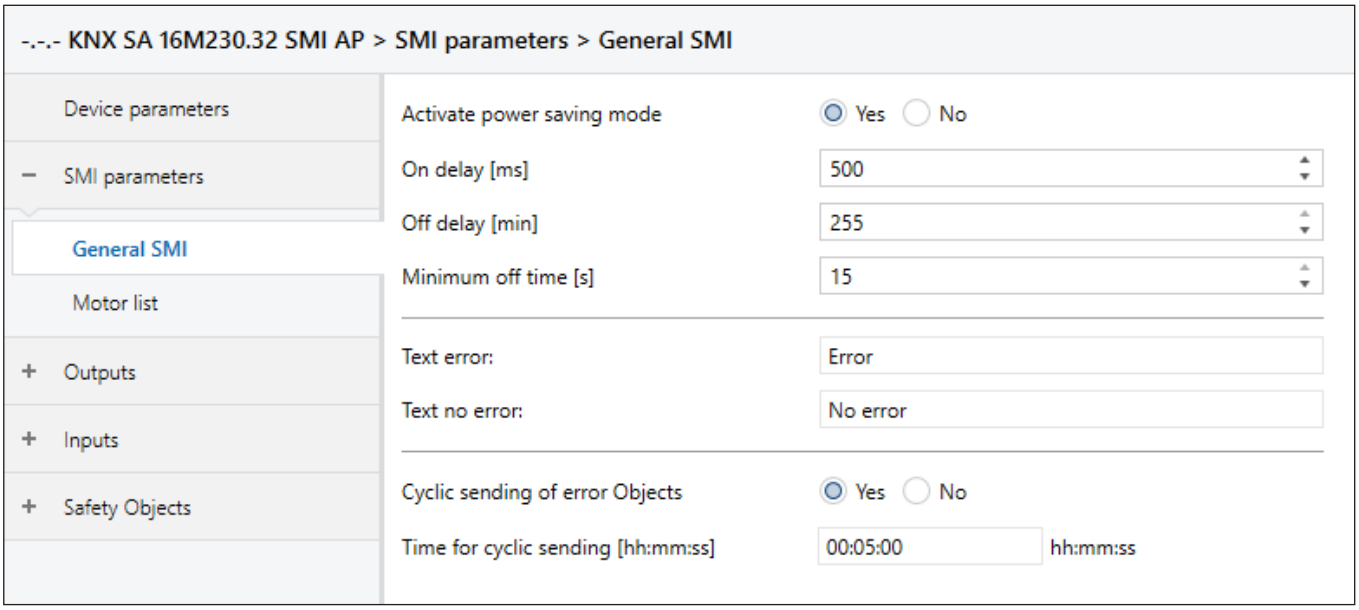


Fig. 20 Parameter dialogue: SMI Parameters → SMI general

Parameter	Function	Values
Activate power saving mode	Switches on the remaining parameters of this function. The power supply of the SMI motors is switched off via a relay.[1]	Yes **No**
On delay [ms]	If one of the motors receives a move command when the power supply is switched off, the power supply must first be switched on. The motors require a certain time to become ready for operation after the voltage is switched on. This time can be parameterised here.	0 : **500** : 2550
Off delay [min]	If none of the connected motors are actuated for the *Off delay* time, the power supply for all motors is switched off.	1 : **255**
Minimum off time [s]	After the power supply is switched off, a minimum off time is maintained before switching on again.	1 : **15**
Text error:	This text is sent to the GO "Output collective error message text" in the event of a fault.[2]	**Error**
Text no error:	This text is sent to the GO "Output collective error message text" when at rest.[2]	**No Error**
Cyclical transmission of error objects	Specifies whether telegrams for fault messages are sent repeatedly. The time interval is set in the parameter *Time for cyclic sending [hh:mm:ss]*.	Yes **No**
Time for cyclical sending [hh:mm:ss]	Telegrams for fault messages can be sent repeatedly. The interval between two consecutive repeats can be parameterised here. Setting the value to 0 causes the telegram to be sent only once.	00:00:00 : **00:05:00** : 23:59:59

[1] this functionality must be released by the motor manufacturer

[2] Maximum 14 characters (= 14 bytes)

7.2.2 Motor list

For the commissioning of the device, an ETS DCA app (*see Section 4.3.3 on page 15*) and a smartphone app (*see Section 4.3.3 on page 15*) are available.

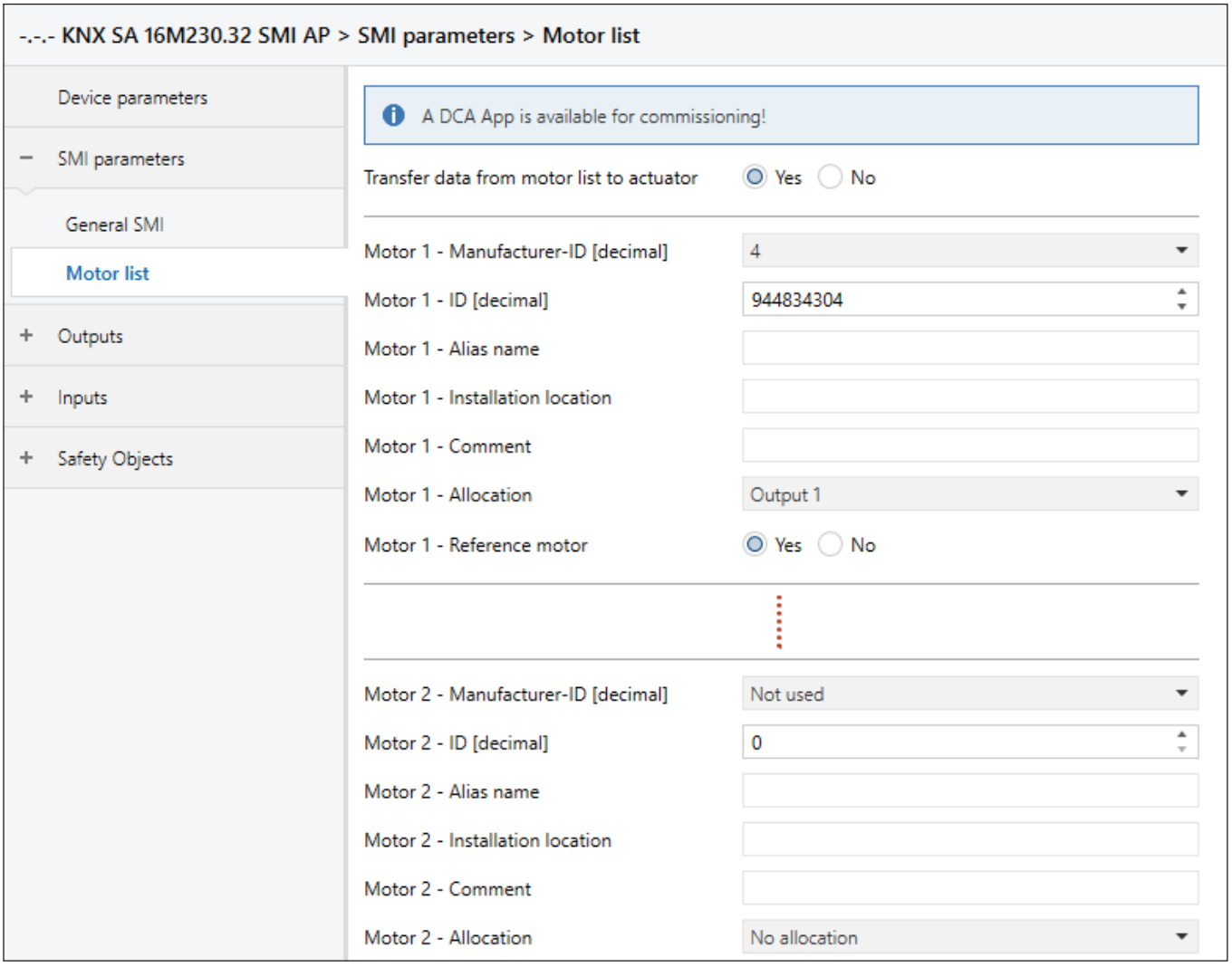


Fig. 21 Parameter dialogue: SMI parameters Motor list

Parameter	Function	Values
Transfer data from motor list to actuator	When loading the device with the ETS, the data from the motor list is transferred to the device and the motor list in the device is overwritten. ATTENTION: Changes in the device which are made with the smartphone app are overwritten.	Yes
		No
Motor n - Manufacturer ID [decimal]	Motor manufacturer ID input. The ID can be found on the motor label.	**Not used** 1 : 15
Motor n - Key ID [decimal]	Key manufacturer ID input. The ID can be found on the motor label.[1]	**0** : 4294967295
Motor n - Alias name	Freely definable alias name [2]	
Motor n - Installation location	Freely definable installation location [2]	
Motor n - Comment	Freely definable comment [2]	
Motor n - Allocation	Specifies the output to which the motor should be allocated.	**Not allocated** Output 1 : Output 16
Cyclical transmission of error objects	Specifies whether the motor should be used as a reference motor for the output. The status of the blind length (among others) is determined via the reference motor. If no reference motor is parameterised, the device itself selects a motor as a reference.	Yes
		No

[1] If the key ID is applied as a hexidecimal, it must be converted into a corresponding decimal value (the hexidecimal value can be entered in the DCA app).

[2] Maximum 31 characters (visible in the smartphone app and the DCA app)

If several motors are allocated to an output, the controlled sun shading products should have the same construction height. Only then is correct positioning via the actuator possible.

7.3 Outputs

7.3.1 Outputs, general

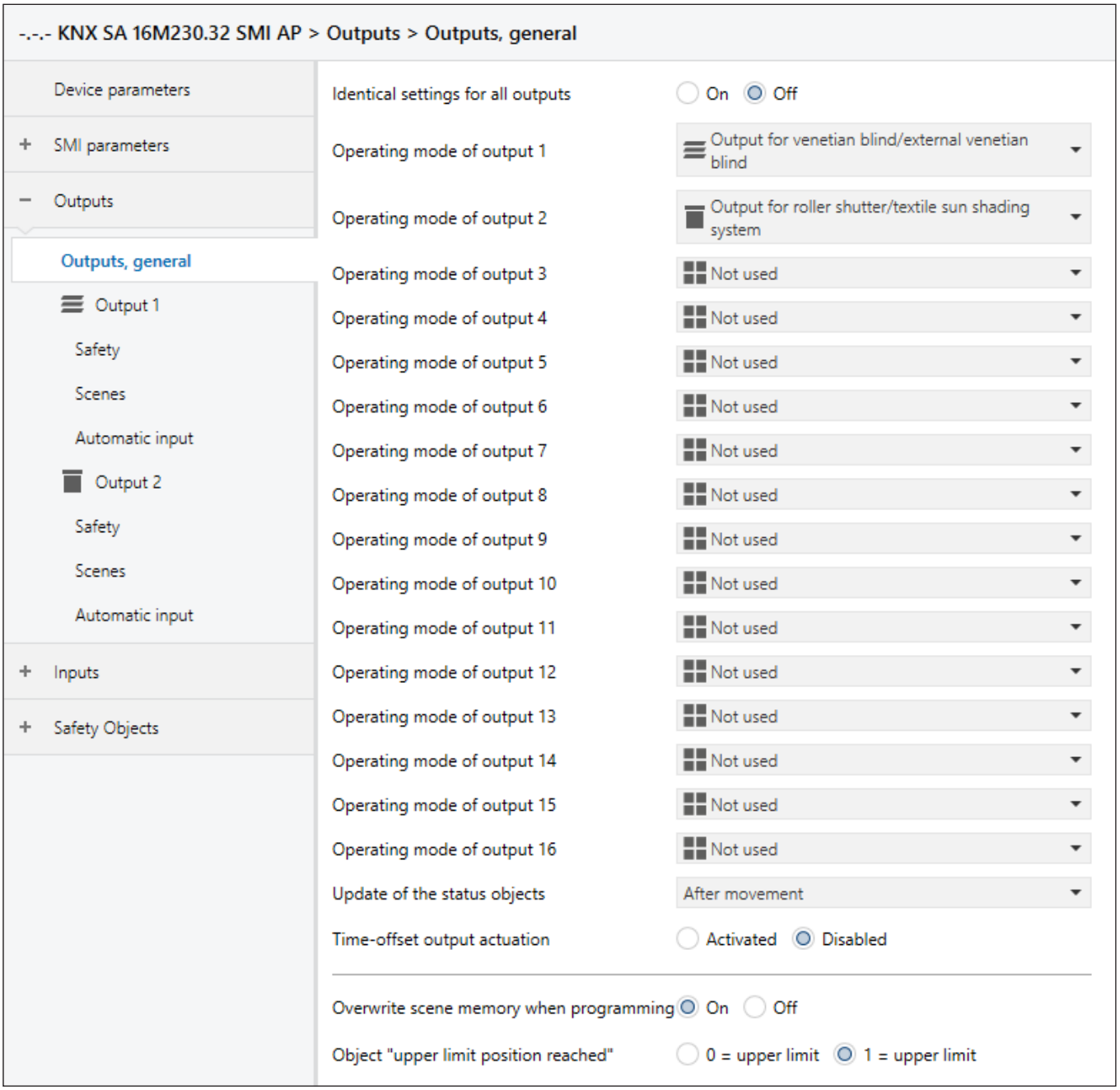


Fig. 22 Parameter dialogue: Outputs → Outputs general

Parameter	Function	Values
Parameterise all outputs identically	Here, the user can specify whether all outputs are to be given identical parameters. The parameters for the individual outputs are then hidden. There is still only one parameter set for all outputs.	**On**
		Off
Operating mode Output n	Operating mode of the output, distinction made between various sun shading product types (see following chapters)	**Not used**
		Output for venetian blind/external venetian blind
		Output for roller shutter/ textile sun shading
Update of the status objects	Here the user can set when the ***status GOs*** of the outputs are updated. After movement, an updated status object is always sent (regardless of the settings).	**After movement**
		During movement: 1 s interval
		During movement: 2 s interval
		During movement: 5 s interval
		During movement: 10 s interval
Time-offset output actuation	If this is set to “Activated”, there is a minimum pause of 20 ms the outputs (= motor groups) of the actuator.	**Disabled**
		Activated
Overwrite scene memory when programming	Here the user can set whether the values for the scenes of the outputs are to be overwritten with the values of the ETS project when the application is loaded with the ETS.	**On**
		Off
Object “upper limit position reached”	Here the user can set what value the ***GO Upper limit position reached*** sends when the upper limit position is reached.	0 = upper limit
		1 = upper limit

7.3.2 Output for venetian blind/external venetian blind

In the `Venetian blind/External venetian blind` operating mode, the KNX SA SMI executes the **Up, Down and Tilt slats** move commands. Each output can be used for controlling an internal or external venetian blind.

The functions of the outputs are explained here using the example of the *Venetian blind/external venetian blind* operating mode. As there are omitted or additional parameters and group objects for the other operating modes, these are explained separately in later chapters.

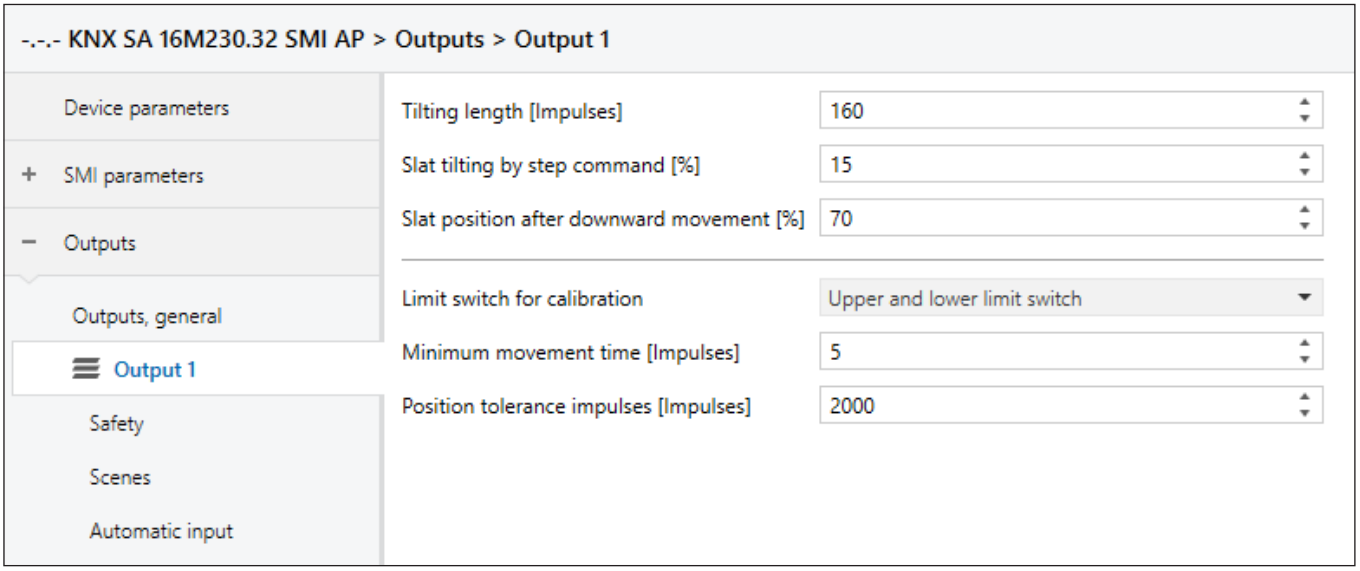


Fig. 23 Parameter dialogue: Outputs → Output n (for venetian blind/external venetian blind)

Parameter	Function	Values
Turn pulses [Pulse]	For this parameter, the turn pulses must be set which a venetian blind or an external venetian blind require to tilt between slat positions 0 and 100% (tilt pulses can be learned with the DCA app).	0 : **160** : 65535
Slat tilting by step command [%]	This parameter defines the percentage by which a sun shading product is raised or lowered after a Stop/Step command telegram. The parameter value is based on the tilt pulse.	0 : **15** : 100
Slat position after downward movement [%]	After a manual operation, it is often useful to automatically turn up the slats of an external venetian blind when the lower limit position is reached. In this way, only one operation is needed to achieve a product position that provides glare control while also permitting visibility to the outside. This parameter defines the slat position that is set after the lower limit position is reached. The parameter value is based on the tilt pulse.	0 : **70** : 100
Limit switches for calibration	Here, you can select which end switches of the product should be evaluated and used for calibration (e.g. after a loss of power)	No limit switch available **Upper and lower limit switches** Upper limit switch Lower limit switch

Minimum travel time [pulses]	This parameter can be used to determine the smallest distance for which a move command will actually be triggered. The value set here must be smaller than all tilt pulses parameterised for the actuator.	0 : **5** : 65535
Position tolerance [pulses]	The tolerance range for the pulse of the product can be defined here. This value is important for the evaluation of internal fault detection. It defines the maximum permissible deviation from the exact position in both directions of travel.	0 : **2000** : 65535

7.3.2.1 Safety

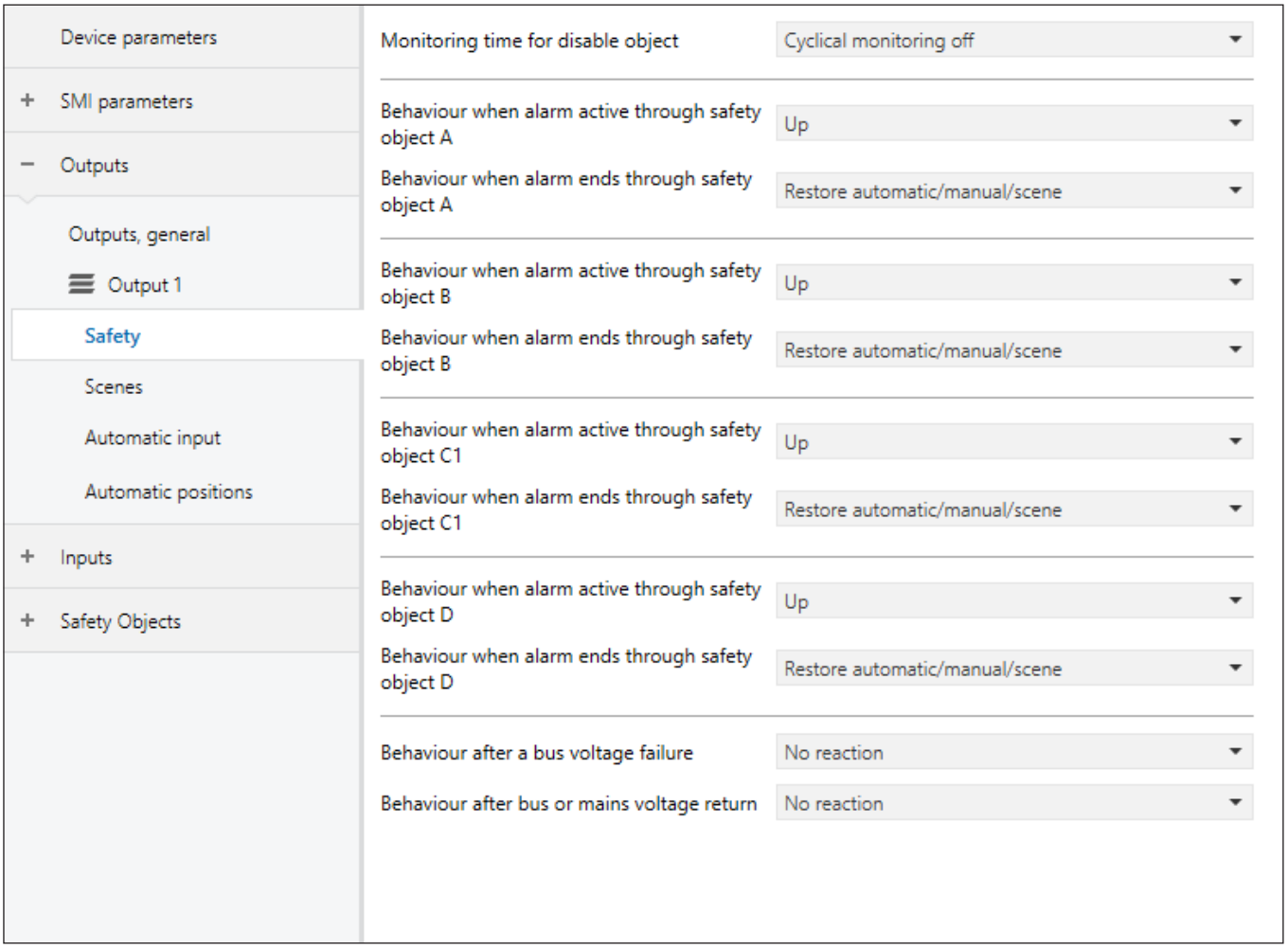


Fig. 24 Parameter dialogue: Outputs → Output n (for venetian blind/external venetian blind) → Safety

The safety objects A, B and D are available on the device once. The safety object C is available separately for each output.

The response to an alarm on the individual safety objects must be parameterised individually for each output.

For the REG devices, the parameters *Safety object\Safety object n\"Behaviour after bus voltage drop"* and *Outputs\Output n\Safety\ "Behaviour after bus voltage drop"* are only available if the parameter *SMI Parameters\SMI general\"Activate power saving mode"* is parameterised with "No" (*see Section 7.2.1 on page 43*).

Parameter	Function	Values
Disable object monitoring time	Monitors whether telegrams are received on the ***GO Disable object*** of the output. The disable object of the output must receive at least one telegram within this time interval. If this time is exceeded without a telegram having been received, the product control is disabled and running movements are stopped. The blocking is cleared after a 0-telegram at the disable object.	**Cyclical monitoring off**
		10 seconds
		1 minute
		2 minutes
		5 minutes
		10 minutes
Behaviour when *alarm active* through safety object A	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object A*** should move in case of alarm.	**0** : 100
Slat position [%]	Slat position, to which ***GO Safety object A*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object A	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour when *alarm active* through safety object B	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object B*** should move in case of alarm.	**0** : 100
Slat position [%]	Slat position, to which ***GO Safety object B*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object B	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour when *alarm active* through safety object Cn	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object Cn*** should move in case of alarm.	**0** : 100

Slat position [%]	Slat position, to which ***GO Safety object Cn*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object Cn	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour when *alarm active* through safety object D	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object D*** should move in case of alarm.	**0** : 100
Slat position [%]	Slat position, to which ***GO Safety object D*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object D	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour after the bus voltage fails	This parameter defines the behaviour of the output after the bus voltage fails.	**No reaction**
		Up
		Down
		Stop
Behaviour after the bus or mains voltage returns	This parameter defines the behaviour of the output after the mains voltage returns.	**No reaction**
		Up
		Down
		Stop

7.3.2.2 Scenes

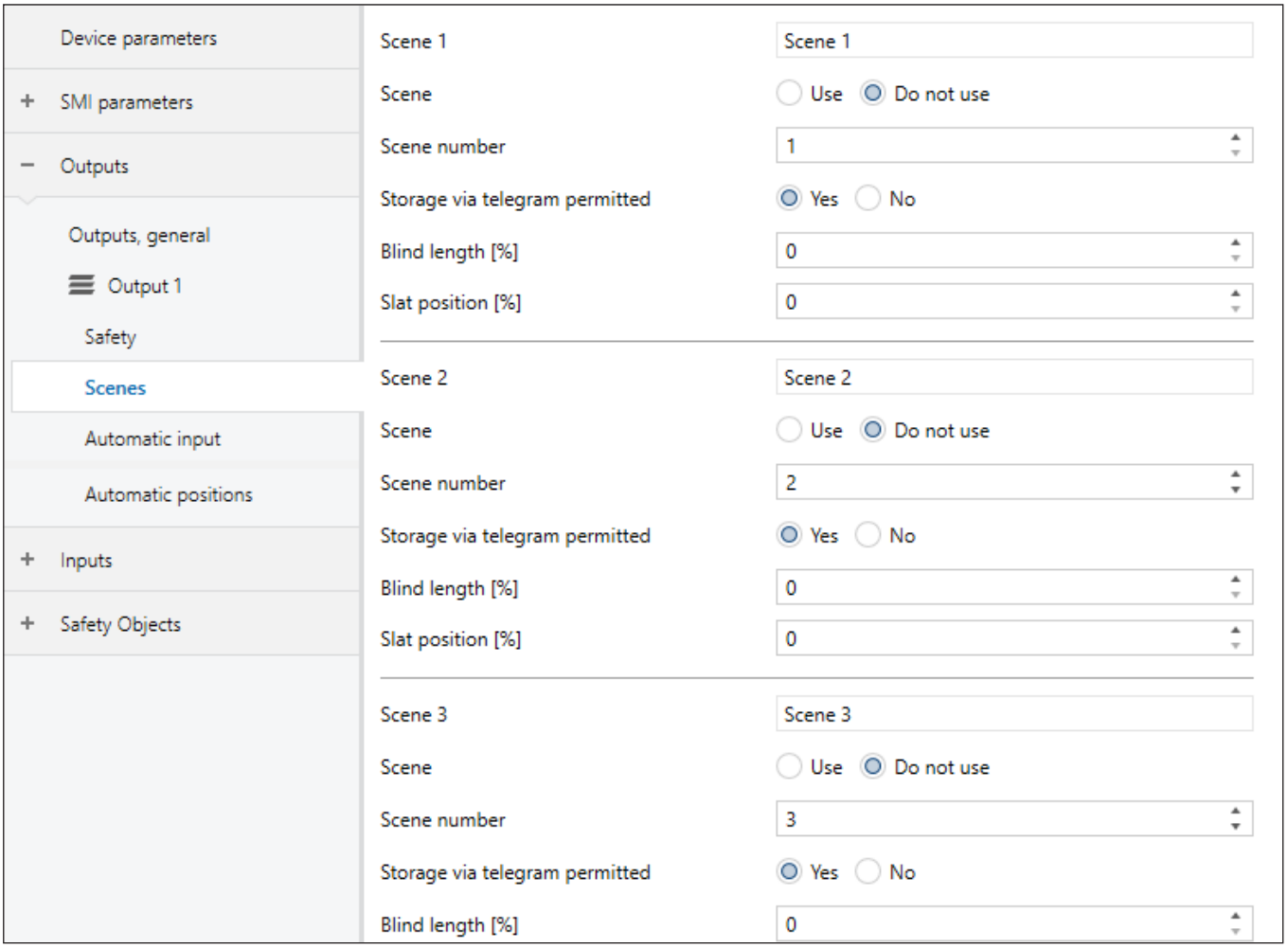


Fig. 25 Parameter dialogue: Outputs → Output n (for venetian blind/external venetian blind) → Scenes

Parameter	Function	Values
Scene n (8 scenes can be defined)	Text as designation of the scene (purely for information). The text may have a maximum of 30 characters.	**Scene n**
Scene	Specifies whether the scene is to be used.	**Do not use**
		Use
Scene number	Scene number that must be received on ***GO Scenes*** of the output for the scene to be executed. Each scene number may only be used once.	**1** : 64
Storage via telegram permitted	Specifies whether the value of the scene should be learned by ***GO Scenes***.	No
		Yes
Blind length [%]	Blind length to which the blind is moved when the scene is activated.	**0** : 100
Slat position [%]	Slat position to which the blind is moved when the scene is activated.	**0** : 100

7.3.2.3 Automatic input

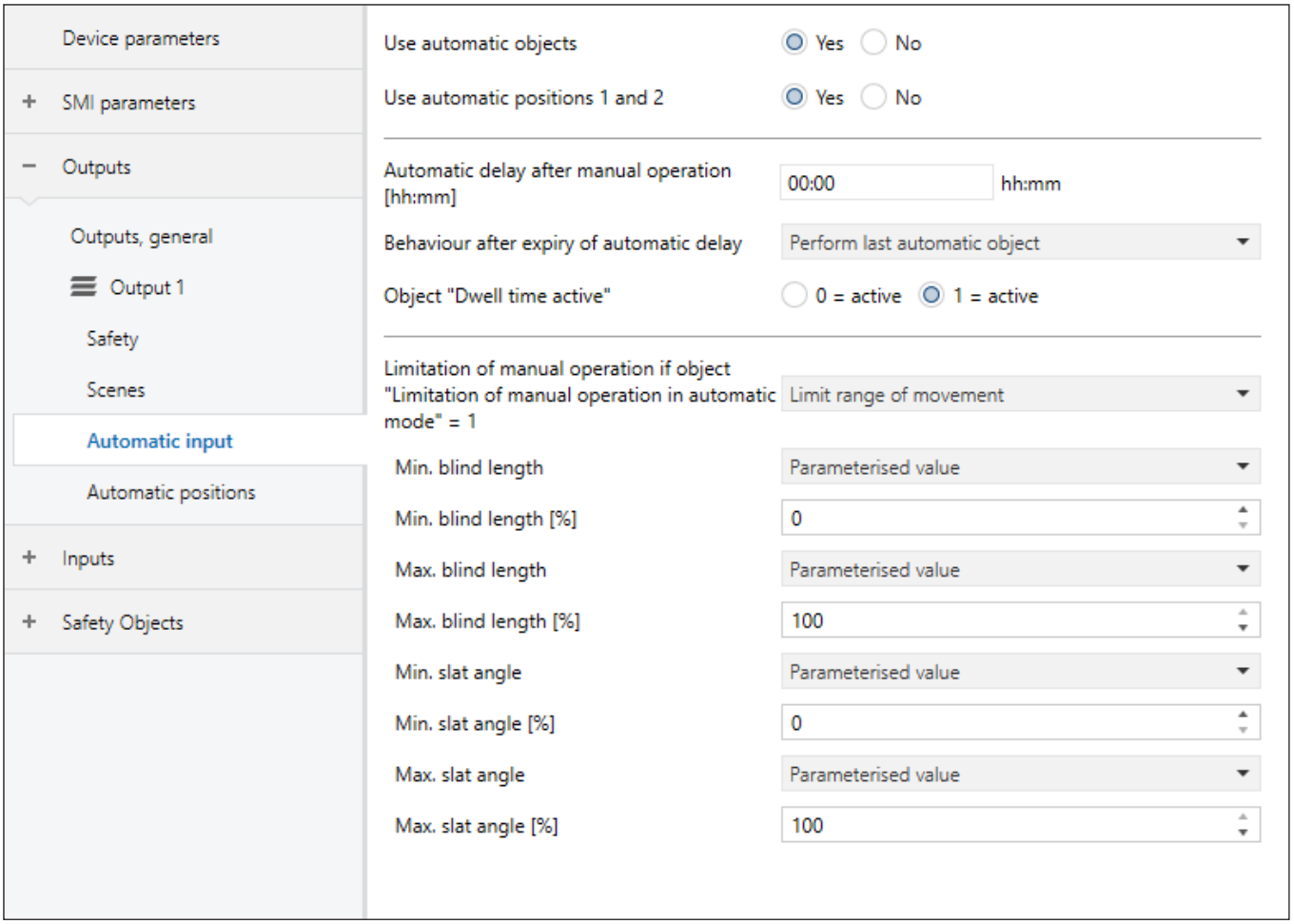


Fig. 26 Parameter dialogue: Outputs → Output n (for venetian blind/external venetian blind) → Automatic input

Parameter	Function	Values
Use automatic objects	Switches on the remaining parameters of this page and the automatic GOs.	**No**
		Yes
Use automatic positions 1 and 2	Specifies whether *automatic positions 1 and 2* are used. Switches on additional parameters.	**No**
		Yes
Automatic delay after manual operation [hh:mm]	After a manual move command the automatic delay starts running. The last position command is repeated after this time expires.	**00:00** : 23:59
Behaviour after expiry of automatic delay	Specifies what action is to take place after the automatic delay (dwell time) has expired.	No reaction
		Up
		Down
		Perform last automatic object
Object “Dwell time active”	Specifies the value which the ***GO Dwell time active*** sends, as long as the automatic delay (dwell time) is running.	0 = active
		1 = active
Limitation of manual operation if object “Limitation of manual operation in automatic mode” = 1	Specifies the range in which the blind length may move if ***GO Limitation of manual operation in automatic mode*** is active.	**Disable manual operation and scenes**
		Disable changing of the blind length
		Limit range of movement

Min. blind length	Specifies the minimum blind length if ***GO Limitation of manual operation in automatic mode*** is active.	No restriction **From automatic blind length object** Parameterised value
Min. blind length [%]	Value used if the parameter *Min. blind length* has been set to *Parameterised value*.	**0** : 100
Max. blind length	Specifies the maximum blind length if **GO Limitation of manual operation in automatic mode** is active.	**No restriction** From automatic blind length object Parameterised value
Max. blind length [%]	Value used if the parameter *Max. blind length* has been set to *Parameterised value*.	0 : **100**
Min. slat angle	Specifies the minimum slat angle if **GO Limitation of manual operation in automatic mode** is active.	No restriction **From automatic slat angle object** Parameterised value
Min. slat angle [%]	Value used if the parameter *Min. slat angle* has been set to *Parameterised value*.	**0** : 100
Max. slat angle	Specifies the maximum slat angle if **GO Limitation of manual operation in automatic mode** is active.	**No restriction** From automatic slat angle object Parameterised value
Max. slat angle [%]	Value used if the parameter *Max. slat angle* has been set to *Parameterised value*.	0 : **100**

7.3.2.4 Automatic positions

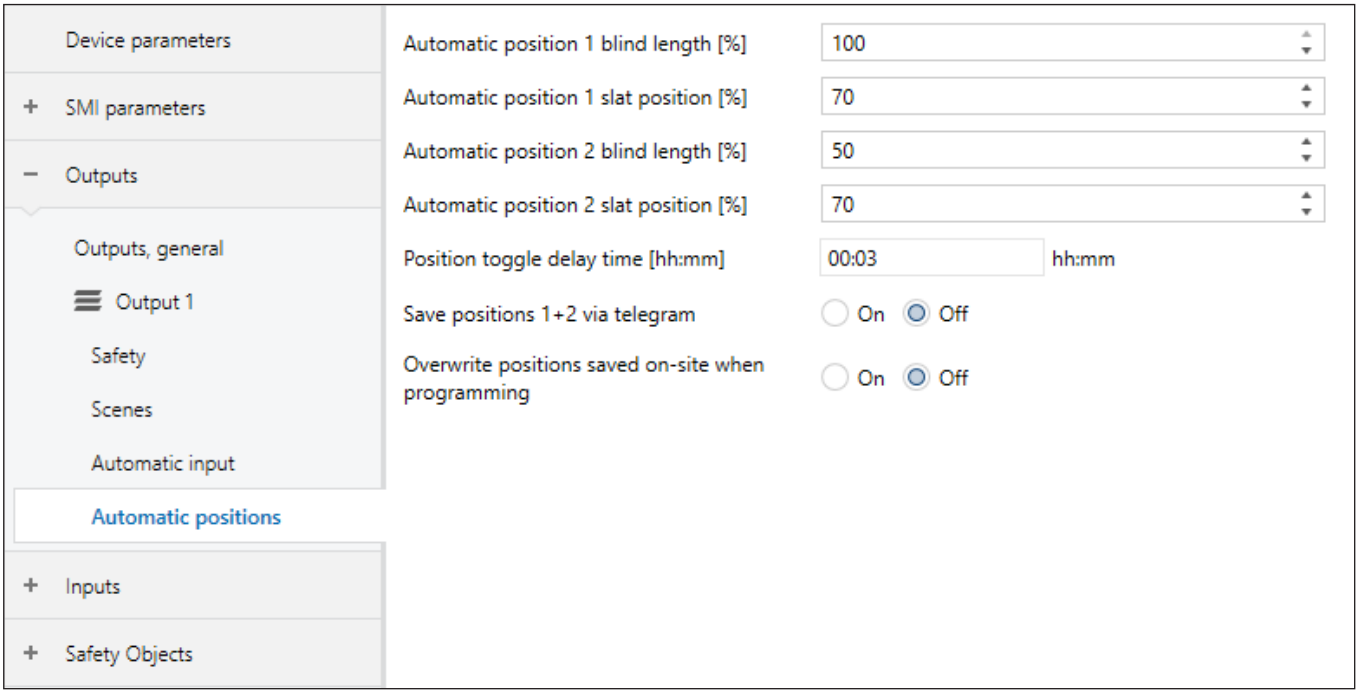


Fig. 27 Parameter dialogue: Outputs → Output n (for venetian blind/external venetian blind) → Automatic positions

Parameter	Function	Values
Automatic position 1 blind length [%]	Specifies the blind length for automatic position 1. If the option *Save positions 1+2 via telegram* is active, the value set here may differ from the value in the device.	0 : **100**
Automatic position 1 slat position [%]	Specifies the slat position for automatic position 1. If the option *Save positions 1+2 via telegram* is active, the value set here may differ from the value in the device.	0 : **70** : 100
Automatic position 2 blind length [%]	Specifies the blind length for automatic position 2. If the option *Save positions 1+2 via telegram* is active, the value set here may differ from the value in the device.	0 : **50** : 100
Automatic position 2 slat position [%]	Specifies the slat position for automatic position 2. If the option *Save positions 1+2 via telegram* is active, the value set here may differ from the value in the device.	0 : **70** : 100
Position toggle delay time [hh:mm]	If a 1-telegram is received on the ***GO Automatic position toggle***, the sun shading product moves to the position that was last received on ***Automatic blind length/Automatic slat position*** after the *Position toggle delay time* expires. If a telegram has not yet been received for ***Automatic blind length/Automatic slat position***, the sun shading product moves to the saved Position 1. If a 0-telegram is received on the ***GO Automatic position toggle***, the sun shading product moves to Position 1 after the *Position toggle delay time* expires. The *Position toggle delay* is always started after the ***Position toggle*** telegram received last, even if the time is already running.	00:00 : **00:03** : 59:59
Store position 1+2 via telegram	On: The current product position is saved after a telegram to the ***GO Save position 1/2***. Off: A telegram to the ***GO Save position 1/2*** causes no change to the position memory.	On **Off**
Overwrite positions stored on-site when programming	On: Positions 1 and 2 saved in the device are overwritten with the parameterised values when the parameters are being programmed. Off: Positions 1 and 2 saved in the device are not overwritten when the parameters are being programmed.	On **Off**

7.3.3 Output for roller shutter/textile sun shading system

In the `Roller shutter/Textile sun shading` operating mode, the KNX SA SMI executes the **Up and Down move commands.**

Each output can be used to control a roller shutter or a textile sun shading system.

The functions of the outputs are explained here in the example of the *Roller shutter/Textile sun shading* operating mode. As there are omitted or additional parameters and group objects for the other operating modes, these are explained separately in later chapters.

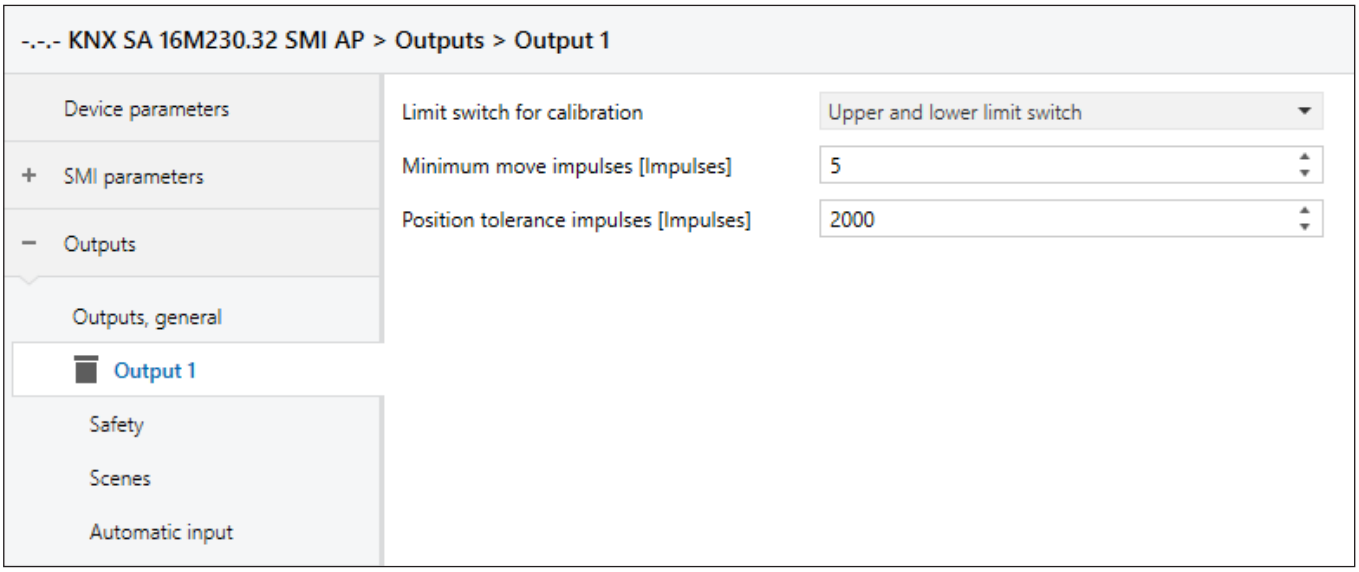


Fig. 28 Parameter dialogue: Outputs → Output n (for roller shutter/Textile sun shading)

Parameter	Function	Values
Limit switches for calibration	Here, you can select which end switches of the product should be evaluated and used for calibration (e.g. after a loss of power)	**No limit switch available**
		Upper and lower limit switches
		Upper limit switch
		Lower limit switch
Minimum travel time [pulses]	This parameter can be used to determine the smallest distance for which a move command will actually be triggered. The value set here must be smaller than all tilt pulses parameterised for the actuator.	0 : **5** : 65535
Position tolerance [pulses]	The tolerance range for the pulse of the product can be defined here. This value is important for the evaluation of internal fault detection. It defines the maximum permissible deviation from the exact position in both directions of travel.	0 : **2000** : 65535

7.3.3.1 Safety

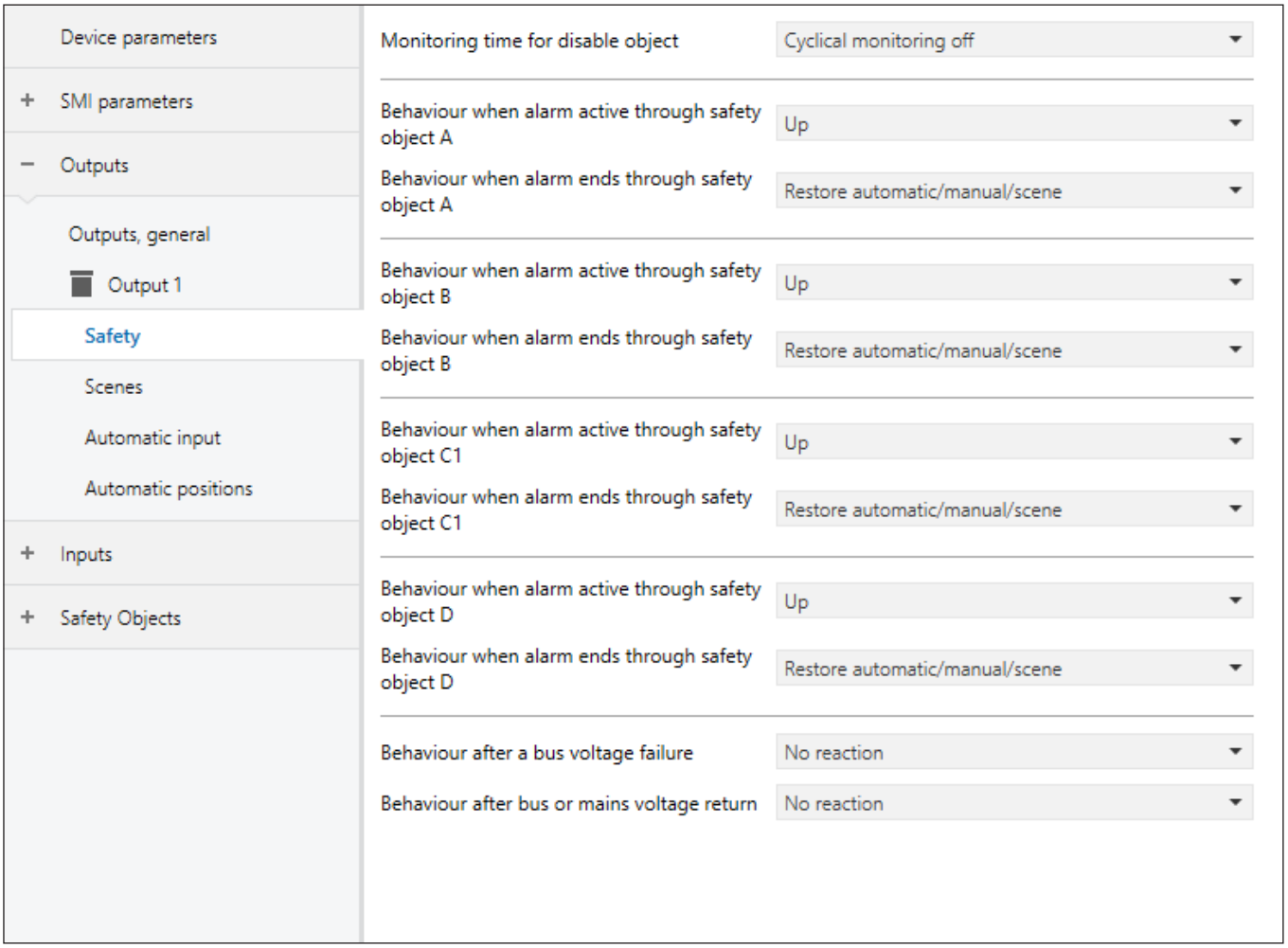


Fig. 29 Parameter dialogue: Outputs → Output n (for roller shutter/Textile sun shading) Safety

The safety objects A, B and D are available on the device once. The safety object C is available separately for each output.

The response to an alarm on the individual safety objects must be parameterised individually for each output.

For the REG devices, the parameters *Safety object\Safety object n\"Behaviour after bus voltage drop"* and *Outputs\Output n\Safety\ "Behaviour after bus voltage drop"* are only available if the parameter *SMI Parameters\SMI general\"Activate power saving mode"* is parameterised with "No" (*see Section 7.2.1 on page 43*).

Parameter	Function	Values
Disable object monitoring time	Monitors whether telegrams are received on the ***GO Disable object*** of the output. The disable object of the output must receive at least one telegram within this time interval. If this time is exceeded without a telegram having been received, the product control is disabled and running movements are stopped. The blocking is cleared after a 0-telegram at the disable object.	**Cyclical monitoring off**
		10 seconds
		1 minute
		2 minutes
		5 minutes
		10 minutes
Behaviour when *alarm active* through safety object A	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object A*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object A	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour when *alarm active* through safety object B	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object B*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object B	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour when *alarm active* through safety object Cn	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object Cn*** should move in case of alarm.	**0** : 100

Behaviour when *alarm ends* through safety object Cn	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour when *alarm active* through safety object D	Output channels execute the move command that has been set for *Alarm active*. Afterwards, only commands from safety objects of higher priority or from the disable object are processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
Blind length [%]	Blind length, to which ***GO Safety object D*** should move in case of alarm.	**0** : 100
Behaviour when *alarm ends* through safety object D	Outputs execute the move command that has been set for Alarm end. Lower priority commands then continue to be processed.	No reaction
		Stop
		Up
		Down
		Move to parameterised position
		Perform last automatic object
		Restore automatic/manual/scene
Behaviour after the bus voltage fails	This parameter defines the behaviour of the output after the bus voltage fails.	**No reaction**
		Up
		Down
		Stop
Behaviour after the bus or mains voltage returns	This parameter defines the behaviour of the output after the mains voltage returns.	**No reaction**
		Up
		Down
		Stop

7.3.3.2 Scenes

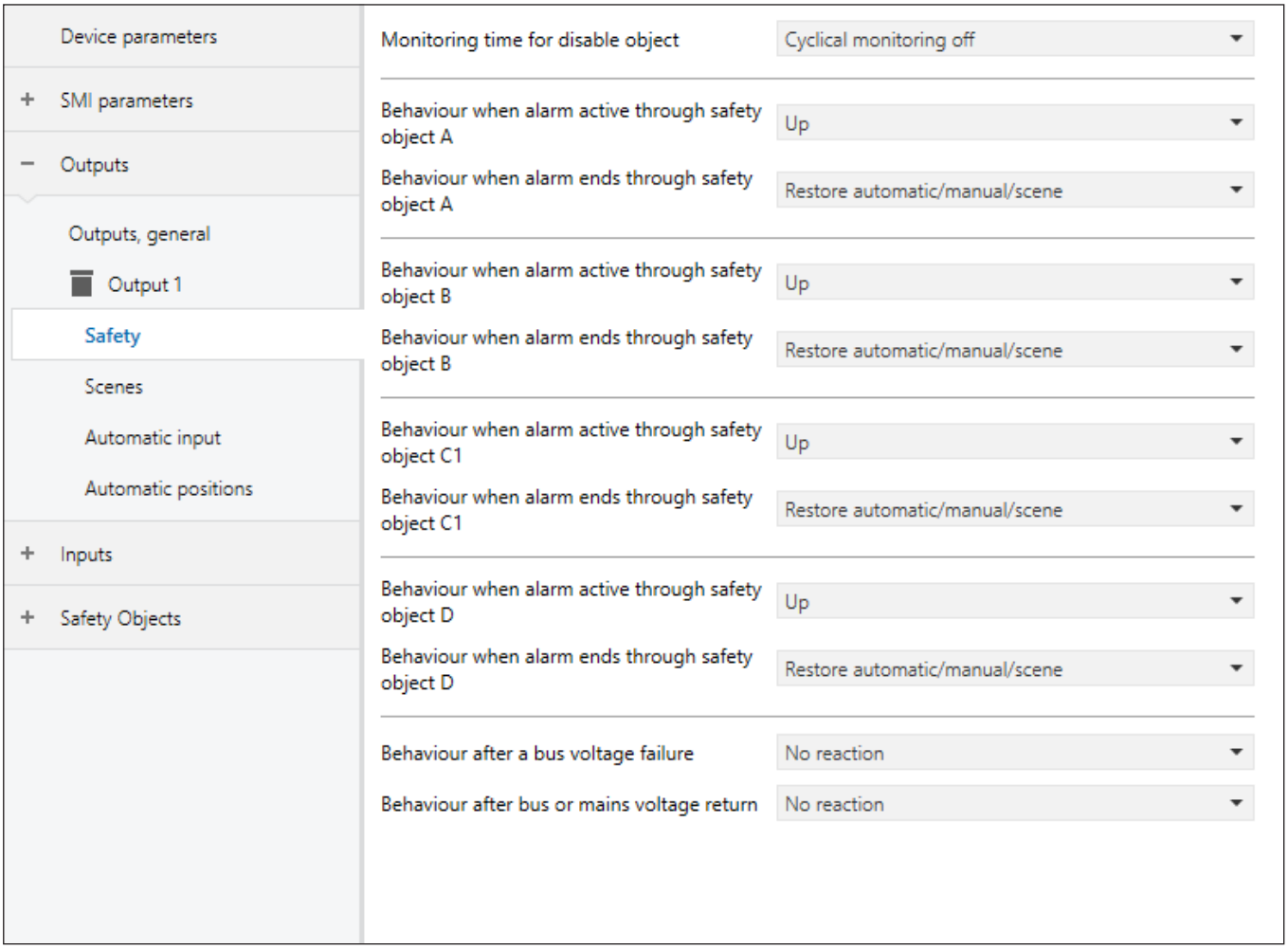


Fig. 30 Parameter dialogue: Outputs → Output n (for roller shutter/Textile sun shading) Scenes

Parameter	Function	Values
Scene n (8 scenes can be defined)	Text as designation of the scene. The text may have a maximum of 30 characters.	**Scene n**
Scene	Specifies whether the scene is to be used.	**Do not use**
		Use
Scene number	Scene number that must be received on ***GO Scenes*** of the output for the scene to be executed. Each scene number may only be used once.	**1** : 64
Storage via telegram permitted	Specifies whether the value of the scene should be learned by ***GO Scenes***.	No
		Yes
Blind length [%]	Blind length to which the blind is moved when the scene is activated.	**0** : 100

7.3.3.3 Automatic input

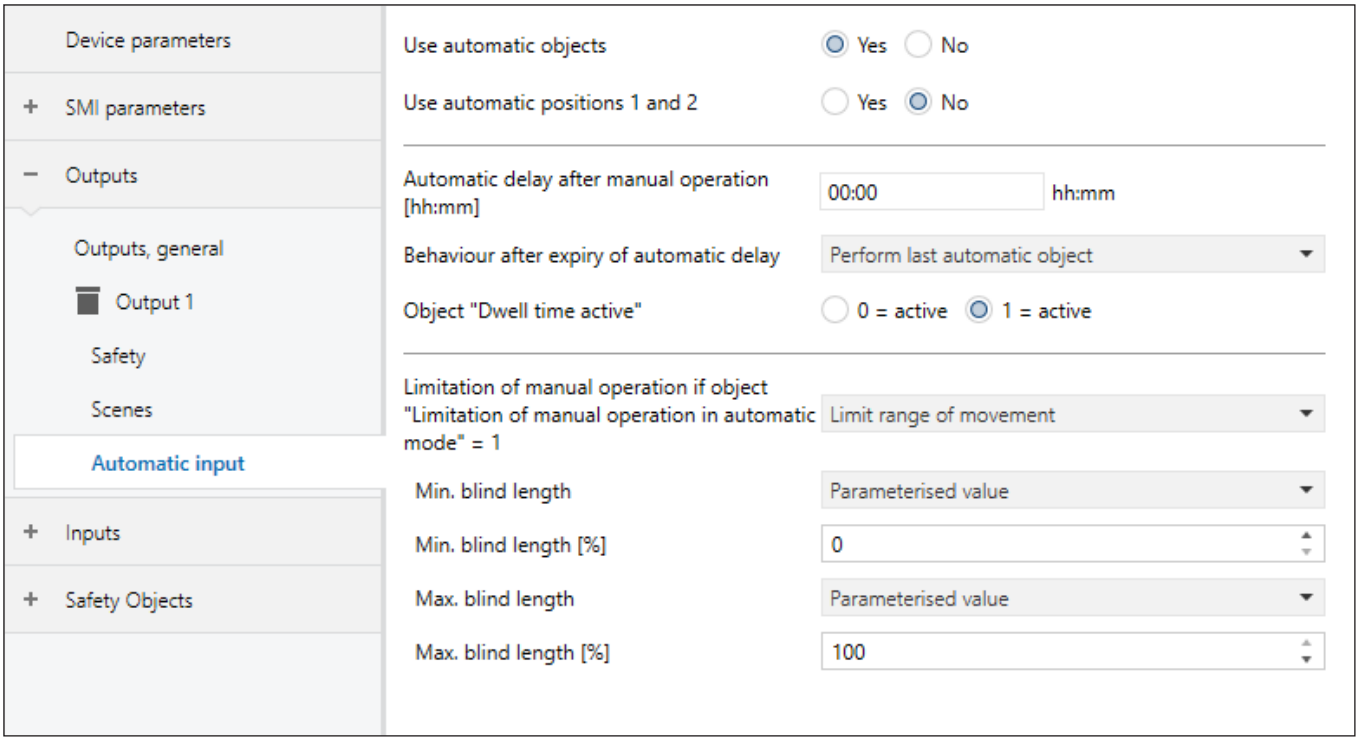


Fig. 31 Parameter dialogue: Outputs → Output n (for roller shutter/Textile sun shading) Automatic input

Parameter	Function	Values
Use automatic objects	Switches on the remaining parameters of this page and the automatic GOs.	**No**
		Yes
Use automatic positions 1 and 2	Specifies whether *automatic positions 1 and 2* are used. Switches on additional parameters.	**No**
		Yes
Automatic delay after manual operation [hh:mm]	After a manual move command the automatic delay starts running. The last position command is repeated after this time expires.	**00:00** : 23:59
Behaviour after expiry of automatic delay	Specifies what action is to take place after the automatic delay (dwell time) has expired.	No reaction
		Up
		Down
		Perform last automatic object
Object “Dwell time active”	Specifies the value which the ***GO Dwell time active*** sends, as long as the automatic delay (dwell time) is running.	0 = active
		1 = active
Limitation of manual operation if object “Limitation of manual operation in automatic mode” = 1	Specifies the range in which the blind length may move if ***GO Limitation of manual operation in automatic mode*** is active.	**Disable manual operation and scenes**
		Disable changing of the blind length
		Limit range of movement
Min. blind length	Specifies the minimum blind length if **GO Limitation of manual operation in automatic mode** is active.	No restriction
		From automatic blind length object
		Parameterised value
Min. blind length [%]	Value used if the parameter *Min. blind length* has been set to *Parameterised value*.	**0** : 100

Max. blind length	Specifies the maximum blind length if **GO Limitation of manual operation in automatic mode** is active.	No restriction **From automatic blind length object** Parameterised value
Max. blind length [%]	Value used if the parameter *Max. blind length* has been set to *Parameterised value*.	**0** : 100

7.3.3.4 Automatic positions

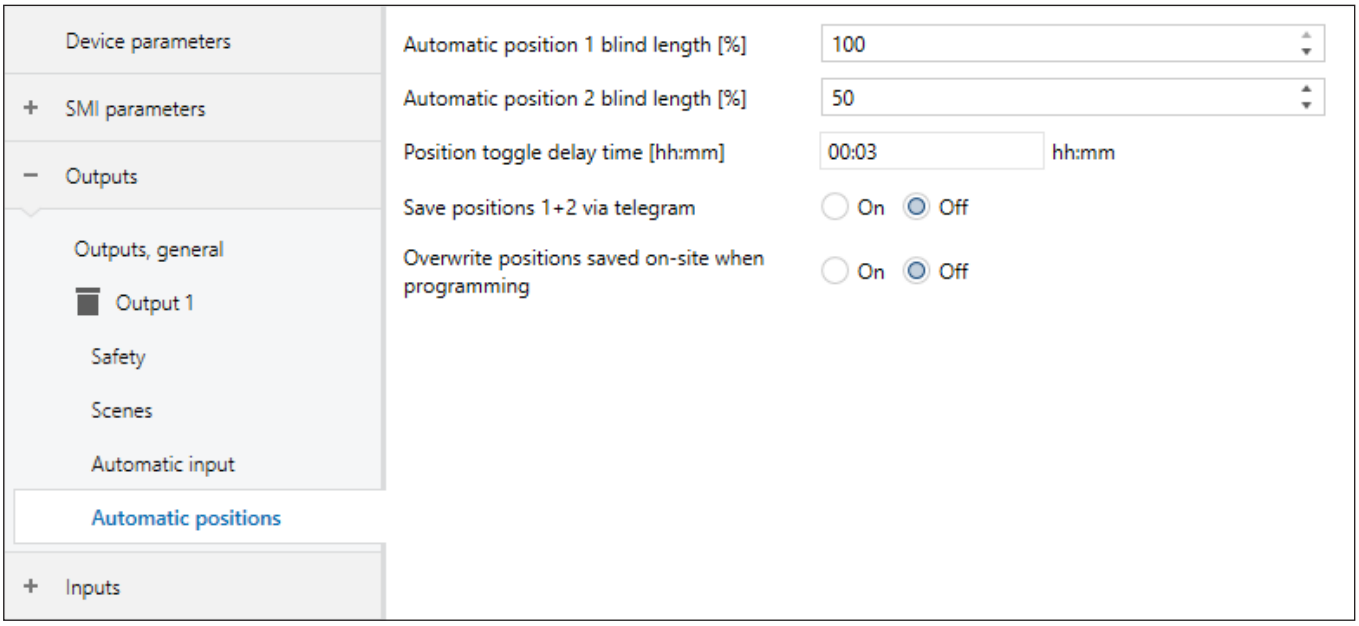


Fig. 32 Parameter dialogue: Outputs → Output n (for roller shutter/Textile sun shading) Automatic positions

Parameter	Function	Values
Automatic position 1 blind length [%]	Specifies the blind length for automatic position 1. If the option *Save positions 1+2 via telegram* is active, the value set here may differ from the value in the device.	0 : **100**
Automatic position 2 blind length [%]	Specifies the blind length for automatic position 2. If the option *Save positions 1+2 via telegram* is active, the value set here may differ from the value in the device.	0 : **50** : 100
Position toggle delay time [hh:mm]	If a 1-telegram is received on the ***GO Automatic position toggle***, the sun shading product moves to the position that was last received on ***Automatic blind length/Automatic slat position*** after the *Position toggle delay time* expires. If a telegram has not yet been received for ***Automatic blind length/Automatic slat position***, the sun shading product moves to the saved Position 1. If a 0-telegram is received on the ***GO Automatic position toggle***, the sun shading product moves to Position 1 after the *Position toggle delay time* expires. The *Position toggle delay* is always started after the ***Position toggle*** telegram received last, even if the time is already running.	00:00 : **00:03** : 59:59

Store position 1+2 via telegram	On: The current product position is saved after a telegram to the ***GO Save position 1/2***. Off: A telegram to the ***GO Save position 1/2*** causes no change to the position memory.	On
		Off
Overwrite positions stored on-site when programming	On: Positions 1 and 2 saved in the device are overwritten with the parameterised values when the parameters are being programmed. Off: Positions 1 and 2 saved in the device are not overwritten when the parameters are being programmed.	On
		Off

7.4 Inputs

The surface mounted devices (AP) are equipped with inputs.
The inputs (input terminals) are always parameterised in pairs.
One input supports different operating modes.

7.4.1 Inputs, general

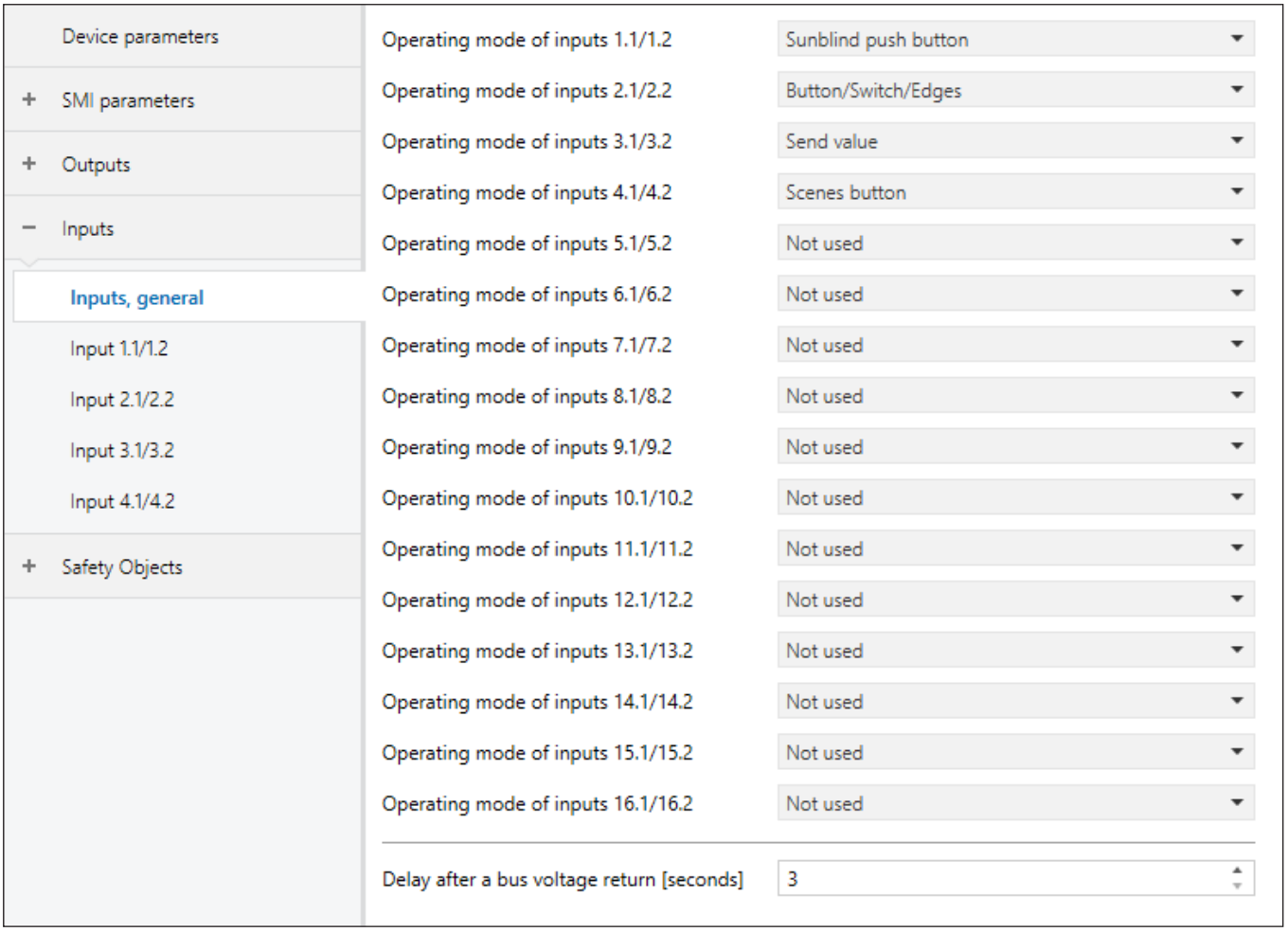


Fig. 33 Parameter dialogue: Inputs → Inputs general

Parameter	Function	Values
Operating mode Inputs n.1/n.2	Specifies the operating mode of a pair of inputs. The Sunblind push button operating mode requires two inputs (Up / Down). The other operating modes require just one input each.	**Not used**
		Sunblind push button
		Button/Switch/Edges
		Sensor
		Scenes button
Delay after a bus voltage return [seconds]	This parameter determines how much time must pass between the return of the bus voltage and the sending of the first telegram.	**3** : 255

7.4.2 Sunblind push button input

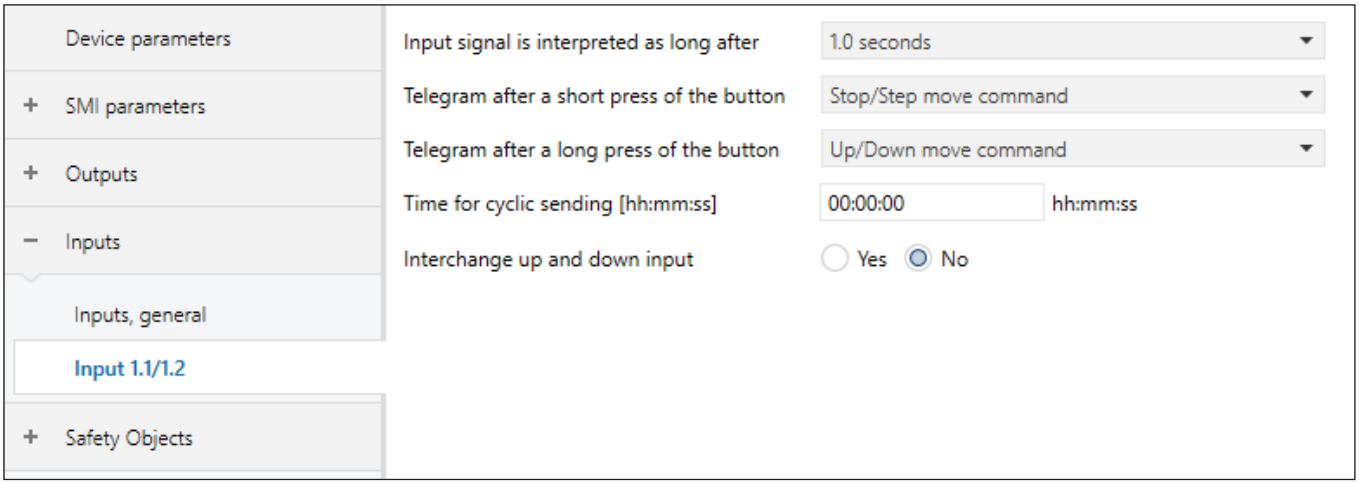


Fig. 34 Parameter dialogue: Inputs → Input n.1/n.2 (Sunblind push button)

Parameter	Function	Values
Input signal is interpreted as long after	If the push button is pressed for at least the set time, the telegram for a long press of the button is sent after the parameterised time. If the button is pressed for a shorter time, the telegram for a short press of the button is sent after the button is released.	0 seconds
		0.4 seconds
		0.5 seconds
		0.6 seconds
		0.8 seconds
		1.0 seconds
		1.2 seconds
		1.5 seconds
		2 seconds
		3 seconds
		4 seconds
		5 seconds
		10 seconds
Telegram after a short press of the button	If the up contact was closed, the ***GO Up/Down move command*** sends a 0-telegram. If the down contact was closed, the ***GO Up/Down move command*** sends a 1-telegram.	No move command
		Up/Down move command
		Stop/Step move command
Telegram after a long press of the button	If the up contact was closed, the ***GO Stop/Step move command*** sends a 0-telegram. If the down contact was closed, the ***GO Stop/Step move command*** sends a 1-telegram.	No move command
		Up/Down move command
		Stop/Step move command
Time for cyclical sending [hh:mm:ss]	Telegrams after a long press of the button can be sent repeatedly as long as the connected contact is closed. The interval between two consecutive repeats can be parameterised here. Setting the value to 0 causes a telegram to be sent only once after a long press of the button.	**00:00:00** : 23:59:59
Switch up and down inputs	The inputs for Up and Down are swapped for each sunblind push button. As a result, a faulty connection can be corrected.	Yes
		No

7.4.3 Button/Switch/Edges input

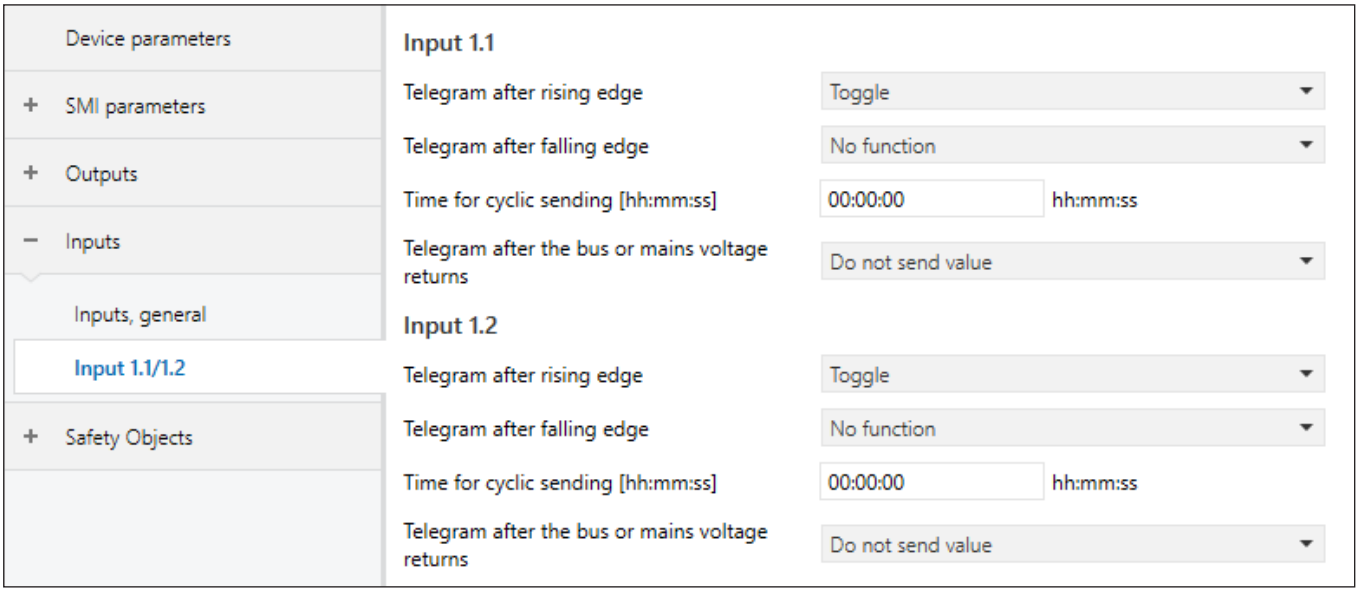


Fig. 35 Parameter dialogue: Inputs → Input n.1/n.2 (Button/Switch/Edges)

Parameter	Function	Values
Telegram after rising edge	No telegram sent after the button is pressed or released or after the bus/mains voltage returns.	No function
	GO Button/Switch/Edges sends 1-telegram(s) once or cyclically	On
	GO Button/Switch/Edges sends 0-telegram(s) once or cyclically	Off
	GO Button/Switch/Edges sends toggle telegram(s) once or cyclically	**Toggle**
Telegram after falling edge	No telegram sent after the button is pressed or released or after the bus/mains voltage returns.	**No function**
	GO Button/Switch/Edges sends 1-telegram(s) once or cyclically	On
	GO Button/Switch/Edges sends 0-telegram(s) once or cyclically	Off
	GO Button/Switch/Edges sends toggle telegram(s) once or cyclically	Toggle
Time for cyclical sending [hh:mm:ss]	Telegrams after a long press of the button can be sent repeatedly as long as the connected contact is closed. The interval between two consecutive repeats can be parameterised here. Setting the value to 0 causes a telegram to be sent only once after a long press of the button. After the bus or network voltage returns, if no new edge is detected, the value determined in the following parameter is sent cyclically.	**00:00:00** : 23:59:59
Telegram after the bus or mains voltage returns	No telegram is sent after the button is pressed or released	**Do not send value**
	An ON telegram is sent	Send ON telegram
	An OFF telegram is sent	Send OFF telegram
	When the *Delay after bus voltage return* expires, the following is sent once: - when the contact is closed, a value corresponding to the telegram after the rising edge - when the contact is open, a value corresponding to the telegram after the falling edge	Send current input state

7.4.4 Sensor input

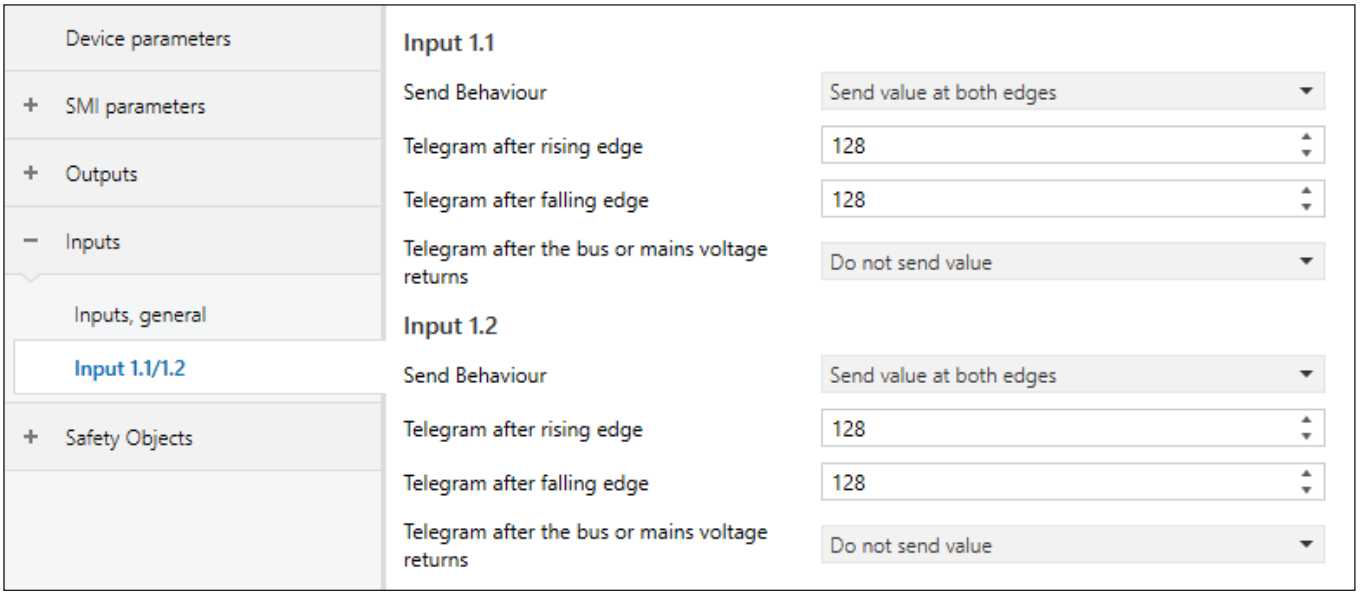


Fig. 36 Parameter dialogue: Inputs → Input n.1/n.2 (Sensor)

Parameter	Function	Values
Send behaviour	***GO Sensor*** sends after the rising edge the value parameterised in *Telegram after rising edge*	Send value at rising edges
	GO Sensor sends after the falling edge the value parameterised in *Telegram after falling edge*	Send value at falling edges
	GO Sensor sends after the rising edge the value parameterised in *Telegram after rising edge* and after the falling edge the value parameterised in *Telegram after falling edge*	**Send value at both edges**
Telegram after rising edge	Value that is sent after the rising edge.	0 : **128** : 255
Telegram after falling edge	Value that is sent after the falling edge.	0 : **128** : 255
Telegram after the bus or mains voltage returns	Specifies what value is sent after the bus or mains voltage returns.	**Do not send value**
		Send value for rising edges
		Send value for falling edges
		Send current input state

7.4.5 Scenes push button input

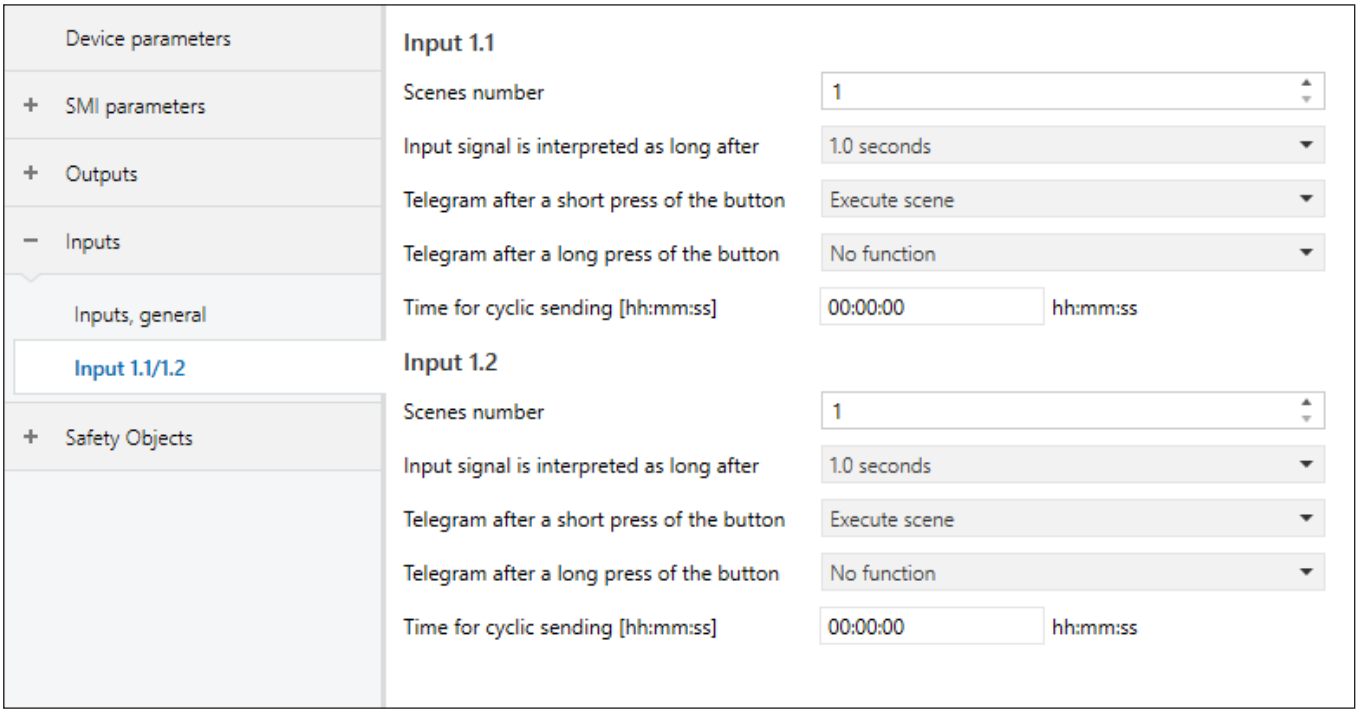


Fig. 37 Parameter dialogue: Inputs → Input n.1/n.2 (Scenes push button)

Parameter	Function	Values
Scenes number	***GO Scenes*** sends the value parameterised here	**1** : 64
Input signal is interpreted as long after	If the push button is pressed for at least the set time, the telegram for a long press of the button is sent after the parameterised time. If the button is pressed for a shorter time, the telegram for a short press of the button is sent after the button is released.	0 seconds
		0.4 seconds
		0.5 seconds
		0.6 seconds
		0.8 seconds
		1.0 seconds
		1.2 seconds
		1.5 seconds
		2 seconds
		3 seconds
		4 seconds
		5 seconds
		10 seconds
Telegram after a short press of the button		No function
	GO Scenes sends the value from *Scenes number* and Execute scene after a short press of the button	**Execute scene**
	GO Scenes sends the value from *Scenes number* and Learn scene after a short press of the button	Learn scene
Telegram after a long press of the button		**No function**
	GO Scenes sends after a long press of the button the value from *Scenes number* and Execute scene	Execute scene
	GO Scenes sends after a long press of the button the value from *Scenes number* and Learn scene	Learn scene
Time for cyclical sending [hh:mm:ss]	Telegrams after a long press of the button can be sent repeatedly as long as the connected contact is closed. The factor for cyclic sending determines the time interval between two consecutive repeats. Setting the value to 0 causes a telegram to be sent only once after a long press of the button.	**00:00:00** : 23:59:59

7.5 Safety objects

The safety functions of the actuators are used to protect controlled systems against damage, such as in the case of a wind alarm.

Four safety group objects are available with different priorities. These safety objects can start or end internal alarms according to the following criteria:

- Bus or mains voltage return
- Bus voltage failure
- Programming of the device
- Cyclical monitoring (Time intervals between received telegrams)
- Contents of the telegrams to safety objects

The safety objects ***A***, ***B*** and ***D*** are available for the actuator once. The safety object ***C*** is available for each output once (***C1***, ***C2***, ...).

For each output, you can set how the ***Safety objects*** (***SO***) are to affect it and which move command is to be performed after activation or deactivation of the respective ***SO***.

For the priorities of the safety objects, see also Chapter *7.5.1*.

Example

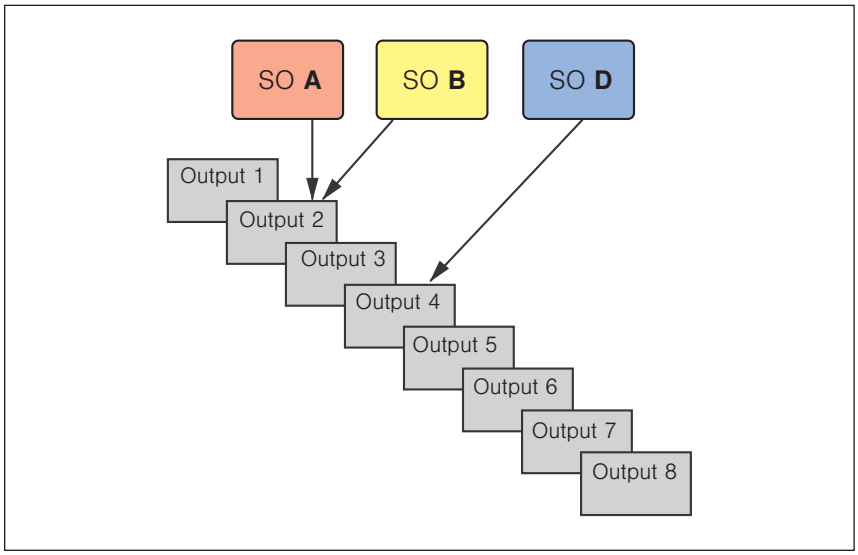


Fig. 38 Allocation example

For example, if safety object ***A*** is activated (1-telegram) while safety object ***B*** is already active, safety object ***B*** is overridden. Output 4 remains unaffected by the change in state of safety objects ***A*** or ***B***.

In the example, the following parameter settings are used:

- ▶ Behaviour after start of alarm from ***SO A***: Raise
- ▶ Behaviour after end of alarm from ***SO A***: Return to previous position
- ▶ Behaviour after start of alarm from ***SO B***: Lower
- ▶ Behaviour after end of alarm from ***SO B***: Return to previous position

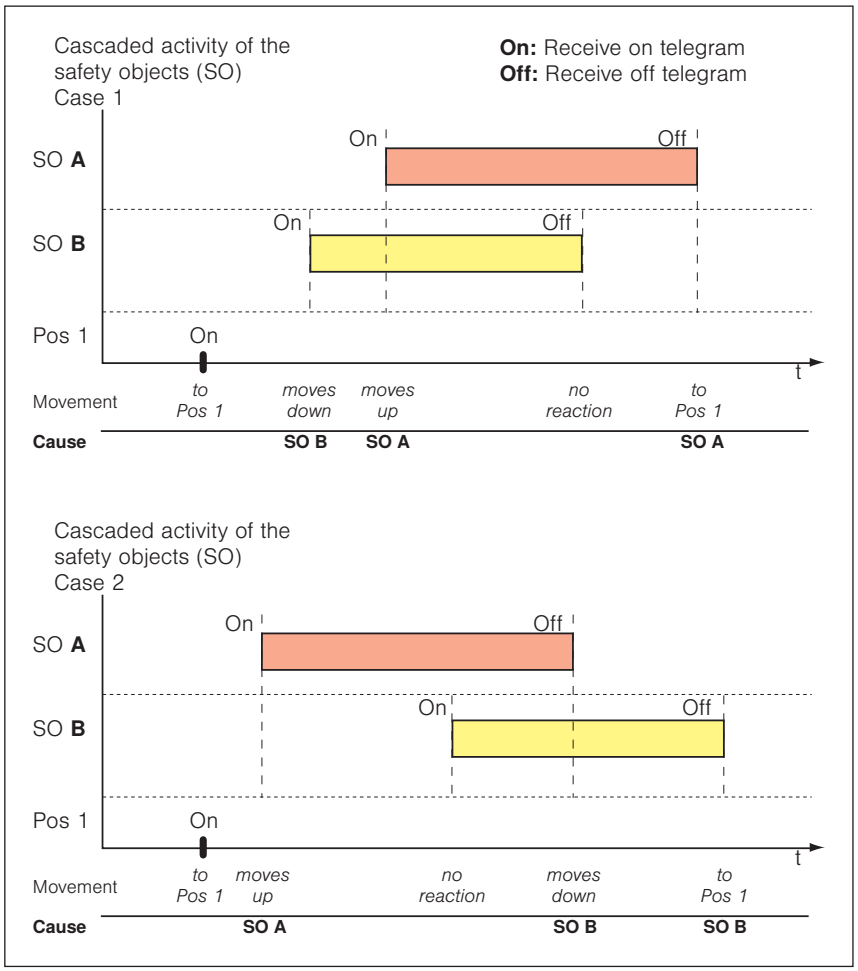


Fig. 39 Example: Behaviour of the safety objects

After a safety object ends, the move command with the lowest priority that is set for this event is executed. This means that it is only executed if no other safety objects are active when an alarm ends.

After a high priority alarm ends, the action that should have been executed when a safety object that is currently still active was first activated is now executed.

7.5.1 Order of priorities

In descending order:

- ***GO Disable object***
  (highest priority, stops all move commands after activation)
- ***GO Safety object A***
- ***GO Safety object B***
- ***GO Safety object C*** (available separately for each input)
- ***GO Safety object D***
- ***GO Limitation of manual operation in automatic mode*** *
- ***GO Move to blind length manually***
  GO Move to slat position manually
  GO Scenes
- Automatic delay after manual operation
- ***GO Move to blind length automatically***
  GO Move to slat position automatically
  GO Move to automatic position 1
  GO Move to automatic position 2

* It is possible that the area which all GOs for manual operation can move to is limited by the ***GO Limitation of manual operation in automatic mode*** and the parameterisation of the automatic input.

For each sun shading output, the behaviour of the product when an alarm starts or ends can be set. An emergency manual operation via smartphone or DCA app as a so-called SMI Broadcast has the highest priority.

7.5.2 Safety objects - General settings in the parameter dialogue

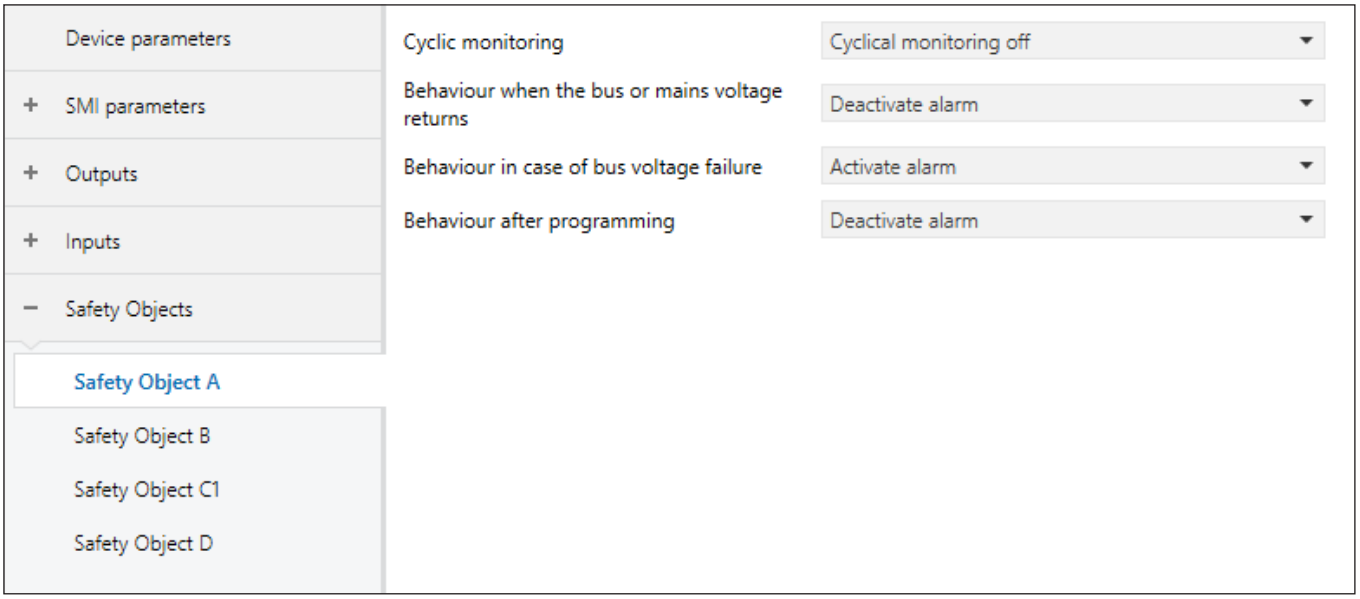


Fig. 40 Parameter dialogue: Safety objects

Parameter	Function	Values
Cyclical monitoring	Monitors whether telegrams are cyclically received on the ***GO Safety object n***. The safety object must receive at least one telegram within this time interval. If this time expires without a telegram having been received, the safety object is activated. After a 0-telegram to the safety object, it is disabled again.	**Cyclical monitoring off**
		10 seconds
		1 minute
		2 minutes
		5 minutes
		10 minutes
Behaviour when the bus or mains voltage returns	This parameter defines the behaviour of the safety objects after the mains voltage returns.	**Deactivate alarm**
		Activate alarm
		No change
Behaviour in case of bus voltage failure	This parameter defines the behaviour of the safety objects after the bus voltage fails. NOTICE: For REG devices, this parameter is disabled when the power saving mode is switched on.	Deactivate alarm
		Activate alarm
		No change
Behaviour after programming	This parameter defines the behaviour of the safety objects after programming.	**Deactivate alarm**
		Activate alarm
		No change

8 Group objects

8.1 Overview

The KNX SA SMI is equipped with a total of 423 group objects (GO). Depending on the actuator model and parameter setting (e.g. product type), the group objects available in each case are shown on the ETS interface.

The following table contains all group objects with the associated specifications.

[1] only in operating mode *Venetian blind/external venetian blind*

No.	Name	Object function	Length	Flags	Data type
1	Output 1	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
2	Output 1	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
3	Output 1	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
4	Output 1	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
5	Output 1	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
6	Output 1	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
7	Output 1	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
8	Output 1	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
9	Output 1	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
10	Output 1	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
11	Output 1	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
12	Output 1	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
13	Output 1	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
14	Output 1	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
15	Output 1	Safety object C1	1 bit	C, W	1 bit, 1.005 alarm
16	Output 1	Disable object	1 bit	C, W	1 bit, 1.005 alarm
17	Output 1	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
18	Output 1	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
19	Output 1	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
20	Output 1	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
21	Output 1	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
22	Output 1	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
23	Output 2	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
24	Output 2	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
25	Output 2	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
26	Output 2	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
27	Output 2	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
28	Output 2	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
29	Output 2	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
30	Output 2	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean

No.	Name	Object function	Length	Flags	Data type
31	Output 2	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
32	Output 2	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
33	Output 2	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
34	Output 2	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
35	Output 2	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
36	Output 2	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
37	Output 2	Safety object C2	1 bit	C, W	1 bit, 1.005 alarm
38	Output 2	Disable object	1 bit	C, W	1 bit, 1.005 alarm
39	Output 2	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
40	Output 2	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
41	Output 2	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
42	Output 2	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
43	Output 2	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
44	Output 2	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
45	Output 3	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
46	Output 3	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
47	Output 3	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
48	Output 3	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
49	Output 3	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
50	Output 3	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
51	Output 3	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
52	Output 3	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
53	Output 3	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
54	Output 3	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
55	Output 3	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
56	Output 3	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
57	Output 3	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
58	Output 3	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
59	Output 3	Safety object C3	1 bit	C, W	1 bit, 1.005 alarm
60	Output 3	Disable object	1 bit	C, W	1 bit, 1.005 alarm
61	Output 3	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
62	Output 3	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
63	Output 3	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
64	Output 3	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
65	Output 3	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
66	Output 3	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
67	Output 4	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
68	Output 4	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
69	Output 4	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)

No.	Name	Object function	Length	Flags	Data type
70	Output 4	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
71	Output 4	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
72	Output 4	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
73	Output 4	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
74	Output 4	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
75	Output 4	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
76	Output 4	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
77	Output 4	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
78	Output 4	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
79	Output 4	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
80	Output 4	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
81	Output 4	Safety object C4	1 bit	C, W	1 bit, 1.005 alarm
82	Output 4	Disable object	1 bit	C, W	1 bit, 1.005 alarm
83	Output 4	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
84	Output 4	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
85	Output 4	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
86	Output 4	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
87	Output 4	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
88	Output 4	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
89	Output 5	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
90	Output 5	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
91	Output 5	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
92	Output 5	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
93	Output 5	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
94	Output 5	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
95	Output 5	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
96	Output 5	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
97	Output 5	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
98	Output 5	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
99	Output 5	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
100	Output 5	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
101	Output 5	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
102	Output 5	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
103	Output 5	Safety object C5	1 bit	C, W	1 bit, 1.005 alarm
104	Output 5	Disable object	1 bit	C, W	1 bit, 1.005 alarm
105	Output 5	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
106	Output 5	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
107	Output 5	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
108	Output 5	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)

No.	Name	Object function	Length	Flags	Data type
109	Output 5	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
110	Output 5	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
111	Output 6	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
112	Output 6	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
113	Output 6	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
114	Output 6	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
115	Output 6	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
116	Output 6	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
117	Output 6	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
118	Output 6	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
119	Output 6	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
120	Output 6	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
121	Output 6	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
122	Output 6	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
123	Output 6	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
124	Output 6	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
125	Output 6	Safety object C6	1 bit	C, W	1 bit, 1.005 alarm
126	Output 6	Disable object	1 bit	C, W	1 bit, 1.005 alarm
127	Output 6	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
128	Output 6	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
129	Output 6	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
130	Output 6	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
131	Output 6	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
132	Output 6	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
133	Output 7	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
134	Output 7	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
135	Output 7	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
136	Output 7	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
137	Output 7	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
138	Output 7	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
139	Output 7	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
140	Output 7	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
141	Output 7	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
142	Output 7	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
143	Output 7	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
144	Output 7	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
145	Output 7	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
146	Output 7	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
147	Output 7	Safety object C7	1 bit	C, W	1 bit, 1.005 alarm

No.	Name	Object function	Length	Flags	Data type
148	Output 7	Disable object	1 bit	C, W	1 bit, 1.005 alarm
149	Output 7	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
150	Output 7	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
151	Output 7	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
152	Output 7	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
153	Output 7	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
154	Output 7	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
155	Output 8	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
156	Output 8	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
157	Output 8	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
158	Output 8	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
159	Output 8	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
160	Output 8	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
161	Output 8	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
162	Output 8	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
163	Output 8	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
164	Output 8	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
165	Output 8	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
166	Output 8	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
167	Output 8	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
168	Output 8	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
169	Output 8	Safety object C8	1 bit	C, W	1 bit, 1.005 alarm
170	Output 8	Disable object	1 bit	C, W	1 bit, 1.005 alarm
171	Output 8	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
172	Output 8	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
173	Output 8	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
174	Output 8	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
175	Output 8	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
176	Output 8	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
177	Output 9	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
178	Output 9	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
179	Output 9	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
180	Output 9	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
181	Output 9	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
182	Output 9	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
183	Output 9	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
184	Output 9	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean

No.	Name	Object function	Length	Flags	Data type
185	Output 9	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
186	Output 9	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
187	Output 9	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
188	Output 9	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
189	Output 9	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
190	Output 9	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
191	Output 9	Safety object C9	1 bit	C, W	1 bit, 1.005 alarm
192	Output 9	Disable object	1 bit	C, W	1 bit, 1.005 alarm
193	Output 9	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
194	Output 9	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
195	Output 9	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
196	Output 9	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
197	Output 9	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
198	Output 9	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
199	Output 10	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
200	Output 10	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
201	Output 10	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
202	Output 10	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
203	Output 10	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
204	Output 10	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
205	Output 10	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
206	Output 10	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
207	Output 10	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
208	Output 10	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
209	Output 10	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
210	Output 10	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
211	Output 10	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
212	Output 10	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
213	Output 10	Safety object C10	1 bit	C, W	1 bit, 1.005 alarm
214	Output 10	Disable object	1 bit	C, W	1 bit, 1.005 alarm
215	Output 10	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
216	Output 10	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
217	Output 10	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
218	Output 10	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
219	Output 10	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
220	Output 10	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
221	Output 11	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
222	Output 11	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
223	Output 11	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)

No.	Name	Object function	Length	Flags	Data type
224	Output 11	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
225	Output 11	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
226	Output 11	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
227	Output 11	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
228	Output 11	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
229	Output 11	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
230	Output 11	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
231	Output 11	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
232	Output 11	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
233	Output 11	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
234	Output 11	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
235	Output 11	Safety object C11	1 bit	C, W	1 bit, 1.005 alarm
236	Output 11	Disable object	1 bit	C, W	1 bit, 1.005 alarm
237	Output 11	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
238	Output 11	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
239	Output 11	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
240	Output 11	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
241	Output 11	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
242	Output 11	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
243	Output 12	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
244	Output 12	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
245	Output 12	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
246	Output 12	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
247	Output 12	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
248	Output 12	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
249	Output 12	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
250	Output 12	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
251	Output 12	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
252	Output 12	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
253	Output 12	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
254	Output 12	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
255	Output 12	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
256	Output 12	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
257	Output 12	Safety object C12	1 bit	C, W	1 bit, 1.005 alarm
258	Output 12	Disable object	1 bit	C, W	1 bit, 1.005 alarm
259	Output 12	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
260	Output 12	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
261	Output 12	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
262	Output 12	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)

No.	Name	Object function	Length	Flags	Data type
263	Output 12	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
264	Output 12	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
265	Output 13	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
266	Output 13	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
267	Output 13	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
268	Output 13	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
269	Output 13	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
270	Output 13	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
271	Output 13	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
272	Output 13	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
273	Output 13	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
274	Output 13	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
275	Output 13	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
276	Output 13	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
277	Output 13	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
278	Output 13	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
279	Output 13	Safety object C13	1 bit	C, W	1 bit, 1.005 alarm
280	Output 13	Disable object	1 bit	C, W	1 bit, 1.005 alarm
281	Output 13	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
282	Output 13	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
283	Output 13	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
284	Output 13	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
285	Output 13	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
286	Output 13	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
287	Output 14	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
288	Output 14	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
289	Output 14	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
290	Output 14	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
291	Output 14	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
292	Output 14	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
293	Output 14	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
294	Output 14	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
295	Output 14	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
296	Output 14	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
297	Output 14	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
298	Output 14	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
299	Output 14	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
300	Output 14	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
301	Output 14	Safety object C14	1 bit	C, W	1 bit, 1.005 alarm

No.	Name	Object function	Length	Flags	Data type
302	Output 14	Disable object	1 bit	C, W	1 bit, 1.005 alarm
303	Output 14	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
304	Output 14	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
305	Output 14	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
306	Output 14	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
307	Output 14	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
308	Output 14	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
309	Output 15	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
310	Output 15	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
311	Output 15	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
312	Output 15	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
313	Output 15	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
314	Output 15	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
315	Output 15	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
316	Output 15	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
317	Output 15	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
318	Output 15	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
319	Output 15	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
320	Output 15	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
321	Output 15	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
322	Output 15	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
323	Output 15	Safety object C15	1 bit	C, W	1 bit, 1.005 alarm
324	Output 15	Disable object	1 bit	C, W	1 bit, 1.005 alarm
325	Output 15	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
326	Output 15	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
327	Output 15	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
328	Output 15	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
329	Output 15	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
330	Output 15	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
331	Output 16	Up/Down move command	1 bit	C, W	1 bit, 1.008 Up/Down
332	Output 16	Stop/Step move command	1 bit	C, W	1 bit, 1.007 step
333	Output 16	Move to blind length manually	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
334	Output 16	Move to slat position manually [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
335	Output 16	Enable automatic objects	1 bit	C, W	1 bit, 1.003 Enable
336	Output 16	Move to blind length automatically	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
337	Output 16	Move to slat position automatically [1]	1 byte	C, W	8 bit unsigned, 5.001 percent (0..100%)
338	Output 16	Move to automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean

No.	Name	Object function	Length	Flags	Data type
339	Output 16	Move to automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
340	Output 16	Save automatic position 1	1 bit	C, W	1 bit, 1.002 Boolean
341	Output 16	Save automatic position 2	1 bit	C, W	1 bit, 1.002 Boolean
342	Output 16	Automatic position toggle	1 bit	C, W	1 bit, 1.002 Boolean
343	Output 16	Limitation of manual operation in automatic mode	1 bit	C, W	1 bit, 1.003 Enable
344	Output 16	Dwell time active	1 bit	C, R, T	1 bit, 1.002 Boolean
345	Output 16	Safety object C16	1 bit	C, W	1 bit, 1.005 alarm
346	Output 16	Disable object	1 bit	C, W	1 bit, 1.005 alarm
347	Output 16	Scenes	1 byte	C, W	Scenes check, 18.001 Scenes check
348	Output 16	Disable scenes	1 bit	C, W	1 bit, 1.003 Enable
349	Output 16	Upper limit position reached	1 bit	C, R, T	1 bit, 1.002 Boolean
350	Output 16	Actual blind length	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
351	Output 16	Slat position status [1]	1 byte	C, R, T	8 bit unsigned, 5.001 percent (0..100%)
352	Output 16	Fault message	1 bit	C, R, T	1 bit, 1.001 switch
353	Input 1	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 1.1 Sensor operating mode: Sensor 1.1 Scenes push button operating mode: Scene 1.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
354	Input 1	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 1.2 Sensor operating mode: Sensor 1.2 Scenes push button operating mode: Scene 1.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
355	Input 1	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
356	Input 2	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 2.1 Sensor operating mode: Sensor 2.1 Scenes push button operating mode: Scene 2.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
357	Input 2	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 2.2 Sensor operating mode: Sensor 2.2 Scenes push button operating mode: Scene 2.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
358	Input 2	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
359	Input 3	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 3.1 Sensor operating mode: Sensor 3.1 Scenes push button operating mode: Scene 3.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check

No.	Name	Object function	Length	Flags	Data type
360	Input 3	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 3.2 Sensor operating mode: Sensor 3.2 Scenes push button operating mode: Scene 3.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
361	Input 3	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
362	Input 4	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 4.1 Sensor operating mode: Sensor 4.1 Scenes push button operating mode: Scene 4.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
363	Input 4	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 4.2 Sensor operating mode: Sensor 4.2 Scenes push button operating mode: Scene 4.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
364	Input 4	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
365	Input 5	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 5.1 Sensor operating mode: Sensor 5.1 Scenes push button operating mode: Scene 5.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
366	Input 5	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 5.2 Sensor operating mode: Sensor 5.2 Scenes push button operating mode: Scene 5.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
367	Input 5	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
368	Input 6	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 6.1 Sensor operating mode: Sensor 6.1 Scenes push button operating mode: Scene 6.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
369	Input 6	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 6.2 Sensor operating mode: Sensor 6.2 Scenes push button operating mode: Scene 6.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
370	Input 6	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable

No.	Name	Object function	Length	Flags	Data type
371	Input 7	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 7.1 Sensor operating mode: Sensor 7.1 Scenes push button operating mode: Scene 7.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
372	Input 7	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 7.2 Sensor operating mode: Sensor 7.2 Scenes push button operating mode: Scene 7.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
373	Input 7	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
374	Input 8	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 8.1 Sensor operating mode: Sensor 8.1 Scenes push button operating mode: Scene 8.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
375	Input 8	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 8.2 Sensor operating mode: Sensor 8.2 Scenes push button operating mode: Scene 8.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
376	Input 8	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
377	Input 9	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 9.1 Sensor operating mode: Sensor 9.1 Scenes push button operating mode: Scene 9.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
378	Input 9	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 9.2 Sensor operating mode: Sensor 9.2 Scenes push button operating mode: Scene 9.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
379	Input 9	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
380	Input 10	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 10.1 Sensor operating mode: Sensor 10.1 Scenes push button operating mode: Scene 10.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
381	Input 10	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 10.2 Sensor operating mode: Sensor 10.2 Scenes push button operating mode: Scene 10.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check

No.	Name	Object function	Length	Flags	Data type
382	Input 10	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
383	Input 11	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 11.1 Sensor operating mode: Sensor 11.1 Scenes push button operating mode: Scene 11.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
384	Input 11	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 11.2 Sensor operating mode: Sensor 11.2 Scenes push button operating mode: Scene 11.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
385	Input 11	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
386	Input 12	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 12.1 Sensor operating mode: Sensor 12.1 Scenes push button operating mode: Scene 12.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
387	Input 12	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 12.2 Sensor operating mode: Sensor 12.2 Scenes push button operating mode: Scene 12.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
388	Input 12	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
389	Input 13	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 13.1 Sensor operating mode: Sensor 13.1 Scenes push button operating mode: Scene 13.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
390	Input 13	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 13.2 Sensor operating mode: Sensor 13.2 Scenes push button operating mode: Scene 13.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
391	Input 13	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
392	Input 14	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 14.1 Sensor operating mode: Sensor 14.1 Scenes push button operating mode: Scene 14.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check

No.	Name	Object function	Length	Flags	Data type
393	Input 14	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 14.2 Sensor operating mode: Sensor 14.2 Scenes push button operating mode: Scene 14.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
394	Input 14	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
395	Input 15	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 15.1 Sensor operating mode: Sensor 15.1 Scenes push button operating mode: Scene 15.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
396	Input 15	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 15.2 Sensor operating mode: Sensor 15.2 Scenes push button operating mode: Scene 15.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
397	Input 15	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
398	Input 16	Sunblind push button operating mode: Up/Down move command Button/Switch/Edges operating mode: On/Off/Toggle 16.1 Sensor operating mode: Sensor 16.1 Scenes push button operating mode: Scene 16.1	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.008 Up/Down 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
399	Input 16	Sunblind push button operating mode: Stop/Step move command Button/Switch/Edges operating mode: On/Off/Toggle 16.2 Sensor operating mode: Sensor 16.2 Scenes push button operating mode: Scene 16.2	1 bit 1 bit 1 byte 1 byte	C, T C, W, T C, T C, T	1 bit, 1.007 step 1 bit, 1.002 Boolean 8 bit unsigned, 5.010 counting impulses (0..255) Scenes check, 18.001 Scenes check
400	Input 16	Sunblind push button operating mode: Enable Button/Switch/Edges operating mode: – Sensor operating mode: – Scenes push button operating mode: –	1 bit	C, W	1 bit, 1.003 Enable
401	All outputs	Output collective error message	1 bit	C, R, T	1 bit, 1.001 switch
402	All outputs	Output collective error message text	14 byte	C, R, T	Character set, 16,000 characters (ASCII)
403	All outputs	Delete Output collective error messages	1 bit	C, W	1 bit, 1.001 switch
420	All outputs	Safety object A	1 bit	C, W	1 bit, 1.005 alarm
421	All outputs	Safety object B	1 bit	C, W	1 bit, 1.005 alarm
422	All outputs	Safety object D	1 bit	C, W	1 bit, 1.005 alarm
423	Device	Actuator available	1 bit	C, R, T	1 bit, 1.002 Boolean

[1] only in operating mode *Venetian blind/external venetian blind*

8.2 Group objects in detail

Below you will find a function description of the group objects used, as well as the possible values. In the column “Must be enabled” you will find the prerequisites for the respective group object to be activated and displayed in the ETS.

8.2.1 Group objects for the actuator outputs

[1] only in operating mode *Venetian blind/external venetian blind*

Name	Object function	Values	Must be enabled in the parameter dialogue
Up/Down move command	If a telegram with the value 0 is received on this GO, the sun shading product is raised. If a telegram with the value 1 is received, the sun shading product is lowered.	0 = UP 1 = DOWN	Outputs \ Outputs general \ Operating mode Output n = e.g. Output for venetian blind/external venetian blind
Stop/Step move command	If a telegram is received on this GO, a moving sun shading product is stopped. In the *Venetian blind/external venetian blind* operating mode, a step command is executed for a stationary sun shading product.	0 = STOP/Open slat tilt 1 = STOP/Close slat tilt	
Move to blind length manually	If a telegram is received on this GO, the sun shading product moves to the height that corresponds to the received value. Once the target position is reached, the slats assume the same position that they had before the movement.	0% (top) ...100% (bottom)	
Move to slat position manually [1]	If a telegram is received on this GO, the slats are positioned in accordance with the received value.	0% (slat OPEN) ...100% (slat CLOSED)	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind
Enable automatic objects	Disable the ***GO Move to automatic positions 1+2***, ***Move to blind length*** and ***Move to slat position***.	0 = Switch off	Outputs \ Outputs general \ Operating mode Output n = e.g. Output for venetian blind/external venetian blind AND Outputs \ Output n \ Automatic input \ Use automatic object = Yes
	Enable the ***GO Move to automatic positions 1+2***, ***Move to blind length*** and ***Move to slat position***. Any ongoing dwell time is ended.	1 = Enable	
Move to blind length automatically	If a telegram is received on this GO, the sun shading product moves to the height that corresponds to the received value. Once the target position is reached, the slats assume the same position that they had before the movement.	0% (top) ...100% (bottom)	
Move to slat position automatically [1]	If a telegram is received on this GO, the slats are positioned in accordance with the received value.	0% (slat OPEN) ...100% (slat CLOSED)	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind AND Outputs \ Output n \ Automatic input \ Use automatic object = Yes
Move to automatic position 1	If 1-telegrams are transmitted to the ***GO Move to automatic position 1***, the connected sun shading product is moved to the blind length and the slat position of automatic position 1.	0 = Blind moves to position 0% 1 = Move to position	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind AND Outputs \ Output n \ Automatic input \ Use automatic object = Yes AND Outputs \ Output n \ Automatic input \ Use automatic positions 1 and 2 = Yes
Move to automatic position 2	If 1-telegrams are transmitted to the ***GO Move to automatic position 2***, the connected sun shading product is moved to the blind length and the slat position of automatic position 2.	0 = Blind moves to position 0% 1 = Move to position	

Name	Object function	Values	Must be enabled in the parameter dialogue
Save automatic position 1	After a 1-telegram is transmitted to the ***GO Save automatic position 1***, the current blind length and slat position are stored in the Position 1 memory of the corresponding output.	1 = Save position	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind AND Outputs \ Output n \ Automatic input \ Use automatic object = Yes AND Outputs \ Output n \ Automatic input \ Use automatic positions 1 and 2 = Yes AND Outputs \ Output n \ Automatic positions \ Save positions 1 and 2 via telegram = Yes
Save automatic position 2	After a 1-telegram is transmitted to the ***GO Save automatic position 2***, the current blind length and slat position are stored in the Position 2 memory of the corresponding output.	1 = Save position	
Automatic position toggle	After a 0-telegram to the ***GO Automatic position toggle***, the product moves to the saved automatic position 1. After a 1-telegram to the ***GO Automatic position toggle***, the product moves to the position that would result from the automatic *Move to blind length* and automatic *Move to slat position* received last. If a *Position toggle delay time* is parameterised, the actions named above are delayed by this delay time. If the same telegram arrives while the delay time is running, it is ignored. The delay time is cancelled in the event of: - opposite telegram to this ***GO*** - a telegram to ***GO Move to automatic position 1 or 2*** - manual commands via GOs, buttons or the smartphone app	0 = Blind moves to saved position 1 1 = move to the position that results from the last received automatic *Move to blind length* and automatic *Move to slat position*	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind AND Outputs \ Output n \ Automatic input \ Use automatic object = Yes AND Outputs \ Output n \ Automatic input \ Use automatic positions 1 and 2 = Yes
Limitation of manual operation in automatic mode	The range of movement of the sun shading product can be limited, or manual operation can be completely disabled. When the limitation is enabled, any ongoing dwell time is ended.	0 = disabled 1 = enabled	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind AND Outputs \ Output n \ Automatic input \ Use automatic object = Yes
Dwell time active	The GO shows when the dwell time for manual operation of the output is still active. The send value can be parameterised.	Is specified by Outputs \ Output n \ Automatic input \ Object “Dwell time active”	Outputs \ Outputs general \ Operating mode Output n = e.g. Output for venetian blind/external venetian blind
Safety object Cn	Activated safety position Cn	0 = No alarm 1 = Alarm	
Disable object	Stops and disables all movements of the output	0 = Enabled 1 = Disable	
Scenes	Execute or save scenes	0 = Activate scene 1 = Learn scene 1...64 = Scene number	
Disable scenes	Disables all scene call-ups of the output. Disabled scene commands are not executed.	0 = Enabled 1 = Disable	
Upper limit position reached	Reports when sun shading product is in the upper limit position.	Is specified by Outputs \ Outputs general \ Object “upper limit position reached”	Outputs \ Outputs general \ Operating mode Output n = e.g. Output for venetian blind/external venetian blind

Name	Object function	Values	Must be enabled in the parameter dialogue
Actual blind length	Sends the current height of the sun shading product. Send behaviour is parameterised by: Outputs general \ Update of the status objects	0% (top) ...100% (bottom)	Outputs \ Outputs general \ Operating mode Output n = e.g. Output for venetian blind/external venetian blind
Slat position status [1]	Sends the current slat position of the sun shading product. Send behaviour is parameterised by: Outputs general \ Update of the status objects	0% (slat OPEN) ...100% (slat CLOSED)	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind
Fault message n [2]	Sends information about an existing output fault Send behaviour is parameterised by: SMI parameters \ SMI general \ Periodic transmission of error objects	0 = No fault 1 = Fault logged	Outputs \ Outputs general \ Operating mode Output n = Output for venetian blind/external venetian blind
Output collective error message	Sends information about an existing fault of the outputs Send behaviour is parameterised by: SMI parameters \ SMI general \ Cyclical transmission of error objects	0 = No fault 1 = Fault logged for at least one output [3]	Always enabled
Output collective error message text	Sends information about an existing output fault Send behaviour is parameterised by: SMI parameters \ SMI general \ Cyclical transmission of error objects	0 = No fault 1 = Fault logged for at least one output [3]	Always enabled
Delete Output collective error messages	Sends information about an existing output fault Send behaviour is parameterised by: SMI parameters \ SMI general \ Cyclical transmission of error objects	0 = No fault 1 = Fault logged for at least one output [3]	Always enabled
Actuator available	Sends information when the actuator is available Send behaviour is parameterised by: Device parameters \ Object "Actuator available" \ Time for cyclic sending	0...1	Device parameters \ Object "Actuator available"

[1] only in operating mode *Venetian blind/external venetian blind*
[2] available once per output
[3] Faults can be displayed via the DCA app (*see Section 5.1.4.7 on page 32*)

8.2.2 Group objects for the actuator inputs

8.2.2.1 Group objects for sunblind push button input

Name	Object function	Values	Must be enabled in the parameter dialogue
Up/Down move command	Sends Up/Down telegram	0 = Up 1 = Down	Inputs \ Inputs general \ Operating mode Input n.1/n.2 = Sunblind push button
Stop/step move command	Sends Reduce / Increase telegram	0 = Reduce 1 = Increase	Inputs \ Inputs general \ Operating mode Input n.1/n.2 = Sunblind push button
Enable	Enables sunblind push button. An Up/Down or Stop/Step telegram is always sent after resetting If False has been received via the bus on ***GO Enable***, telegrams are no longer sent on the ***GOs Up/Down move command*** and ***Stop/Step move command*** until a TRUE value is received again on ***GO Enable***	0 = Disable 1 = Enable	Inputs \ Inputs general \ Operating mode Input n.1/n.2 = Sunblind push button

8.2.2.2 Group objects for Button/Switch/Edges input

Name	Object function	Values	Must be enabled in the parameter dialogue
On/Off/Toggle n.1/n.2	***GO Button/Switch/Edges*** sends 1-telegram(s) once or cyclically.	On	Inputs \ Inputs general \ Operating mode input n.1/n.2 = Button / Switch / Edge
	GO Button/Switch/Edges sends 0-telegram(s) once or cyclically.	Off	
	GO Button/Switch/Edges sends toggle telegram(s) once or cyclically.	Toggle	

8.2.2.3 Group objects for sensor input

Name	Object function	Values	Must be enabled in the parameter dialogue
Sensor n.1/n.2	Sends parameterised values	0...**128**...255	Inputs \ Inputs general \ Operating mode Input n.1/n.2 = Sensor

8.2.2.4 Group objects for Scenes push button input

Name	Object function	Values	Must be enabled in the parameter dialogue
Scene n.1/n.2	Sends parameterised values	0 = Activate scene 1 = Learn scene 1...64 = Scene number	Inputs \ Inputs general \ Operating mode Input n.1/n.2 = Scenes push button

8.2.3 Group objects for the safety objects

Name	Object function	Values	Must be enabled in the parameter dialogue
Safety object A / B / D	Receives external alarm	0 = No alarm 1 = Alarm	Always enabled

For safety objects Cn see *chapter 8.2.1 Group objects for the actuator outputs on page 89*

9 Planning examples

9.1 Operating venetian blinds with push buttons

One venetian blind is connected to a KNX SA 16M230.32 SMI AP and one to a KNX SA 16M230 SMI REG. A sunblind push button is connected to the KNX SA 16M230.32 SMI AP. This button is used to move the three venetian blinds to any blind length and any slat position.

A long press of the button (>1 s) should start the movement of the product to the limit position, while a short press of the button should cause the slats to tilt.

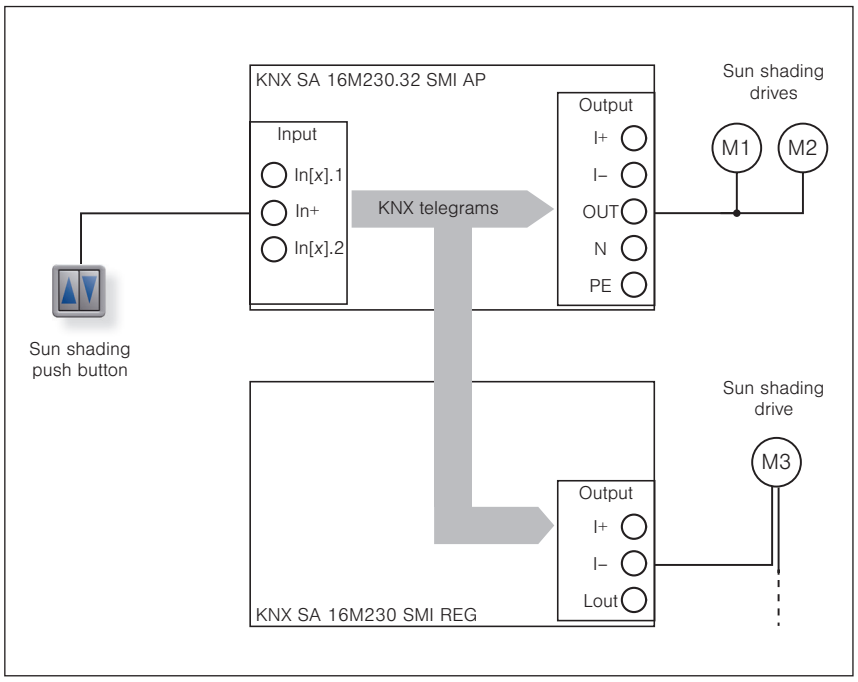


Fig. 41 “Operating venetian blinds” planning example

9.1.1 Settings

- Use the factory settings for both KNX SA SMIs.
- Parameterise `tilt pulses`

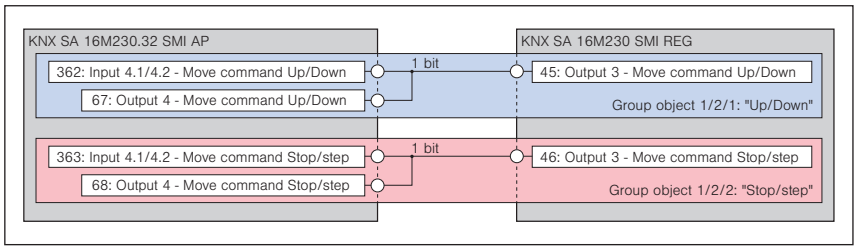


Fig. 42 Links in the “Operating venetian blinds” planning example

9.2 Connection to an automation

Connection of a KNX SA SMI sun shading actuator to an automation (here BAline KNXMCM) including a visualisation and a tactile sensor. Overview of the connections via group objects.

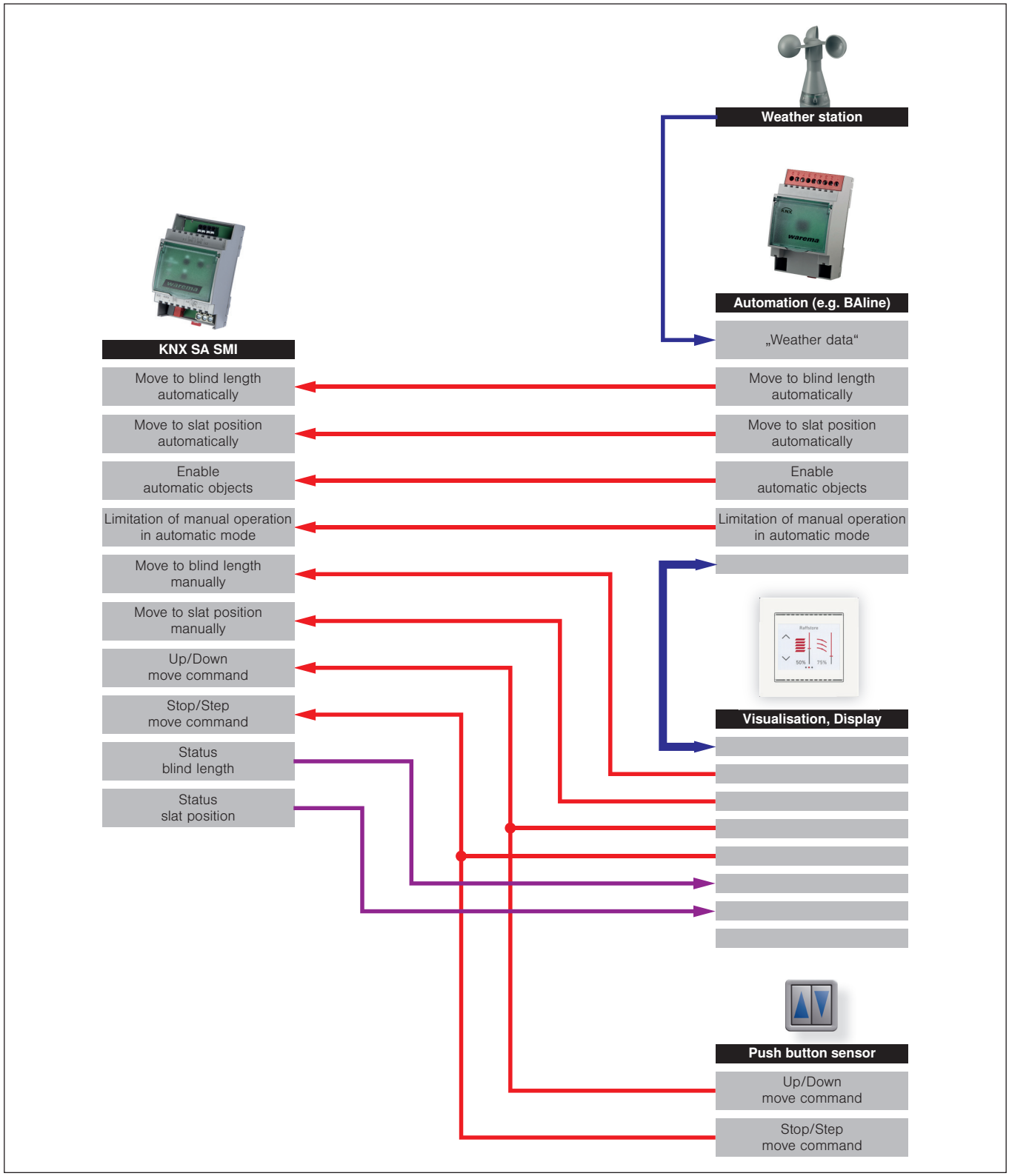


Fig. 43 “Connection to an automation” planning example

10 Control functions

The following diagrams show the behaviour of the actuators depending on the different conditions of the automatic group objects.

Example After a manual move command **M**, a set dwell time begins. If the dwell time has elapsed, the last automatic move command **A** is executed.

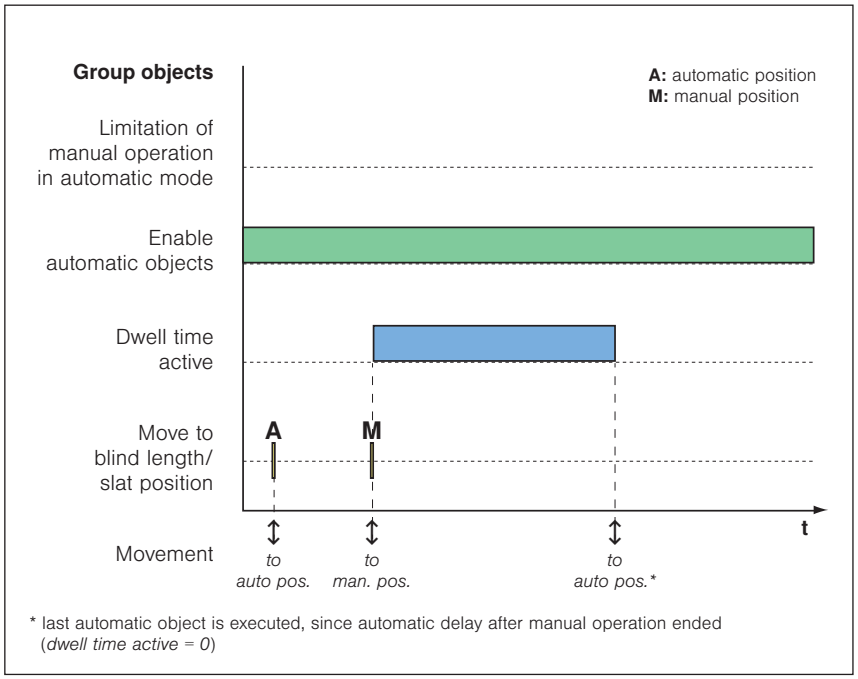


Fig. 44 Automatic objects enabled

Example If the ***GO Enable automatic objects*** is set to 0, all automatic commands ($\mathbf{A_2}$) from this point on are ignored. The last automatic move command $\mathbf{A_1}$ is also not executed once the dwell time has elapsed.

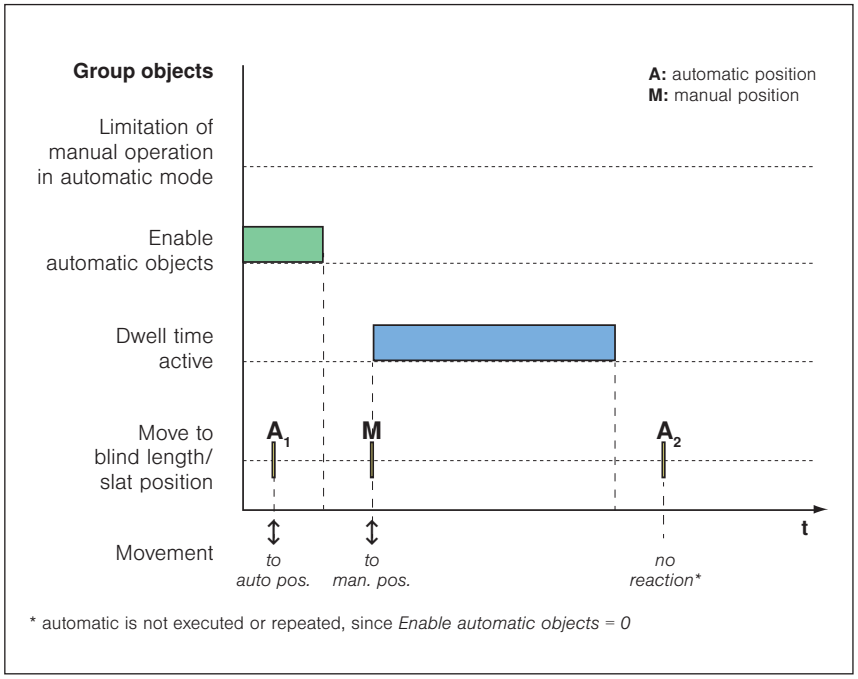


Fig. 45 Automatic objects disabled

Example If the GO ***Enable automatic objects*** is set to 0 and then set back to 1, any still ongoing dwell time is ended. The last automatic move command **A** is executed.

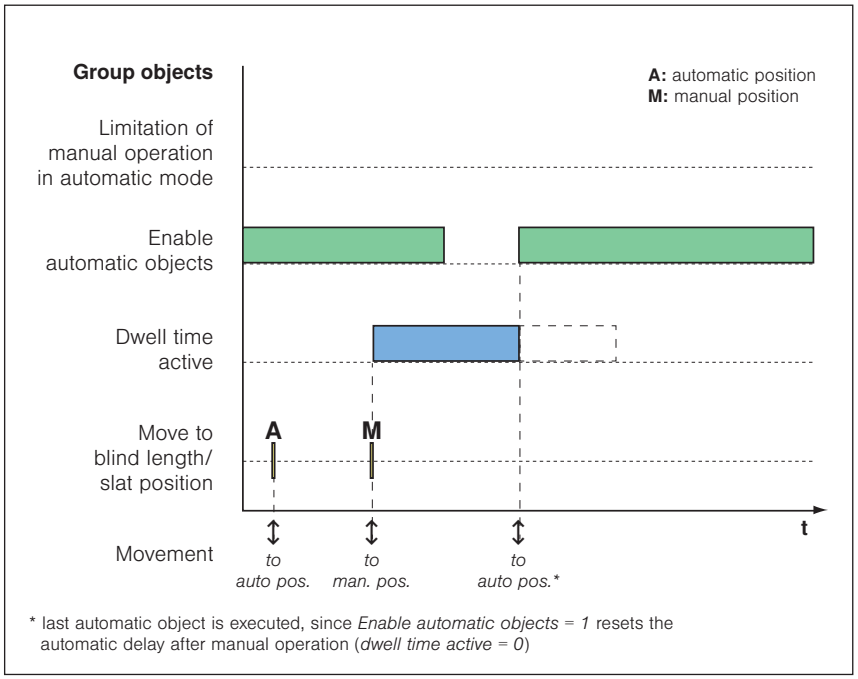


Fig. 46 Enable automatic objects when dwell time is ongoing

Example If a repeated 1 is received on the (still active) GO ***Enable automatic objects***, any still ongoing dwell time is ended. The last automatic move command ($\mathbf{A_2}$ in the example) is executed.

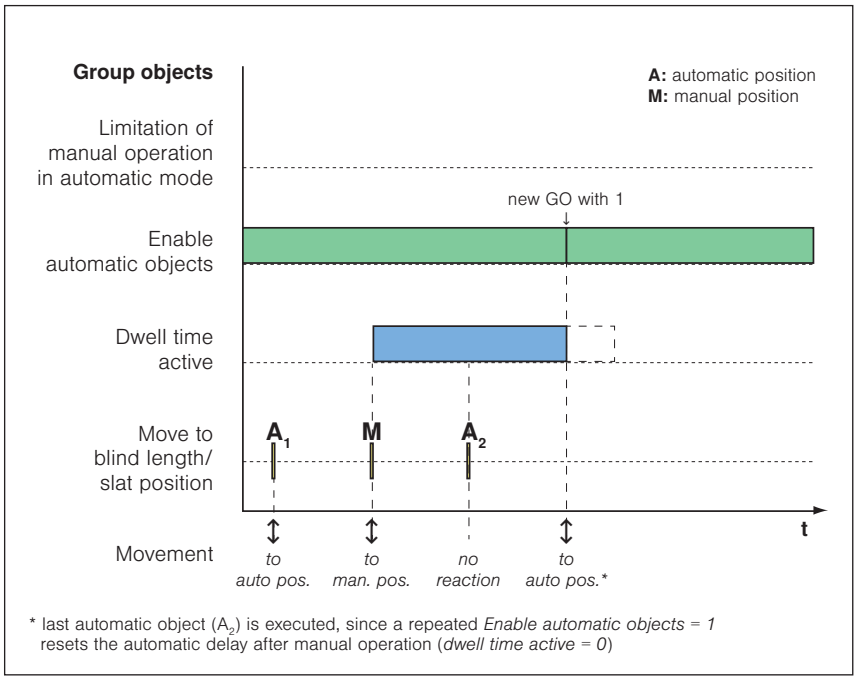


Fig. 47 Enable automatic objects with repeated GO

Example If the ***GO Limitation of manual operation in automatic mode*** is active, manual movements are only possible in the parameterised area ($\mathbf{M_1}$ in the example).
A 0 on the ***GO Enable automatic objects*** will disable the automatic object ***Limitation of manual operation in automatic mode***. From this point on, manual move commands are once again executed without any limitation ($\mathbf{M_2}$ in the example).

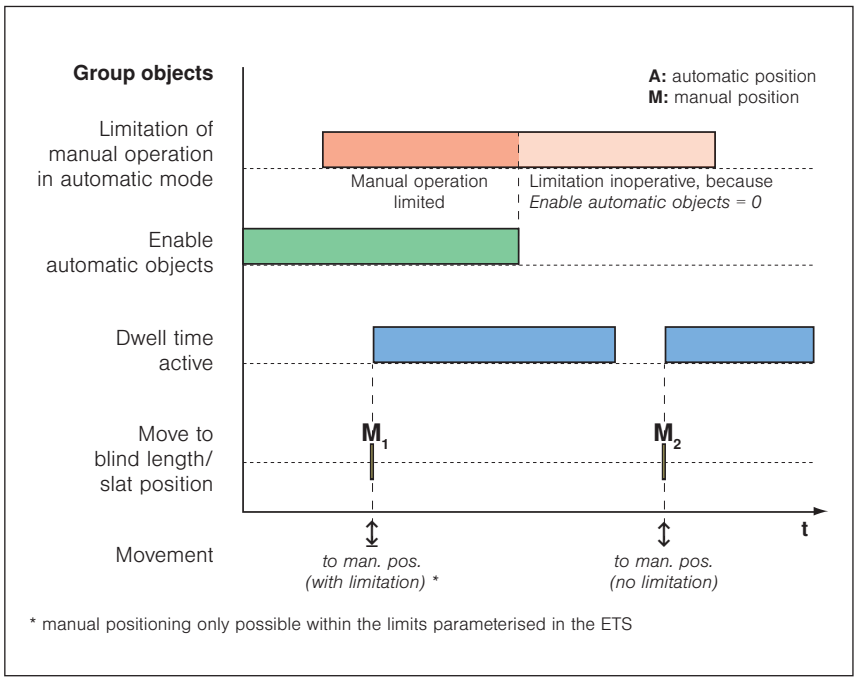


Fig. 48 Limitation of manual operation

11 Index

WAREMA Renkhoff SE • Hans-Wilhelm-Renkhoff-Straße 2 • 97828 Marktheidenfeld • Germany
www.warema.de • info@warema.de