CHORUS

GEWISS

KNX 6-channel push-button panel

with shutter actuator

GW1x785A

Technical Manual

Contents

1 Introduction

This manual describes the functions for the device "**KNX 6-channel push-button panel with shutter actuator** " (GW10785A, GW12785A, GW14785A) and how they are set and configured using the ETS configuration software.

2 Application

This push-button panel is a command device with 6 channels, which can be used individually or combined and with an actuator with motor command functions for shutters/louvres.
Each channel has two LEDs for light signalling, one amber and one green.
The buttons can perform the following functions:

- load ON / OFF commands
- timed commands
- dimmer management (single or double button)
- curtain / shutter management (single or double button)
- scene management

The motor command actuator can perform the following functions:

- shutter movement
- shutter stop and venetian blind movement
- movement through a priority command (forcing)
- scenes
- movement generated by weather alarms

A function can be associated to each channel by means of a specific parameter, as described below.

2.1 Association limits

The maximum number of communication objects available is 164.
The maximum number of associations that the device can store is 204.
The maximum number of group addresses is 204.

3 *“6-channel push-button panel”* menu

The **6-channel push-button panel** menu contains the application parameters for all the input channels implemented by the device.
The main operator parameters for the device (fig. 3.1).

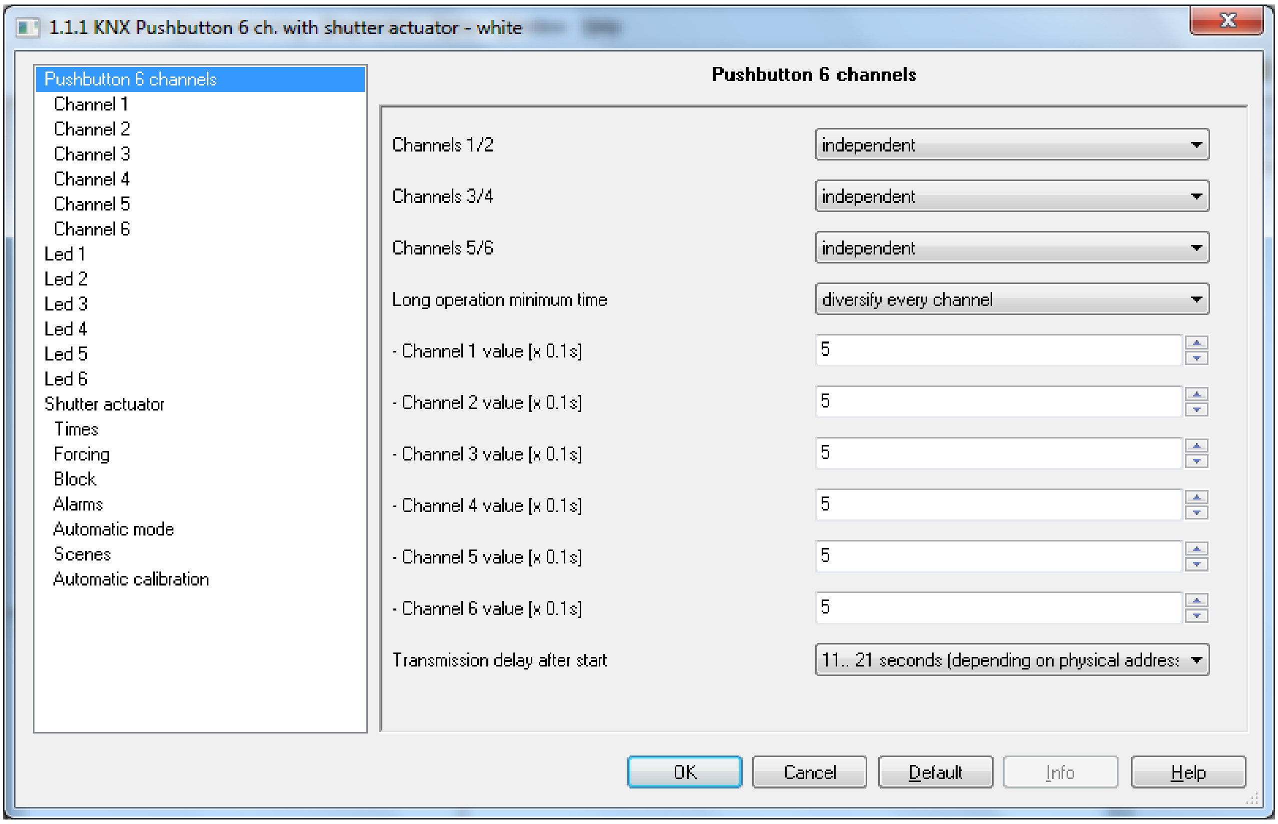


Fig. 3.1

3.1 *Parameters*

➢ *3.1.1. Channels X/Y*

Using these parameters, it is possible to select for each of the 6 input channels implemented by the push-button panel if they should carry out an independent function or if it should be combined two at a time to contribute towards carrying out a shared function. The values that can be set are:

- **independent** **(default value)**
- coupled

The database structure will vary depending on the values set for the above parameters: an independent setting menu for each channel if the set value is independent and a common menu if a combined value is set.

➢ *3.1.2 Long operation minimum time*

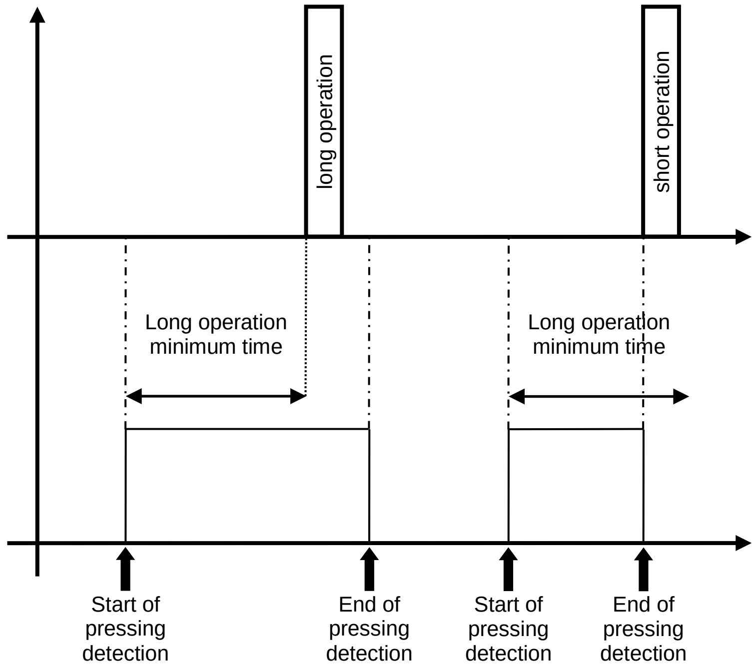
Many functions that the independent or combined channels can carry out foresee the differentiation between a short operation and a long operation. This parameter can be used to define a single time value for all channels or a different one for each of the channels; The values that can be set are:

- **the same for all channels (default value)**
- diversify every channel

The following parameters are displayed depending on the selected value: **“Value [x 0.1s]”** (if an identical value is set for all channels) or **“Channel 1 value [x 0.1s]”, “Channel 2 value [x 0.1s]”, “Channel 3 value [x 0.1s]”, “Channel 4 value [x 0.1s]”, “Channel 5 value [x 0.1s]” and “Channel 6 value [x 0.1s]”** (if a different value is set for each channel), which determine the minimum effective time during which the device must detect the pressing of the button to differentiate between the short operation and the long operation associated with the channel.
The possible values are:

- from 3 to 150 with step 1, **5 (default value)**

The following example shows the meaning of the above parameters



➢ *3.1.3 Transmission delay after start*

This parameter is used to define the time that must pass after which the device may transmit the telegrams on the bus following a drop/recovery of the bus supply voltage, to ensure that, with multiple devices in the line, the telegrams sent by the various devices do not collide when the bus voltage is restored.
The values that can be set are:

- **11.. 21 seconds (depending on physical address) (default value)**
- 5.. 9 seconds
- 11 seconds
- 13 seconds
- 15 seconds
- 17 seconds
- 19 seconds
- 21 seconds
- *no delay*

Setting the values **11.. 21 seconds (depending on physical address)** and **5.. 9 seconds**, the device automatically calculates the transmission delay according to an algorithm that examines the physical address of the device itself; the presented values (11/21 or 5/9) indicate the extremes of the value interval that can be calculated.

4 “Channel x” menu (independent channels)

If channel operation is independent, a dedicated menu is shown for each channel called **Channel x** (x is the channel index). The menu structure changes based on the value set for the **“Matched function”** parameter. For the sake of simplicity, the parameters enabled according to the value set for the above parameter are listed in the following paragraphs.
Figure 4.1 shows the basic structure of the menu:

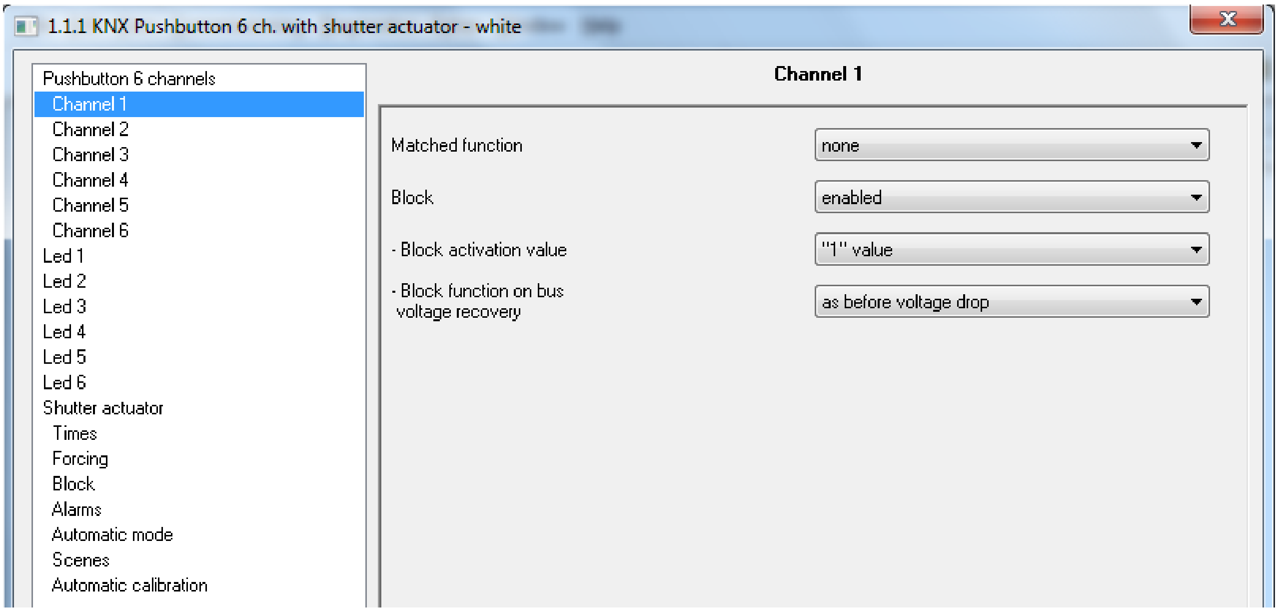


Fig 4.1

4.1 Parameters

➢ 4.1.1 Matched function

This is used to define the function implemented by the channel; The values that can be set are:

- ***none (default value)***
- ***edges/sequence commands***
  See chapter 5 Function “***edges/sequence commands***”
- ***1 push button + stop dimmer***
  See chapter 6 Function “***1 push button + stop dimmer***”
- ***cyclic sending 1 push button dimmer***
  See chapter 7 Function “***cyclic sending 1 push button dimmer***”
- ***1 push button shutter control***
  See chapter 8 Function **“*1 push button shutter control***”
- ***scene management***
  See chapter 9 Function “***scene management***”
- ***switching sequences***
  See chapter 10 Function “***switching sequences***”
- ***actuator local command***
  See chapter 11 Function “***actuator local command***”

4.1.2 Block

To inhibit the channel when sending commands associated with pressing/release the button, the block function must be activated: this function inhibits the detection of button pressing/releasing, preventing the device from sending the telegrams associated with these events on the bus; if activated, any change in status that occurs will not be interpreted until a block deactivation command is received.
This parameter is used to enable/disable the function and can have the following values:

- **disabled (default value)**
- enabled

Setting the value **enabled** displays the parameters **"Block activation value"** and **"Block function on bus voltage recovery"** and the communication object **Ch.x - Block** through which it is possible to activate the function via a bus command.

The parameter **"Block activation value"** makes it possible to set which logic value the bit received via bus telegram should assume to activate the block function; The values that can be set are:

- value "0"
- **value "1" (default value)**

The parameter **"Block function on bus voltage recovery"** is used to set the status of the block function on recovery; The values that can be set are:

- disabled
- enabled
- **as before voltage drop (default value)**

5 Function *"edges/sequence commands"*

This function is used to set the type and number of commands to send after a status change has been detected, for up to a total of eight commands per channel; it is possible to differentiate the type of command depending on the event that is detected (pressing and releasing) and delay the sending of the commands with a fixed settable time.
The basic structure of the menu is as follows (Fig. 5.1):

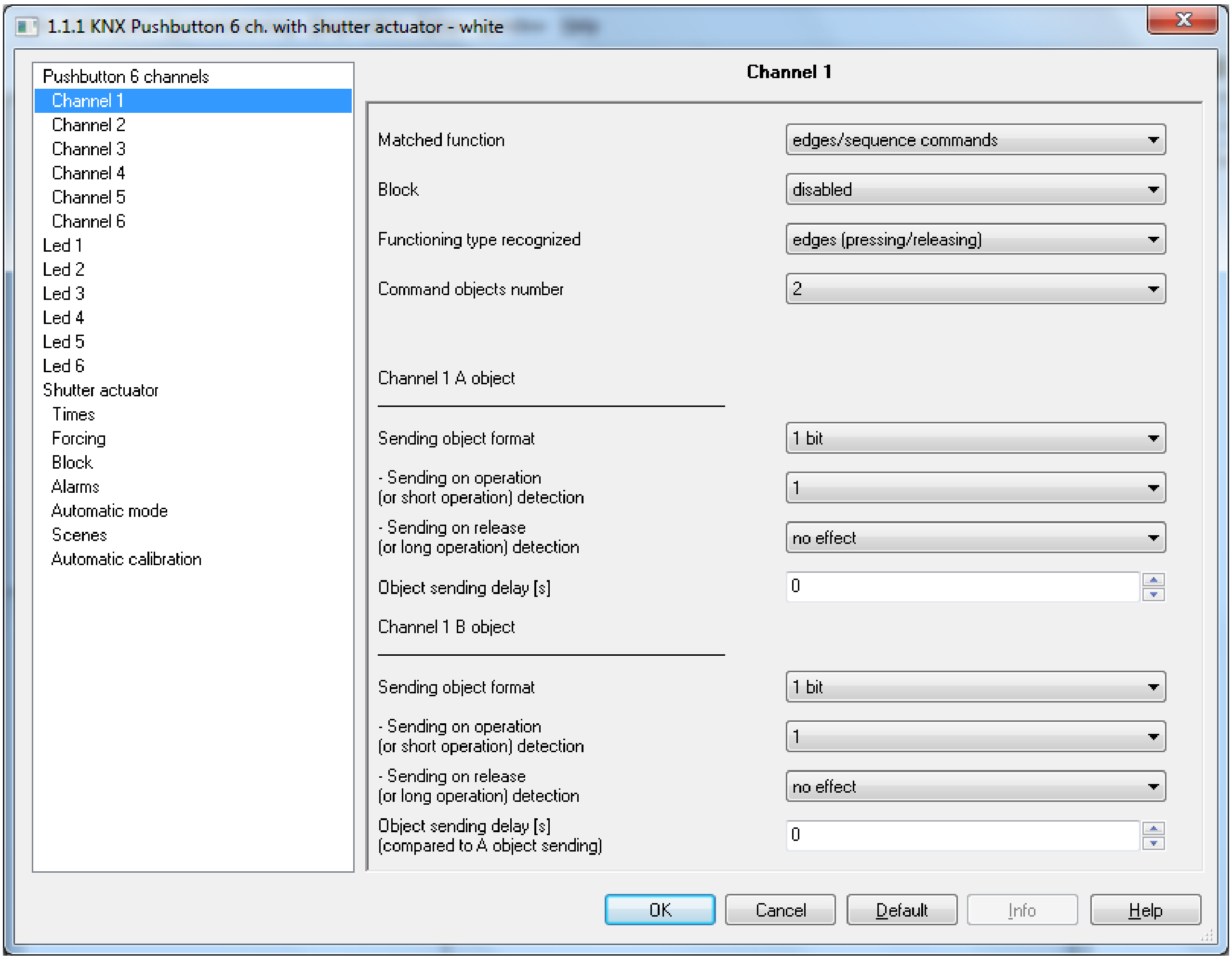


Fig. 5.1

5.1 *Parameters*

➢ *5.1.1 Functioning type recognized*

This is used to define which type of button operation generates the sending of the sequence commands; The values that can be set are:

- **edges (pressing/releasing)** **(default value)**
- short operation/long operation

➢ *5.1.2 Command objects number*

This is used to define how many communication objects are managed by the channel in question; The values that can be set are:

- **1 (default value)**, 2, …, 8

Depending on the set value, the following parameters appear for each of the selected object **"Sending object format"**, **"Sending on operation (or short operation) detection"**, **"Sending on release (or long operation) detection"** and **"Object sending delay [s]"** grouped in the subset **Channel x z object** (z is the index of the object associated with the channel, between **A** and **H)**.

The parameter **"Sending object format"** makes it possible to set the format and code of the bus telegram that will be sent by the device. The values that can be set are:

- **1 bit** **(default value)**
- 2 bits
- 1 byte unsigned value
- 1 byte signed value
- 1 byte percentage value
- 1 byte HVAC mode
- 2 bytes unsigned value
- 2 bytes signed value
- 4 bytes unsigned value
- 4 bytes signed value
- 14 bytes

The value set for this item will change as a result the values set for the parameters **"Sending on operation (or short operation) detection"** and **"Sending on release (or long operation) detection"**.
The parameter **"Sending on operation (or short operation) detection"** is used to set the command or the value to send when pressing of the button associated with the channel is detected.
The parameter **"Sending on release (or long operation) detection"** is used to set the command or the value to send when the release of the button associated with the channel is detected.

– If the format of the object to send is **1 bit**, the communication object ***Ch.x - z object 1 bit value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect** **(release detection default value)**
  - 0
  - **1** **(pressing detection default value)**
  - cyclic switching

  Selecting the value **cyclic switching**, the parameter **"Status feedback object"** will be shown, which makes it possible to enable and display the communication object ***Ch.x - z object feedback status***; by enabling this object, when the status feedback telegram is received for the object in question, the command that the push-button panel will send (via the object ***Ch.x - z object 1 bit value***) when the event associated with the cyclic sending is detected will be the opposite of the value generated by the most recent event between the bus value received on object ***Ch.x - z object status feedback*** and the last sent value (via the object ***Ch.x - z object 1 bit value***).

  The **"Status feedback object"** may have the following values:

  - **disabled** **(default value)**
  - enabled

  Selecting the value **enabled** displays the communication object ***Ch.x - z object status feedback.*** Each time bus voltage is reset, the device sends a status reading command for this object to update the push-button panel about the status of the connected devices.

– If the format of the object to send is **2 bits**, the communication object ***Ch.x - z object 2 bits value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect** **(release default value)**
  - **activate on (down) forcing** **(default pressing value)**
  - activate off (up) forcing

- disable forced positioning
- cyclical switching forcing on/forcing off
- cyclical switching forcing on/deactivate forcing
- cyclical switching forcing off/deactivate forcing

By selecting **cyclical switching**, in this case no communication object will be displayed as the device is always updated about the function activation status.

– If the format of the object to send is **1 byte unsigned**, the communication object ***Ch.x - z object 1 byte value*** will be visible and the values that can be set for the two above parameters are:

- **no effect (release default value)**
- **value sending (pressing default value)**

By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (0 .. 255)"** which can assume the following values:

- from **0 (default value)** to 255

– If the format of the object to send is **1 byte signed**, the communication object ***Ch.x - z object 1 byte value*** will be visible and the values that can be set for the two above parameters are:

- **no effect (release default value)**
- **value sending (pressing default value)**

By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (-128 .. 127)"** which can assume the following values:

- from -128 to 127, **0 (default value)**

– If the format of the object to send is **1 byte percentage**, the communication object ***Ch.x - z object 1 byte value*** will be visible and the values that can be set for the two above parameters are:

- **no effect (release default value)**
- **value sending (pressing default value)**

By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (0% .. 100%)"** which can assume the following values:

- from **0 (default value)** to 100

– If the format of the object to send is **1 byte HVAC mode**, the communication object ***Ch.x - z object 1 byte value*** will be visible and the values that can be set for the two above parameters are:

- **no effect (release default value)**
- auto
- **comfort (pressing default value)**
- precomfort
- economy
- off (building protection)
- cyclical switching (thermostat)
- cyclical switching (timed thermostat)

By selecting **cyclical switching**, in this case no communication object will be displayed as the device is always updated about the function activation status.

By selecting the value **cyclical switching (thermostat)**, each time the associated event is detected (pressing/releasing) the device sends a new thermoregulation mode (HVAC), following the order *Comfort→ Precomfort→ Economy→ Off→ Comfort* …; By selecting the value **cyclical switching**

(chronothermostat), each time the associated event is detected (pressing/releasing) the device sends a new thermoregulation mode (HVAC), following the order *Comfort→ Precomfort→ Economy→ Off→ Auto→ Comfort* …

– If the format of the object to send is **2 bytes unsigned**, the communication object ***Ch.x - z object 2 bytes value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect (release detection default value)**
  - **value sending (pressing detection default value)**

  By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (0 .. 65535)"** which can assume the following values:

  - from **0 (default value)** to 65535

– If the format of the object to send is **2 bytes signed**, the communication object ***Ch.x - z object 2 bytes value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect (release detection default value)**
  - **value sending (pressing detection default value)**

  By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (-32768 .. +32767)"** which can assume the following values:

  - from -32768 to +32767, **0 (default value)**

– If the format of the object to send is **4 bytes unsigned**, the communication object ***Ch.x - z object 4 bytes value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect (release detection default value)**
  - **value sending (pressing detection default value)**

  By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (0 .. 4294967295)"** which can assume the following values:

  - from **0 (default value)** to 4294967295

– If the format of the object to send is **4 bytes signed**, the communication object ***Ch.x - z object 4 bytes value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect (release detection default value)**
  - **value sending (pressing detection default value)**

  By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (-2147483648 .. 2147483647)"** which can assume the following values:

  - from -2147483648 to 2147483647, **0 (default value)**

– If the format of the object to send is **14 bytes**, the communication object ***Ch.x - z object 14 bytes value*** will be visible and the values that can be set for the two above parameters are:

  - **no effect (release detection default value)**
  - **value sending (pressing detection default value)**

  By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **"Value (ISO characters 8859-1)"** which can assume the following values:

  - 14 alphanumeric characters with ISO/IEC coding 8859-1

➢ *5.1.3 Object sending delay (0..255 seconds)*

This is used to set the delay between the detection of the event associated with sending the command and the effective sending of the command on the bus.
With regard to the objects that range from index B to index H, this parameter indicates the delay between sending the command/value associated with the object with the previous index (z-1) and sending the command/value associated with the object to which the parameter refers; The delay to which reference is made in these cases is calculated from the moment in which the command/value is sent that is associated with the object with the previous index (z-1) and not from the moment in which the event that generates sending is detected (pressing/releasing or short operation).
The set delay will only be executed if the event in progress, associated to the object to which the parameter refers, is associated with any value other than **no effect**; otherwise, the delay is ignored.
The parameter may assume the following values:

- from **0 (default value)** to 255 seconds

NOTE : If a command sequence is being sent with delays, activated by the detection of a certain event (pressing/releasing), the detection of the opposite event will stop the sending of the sequence only if at least one of the actions associated with the detection of the last event is different than **no effect**; otherwise, the command/value sequence will be continue to be sent until the last command/value is sent.

6 Function *“1 push button + stop dimmer”*

This is used to configure the channel to control a dimmer with a single button, increasing and decreasing dimmer brightness always using the same button.
For sending on/off telegrams and brightness control telegrams.
As there is only one button to manage the On/Off and brightness control functions, the operation is managed by differentiating between short operations and long operations:

– a long operation is transformed into a brightness control command. When released, an adjustment stop telegram is sent to stop the brightness increase/decrease operation for the dimmer and to fix the brightness value reached at the moment the stop control command was received.
– a short operation is transformed into an on/off command.

Using this type of function, brightness control depends on the so-called brightness control characteristic curve, which varies from actuator to actuator, based on how the manufacturer designed the curve that regulates power, and as a result brightness. This means that the speed with which brightness reaches its maximum and minimum value does not depend on the commands sent from the push-button panel, but the latter regulates the brightness itself by stopping its increase/decrease based on the desired value. The communication objects that this function enables are ***Ch.x - Switching*** and ***Ch.x - Brightness dimming***.

The basic structure of the menu is as follows (Fig. 6.1):

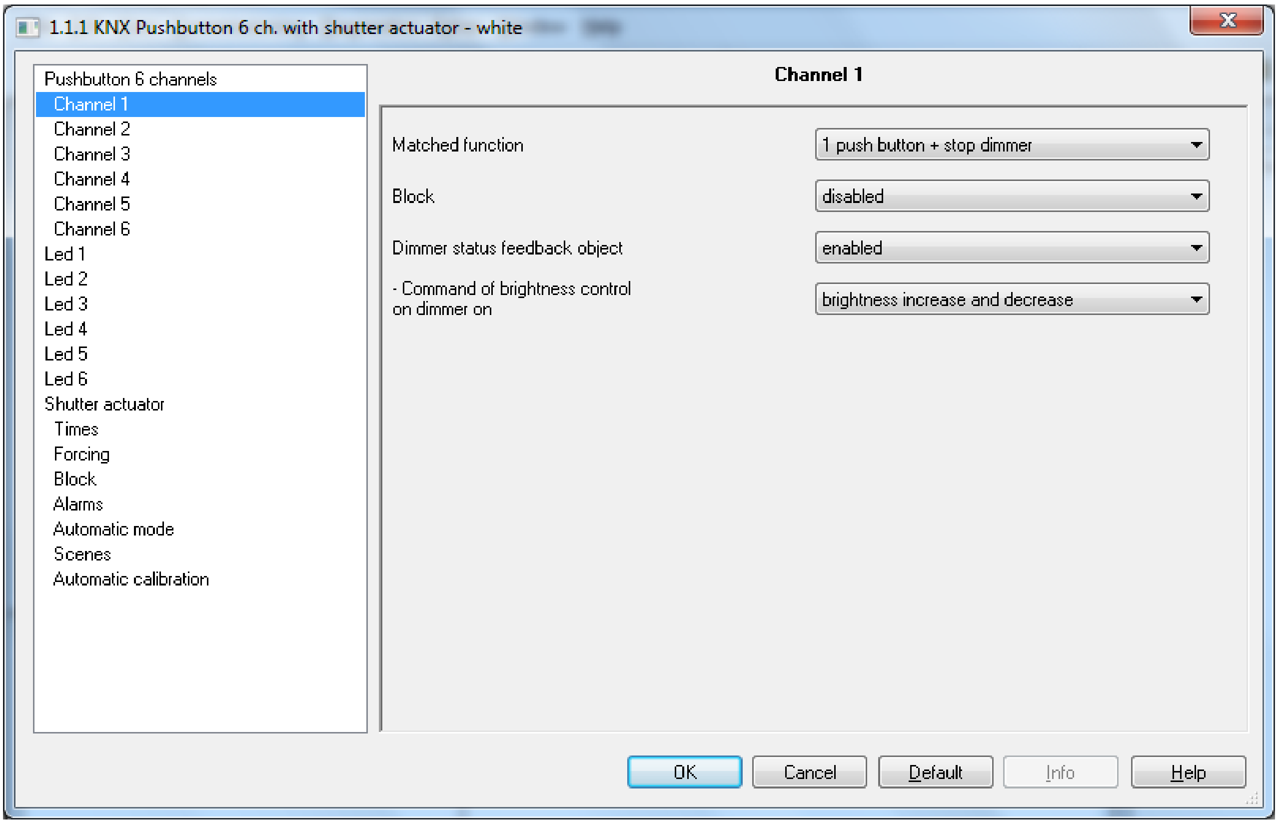


Fig. 6.1

The push-button panel foresees that if the command to be sent is the opposite of the last command sent, this is transformed into:

– long operation: if the last sent command was an off command or a decrease brightness command, the new command will be an increase brightness command; vice versa, if the last sent command was an on command or an increase brightness command, the new command will be a decrease brightness command. In both cases, when released, an adjustment stop telegram is sent to stop the brightness increase/decrease operation for the dimmer and to fix the brightness value reached at the moment the stop control command was received.
– short operation: if the last sent command was an on command, the new command will be an off command; vice versa, if the last sent command was an off command, the new command will be an on

command; the brightness increase/decrease control commands in this case do not determine the value of the last command sent to distinguish the value of the new command to be sent.

This behaviour is changed if the user enables the communication object ***Ch.x - Dimmer status feedback***, via the parameter **“Dimmer status feedback object”**; this parameter may have the following values:

- **disabled (default value)**
- enable

Selecting the value **enable** displays the parameter **“Brightness control commands with dimmer on”** and the communication object ***Ch.x - Dimmer status feedback***, which makes it possible to receive status feedback from the controlled dimmer actuator; the behaviour of the push-button panel is modified as follows:

– long operation: the command that the push-button panel sends depend on the parameter **“Brightness control commands with dimmer on”**, which can assume the following values:

  - only brightness increase
  - only brightness decrease
  - **brightness increase and decrease (default value)**

  By setting **brightness increase and decrease**, if the value of the last two events "last sent command" and "dimmer status feedback" is ON, the new brightness adjustment command to be sent will be the opposite of the last sent command; when released, an adjustment stop telegram is sent to stop the brightness increase/decrease operation for the dimmer and to fix the brightness value reached at the moment the stop control command was received; if the value of the last of the two events "last sent command" and "dimmer status feedback" is OFF, the first command to be sent is increase brightness value, followed by sending the command opposite of the last one sent.

– short operation: if the value of the last of the two events "last sent command" and "dimmer status feedback" is ON, the new command will be an off command; vice versa, if the value of the last of the two events "last sent command" and "dimmer status feedback" is OFF, the new command will be an on command.

If the feedback object is enabled, each time bus voltage is reset, the device sends a status reading command for this object to update the push-button panel about the status of the connected devices.

7 Function *"cyclic sending 1 push button dimmer"*

This is used to configure the channel to control a dimmer with a single button, increasing and decreasing dimmer brightness always using the same button, with defined and settable control steps.
As there is only one button to manage the On/Off and brightness control functions, the operation is managed so that each time the button is pressed the opposite command is sent in comparison to the last sent command and by differentiating between short operations and long operations:

– a long operation is transformed into a brightness control command. No telegram is sent when released.
– a short operation is transformed into an on/off command.

Unlike the function ***1 push button + stop dimmer***, it is possible to define both the brightness various steps as well as the time that passes between sending different commands, if the long operation continues over time; therefore the sending of the control stop telegram is not necessary when releasing the button, even though the control follows the power/brightness characteristic curve, as it is the command that is sent from the push-button panel that determines the percentage variation. The communication objects that this function enables are ***Ch.x - Switching*** and ***Ch.x - Brightness dimming***.

The structure of the menu is as follows (Fig. 7.1):

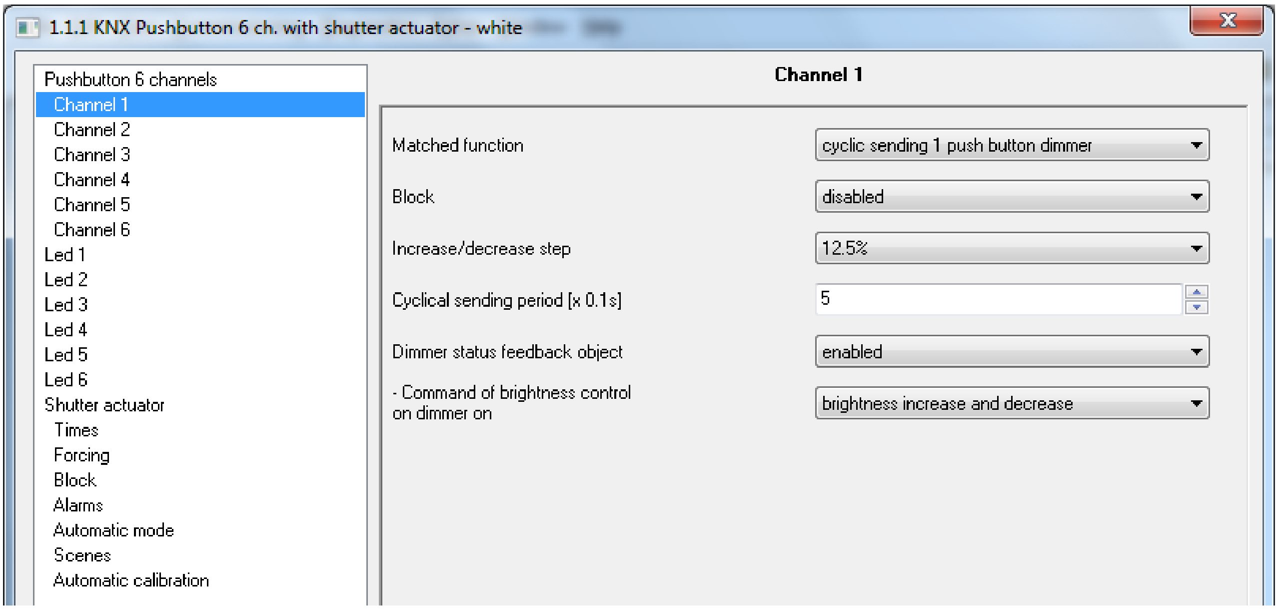


Fig. 7.1

7.1 Parameters

➢ 7.1.1 Increase/decrease step

This is used to set the percentage value of the brightness variation associated with the brightness increase/decrease commands. In this way, as soon as a long operation is detected, the device sends the first increase/decrease command with the set percentage; The values that can be set are:

- 100%
- 50%
- 25%
- **12.5% (default value)**
- 6.25%
- 3.125%
- 1.56%

If pressing is maintained, the device will cyclically send the command until release is detected.

➢ 7.1.2 Cyclical sending period [x 0.1s]

This is used to set the time that passes between sending subsequent increase/decrease commands if pressing is maintained. When released, no telegram is sent but only the cyclical sending of the brightness control commands is stopped.
The values that can be set are:

- from 3 to 50, **5 (default value)**

In summary, if a long operation is detected, the device sends the first increase/decrease command with the set percentage and, if the pressing is maintained, it will cyclically send the command until release is detected.

EXAMPLE: for example, if the value for item **Long operation minimum time** in the ***Main*** menu is set to ***0.5 sec***, the parameter **Increase/decrease step** is set to **12.5%** and parameter **Cyclical sending period [x 0.1s]** is set to **3** (0.3 sec) and pressing is detected:

– 0.5 seconds after detecting the pressing of the button, a long operation is recognized and as a result the first 12.5% brightness increase/decrease telegram is sent.
– from this moment, for every 0.3 seconds that pressing is continued, the device will send a new 12.5% brightness increase/decrease command until the release of the button is detected
– when released, no telegram is sent but the cyclical sending is stopped

➢ 7.1.3 Dimmer status feedback object

Refer to chapter 6 for the settings of this parameter

8 Function *“1 push button shutter control”*

This is used to configure the channel to control a shutter with a single button, regulating the upward and downward travel of the shutter and, depending on the device version, controlling louvres opening/closing.
As only one button manages the louvres up/down and control functions, the operation is managed so that each time the button is pressed, a command is sent that is the opposite of the last movement signal received by the actuator that manages the shutter; a differentiation is made between short and long operations:

– a long operation is transformed into an up/down movement command. If the last received movement signal was “up”, the new command will be a down command, and vice versa.
– a short operation is transformed into a louvres control command. If the last received movement signal was “up”, the new command will be a closing louvres control command however, if the last received movement signal was “down”, the new command will be an opening louvres control command. If the shutter is moving, the louvres control command will only stop the shutter up/down movement.

The communication objects this function enables are ***Ch.x - Shutter movement***, ***Ch.x - Shutter stop/Louvres control*** and ***Ch.x - Movement feedback***.

The structure of the menu is as follows (fig. 8.1):

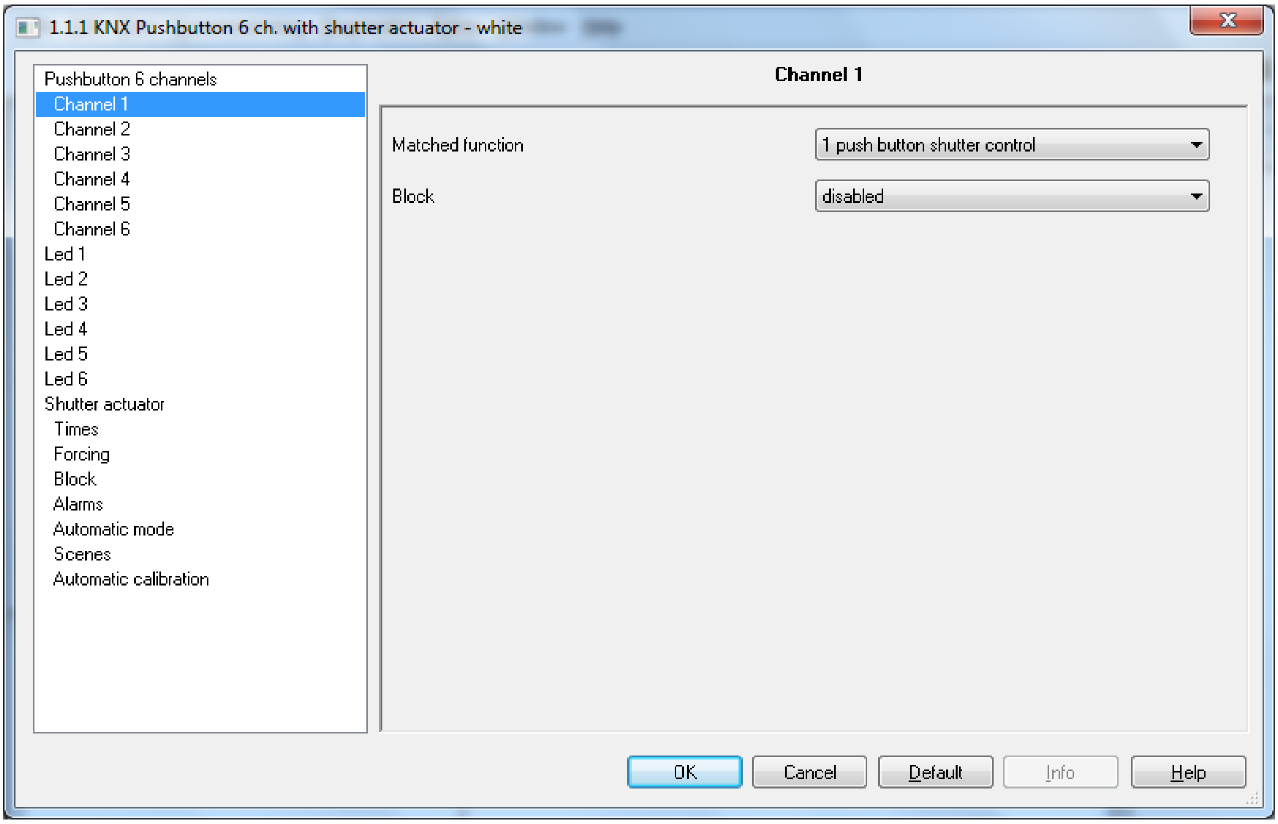


Fig. 8.1

9 Function *“scene management”*

This is used to configure the channel to send scene memorizing and execution commands, with the possibility of sending the scene memorizing command following a command received from the bus. Only one scene can be managed for each channel.
A differentiation is made between short and long operations:

– a long operation is transformed into a scene storing command.
– a long operation is transformed into a scene execution command.

The communication objects that this function enables are ***Ch.x - Scene*** and ***Ch.x - Scene storing trigger***.
When a long operation is recognized, to provide the user with a visual confirmation that the scene storing command was sent, night signalling will be deactivated for a brief period (blink); If night lighting is disabled, when the long operation is detected, a light signalling (blink) is briefly activated with the currently selected color; this effect has priority over all the light effects activated by the bus (see chapter 13 “Led X menu”); The structure of the menu is as follows (fig. 9.1):

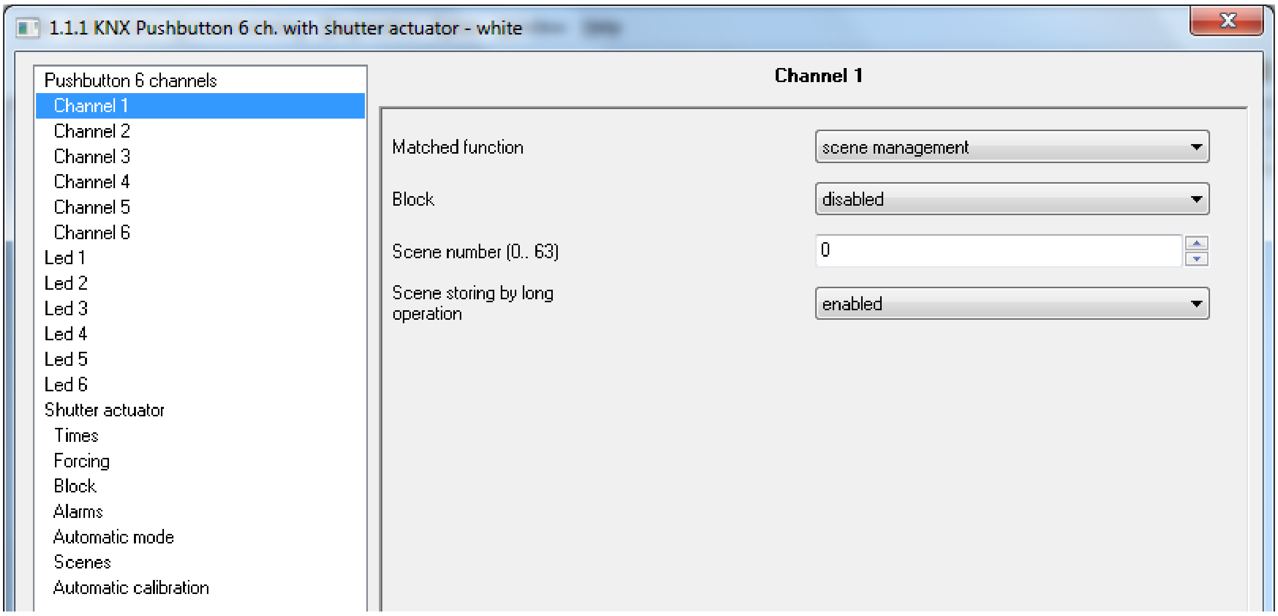


Fig. 9.1

9.1 Parameters

➢ *9.1.1 Scene number (0..63)*

This is used to set the value of the scene to be recalled/stored and as a result the relative values that are sent via the object ***Ch.x - Scene***. The possible values are:

- from **0 (default value)** to 63

➢ *9.1.2 Scene storing by long operation*

This enables the sending of a scene memorizing command when a long operation is recognized.
The values that can be set are:

- disabled
- **enabled (default value)**

The device will send the scene storing command after a long operation is detected only if the value **enabled** is selected; by selecting the value **disabled**, a long operation is not recognized and the long operation triggers the sending of the scene execution command (as for a short operation).
Independently of the value set for the above parameter, it is possible to indirectly generate the sending of the scene memorizing command after receiving a bus telegram on the object ***Ch.x - Scene storing trigger*** (both with value “1” as well as with value “0”); each time the device receives a telegram on that object, a scene memorization telegram will be sent immediately.

10 Function *“switching sequences”*

This is used to send a command sequence after a certain pressing has been detected.
The structure of the menu is as follows (fig. 10.1):

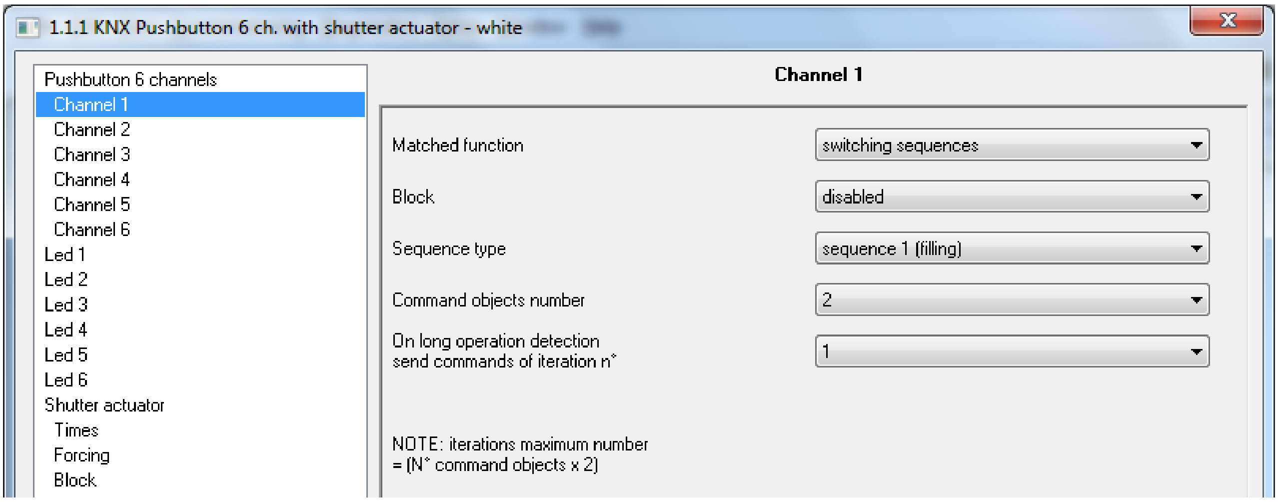


Fig. 10.1

10.1 Parameters

➢ *10.1.1 Command objects number*

This is used to set the number of commands that make up the sequence itself; based on the value set for this item, the communication objects ***Ch.x - Sequence z*** are enabled (with **z** between A and H).
The values that can be set are:

- from **2 (default value)** to 8

➢ *10.1.2 Sequence type*

This is used to set the type of sequence to be sent.
The values that can be set are:

- **sequence 1 (filling)** **(default value)**
- sequence 2 (sum)
- sequence 3 (random)

Sequence 1 (filling) consists in: each time pressing is detected (edge) the device sends a sequence that follows the filling progress on the communication objects. This sequence consists in activating one communication object a time, in cascade, until all the objects have the logical value “1” and in deactivating the objects in cascade until they again have the logical value “0”. Taking into consideration a sequence that includes 3 commands, at each iteration, the sent commands will be:

Edge no.	Value sent on ***Ch.x - C sequence***	Value sent on ***Ch.x - B sequence***	Value sent on ***Ch.x - A sequence***
1st edge	0	0	1
2nd edge	0	1	1
3rd edge	1	1	1
4th edge	0	1	1
5th edge	0	0	1
6th edge	0	0	0

Once the 6th edge is detected, the sequence will start from the beginning

The table shows how, considering the increasing/decreasing trend of the sequence, the most significant bit of the sequence, in this particular case, is the one for the communication object ***Ch.x - C sequence*** whereas the least significant is always the one for the object ***Ch.x - A sequence***.

Sequence 2 (sum) consists in: each time pressing is detected (edge) the device sends a sequence that follows the sum progress on the communication objects. This sequence consists in counting the detected edges and converting this value into a binary format, distributing it on the enabled communication objects. Taking into consideration a sequence that includes 3 commands, at each iteration, the sent commands will be:

Edge no.	Value sent on ***Ch.x - C sequence***	Value sent on ***Ch.x - B sequence***	Value sent on ***Ch.x - A sequence***
1st edge	0	0	1
2nd edge	0	1	0
3rd edge	0	1	1
4th edge	1	0	0
5th edge	1	0	1
6th edge	1	1	0
7th edge	1	1	1
8th edge	0	0	0

Once the 8th edge is detected, the sequence will start from the beginning

The table shows how the trend of the sent commands depends on the count of the detected edge; in fact it starts with the binary coding of value 1 up to, in this specific case, the coding of value 7 and then the counting starts again for the next edge. Also in this case, the most significant bit in the sequence is the one for the communication object ***Ch.x - C sequence*** whereas the least significant is always the one for object ***Ch.x - A sequence***.

Sequence 3 (free) allows the user to directly set the value for each command for each set edge; this setting enables the parameter **"Number of sequence iterations"** and the configuration menu **z object channel x** (one for each enabled command). The parameter **"Number of sequence iterations"** allows to set the number of iterations (edges) that make up the sequence; The values that can be set are:

- from **2 (default value)** to 16

Based on the value set for this item, the **Channel x z object** menu will display or hide the parameters **"Iteration 1 object value"**, **"Iteration 2 object value"**, **"Iteration 3 object value"**, **"Iteration 4 object value"**, **"Iteration 5 object value"**, **"Iteration 6 object value"**, **"Iteration 7 object value"**, **"Iteration 8 object value"**, **"Iteration 9 object value"**, **"Iteration 10 object value"**, **"Iteration 11 object value"**, **"Iteration 12 object value"**, **"Iteration 13 object value"**, **"Iteration 14 object value"**, **"Iteration 15 object value"** and **"Iteration 16 object value"**, which can assume the following values:

- value "0"
- **value "1" (default value)**

The structure of the menu **Channel x z object** is as follows(fig. 10.2):

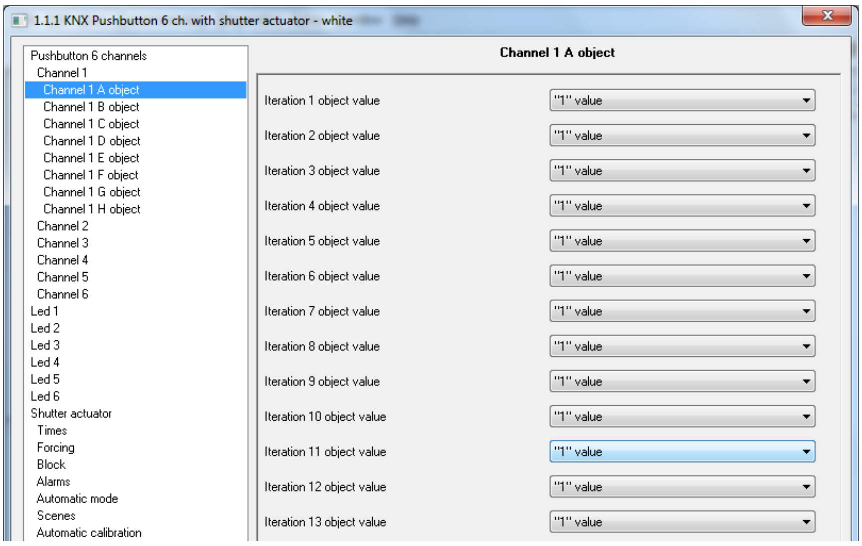


Fig. 10.2

➢ *10.1.3 On long operation detection, send commands of iteration n°*
This is used to define which iteration of the sequence is to be sent when a long operation is detected; The values that can be set are:
- from 1 to 256, **1 (default value)**

EXAMPLE: with reference to the previous table, supposing that the value set by the user is **3**, when a long operation is detected the device will send:

Edge no.	Value sent on ***Ch.x* - *C sequence***	Value sent on ***Ch.x* - *B sequence***	Value sent on ***Ch.x* - *A sequence***
1st edge	0	0	1
2nd edge	0	1	1
3rd edge	1	1	1
4th edge	0	1	1
5th edge	0	0	1
6th edge	0	0	0

"Filling" sequence

Edge no.	Value sent on ***Ch.x* - *C sequence***	Value sent on ***Ch.x* - *B sequence***	Value sent on ***Ch.x* - *A sequence***
1st edge	0	0	1
2nd edge	0	1	0
3rd edge	0	1	1
4th edge	1	0	0
5th edge	1	0	1
6th edge	1	1	0
7th edge	1	1	1
8th edge	0	0	0

"Sum" sequence

Once the long operation is detected and the sequence relative to the set iteration is sent, the next time a short operation is detected the sequence related to the iteration following the one associated with the long operation will be sent (in this example, the sequence association with iteration 4 will be sent).
In summary, the value set for the parameter **"On long operation detection, send commands of iteration n°"** defines both the sequence to be sent as well as the value with which the iteration counter should start when a long operation is detected.

11 *“Actuator local command”* function

This is used to dedicate the channel to controlling the actuator on the device without having to make additional logical connections via the ETS software. No communication object is enabled by this function.
The structure of the menu is as follows (fig. 11.1):

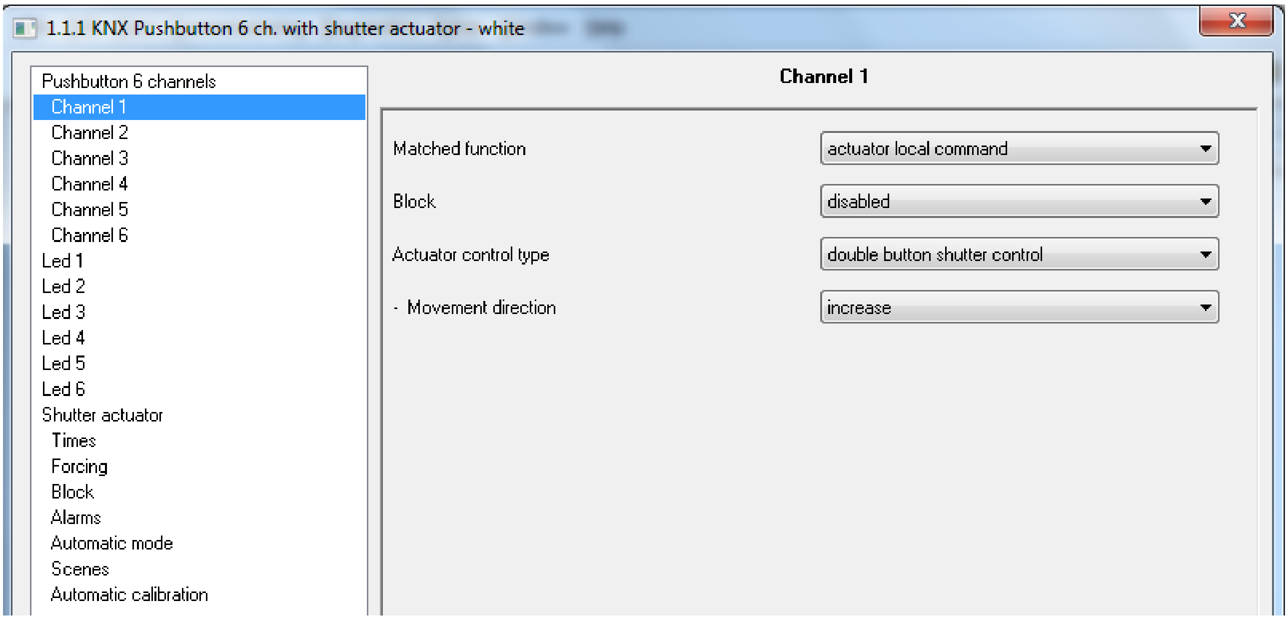


Fig. 11.1

11.1 *Parameters*

➢ *11.1.1 Actuator control type*

As the motor command actuator on board the device implements various types of operations (up/down movements, percentage positions, automatic calibration and scenes) and functions with different priorities (forcing and block), it is necessary to define which of the following functions the push-button associated with the channel must perform using the parameter **“Actuator control type”**; the values that can be set are:

- **single button shutter control (default value)**
- double button shutter control
- set percentage position
- automatic calibration
- scene
- forced positioning
- block
- local command pushbutton

the difference between the values **single button shutter control** and **local command pushbutton** is that the first acts as a command received by the bus on the object ***Actuator movement*** and ***Actuator louvres adjustment/stop*** or ***Actuator stop*** (and as a result, has a lower priority than the meteo alarm functions, automatic mode, forcing or block of the actuator itself), whereas the second directly switches the relay, ignoring any active function, whose activation status is not is not changed in any manner.
Various parameters are displayed depending on the selected value:

– If the actuator control type is **single button shutter control, automatic calibration** or **local command pushbutton**, no new parameter is displayed.

– If the actuator control type is **double button shutter control**, the parameter **“Movement direction”** is displayed, which is used to define which movement direction to manage with the button; a long operation will produce a movement of the load, a short operation will stop the movement or control of the louvres. The values that can be set are:

- **Up (default value for channels with uneven index)**
- **Down (default value for channels with even index)**

– If the actuator control type is **set percentage position**, this displays the parameters **“Shutter/venetian blind percentage position”** and **“Louvres percentage position”**. The first makes it possible to set the percentage position that the shutter/venetian blind controlled by the actuator must assume when the push-button is pressed; the second, if the actuator operating mode is venetian blind, makes it possible to define the percentage position that the venetian blind louvres controlled by the actuator must assume when the push-button is pressed. The values that can be set for both parameters are:

  - from **0% (default value)** to 100% with step of 5%

– If the actuator control type is **forced positioning**, this displays the parameters **“Sending on pressing detection”** and **“Sending on release detection”**.
The parameter **“Sending on pressing detection”** is used to set the command to send to the on-board motor command actuator after the pressing of the button associated with the channel has been detected.
The parameter **“Sending on release detection”** is used to set the command to send to the on-board motor command actuator after the release of the button associated with the channel has been detected; The values that can be set for the two above parameters are:

  - **activate forcing down (default pressing value)**
  - activate forcing up
  - disable forced positioning
  - forc up/forc down cyclic switching
  - cyclical switching forcing down/deactivate forcing
  - cyclical switching forcing up/deactivate forcing
  - **no effect (release default value)**

  By selecting **cyclical switching**, in this case no communication object will be displayed as the device is always updated about the function activation status.

– If the actuator control type is **block**, this displays the parameters **“Sending on pressing detection”** and **“Sending on release detection”**.
The parameter **“Sending on pressing detection”** is used to set the command to send to the on-board motor command actuator after the pressing of the button associated with the channel has been detected.
The parameter **“Sending on release detection”** is used to set the command to send to the on-board motor command actuator after the release of the button associated with the channel has been detected; The values that can be set for the two above parameters are:

  - deactivation
  - **activation (pressing default value)**
  - cyclical switching
  - **no effect (release default value)**

  By selecting **cyclical switching**, in this case no communication object will be displayed as the device is always updated about the function activation status.

– If the type of actuator control is **scene**, this displays the parameters **“Scene number (0.. 63)”** and **“Scene storing by long operation”**.
The parameter **“Scene number (0.. 63)”** is used to set the value of the scene to be recalled/stored; if this value does not coincide with what is associated with the relative parameter in the **Scene** menu of the motor command actuator channel, no scene will be recalled/memorized. The possible values are:

  - from **0 (default value)** to 63

  Scene storing by long operation enables the sending of a scene memorizing command when a long operation is recognized. The values that can be set are:

- disabled
- **enabled (default value)**

The device will send the scene storing command after a long operation is detected only if the value **enabled** is selected; by selecting the value **disabled**, a long operation is not recognized and the long operation triggers the sending of the scene execution command (as for a short operation).

➢ *11.1.2 Sending on pressing detection*

If any other value other than **scene** is selected for the previously described parameter, this parameter will be displayed that makes it possible to set the command to be sent to the on-board actuator after the pressing of the button associated with the channel has been detected.

➢ *11.1.3 Sending on release detection*

If any other value other than **scene** is selected for the parameter described in 11.1.1, this parameter will be displayed that makes it possible to set the command to be sent to the on-board actuator after the release of the button associated with the channel has been detected.

12 *“Channel X/Y” (coupled channels)* menu

If the channel operation is matched, a dedicated menu is displayed for each channel pair, called **Channel x/y**. The menu structure changes based on the value set for the **“Matched function”** parameter. For the sake of simplicity, the parameters enabled according to the value set for the above parameter are listed in the following paragraphs.
The basic structure of the menu is as follows (Fig. 12.1):

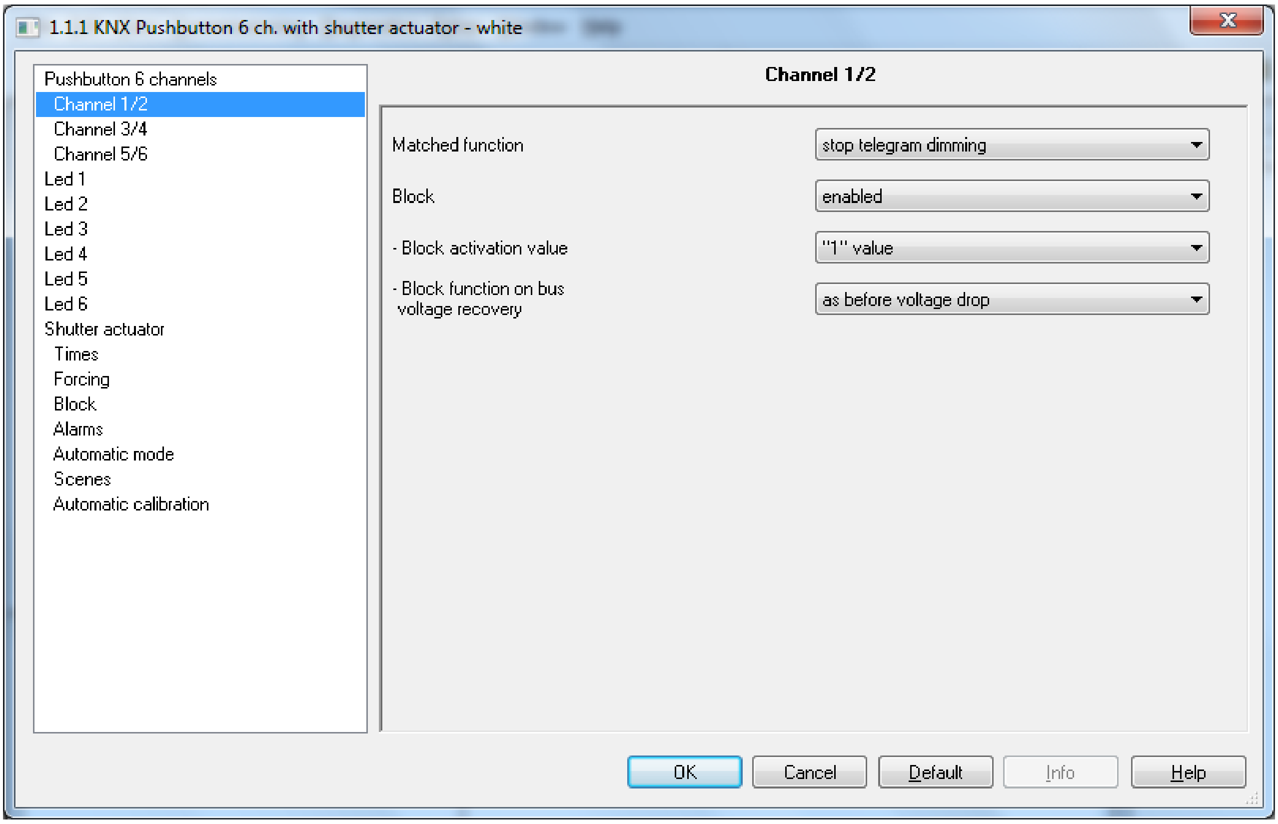


Fig. 12.1

12.1 Parameters

➢ *12.1.1 Matched function*

This is used to define the function implemented by the combined channels; The values that can be set are:

- **stop telegram dimming (default value)**
  (See chapter 6 Function “1 push button + STOP dimmer”)

- cyclic telegram dimming
  (See chapter 7 Function “cyclic sending 1 push button dimmer”)

- shutter control
  (See paragraph 8 Function “1 push button shutter control”)

➢ *12.1.2 Block*

Refer to paragraph 4.1.2 for the settings of this parameter

13 *“Led X”* menu

This is used to define and personalize the operation of the signalling LEDs associated with the channel.
The signalling LED can assume an amber or green color, one of them can be used for the night lighting function or signalling of motor control actuator movement in progress or both can be independently managed via the relative communication objects. The communication objects enabled by this function are ***Led x - Effect 1***, ***Led x - Effect 2***, ***Led x - Effect 3***, ***Led x - Effect 4*** and ***Led x - Effect 5***.
The basic structure of the menu is as follows (fig. 13.1):

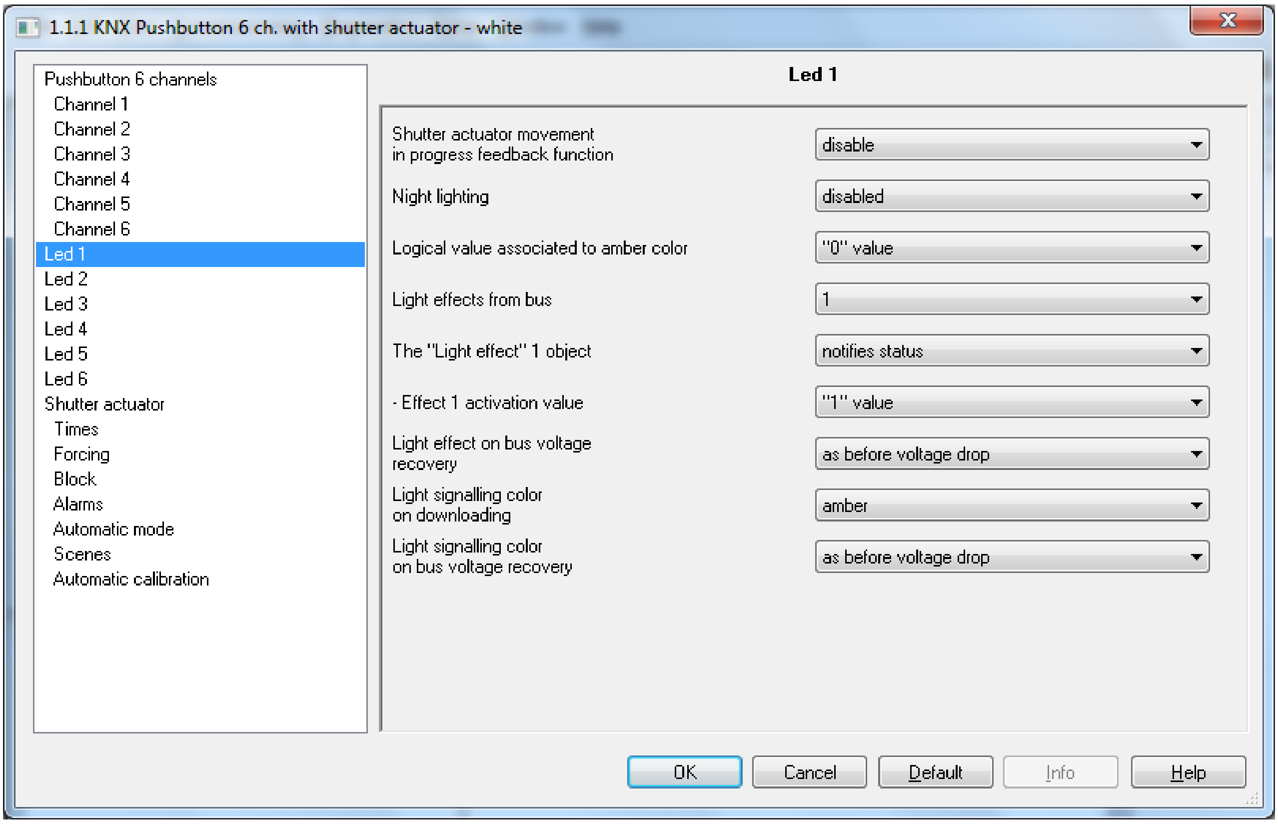
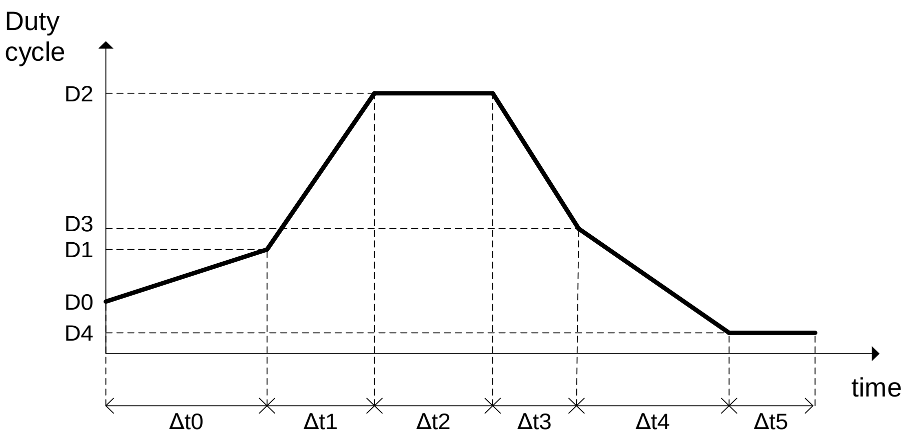


Fig. 13.1

Given how complex it is for the final user to understand all the parameters that are theoretically configurable to obtain the desired light effect (see figure below), a predefined set of light effects is defined, limiting as far as possible the number of parameters that can be configured for each effect.



13.1 Parameters

➢ 13.1.1 Shutter actuator movement in progress feedback function

As the actuator on board the device does not have its own status light signalling LED, the LED associated with channel x can carry out this function; this parameter therefore makes it possible to configure the actuator status signalling led and can assume the following values:

- **Disabled (default value)**
- enable

➢ 13.1.2 Night lighting

This is used to enable and define the night lighting color associated with channel x; The values that can be set are:

- disabled
- **enable amber signalling (default value)**
- enable green signalling

Selecting the value **disabled**, displays the parameter **"LED color for signalling movement in progress"**; Selecting a value other than **disabled** displays the parameter **"LED brightness percentage value for night lighting"**.

Also, selecting the value **disabled**, displays the communication object ***Led x - Color choice*** that allows to change from the bus the signalling LED color and the parameters **"Logical value associated to amber color"**, **"Light signalling color on downloading"** and **"Light signalling color on bus voltage recovery"**. Selecting a value other than **disabled** displays the parameter **"LED brightness percentage value for night lighting"**.

If night lighting is active, the coloring of the light effects activated by the bus command (via objects ***Led x - Effect 1*** etc.) is associated to a LED not used for signalling; vice versa, if night lighting is disabled, the coloring of these effects depends on the communication object ***Led x - color choice*** and on the parameter value **Px "Light signalling color on bus voltage recovery"**.

➢ 13.1.3 LED color for signalling movement in progress

It allows to define the color used for signalling load movement in progress the possible values are:

- amber
- **green (default value)**

➢ 13.1.4 LED brightness percentage value for night lighting

This is used to define the LED brightness intensity percentage with the night lighting function; The values that can be set are:

from 5% to **100% (default value)**

➢ 13.1.5 Movement direction to notify

This parameter defines the direction of the movement in progress, which is signalled by the LED x; the possible values are:

- **increase (default value for led with uneven index)**
- **decrease (default value for led with even index)**
- both

If there is a movement in progress in the direction defined in the parameter **"Movement direction to notify"**, the signalling led is on fixed; vice versa, the signalling LED is off (localisation LED on, if enabled).

If the communication objects for meteo alarm management are enabled, when the alarm condition is detected, the green and amber LEDs will flash alternatively to signal the alarm in progress condition.

With the parameter **"Night lighting"**, it is possible to enable and define the color of the night lighting associated with channel x; The values that can be set are:

- disabled
- **amber signalling enabled (default value)**
- enables green signalling

selecting the value **disabled**, displays the communication object ***Led x - Color choice*** that allows to change from the bus the signalling LED color and the parameters **“Logical value associated to amber color”**, **“Light signalling color on downloading"** and **“Light signalling color on bus voltage recovery”**.
Selecting a value other than **disabled** displays the parameter **"LED brightness percentage value for night lighting"**.

➢ *13.1.6 Logical value associated to amber color*

This defines which logical value received on the communication object ***Led x - Color choice*** is associated with the amber color; as a result, the opposite logical value is associated with the green color. The values that can be set are:

- **value “0” (default value)**
- value “1”

➢ *13.1.7 Light effects from bus*

This is used to enable various communication objects for the activation of the light signalling via bus telegram; The values that can be set are:

- **none (default value)**
- 1
- 2
- 3
- 4
- 5

Based on the number of selected effects, this will display the parameters **“The “Light effect” 1 object”**, **“Effect 1 activation value”**, **“The “Light effect” 2 object”**, **“Effect 2 activation value”**, **“The “Light effect” 3 object”**, **“Effect 3 activation value”**, **“The “Light effect” 4 object”**, **“Effect 4 activation value”** and **“The “Light effect” 5 object”**, **“Effect 5 activation value”**.

➢ *13.1.8 The “Light effect” x object*

This is used to associate the light effect to display via the bus communication objects ***Led x - Effect 1***, ***Led x - Effect 2***, ***Led x - Effect 3***, ***Led x - Effect 4*** and ***Led x - Effect 5***; via these communication objects, it is possible to activate/deactivate the set light signalling from the bus. The values that can be set for this parameter are:

- **notifies status** **(default value)**
- switching night signalling off (visible only if green/amber signalling is enabled)
- activate fast cyclic blinking
- activate slow cyclic blinking
- executes down ramp
- executes heartbeat 1
- executes heartbeat 2
- executes jellyfish
- executes blinking
- executes heartbeat 3
- executes heartbeat 4
- executes fast blinking
- executes slow blinking
- executes very slow blinking
- executes 3 blinkings
- executes personalized effect

if the selected value is **personalized effect**, this will display the new configuration menu **Personalize effect y**, with 1≤Y≤5 (see paragraph 13.2)

13.1.9 Effect x activation value

This is used to define which logic value received via the objects ***Led x - Effect 1***, ***Led x - Effect 2***, ***Led x - Effect 3***, ***Led x - Effect 4*** and ***Led x - Effect 5*** activates the associated light signalling. The values that can be set for this parameter are:

- value “0”
- **value “1” (default value)**

Via the communication objects ***Led x - Effect 1*****,** ***Led x - Effect 2*****,** ***Led x - Effect 3*****,** ***Led x - Effect 4*** **and** ***Led x - Effect 5***, it is possible to activate/deactivate the associated light effect via bus commands; by activating a light effect that is different than the one already active, the new effect will be implemented and the old effect will be deactivated. This means that only one effect may be active and, once it is deactivated, the signalling led will deactivate and the night lighting will activate without having to deactivate the previously activated light effects; to deactivate the led, the active light effect must be deactivated.

13.1.10 Light effect on bus voltage recovery

Makes it possible to set the light signalling effect that is activated when the bus voltage is recovered. The values that can be set are:

- no effect
- light effect 1
- light effect 2
- light effect 3
- light effect 4
- light effect 5
- **as before voltage drop (default value)**

Selecting the value **no effect**, if night lighting is active, when bus voltage is recovered the night lighting will be activated. By selecting any value other than **no effect**, if night lighting is active, the selected effect will be replicated when the bus voltage is recovered with coloring that is NOT dedicated to night lighting.

13.1.11 Light signalling color on downloading

This parameter is visible if night lighting is disabled, and is used to preset the color of the light signalling color on downloading the application parameters via ETS. The values that can be set are:

- **amber (default value)**
- green

13.1.12 Light signalling color on bus voltage recovery

This parameter is visible if night lighting is disabled, and is used to preset the color of the light signalling color on bus voltage recovery. The values that can be set are:

- amber
- green
- **as before voltage drop (default value)**

13.2 "Personalize effect y" menu

This menu is displayed if the value for parameter **""Light effect" y object"** in menu **Led x** is set to **execute personalized effect**.
In this menu, all the parameters used for creating the light effects are visible and configurable; in this way, the user can create a personalized light effect. These light effects are activated/deactivated by the objects ***Led x - Effect y***.

The basic structure of the menu is as follows (fig. 13.2):

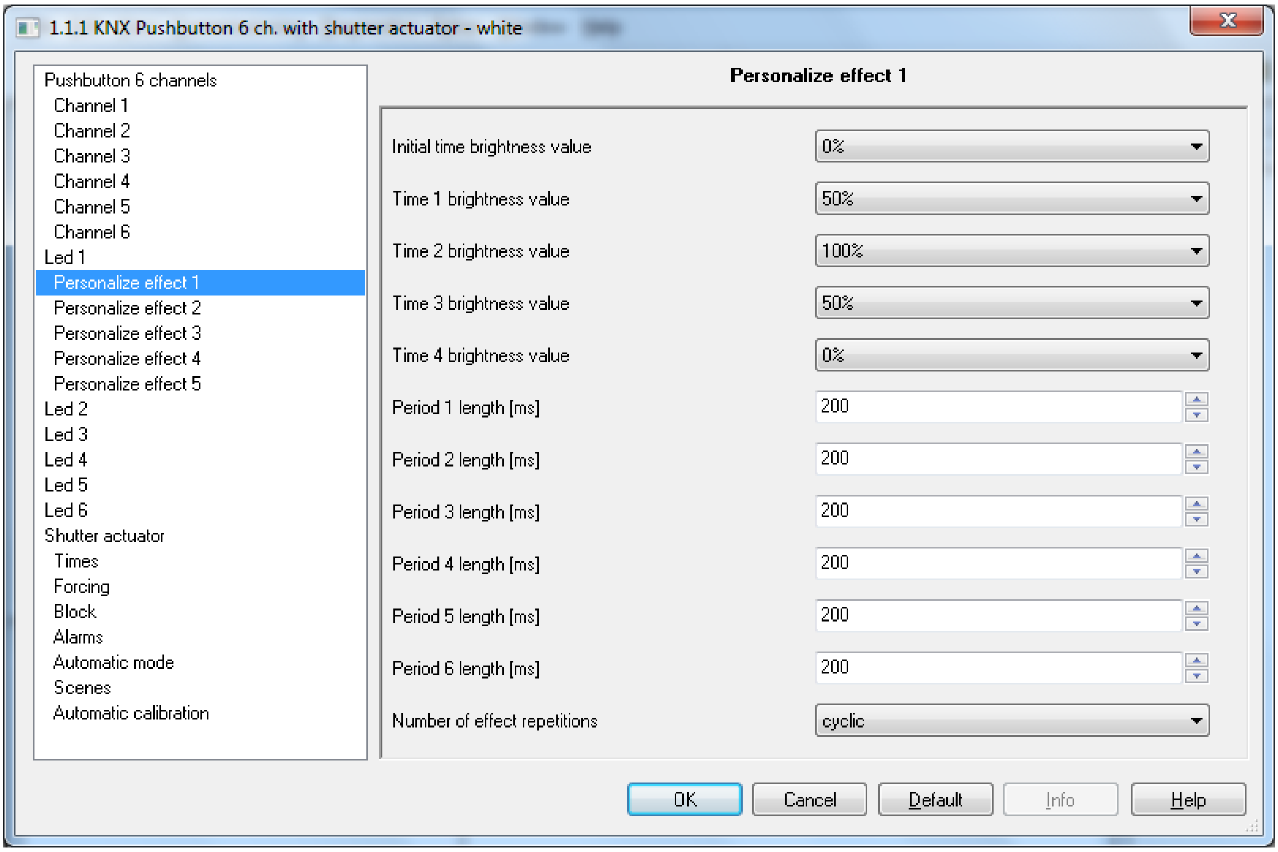
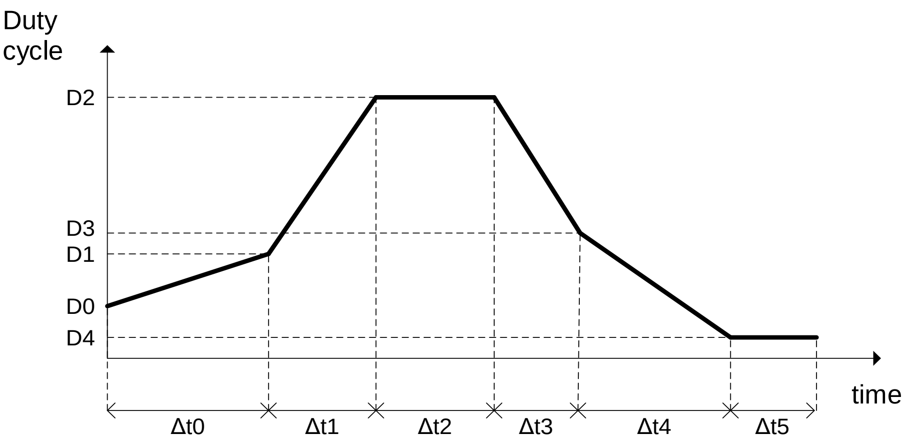


Fig. 13.2

Using the figure shown below as a reference, all the variables that create the light effect can be set by the user in this configuration menu.



The available variables are:

- 6 parameters Δ_t0, Δ_t1, Δ_t2, Δ_t3, Δ_t4, Δ_t5 expressed in ms (0-65535) that define the duration of the brightness control ramp between the value $\Delta_t(n)$ and $\Delta_t(n+1)$ or, in this case Δ_t2, the interval for which the brightness D2 will be maintained
- 5 parameters D0, D1, D2, D3, D4 (0-255) that define LED brightness values (duty-cycle). The values that can be set for these parameters will be displayed to the user as a percentage value between 0% and 100%, according to the proportion D(n)= Parameter*255/100
- 1 parameter that defines the number of cycles for **repeating** the effect (1 .. 254);

The parameters used to define the brightness values to reproduce are **"Initial time brightness value"** (D0), **"Time 1 brightness value"** (D1), **"Time 2 brightness value"** (D2), **"Time 3 brightness value"** (D3) and **"Time 4 brightness value"** (D4), which may be assigned the following values:

- from 0% to 100%, **0 % (default value D0-D4)**, **50% (default value D1-D3)** and **100% (default value D2)**

The parameters used to define the duration of the control ramp between a brightness value and the next are **"Period 1 length [ms]"** (Δ_t0), **"Period 2 length [ms]"** (Δ_t1), **"Period 3 length [ms]"** (Δ_t2), **"Period 4 length [ms]"** (Δ_t3), **"Period 5 length [ms]"** (Δ_t4) and **"Period 6 length [ms]"** (Δ_t5), which can assume the following values:

from 0 to 65535, **200 (default value)**

➢ *13.2.1 Number of effect repetitions*

This defines how many times the set light effect must be repeated when an activation command is received from the bus; The values that can be set are:

- 1, 2, .. 254, **cyclic (default value)**

14 *“Shutter actuator”* menu

The **Shutter actuator** menu includes the parameters that define the general behaviours of the shutter actuator implemented in the device.
The priority of the functions implemented by the shutter actuator channel is shown in the following table:

Function	**Priority**	
Movement	1	Low
Stop (/louvres control)	1	
Position command	1	
Louvres position command	1	
Scene	1	
Behaviour at the end of automatic calibration	1	
Behaviour upon deactivation of wind alarm 1	1	
Behaviour upon deactivation of wind alarm 2	1	
Behaviour upon deactivation of wind alarm 3	1	
Behaviour on rain alarm deactivation	1	
Behaviour on ice alarm deactivation	1	
Behaviour at block deactivation	1	
Behaviour on forced positioning disactivation	1	
Automatic calibration	2	
Automatic mode	3	
Weather alarms	4	
Lock	5	
Forcing	6	
Actuator local command (if “local command button” function)	7	
Actuator behaviour at bus voltage recovery	8	
Alarm status when BUS voltage is reset	9	
Value of lock object when BUS voltage is reset	10	
Forced positioning status on bus voltage recovery	11	
230V voltage fall status (stop/no action)	12	
BUS voltage fall status (stop/no action)	12	High

The basic structure of the menu is as follows (fig. 14.1):

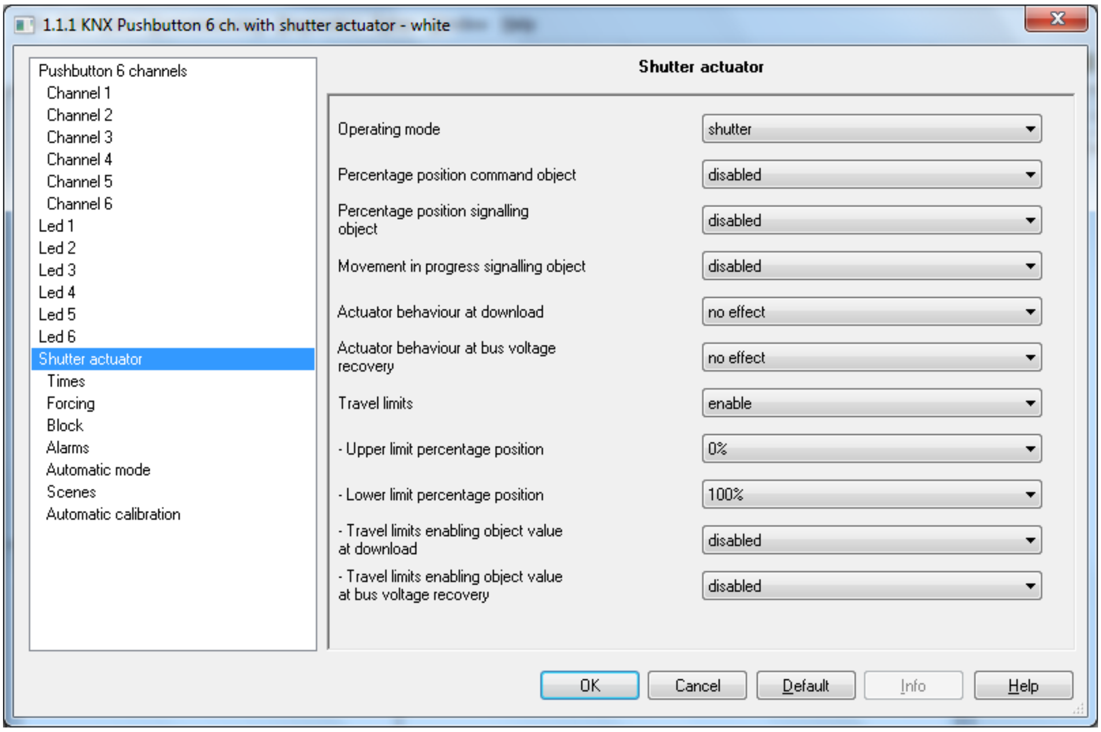


Fig. 14.1

14.1 Parameters

➢ *14.1.1 Operating mode*

This parameter may have the following values:

- **shutter** (default value)
- venetian blind

A different communication object is associated to the two values with regard to stopping shutter or venetian blind movement; selecting shutter, the visible object is ***Actuator stop***, selecting venetian blind, the object is ***Actuator louvres adjustment/stop***. If any value is received on both objects while the load is movement will cause it to stop immediately; the reception of the object with the load stopped in the case of **shutter** does not involve any action, in the case of **venetian blind** it involves a regulation step of the opening regulation (if the object assumes the value "0") or when closing (if the object assumes the value "1").
Selecting the value **venetian blind** displays the **Louvres adjustment** menu.

Using the communication object (always displayed) ***Actuator movement*** it is possible to move the shutter or venetian blind up or down, then stopping it via object ***Actuator stop*** (or ***Actuator louvres adjustment/stop***) or when the movement time has expired.

➢ *14.1.2 Percentage position command object*

This parameter is used to enable the communication object through which the percentage position of the shutter/venetian blind is set, leaving the actuator with the task of executing the necessary movement; The values that can be set are:

- **disabled** (default value)
- enabled

Selecting the value **enabled** displays the communication object ***Actuator position command*** which is used to specify the percentage position to use for actuation.

➢ *14.1.3 Percentage position signalling object*

The device can signal the percentage position of the load controlled by the actuator following a movement according to the value set by this parameter which can assume the following values:

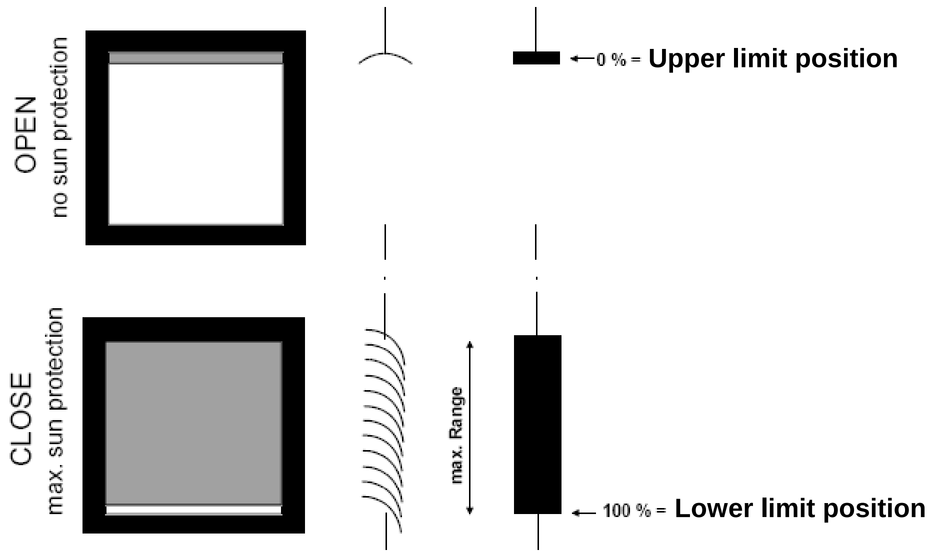
- **disabled (default value)**
- enabled on demand only
- enabled on change and switching on

setting any other value than **disabled** displays the object ***Actuator position signalling*** through which the device percentage position at which the load is located with a communication object.
Setting instead **active on change and when switching on**, the transmission occurs each time that there is a change in the position of the load and at bus voltage recovery.

With regard to the percentage position of the object, the following convention is used:

- 0% →shutter/venetian blind completely raised
- 100%→shutter/venetian blind completely lowered

An example is shown below :



venetian blind shutter

➢ *14.1.4 Movement in progress signalling object*

This makes it possible to enable movement in progress signalling object using this parameter, which can assume the following values:

- **disabled (default value)**
- enabled on demand only
- enabled on variation

Setting any other value than **disabled** displays the actuator movement signalling object through which the devices signals with a value equal to "1" that a downward movement of the load is in progress and with a value equal to "0" that an upward movement is in progress. Setting the value "active on change", the transmission occurs each time that a load movement is in progress.

➢ *14.1.5 Actuator behaviour at downloading*

It is possible to define the actuator behaviour after the application is downloaded from the ETS software via this parameter, which can assume the following values:

- **no effect (default value)**
- automatic calibration execution

➢ *14.1.6 Actuator behaviour at bus voltage recovery*

The behaviour of the actuator when the bus voltage is recovered is determined by this parameter, which can assume the following values:

- **no effect (default value)**
- up movement
- down movement
- automatic calibration execution
- percentage position

Setting the value **percentage position** displays the parameters **"Percentage position at bus voltage recovery"** with the following values

- from **0% (default value)** to 100% with step of 5%

and if the operating mode is **venetian blind**, the parameter **"Louvres percentage position on bus voltage recovery"** with the following values:

- from **0% (default value)** to 100% with step of 5%

Before going to the desired position, the actuator automatically executes the calibration operation.

➢ *14.1.7 Travel limits*

In some applications, it is useful to limit the object travel to a certain interval to prevent it from being damaged or damaging other objects; the parameter **“Travel limits”** makes it possible to enable load travel limitation by using the communication object ***Actuator travel limits enabling*** which can assume the following values:

- **disable (default value)**
- enable

Selecting the value **enable** displays both the communication objects ***Actuator travel limits enabling***, ***Actuator upper travel limit***, ***Actuator lower travel limit*** and the parameters **“Upper limit percentage position”** and **“Lower limit percentage position”**; the latter are used to set the load travel interval limits once the limitation is enabled through the object ***Actuator travel limits enabling***. The values that can be set are:

- from **0% (upper limit default value)** to **100% (upper limit default value)** with step of 5%

The travel limits can be modified via the communication objects ***Actuator upper travel limit*** and ***Actuator lower travel limit***.

Setting the value **enabled** for the parameter **“Travel limits”** it is possible to define the initial value of the object ***Actuator travel limits enabling*** on downloading from ETS via the parameter **“Value of object enabling travel limits on downloading”** and at bus voltage recover with the parameter **“Value of object enabling travel limits on bus voltage recovery”**; The values that can be set are:

- **disabled (default value)**
- enabled
- as before voltage drop

The value **“as before voltage drop ”** is available <u>only</u> for the parameter **“Value of object enabling travel limits on bus voltage recovery”**.
The travel limits are only using for movements initiated by a communication object with a lower priority than the weather alarms.

14.2 “Times” function

The **Times** menu includes the parameters that define the characteristic time values of the load connected to the actuator.
The basic structure of the menu is as follows (fig. 14.2):

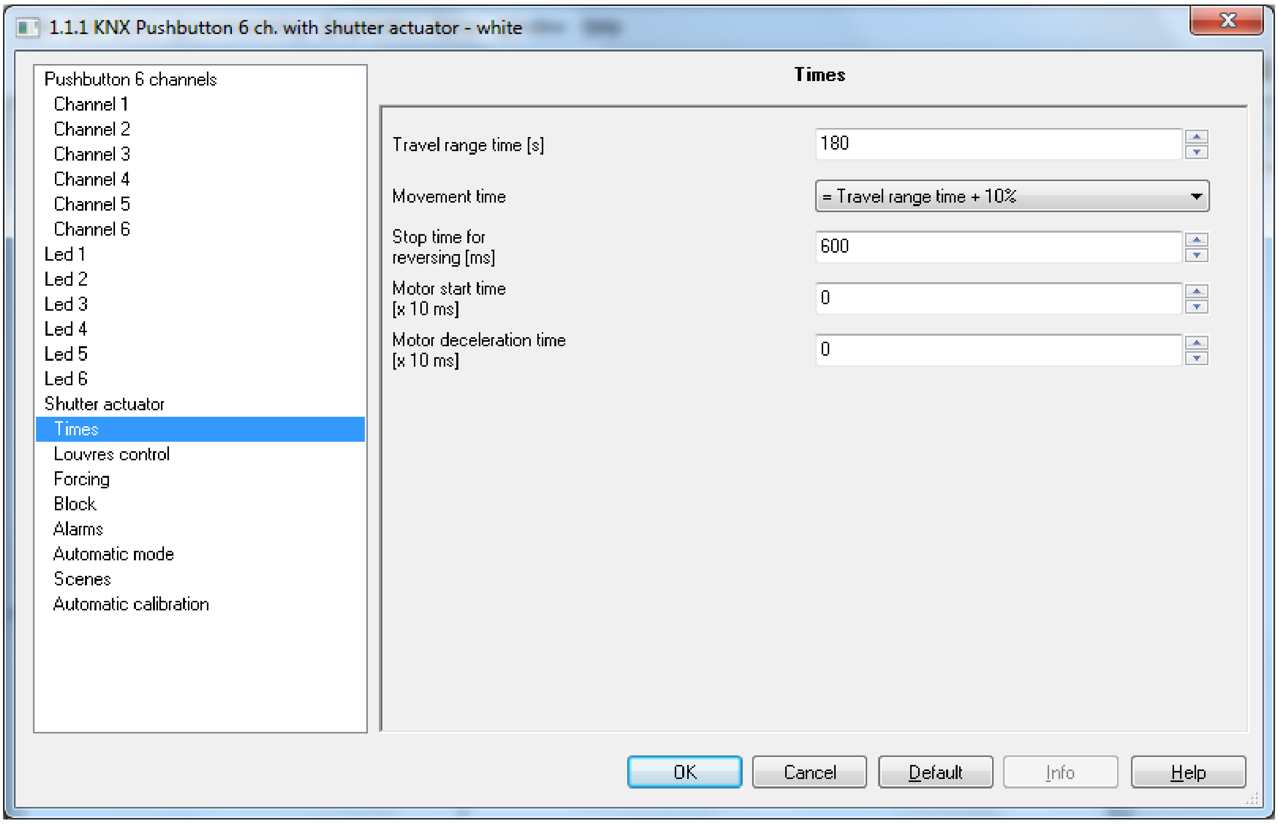


Fig. 14.2

➢ *14.2.1 Travel range time [s]*

The device calculates the percentage positions and executes partial movements based on the total load travel time, which is the time the load requires to move from the “fully up” position (0%) to the “fully closed” position (100%); this value must be set for the parameter **“Travel time [s]”** which can assume the following values:

- from **1** to **3000** with step of 1, **180 (default value)**

➢ *14.2.2 Movement time*

Normally the device continues to power the motor contact even if the limit switch has been reached, to make sure that the load has reached the limit switch position. It is possible to set this timeout value via this parameter which can assume the following values:

- = Travel range time
- = Travel range time + 2%
- = Travel range time + 5%
- **= Travel range time + 10% (default value)**
- = Travel range time + 20%

➢ *14.2.3 Stop time for reversing [ms]*

To prevent damaging the motor, a delay must be entered between receiving a command for movement in the opposite direction of the current movement and the effective reversal of direction; this time is normally provided by the shutter/venetian blind manufacturer and must be rigorously observed. This parameter may have the following values:

- from **50** to **10000**, **600 (default value)**

➢ *14.2.4 Motor start time [x 10 ms]*

From the point of view of keeping the precision of the correct position of the load unchanged over time, following the execution of scenes or percentage value commands, to rectify a possible start delay introduced by the motor (delay between the moment when the motor is powered and the moment when it starts to run), it is possible to add a time extension, which is calculated to reach the requested percentage position; this value, which is set via this parameter , can assume the following values:

- from **0 (default value)** to 255

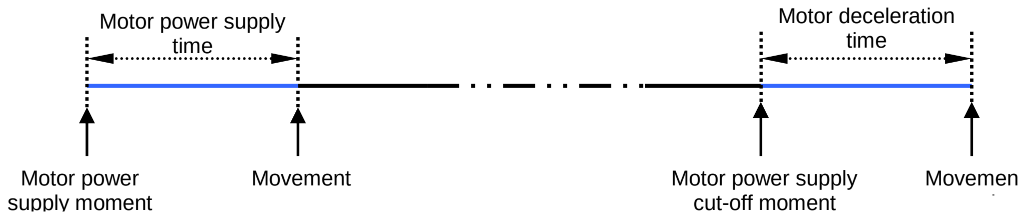
The times set for the parameters **"Motor start time [ms]"** and **"Motor deceleration time [ms]"** are only used to calculate the time required to reach a certain intermediate percentage position (following a specific command, or following the reproduction of a position associated with a scene or prior to a status) and are not used for any movements carried out via the object ***Actuator movement***; the factor set for the above listed parameters must be multiplied by 10 milliseconds.

➢ *14.2.5 Motor deceleration time [x 10 ms]*

Furthermore, some motors, after power is cut off, continue to move for a brief period of time. This parameter can be used to set the time value that is subtracted from the calculated time to reach an intermediate percentage position requested by the user, so that the adjustment is more precise. The values that can be set are:

- from **0 (default value)** to 255

The following figure shows an example of these two parameters



The times set for the parameters **"Motor start time [ms]"** and **"Motor deceleration time [ms]"** are only used to calculate the time required to reach a certain intermediate percentage position (following a specific command, or following the reproduction of a position associated with a scene or prior to a status) and are not used for any movements carried out via the object ***Actuator movement***; the factor set for the above listed parameters must be multiplied by 10 milliseconds.

14.3 "Louvres control" function

The menu **Louvres adjustment**, which is displayed if the operating mode set in the **Motor command actuator** menu is **venetian blind**, contains the parameters that define the characteristic time values of the venetian blind connected to the actuator, making it possible to control the louvres through various bus commands or automatically at the end of a specific movement.

The basic structure of the menu is as follows (fig. 14.3):

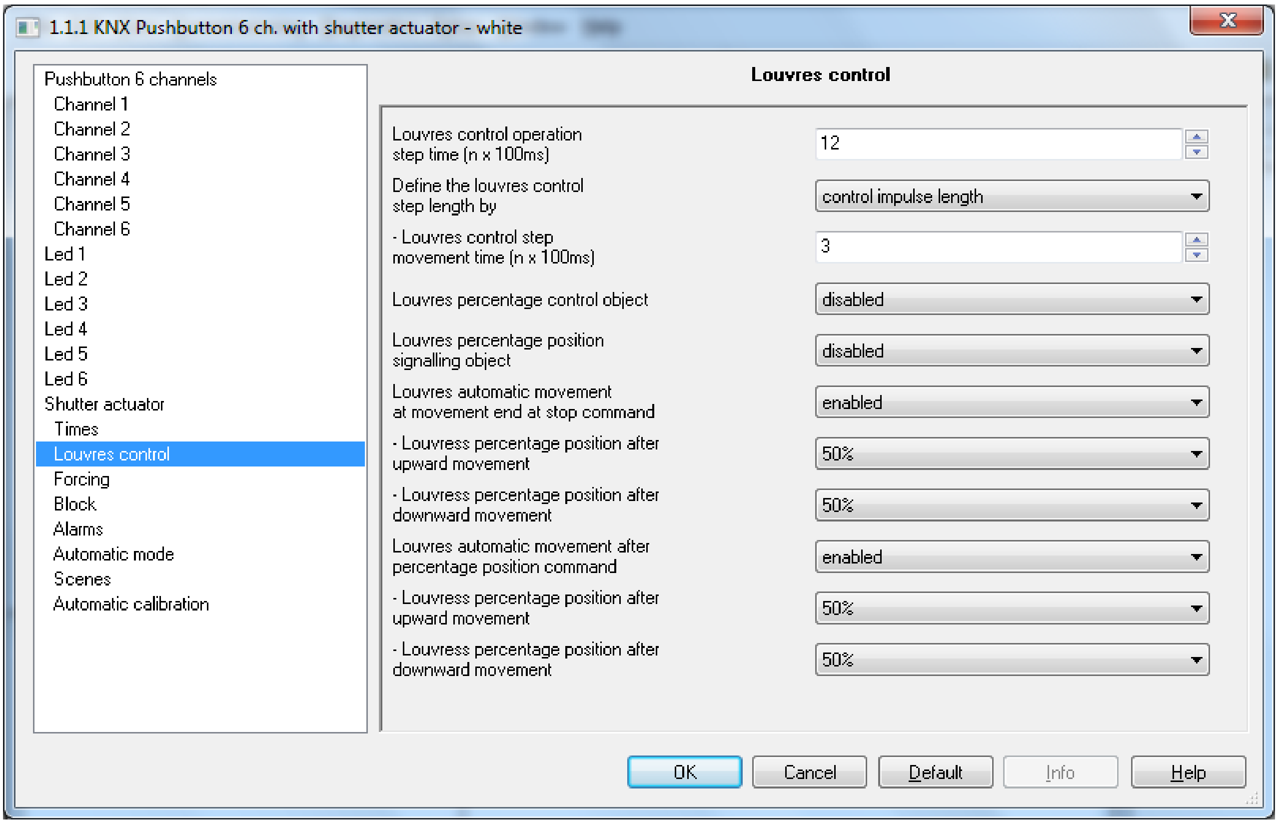


Fig. 14.3

➢ *14.3.1 Louvres control operation step time (n x 100ms)*

This parameter is used to set the factor that, when multiplied by the base (100 milliseconds), represents the duration of the louvres adjustment from the "completely open" condition to the "completely closed" condition; The values that can be set are:

- from **1** to **120**, **12 (default value)**

➢ *14.3.2 Define the louvres control step length by*

This is used to define if the duration of a louvres adjustment set is based on time or based on the number of steps necessary for the louvres to pass from completely open to completely closed. The parameter may have the values:

- **control impulse lenght (default)**
- number of control impulses

In the first case, the parameter **"Louvres adjustment step activation time (n x 100ms)"** will become available, which can be used to set the factor that, when multiplied by the base (100 milliseconds), represents the period of power supply to the motor that corresponds to a louvres adjustment step; the values that can be set are:

- from **1** to **120**, **3 (default value)**

In the second case the parameter **"Number of steps for total louvres closure/opening"** will become available. The parameter may have the values:

- from **1** to **15**, **4 (default value)**

➢ *14.3.4 Louvres percentage control object*

It is possible to regulate the louvres position by setting from the bus the percentage value to which they should be brought according to the following convention:

0%→ louvres completely open,
100%→ louvres completely closed,

This parameter may have the following values:

- **disabled (default value)**
- enabled

Selecting **enabled** displays the communication object ***Actuator louvres adjustment command*** that makes percentage adjustment of the louvres from the bus possible.

➢ *14.3.5 Louvres percentage position signalling object*

This makes it possible to enable the louvres percentage position signalling, which can assume the following values:

- **disabled (default value)**
- enabled on demand only
- enabled on change and switching on

Selecting a value other than **disabled** displays the communication object ***Actuator louvres position signalling***.

➢ *14.3.6 Louvres automatic movement at movement end at stop command*

The device can be configured to automatically adjust the louvres at the end of a movement after receiving a stop command via this parameter, which can assume the following values:

- **disabled (default value)**
- enabled
- enabled after movement up
- enabled after movement down

Selecting the value **enabled after movement down** or **enabled** displays the parameter **"Louvres percentage position after downward movement"**, which makes it possible to set the percentage value of the louvres after a downward movement; selecting the value **enabled after upward movement** or **enabled** displays the parameter **"Louvres percentage position after upward movement"**, which makes it possible to set the percentage value of the louvres after an upward movement. Both the values can assume the following values:

- from **0% (up default value)** to **100% (down default value)** with step of 5%

It is possible to set automatic louvres adjustment following the execution of a percentage position movement command **"Louvres automatic movement after percentage position command".** The parameter may have the following values:

- **disabled (default value)**
- enabled
- enabled after movement up
- enabled after movement down

Selecting the value **enabled after movement down** or **enabled** displays the parameter **"Louvres percentage position after downward movement"**, which makes it possible to set the percentage value of the louvres after a downward movement; selecting the value **enabled after upward movement** or **enabled** displays the parameter **"Louvres percentage position after upward movement"**, which makes it possible to set the percentage value of the louvres after an upward movement. Both the values can assume the following values:

- from **0% (up default value)** to **100% (down default value)** with step of 5%,

14.4 “Forcing” function

It is possible to force the load status in a certain condition (settable) after receiving the communication object ***Actuator priority command*** that activates the forcing function; until it is deactivated, any command received on all other input communication objects will not be executed. The forcing function has the highest priority over all others.

The basic structure of the menu is as follows (fig. 14.4):

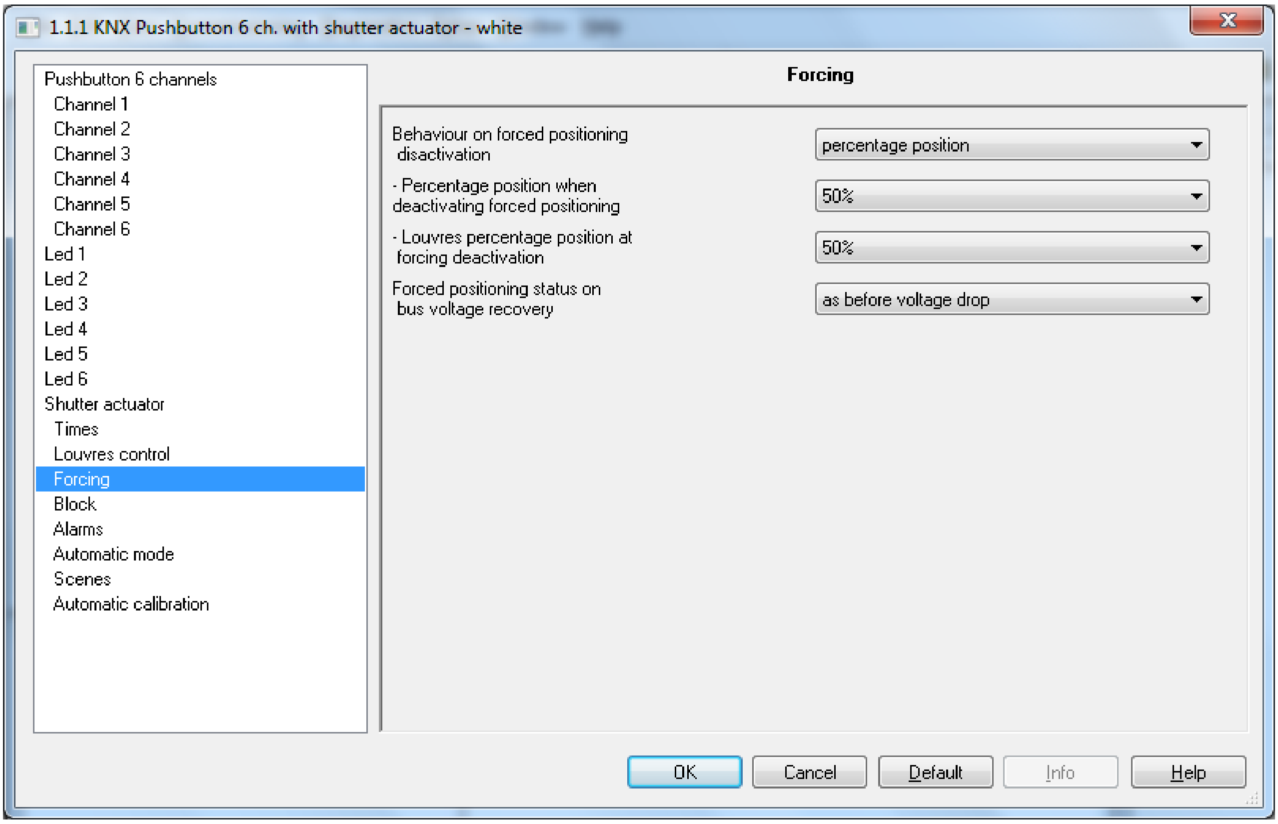


Fig. 14.4

The semantics of the command follows what is shown in the following table:

bit1	bit 0	
0	0	Deactivates forcing
0	1	Deactivates forcing
1	0	Forcing UP
1	1	Forcing DOWN

The forcing activation command has higher priority than any other bus command, including the block function.

➢ *14.4.1 Behaviour on forced positioning disactivation*

This parameter can be used to set actuator behaviour once forcing is deactivated; The values that can be set are:

- **no effect (default value)**
- up movement
- down movement
- back to previous position
- follows last command received
- percentage position
- stop

The option **percentage position** displays the parameters **“Percentage position when deactivating forced positioning”**, which can be used to select the desired value; The values that can be set are:

- from **0%** to **100%** with a step of 5%, **50% (default value)**

and, if the operating mode is **venetian blind**, the parameter **Louvres percentage position at forcing deactivation** which can be used to select the desired value; The values that can be set are:

- from **0%** to **100%** with a step of 5%, **50% (default value)**

➢ *14.4.2 Forced positioning status on bus voltage recovery*

This parameter is used to determine the status of forcing on bus voltage recovery. This parameter is useful if the function is active when the bus voltage drops and you want to have the actuator behaviour not be changed after voltage failure. The parameter may assume the following values:

- deactivated
- **as before voltage drop (default value)**

If the value **deactivate** is selected (and forcing was active before the bus voltage drop), when bus voltage is recovered the forcing function will be deactivated and the actuator will execute what is set for parameter “**Behaviour on forced positioning disactivation”.** If the value set for this last parameter is **follows last command received**, the actuator will execute the last command received before the bus voltage drop that, as a result, must be stored to the non-volatile memory.
If the value **as before bus voltage drop** is selected (and forcing was activated before bus voltage drop), when the bus voltage is recovered the forcing function is reactivated and the actuator reproduces the position prior to the voltage drop. If a forcing deactivation command is received, if the parameter “**Behaviour on forced positioning disactivation”** takes on the value **follows last command received**, the actuator executes the last command received before the bus voltage drop that, as a result, must be stored to the non-volatile memory.

14.5 “Block” function

It is possible to block the device in a certain (settable) condition after receiving the communication object ***Actuator block*** that activates the block function; until this is deactivated, any command received on all other input communication objects will not be executed, with the exception of on-off forcing activation commands. The block function is the function with the highest priority, with the exception of the forcing function.
The basic structure of the menu is as follows (fig. 14.5):

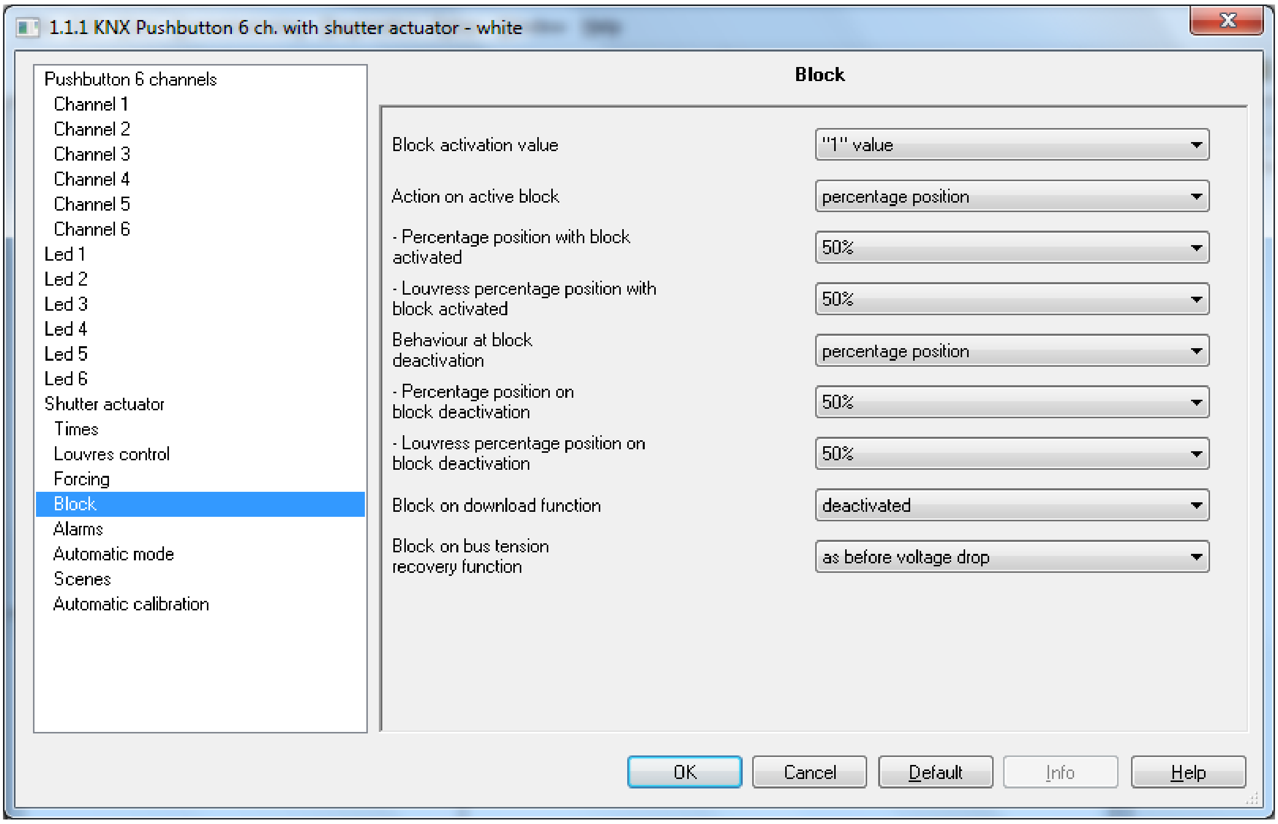


Fig. 14.5

➢ *14.5.1 Block activation value*

This determines which logic value the communication object must have to activate the block function; the value opposite of the set value will deactivate the function. The values that can be set are:

- value “0”
- **value “1” (default value)**

➢ *14.5.2 Action with active block*

This parameter is used to set the behaviours of the actuator channel upon activation of the block and the values it can assume are as follows:

- **no effect (default value)**
- up movement
- down movement
- percentage position
- stop

The option **percentage position** displays the parameter **“Percentage position with block activated”**, which makes it possible to define the percentage position of the load and, if the operation is venetian blind, the parameter **“Louvres percentage position with block activated”**, which makes it possible to define the percentage position of the venetian blind louvres; The values that can be set are:

- from **0%** to **100%** with a step of 5%, **50% (default value)**

➢ *14.5.3 Behaviour at block deactivation*
This is used to set the behaviour of the actuator channel after block deactivation; the possible values are:

- **no effect (default value)**
- up movement
- down movement
- back to previous position
- follows last command received
- percentage position
- stop

The option **percentage position** displays the parameter **"Percentage position on block deactivation"**, which makes it possible to define the percentage position of the load and, if the operation is venetian blind, the parameter **"Louvres percentage position on block deactivation"**, which makes it possible to define the percentage position of the venetian blind louvres;

- from **0%** to **100%** with a step of 5%, **50% (default value)**

➢ *14.5.4 Block on download function*
This defines the logic value assumed by the communication object "**Channel Y - Block**" on ETS downloading; the possible values are:

- **deactivated (default value)**
- active

➢ *14.5.5 Block on bus tension recovery function*
This is used to set the logic value assumed by the communication object ***Actuator block*** each time the bus voltage is recovered; the possible values are:

- deactivated
- active
- **as before voltage drop (default value)**

14.6 “Alarms” function

To prevent damage from atmospheric events to the load connected to the actuator, a specific “**Alarms**” menu configuration is available that makes it possible to interface dedicated communication objects with weather sensors (rain sensor, wind sensor etc.), permitting the device to behave in a certain manner when particular weather conditions occur. This function can be enabled via the parameter **“Alarm functions”**, which can assume the following values:

- **deactivated (default value)**
- active

Selecting **active** displays all the configuration parameters and communication objects necessary to manage the function.

The basic structure of the menu is as follows (fig. 14.6):

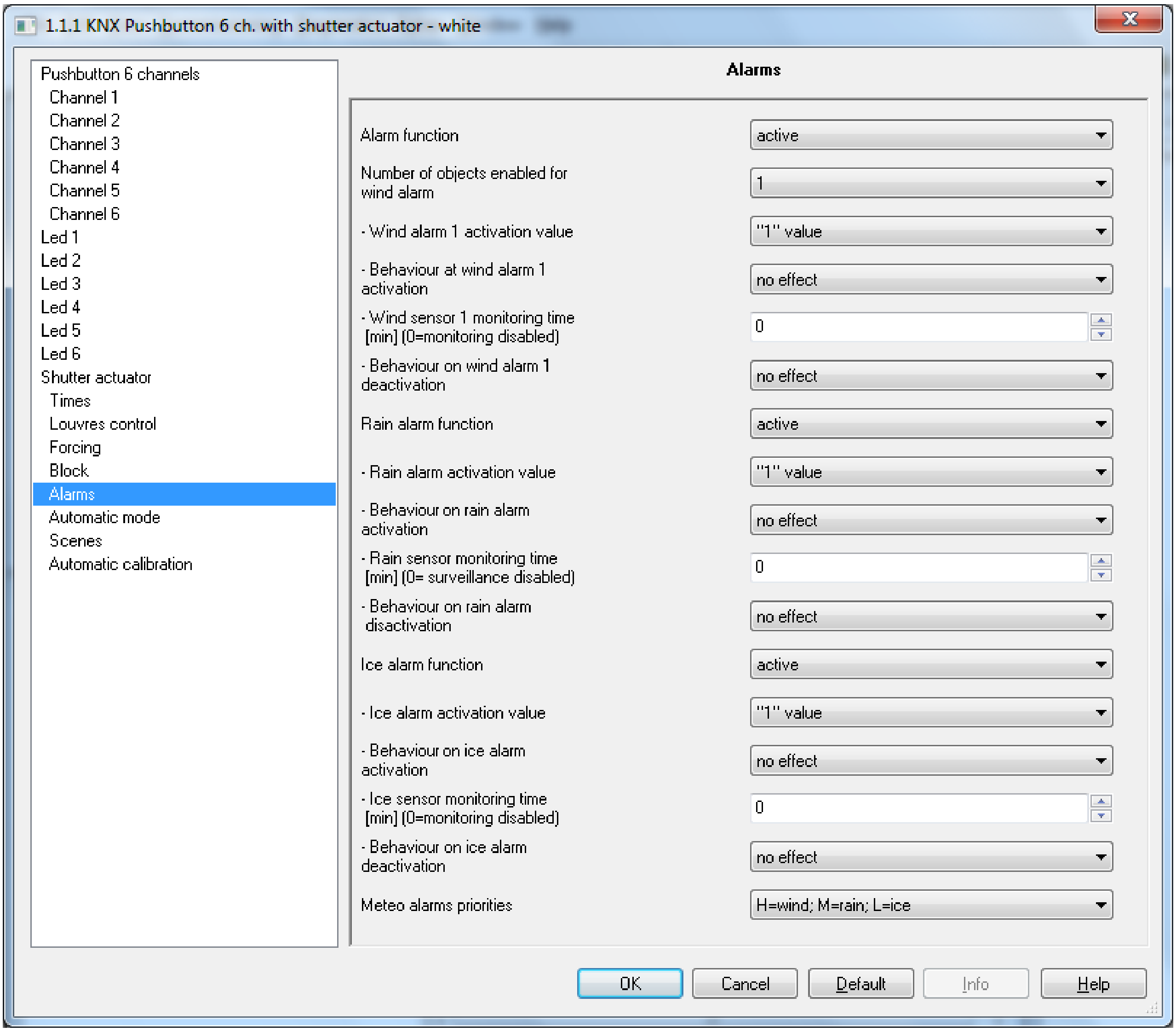


Fig. 14.6

14.6.1 Number of objects enabled for wind alarm

This parameter can be used to define how many communication objects the user wants to use to monitor the wind; The values that can be set are:

- **0 (default value)**, 1, 2, 3

Based on the number of enabled objects, this will display the communication objects ***Wind alarm 1 actuator***, ***Wind alarm 2 actuator*** and ***Wind alarm 3 actuator*** and the parameters **Px "Wind alarm 1 activation value 1"**, **Px "Wind alarm 2 activation value 2"** and **Px "Wind alarm 3 activation value 3"**; These parameters are used to define the value of the communication object associated with the wind alarm; The values that can be set are:

- value "0"
- **value "1" (default value)**

14.6.2 Behaviour at wind alarm x activation

For each enabled wind alarm, the following parameters are displayed **"Behaviour at wind alarm 1 activation"**, **"Behaviour at wind alarm 2 activation"** and **"Behaviour at wind alarm 3 activation"**, which are used to define the behaviour of the device when the wind alarm is active as well as the parameter **"Wind sensor 1 monitoring time [min] (0=monitoring disabled)"**, **"Wind sensor 2 monitoring time [min] (0=monitoring disabled)"** and **"Wind sensor 3 monitoring time [min] (0=monitoring disabled)"**, which are used to monitor the n-th wind sensor.
The parameters **"Behaviour at wind alarm 1 activation"**, **"Behaviour at wind alarm 2 activation"** and **"Behaviour at wind alarm 3 activation"** can assume the following values:

- **no effect (default value)**
- up movement
- down movement
- percentage position
- stop

When the parameter takes on the value **percentage position**, this displays the parameters **"Percentage position with wind alarm 1 active"**, **"Percentage position with wind alarm 2 active"** and **"Percentage position with wind alarm 3 active"**, which makes it possible to define the percentage position of the shutter in case of an n-th wind alarm and, if the operation is venetian blind, the parameter **"Louvres percentage position with wind alarm 1 active"**, **"Louvres percentage position with wind alarm 2 active"** and **"Louvres percentage position with wind alarm 3 active"**, which makes it possible to define the percentage position of the venetian blind louvres. The values that can be set for both parameters are:

- from **0% (default value)** to 100% with step of 5%

14.6.3 Wind sensor x monitoring time [min] (0=monitoring disabled)

The parameters **"Wind sensor 1 monitoring time [min] (0=monitoring disabled)"**, **"Wind sensor 2 monitoring time [min] (0=monitoring disabled)"** and **"Wind sensor 3 monitoring time [min] (0=monitoring disabled)"** can assume the following values:

- from **0 (default value)** to 60

By selecting the value **0**, the n-th wind alarm object will not be monitored. If the n-th wind alarm communication object is not received in the period corresponding to the monitoring time, the actuator executes the same action corresponding to the n-th alarm event.

14.6.4 Behaviour on wind alarm x deactivation

Parameters **"Behaviour on wind alarm 1 deactivation"**, **"Behaviour on wind alarm 2 deactivation"** and **"Behaviour on wind alarm 3 deactivation"** are used to set the behaviour of the device when the n-th wind alarm is deactivated; The values that can be set are:

- **no effect (default value)**
- up movement
- down movement
- return to previous position

- follows last command received
- percentage position
- stop

The option **set percentage value** displays the parameters **“Percentage position upon wind alarm 1 deactivation”**, **“Percentage position upon wind alarm 2 deactivation”** and **“Percentage position upon wind alarm 3 deactivation”**, which are used to define the percentage position of the load and, if the operation is venetian blind, the parameters **“Louvres percentage position upon wind alarm 1 deactivation”**, **“Louvres percentage position upon wind alarm 2 deactivation”** and **“Louvres percentage position upon wind alarm 3 deactivation”**, which define the percentage position of the venetian blind louvres. The values that can be set for both parameters are:

- from **0% (default value)** to 100% with step of 5%

The relative priority for the wind alarms is as follows:
Wind alarm 1 < Wind alarm 2 < Wind alarm 3.
Behaviour on alarm deactivation will be effectively implemented if and only if a wind alarm or another lower priority alarm is not active.

➢ *14.6.5 Rain alarm function*
In addition to the wind alarm, it is possible to enable a communication object for monitoring the rain event via the parameter **“Rain alarm function”**, which can assume the following values:

- **deactivated (default value)**
- active

Enabling the rain alarm displays the communication object ***Actuator rain alarm*** and the parameters **“Rain alarm activation value”**, which is used to define the value of the communication object associated with the rain alarm, the parameter **“Behaviour on rain alarm activation"**, which is used to define the behaviour of the device when the rain alarm is active and the parameter **“Rain sensor monitoring time [min] (0=monitoring disabled)”**, which is used to define the rain sensor monitoring time.

➢ *14.6.6 Rain alarm activation value*
This parameter may have the following values:

- value “0”
- **value “1” (default value)**

➢ *14.6.7 Behaviour at rain alarm activation*
This parameter may have the following values:

- **no effect (default value)**
- up movement
- down movement
- percentage position
- stop

When the parameter takes on the value **percentage position**, this displays the parameters **“Percentage position with active rain alarm”**, which is used to define the percentage position of the load and, if the operation is venetian blind, the parameter **“Louvres percentage position with active rain alarm”**, which is used to define the percentage position of the venetian blind louvres. The values that can be set for both parameters are:

- from **0% (default value)** to 100% with step of 5%

➢ *14.6.7 Rain sensor monitoring time [min] (0=surveillance disabled)*
This parameter may have the following values:

- from **0 (default value)** to 60

By selecting the value **0**, the object enabled for the rain alarm will not be monitored. The meaning of the monitoring time is similar to the one for the wind alarm:

➢ *14.6.8 Behaviour on rain alarm deactivation*
This parameter is used to set the behaviour of the device when the rain alarm is deactivated; The values that can be set are:

- **no effect (default value)**
- up movement
- down movement
- back to previous position
- follows last command received
- percentage position
- stop

The option **set percentage position** displays the parameters **"Percentage position at rain alarm deactivation"**, which makes it possible to define the percentage position of the load and, if the operation is venetian blind, the parameter **"Louvres percentage position at rain alarm deactivation"**, which makes it possible to define the percentage position of the venetian blind louvres. The values that can be set for both parameters are:

- from **0% (default value)** to 100% with step of 5%

➢ *14.6.9 Ice alarm function*
This parameter may have the following values:

- **deactivated (default value)**
- active

Enabling the ice alarm displays the communication object ***Actuator ice alarm*** and the parameters **"Ice alarm activation value"**, which is used to define the value of the communication object associated with the ice alarm, the parameter **"Behaviour on ice alarm activation"**, which is used to define the behaviour of the device when the ice alarm is active and the parameter **"Ice sensor monitoring time [min] (0=monitoring disabled)"**, which is used to define the ice sensor monitoring time.

➢ *14.6.10 Ice alarm activation value*
This parameter may have the following values:

- value "0"
- **value "1" (default value)**

➢ *14.6.11 Behaviour at ice alarm activation*
This parameter may have the following values:

- **no effect (default value)**
- up movement
- down movement
- percentage position
- stop

When the parameter takes on the value **percentage position**, this displays the parameters **"Percentage position with active ice alarm"**, which is used to define the percentage position of the load and, if the operation is venetian blind, the parameter **"Louvres percentage position with active ice alarm",** which is used to define the percentage position of the venetian blind louvres. The values that can be set for both parameters are:

- from **0% (default value)** to 100% with step of 5%

➢ *14.6.12 Ice sensor monitoring time [min] (0=monitoring disabled)*
This parameter may have the following values:

- from **0 (default value)** to 60

By selecting the value **0**, the object enabled for the ice alarm will not be monitored. The meaning of the monitoring time is similar to the one for the wind alarm:

➢ *14.6.13 Behaviour on ice alarm deactivation*
This parameter is used to set the behaviour of the device when the ice alarm is deactivated; The values that can be set are:

- **no effect (default value)**
- up movement
- down movement
- back to previous position
- follows last command received
- percentage position
- stop

The option **set percentage position** displays the parameters **"Percentage position at ice alarm deactivation"**, which makes it possible to define the percentage position of the load and, if the operation is venetian blind, the parameter **"Louvres percentage position at ice alarm deactivation"**, which makes it possible to define the percentage position of the venetian blind louvres. The values that can be set for both parameters are:

- from **0% (default value)** to 100% with step of 5%

➢ *14.6.14 Meteo alarms priorities*
This parameter is used to define the intrinsic priority of the meteo alarm events; The values that can be set are:

- **H= wind M=rain L=ice (default value)**
- H= wind M=ice L=rain
- H= ice M=rain L=wind
- H= ice M=wind L=rain
- H= rain M=wind L=ice
- H= rain M= ice L= wind

The term H (High) indicates the meteo alarm with the highest priority, M (Medium) the meteo alarm with medium priority and L (Low) low priority.

In general, all defined behaviours on alarm deactivation are effectively implemented if and only if an alarm with a lower priority is not active. Otherwise, the actuator will return the shutter/venetian blind to the position corresponding to the highest priority alarm still active.

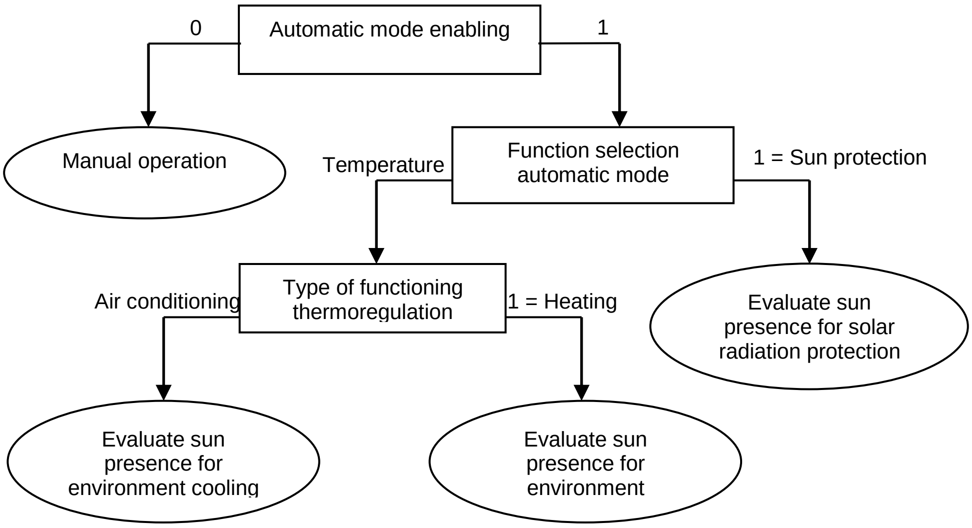
On bus voltage recovery, it may be necessary to reset an alarm status.

14.7 “Automatic mode” function

The device is able to perform autonomous activations to take advantage of sunlight to heat the environment, for example; it is possible to define the position to which the load must be moved if the user requests protection against the direct light of the sun's rays, requests making use of the sun to heat the environment or requests suitable positioning of the load to protect against the sunlight to keep the environment cool.
Automatic operation is enabled via the communication object **“Actuator automatic mode enabling"**; in this operating mode, the objects that permit 'manual' control (including automatic calibration) of the load are ignored whereas the higher priority commands (meteo alarms, block and forcing) are executed.

In automatic operation, the device evaluates how to command the load both based on the value of the object “**Actuator automatic mode function selection**”, which makes it possible to activate the sunlight protection function (value 1) or the thermoregulation function (value 0), or based on the presence of the sun communicated by the communication object “**Actuator sun presence**”.

The block diagram shown below demonstrates this type of operation:



The activation of automatic mode enables the **Automatic mode** menu, which includes the parameters shown in fig. 14.7.

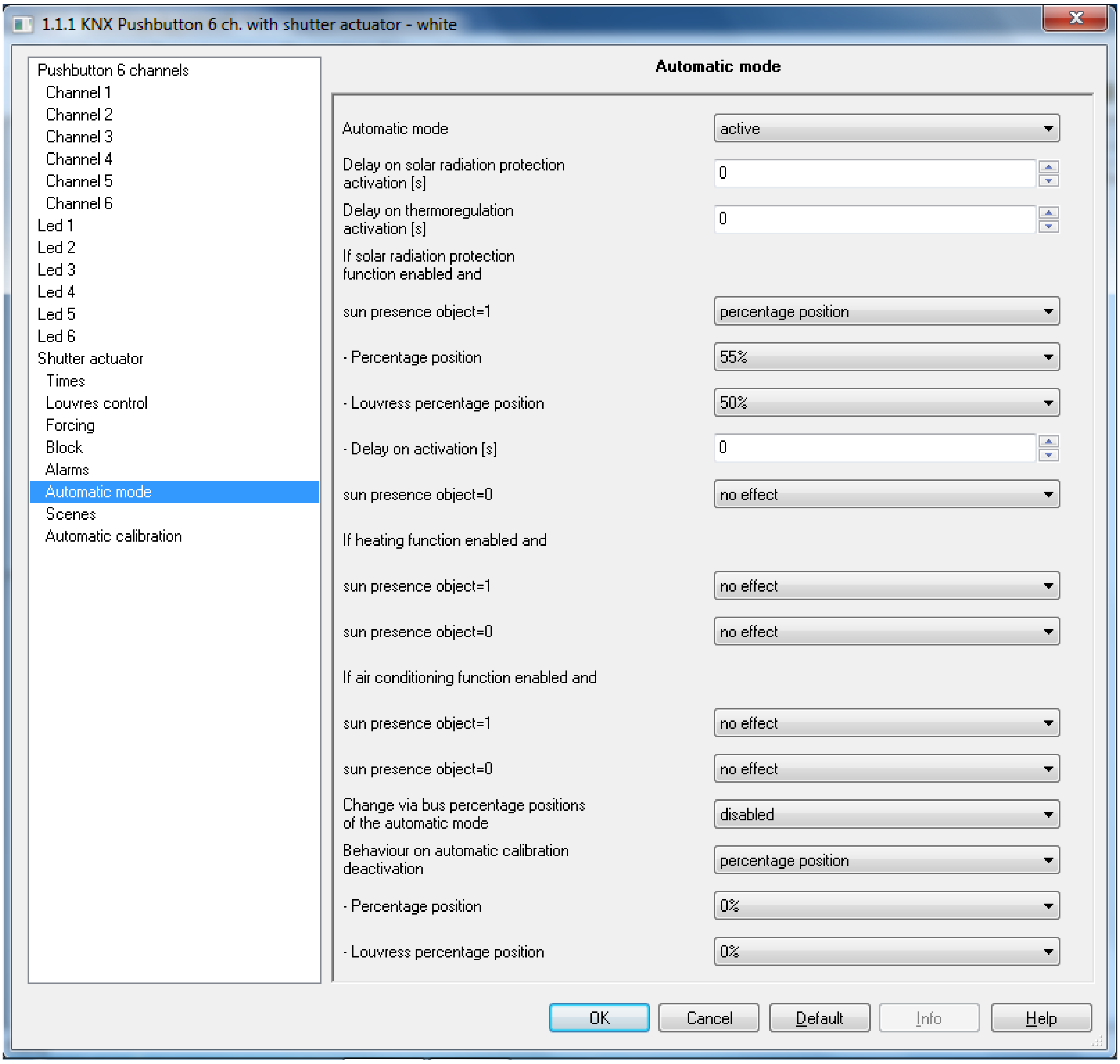


Fig. 14.7

➢ *14.7.1 Delay on solar radiation protection activation [s]*

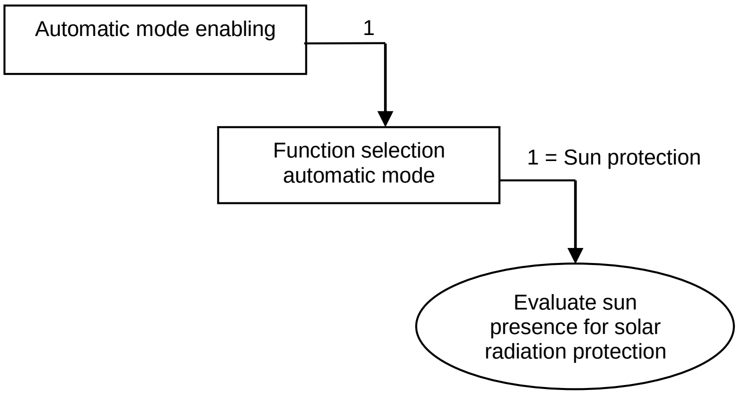
The activation of the sun protection function via “**Actuator automatic mode function selection**” can be delayed by a time that ranges from 0 to 3600 seconds (0 is the default).

➢ *14.7.2 Delay on thermoregulation activation [s]*

The activation of the thermoregulation function via “**Actuator automatic mode function selection**” can be delayed by a time that ranges from 0 to 3600 seconds (0 is the default).

➢ *14.7.3 If solar radiation protection function enabled and*

In this section, it is possible to determine the load's behaviour based on the presence of the sun, communicated by the communication object “**Actuator sun presence**” and by the fact that the sun protection was activated via the communication object “**Actuator automatic mode function selection**”
The block diagram shown below demonstrates the condition for this type of operation:



For sun protection, it is possible to define the behaviour of the device in the presence of and without sun.
Using the parameter **“sun presence object=1”** it is possible to define device behaviour when the set operation is "solar radiation protection" and the value of the object “Actuator sun presence” is “1” (sun presence); the values that can be set are:

- **no effect**
  this is the default and does not carry out any action
- **percentage position**
  this displays the parameters **“Percentage position”** which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **“Louvres percentage position”** which makes it possible to define the percentage position of the venetian blind louvres. For percentage values can be set from 0 to 100 and 50% is the default.
  By using the parameter **“Delay on activation [s] ”,** if it possible to define the delay time between receiving the sun presence object = 1 and the effective movement command. The parameter value can range between 0, which is the default, and 3600.

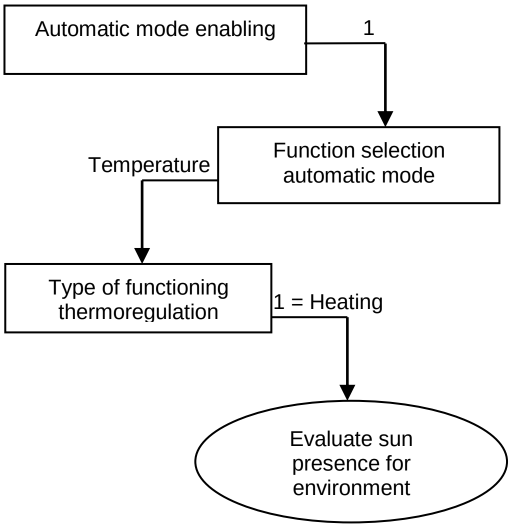
Using the parameter **“sun presence object=0”** it is possible to define device behaviour when the set operation is "solar radiation protection" and the value of the object “Actuator sun presence” is “0” (no sun). The values that can be set are:

- **no effect**
  this is the default and does not carry out any action
- **percentage position**
  this displays the parameters **“Percentage position”** which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **“Louvres percentage position”** which makes it possible to define the percentage position of the venetian blind louvres. For percentage values can be set from 0 to 100 and 50% is the default.
  By using the parameter **“Delay on activation [s] ”,** if it possible to define the delay time between receiving the sun presence object = 0 and the effective movement command. The parameter value can range between 0, which is the default, and 3600.

➢ *14.7.4 If heating function enabled and*

In this section, it is possible to determine the load's behaviour based on the presence of the sun, communicated by the communication object “**Actuator sun presence**” and by the fact that the heating thermoregulation was activated via the communication object “**Actuator automatic mode function selection**”

The block diagram shown below demonstrates the condition for this type of operation:



For the heating function, it is possible to define the behaviour of the device in the presence of and without sun.
Using the parameter **"sun presence object=1"** it is possible to define device behaviour when the set operation is "Heating" and the value of the object "Actuator sun presence" is "1" (sun presence); the values that can be set are:

- **no effect**
  this is the default and does not carry out any action
- **percentage position**
  this displays the parameters **"Percentage position"** which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **"Louvres percentage position"** which makes it possible to define the percentage position of the venetian blind louvres. For percentage values can be set from 0 to 100 and 50% is the default.
  By using the parameter **"Delay on activation [s] ",** if it possible to define the delay time between receiving the sun presence object = 1 and the effective movement command. The parameter value can range between 0, which is the default, and 3600.

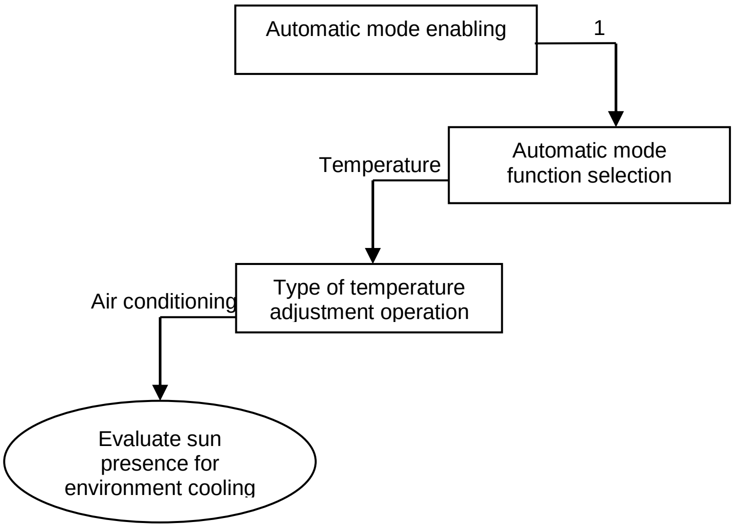
Using the parameter **"sun presence object=0"** it is possible to define device behaviour when the set operation is "Heating" and the value of the object "Actuator sun presence" is "0" (no sun); The values that can be set are:

- **no effect**
  this is the default and does not carry out any action
- **percentage position**
  this displays the parameters **"Percentage position"** which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **"Louvres percentage position"** which makes it possible to define the percentage position of the venetian blind louvres. For percentage values can be set from 0 to 100 and 50% is the default.
  By using the parameter **"Delay on activation [s] ",** if it possible to define the delay time between receiving the sun presence object = 0 and the effective movement command. The parameter value can range between 0, which is the default, and 3600.

➢ *14.7.5 If air conditioning function enabled and*

In this section, it is possible to determine the load's behaviour based on the presence of the sun, communicated by the communication object "**Actuator sun presence**" and by the fact that the air conditioning thermoregulation was activated via the communication object "**Actuator automatic mode function selection**"

The block diagram shown below demonstrates the condition for this type of operation:



For the air conditioning function, it is possible to define the behaviour of the device in the presence of and without sun.
Using the parameter **“sun presence object=1”** it is possible to define device behaviour when the set operation is "Air conditioning" and the value of the object “Actuator sun presence” is “1” (sun presence); the values that can be set are:

- **no effect**
  this is the default and does not carry out any action
- **percentage position**
  this displays the parameters **“Percentage position”** which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **“Louvres percentage position”** which makes it possible to define the percentage position of the venetian blind louvres. For percentage values can be set from 0 to 100 and 50% is the default.
  By using the parameter **“Delay on activation [s] ”,** if it possible to define the delay time between receiving the sun presence object = 1 and the effective movement command. The parameter value can range between 0, which is the default, and 3600.

Using the parameter **“sun presence object=0”** it is possible to define device behaviour when the set operation is "Air conditioning" and the value of the object “Ch.x - Sun presence” is “0” (no sun). The values that can be set are:

- **no effect**
  this is the default and does not carry out any action
- **percentage position**
  this displays the parameters **“Percentage position”** which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **“Louvres percentage position”** which makes it possible to define the percentage position of the venetian blind louvres. For percentage values can be set from 0 to 100 and 50% is the default.
  By using the parameter **“Delay on activation [s] ”,** if it possible to define the delay time between receiving the sun presence object = 0 and the effective movement command. The parameter value can range between 0, which is the default, and 3600.

➢ *14.7.6 Change via bus percentage positions of the automatic mode*

It is possible to change the position of the shutters or the venetian blind louvres relative to a certain automatic operation set via the ETS parameters described above, through dedicated communication objects, enabling the parameter **“Modification of the percentage positions relative to automatic mode from the bus”** disabled by default, which displays the communication objects “**Actuator automatic mode position command**” and, if the operating mode is "venetian blind", the object “**Actuator automatic mode louvres adjustment command**”.
The possible cases are indicated below:

- If the currently active mode is ***sun protection*** (paragraph 11.7.3), the above mentioned objects are used to modify both the position as well as the adjustment of the louvres and also the percentage values of *Percentage position* and louvres *Percentage position* are updated (the latter if venetian blind mode is active), defined by the database parameters previously set via ETS.

- If the currently active mode is ***heating*** (paragraph 11.7.4), the above mentioned objects are used to modify both the position as well as the adjustment of the louvres and also the percentage values of *Percentage position* and louvres *Percentage position* are updated (the latter if venetian blind mode is active), defined by the database parameters previously set via ETS.

- If the currently active mode is ***air conditioning*** (paragraph 11.7.5), the above mentioned objects are used to modify both the position as well as the adjustment of the louvres and also the percentage values of *Percentage position* and louvres *Percentage position* are updated (the latter if venetian blind mode is active), defined by the database parameters previously set via ETS.

➢ *14.7.7 Behaviour on automatic calibration deactivation*

This parameter is used to set the device behaviour when automatic mode is deactivated (receipt of value 0 on the Actuator automatic mode enabling communication object). The values that can be set are:

- **no effect**
  this is the default value that, upon deactivation of the automatic mode, does not carry out any action.
- **up movement**
  upon deactivation of the automatic mode it commands an up movement.
- **down movement**
  upon deactivation of the automatic mode it commands a down movement.
- **back to previous position**
  upon deactivation of the automatic mode, the load for the relative channel returns to the position prior to deactivation
- **follows last command received**
  upon deactivation of the automatic mode, the last received command is executed.
- **percentage position**
  in this case, when the automatic mode is deactivated the parameters **"Percentage position on deactivation "** are displayed, which makes it possible to define the percentage position of the shutter and, if the operation is venetian blind, the parameter **"Louvres percentage position upon deactivation of automatic mode"** which makes it possible to define the percentage position of the Venetian blind louvres. The percentage values can be set for both parameters range from 0, which is the default, to 100.
- **stop**
  a stop is commanded when automatic mode is deactivated.

14.8 “Scenes” function

The scene function is used to replicate a certain position that was preset or previously stored after receiving the communication object ***Actuator scene***. The device is able to memorize and execute 8 scenes.
The basic structure of the menu is as follows (fig. 14.8):

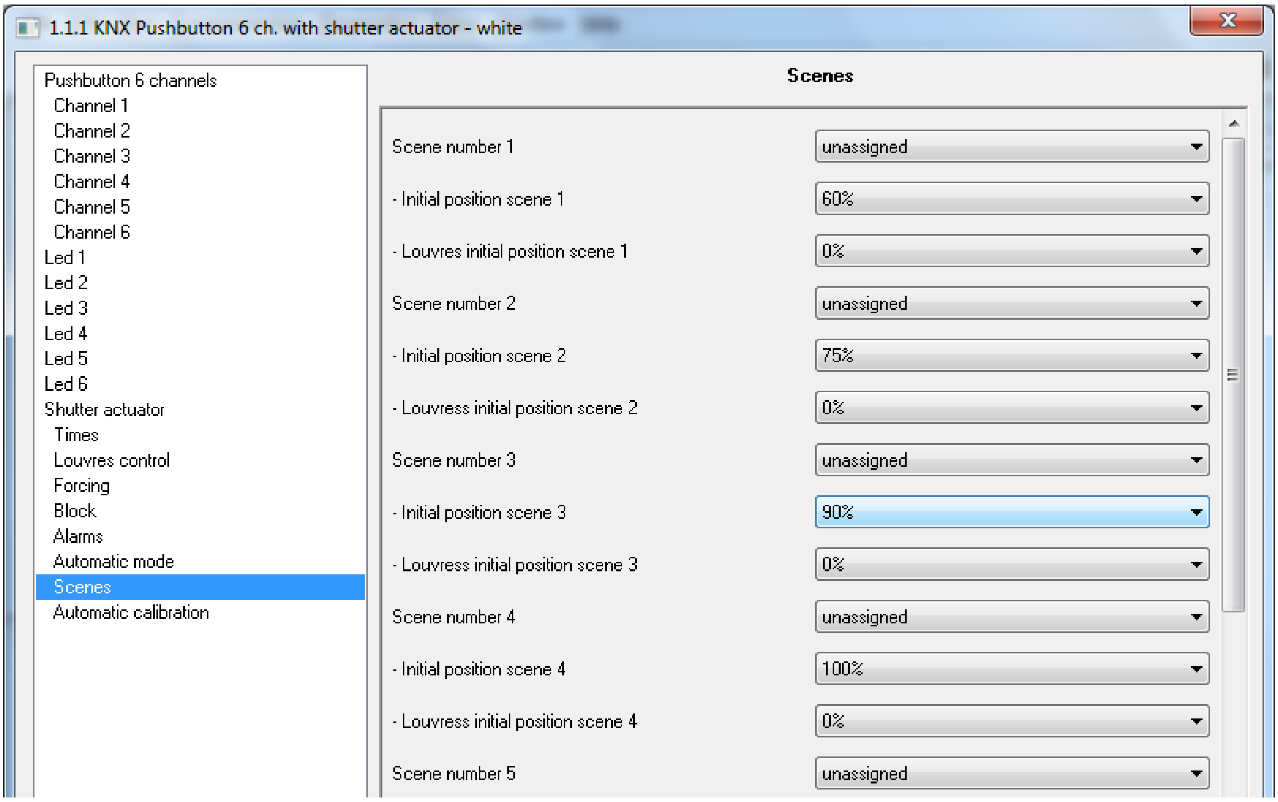


Fig. 14.8

➢ *14.8.1 Scene number i*

The parameters **“Scene number 1”**, **“Scene number 2”**, **“Scene number 3”**, **“Scene number 4”**, **“Scene number 5”**, **“Scene number 6”**, **“Scene number 7”** and **“Scene number 8”** are used to set the numerical value used to identify and therefore execute/store the n-th scene; the possible values are:

- **not assigned (default value)**
- 0, 1, ..., 63

➢ *14.8.2 Initial position scene x*

The parameters **“Initial position scene 1”**, **“Initial position scene 2”**, **“Initial position scene 3”**, **“Initial position scene 4”**, **“Initial position scene 5”**, **“Initial position scene 6”**, **“Initial position scene 7”** and **“Initial position scene 8”** are used to define the percentage position of the load that the actuator must replicate after receiving the communication object for executing the n-th scene; the possible values are:

- **0%** (default value for scene 1), 5%, 10%, **15 %** (default value for scene 2), 20 %, 25%, **30%**(default value for scene 3), **45%**(default value for scene 4), **60%**(default value for scene 5), ..**75%**(default value for scene 6).. **90%**(default value for scene 7) **100%** (default value for scene 8)

➢ 14.8.3 Louvres initial position scene x

In the case of "venetian blind" operating mode, the parameters **"Louvres initial position scene 1"**, **"Llouvres initial position scene 2"**, **"Louvres initial position scene 3"**, **"Louvres initial position scene 4"**, **"Louvres initial position scene 5"**, **"Louvres initial position scene 6"**, **"Louvres initial position scene 7"** and **"Louvres initial position scene 8"** are used to define the position of the louvres of the connected load that the actuator must replicate after receiving the communication object for executing the n-th scene; the possible values are:

- from **0% (default value)** to 100% with step of 5%

➢ 14.8.4 Scene storing enabling

This parameter is used to enable/disable the scene storing possibility via the communication object ***Actuator scene***; The parameter may assume the following values:

- disabled
- **enabled (default value)**

Selecting **enable** displays the communication object ***Actuator scenes storing enabling*** which makes it possible to enable/disable via the bus the possibility to store scenes via the communication object ***Actuator scene***.

Scene storing includes memorizing the percentage position of the louvres (for venetian blinds). Therefore, executing a scene must also include reproducing the previously memorized louvres position.

14.9 "Automatic calibration" function

The actuator calculates the percentage position of the connected load based on the travel range time set by the user and in function of the movement commands it executes. Even if the parameters that determine operating times are set accurately, there may be small variations between the position calculated by the actuator and the one that actually occurs, due to climatic events, for example, or mechanical load tolerances. The device can zero out these variations by executing movements towards the reference positions, that is towards those identified by the two limit switches. The automatic calibration function enables the device to perform automatic movements at end of travel to guarantee the correct movement of the load even after various manoeuvres; the basic structure of the menu is as follows (fig. 14.9):

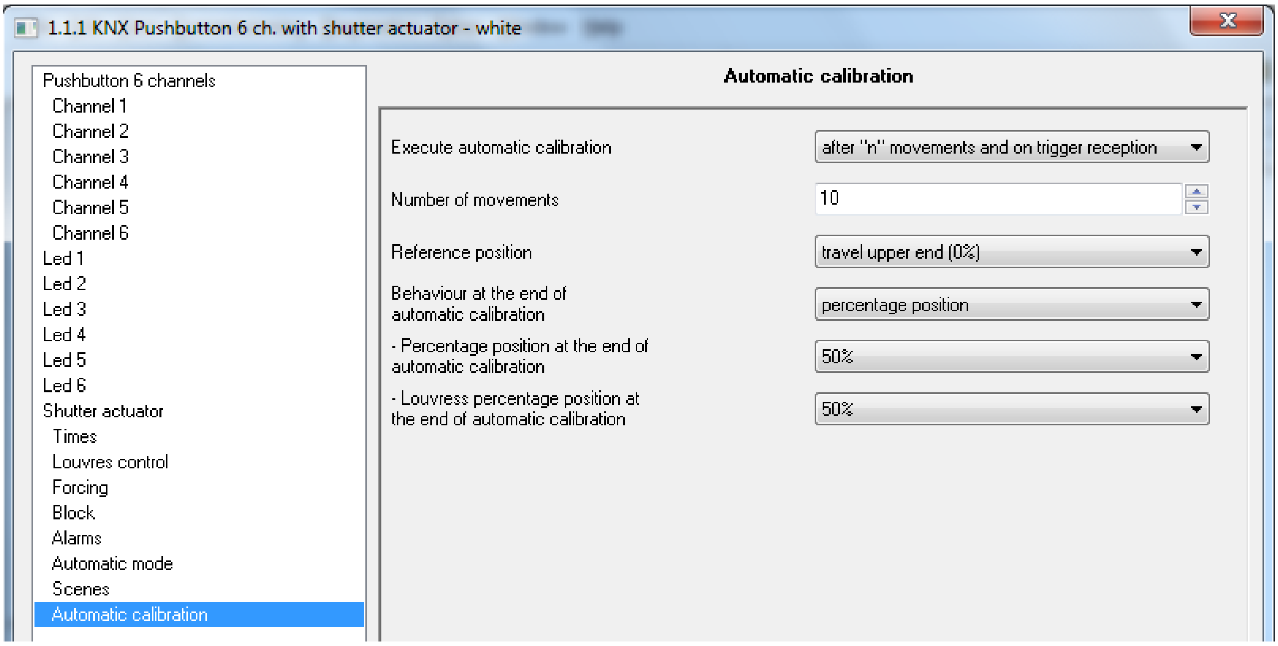


Fig. 14.9

➢ 14.9.1 Execute automatic calibration

Once the device executes a movement to the limit switch, the movement times can be correctly recalculated and any misalignment errors are cancelled. The device performs this operation each time the limit switch is reached, but automatic calibration can be activated via the parameter **“Execute automatic calibration”,** which can assume the following values:

- **never, on downloading or bus voltage recovery (default value)**
- after n movements
- at trigger reception
- after n movements and on trigger reception

The value **never, on downloading or bus voltage recovery** is selected when an automatic calibration should not be executed except on downloading or on bus voltage recovery, according to what is defined in the parameters **“Actuator behaviour”** and **“Actuator behaviour at bus voltage recovery”** in the **Shutter actuator** menu.
Selecting the value **after n movements** or **after n movements and on trigger reception**, via the parameter **“Number of movements”** it is possible to set after how many movements automatic calibration is performed; each time the device reaches the limit switch, the movement counter is reset. the values that can be set are:

- from **1** to **30**, **10 (default value)**

Selecting the value **on trigger reception** or **after n movements and on trigger reception** displays the communication object ***Actuator automatic calibration trigger***. After reception of the communication object with value “0” or “1” the device executes automatic calibration.

➢ *14.9.2 Reference position*

This parameter is used to set which limit switch the device must use as a reference for automatic calibration; The available values are:

- **upper limit (0%) (default value)**

➢ *14.9.3 Behaviour at the end of automatic calibration*

When the calibration operation is terminated, it is possible to define the device behaviour via this parameter, which can assume the following values

- **stays in the reference position (default value)**
- back to previous position
- percentage position

Selecting the option **percentage position** displays the parameter **“Percentage position end of automatic calibration”** which is used to define the percentage position of the shutter and, if the operation is venetian blinds, the parameter **“Louvres percentage position at the end of automatic calibration”** which makes it possible to define the percentage position of the venetian blind louvres at the end of automatic calibration. The values that can be set for both parameters are:

- from **0%** to **100%** with a step of 5%, **50% (default value)**

15 Communication objects

The communication objects are listed in the following table (this example shows only general objects and those relative to channel 1 and 1/2 of the push-button panel and those of the motor command actuator) (Fig. 15.1):

Number	Name	Object Function	Leng...	C	R	W	T	U	Data Type	Priority
0	Ch.1 - Block	Switching On/Off	1 bit	C	-	W	-	-		Low
0	Ch.1/2 - Block	Switching On/Off	1 bit	C	-	W	-	-		Low
1	Ch.1 - A Sequence	On/Off	1 bit	C	R	-	T	-		Low
1	Ch.1 - Shutter movement	Up/Down	1 bit	C	R	-	T	-		Low
1	Ch.1 - Switch	On/Off	1 bit	C	R	-	T	-		Low
1	Ch.1/2 - Switch	On/Off	1 bit	C	R	-	T	-		Low
1	Ch.1/2 - Shutter movement	Up/Down	1 bit	C	R	-	T	-		Low
1	Ch.1 - Scene	Execute/Store	1 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 14 bytes value	Characters ISO 8859-1	14 B...	C	R	-	T	-		Low
1	Ch.1 - A object 4 bytes value	Signed value	4 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 4 bytes value	Unsigned value	4 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 2 bytes value	Signed value	2 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 2 bytes value	Unsigned value	2 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 1 byte value	HVAC mode	1 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 1 byte value	% Value	1 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 1 byte value	Signed value	1 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 1 byte value	Unsigned value	1 Byte	C	R	-	T	-		Low
1	Ch.1 - A object 2 bits value	On/Off forced positioning	2 bit	C	R	-	T	-		Low
1	Ch.1 - A object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
1	Ch.1 - A Sequence	On/Off	1 bit	C	R	-	T	-		Low
2	Ch.1 - Shutter stop/Louvres control	Stop/step	1 bit	C	R	-	T	-		Low
2	Ch.1 - Brightness dimming	Increase/Decrease	4 bit	C	R	-	T	-		Low
2	Ch.1/2 - Shutter stop/Louvres control	Stop/step	1 bit	C	R	-	T	-		Low
2	Ch.1/2 - Brightness dimming	Increase/Decrease	4 bit	C	R	-	T	-		Low
2	Ch.1 - Scene storing trigger	Store	1 bit	C	-	W	-	-		Low
2	Ch.1 - B object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
2	Ch.1 - B sequence	On/Off	1 bit	C	R	-	T	-		Low
3	Ch.1 - C object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
3	Ch.1 - C sequence	On/Off	1 bit	C	R	-	T	-		Low
4	Ch.1 - D object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
4	Ch.1 - D sequence	On/Off	1 bit	C	R	-	T	-		Low
5	Ch.1 - E object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
5	Ch.1 - E sequence	On/Off	1 bit	C	R	-	T	-		Low
6	Ch.1 - F object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
6	Ch.1 - F sequence	On/Off	1 bit	C	R	-	T	-		Low
7	Ch.1 - G object 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
7	Ch.1 - G sequence	On/Off	1 bit	C	R	-	T	-		Low
8	Ch.1 - Hobject 1 bit value	1/0 value	1 bit	C	R	-	T	-		Low
8	Ch.1 - H sequence	On/Off	1 bit	C	R	-	T	-		Low
9	Ch.1 - Movement feedback	Increase/Decrease	1 bit	C	-	W	-	-		Low
9	Ch.1 - Dimmer status feedback	On/Off status	1 bit	C	-	W	T	U		Low
9	Ch.1 - A object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
10	Ch.1 - B object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
11	Ch.1 - C object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
12	Ch.1 - D object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
13	Ch.1 - E object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
14	Ch.1 - F object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
15	Ch.1 - G object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
16	Ch.1 - H object status feedback	On/Off status	1 bit	C	-	W	T	U		Low
102	Led 1 - Effect 1	Switching On/Off	1 bit	C	-	W	-	U		Low
102	Led 1 - Effect 1	Switching On/Off	1 bit	C	-	W	-	U		Low
103	Led 1 - Effect 2	Switching On/Off	1 bit	C	-	W	-	U		Low
104	Led 1 - Effect 3	Switching On/Off	1 bit	C	-	W	-	U		Low
105	Led 1 - Effect 4	Switching On/Off	1 bit	C	-	W	-	U		Low
106	Led 1 - Effect 5	Switching On/Off	1 bit	C	-	W	-	U		Low
107	Led 1 - Color choice	1=green/0=amber	1 bit	C	-	W	-	-		Low

Number	Name	Object Function	Leng...	C	R	W	T	U	Data Type	Priority
138	Actuator movement	Up/Down	1 bit	C	-	W	-	-		Low
139	Actuator shutter stop/louvres control	Stop/step	1 bit	C	-	W	-	-		Low
139	Actuator stop	Stop	1 bit	C	-	W	-	-		Low
140	Actuator position command	% Value	1 Byte	C	-	W	-	-		Low
141	Actuator louvres control command	% Value	1 Byte	C	-	W	-	-		Low
142	Actuator block	Switching On/Off	1 bit	C	-	W	-	-		Low
143	Actuator priority command	Forced positioning Up/Down	2 bit	C	-	W	-	-		Low
144	Actuator scene	Execute/Store	1 Byte	C	-	W	-	-		Low
145	Actuator wind alarm 1	Alarm input	1 bit	C	-	W	-	-		Low
146	Actuator wind alarm 2	Alarm input	1 bit	C	-	W	-	-		Low
147	Actuator wind alarm 3	Alarm input	1 bit	C	-	W	-	-		Low
148	Actuator rain alarm	Alarm input	1 bit	C	-	W	-	-		Low
149	Actuator ice alarm	Alarm input	1 bit	C	-	W	-	-		Low
150	Actuator automatic mode enabling	Enable/Disable	1 bit	C	-	W	-	-		Low
151	Actuator automatic mode selection	Sun protection/Thermoreg.	1 bit	C	-	W	-	-		Low
152	Actuator thermoregulation functioning type	Heating/Air conditioning	1 bit	C	-	W	-	-		Low
153	Actuator sun presence	True/False	1 bit	C	-	W	-	-		Low
154	Actuator automatic mode position comm...	% Value	1 Byte	C	-	W	-	-		Low
155	Actuator auto mode louvres control com...	% Value	1 Byte	C	-	W	-	-		Low
156	Actuator automatic calibration trigger	Calibration request	1 bit	C	-	W	-	-		Low
157	Actuator movement signalling	Increase/Decrease	1 bit	C	R	-	-	-		Low
158	Actuator position feedback	% Value	1 Byte	C	R	-	T	-		Low
159	Actuator louvres position feedback	% Value	1 Byte	C	R	-	-	-		Low
160	Actuator travel limits enabling	Enable/Disable	1 bit	C	-	W	-	-		Low
161	Actuator lower travel limit	% Value	1 Byte	C	-	W	-	-		Low
162	Actuator upper travel limit	% Value	1 Byte	C	-	W	-	-		Low
163	Actuator scenes storing enabling	Enable/Disable	1 bit	C	-	W	-	-		Low

Fig. 15.1

15.1 Communication object table

The following tables summarize all the communication objects with their ID number, the name and function displayed in ETS, plus a brief description of the function performed and the type of Datapoint used.

➢ 15.1.1 Communication objects with input functions

The following table contains all the objects with an input function:

Communication object no.						Object name	Object function	Description	Datapoint type
Ch.1	Ch.2	Ch.3	Ch.4	Ch.5	Ch.6				
0	17	34	51	68	85	Ch.x - Block	Switching On /Off	Used to activate/deactivate the block function	1.003 DPT_Enable
0	0	34	34	68	68	Ch.x/y - Block	Switching On /Off	Used to activate/deactivate the block function	1.003 DPT_Enable
2	19	36	53	70	87	Ch.x - Scene storing trigger	Store	Receives the request (trigger) to send a scene storing message	1.017 DPT_Trigger

9	26	43	60	77	94	Ch.x - Dimmer status feedback	On/Off status	Receives the dimmer status feedback	1.001 DPT_Switch
9	26	43	60	77	94	Ch.x - A status feedback	On/Off status	Receives the actuator status feedback for A object cyclic switching	1.001 DPT_Switch
9	26	43	60	77	94	Ch.x - Movement feedback	Increase/Decrease	Receives the feedback about the current movement direction of the motor command actuator	1.008 DPT_UpDown
10	27	44	61	78	95	Ch.x - B status feedback	On/Off status	Receives the actuator status feedback for B object cyclic switching	1.001 DPT_Switch
11	28	45	62	79	96	Ch.x - C status feedback	On/Off status	Receives the actuator status feedback for C object cyclic switching	1.001 DPT_Switch
12	29	46	63	80	97	Ch.x - D status feedback	On/Off status	Receives the actuator status feedback for D object cyclic switching	1.001 DPT_Switch
13	30	47	64	81	98	Ch.x - E status feedback	On/Off status	Receives the actuator status feedback for E object cyclic switching	1.001 DPT_Switch

14	31	48	65	82	99	Ch.x - F status feedback	On/Off status	Receives the actuator status feedback for F object cyclic switching	1.001 DPT_Switch
15	32	49	66	83	100	Ch.x - G status feedback	On/Off status	Receives the actuator status feedback for G object cyclic switching	1.001 DPT_Switch
16	33	50	67	84	101	Ch.x - H status feedback	On/Off status	Receives the actuator status feedback for H object cyclic switching	1.001 DPT_Switch
102						Led 1 - Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
103						Led 1 - Effect 2	Switching On /Off	Switching On /Off light effect 2	1.001 DPT_Switch
104						Led 1 - Effect 3	Switching On /Off	Switching On /Off light effect 3	1.001 DPT_Switch
105						Led 1 - Effect 4	Switching On /Off	Switching On /Off light effect 4	1.001 DPT_Switch
106						Led 1 - Effect 5	Switching On /Off	Switching On /Off light effect 5	1.001 DPT_Switch
108						Led 2 - Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
109						Led 2 - Effect 2	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
110						Led 2 - Effect 3	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
111						Led 2 - Effect 4	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
112						Led 2 - Effect 5	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
114						Led 3 - Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch

115	Led 3 - Effect 2	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
116	Led 3 - Effect 3	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
117	Led 3 - Effect 4	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
118	Led 3 - Effect 5	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
120	Led 4 - Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
121	Led 4 - Effect 2	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
122	Led 4 - Effect 3	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
123	Led 4 - Effect 4	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
124	Led 4 - Effect 5	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
126	Led 5 - Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
127	Led 5 - Effect 2	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
128	Led 5 - Effect 3	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
129	Led 5 - Effect 4	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
130	Led 5 - Effect 5	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
132	Led 6 - Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
133	Led 6 - Effect 2	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
134	Led 6 - Effect 3	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
135	Led 6 - Effect 4	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch

136	Led 6 - Effect 5	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
107	Led 1 - Color choice	1=green/0=amber	Selects the light signalling color	1.001 DPT_Switch
113	Led 2 - Color choice	1=green/0=amber	Selects the light signalling color	1.001 DPT_Switch
119	Led 3 - Color choice	1=green/0=amber	Selects the light signalling color	1.001 DPT_Switch
125	Led 4 - Color choice	1=green/0=amber	Selects the light signalling color	1.001 DPT_Switch
131	Led 5 - Color choice	1=green/0=amber	Selects the light signalling color	1.001 DPT_Switch
137	Led 6 - Color choice	1=green/0=amber	Selects the light signalling color	1.001 DPT_Switch
138	Actuator movement	Up/down	Up/down movement	1.008 DPT_UpDown
139	Actuator stop	Stop	Movement stop	1.017 DPT_Trigger
139	Actuator louvres adjustment/stop	Stop/Step	Movement stop/louvres control	1.007 DPT_Step
140	Actuator position command	% Value	Sets the percentage position	5.001 DPT_Scaling
141	Actuator louvres adjustment command	% Value	Sets the louvres percentage position	5.001 DPT_Scaling
142	Actuator block	Switching On /Off	Blocks the position of the actuator in a parameterizable state	1.003 DPT_Enable
143	Actuator priority command	Up/down forcing	Forces the load status up or down	2.008 Direction1DPT_Switch_Control
144	Actuator scene	Execute/Store	Makes it possible to store/execute scenes	18.001 DPT_SceneControl
145	Wind alarm 1 actuator	Alarm input	Wind alarm input	1.005 DPT_Alarm

146	Wind alarm 2 actuator	Alarm input	Wind alarm input	1.005 DPT_Alarm
147	Wind alarm 3 actuator	Alarm input	Wind alarm input	1.005 DPT_Alarm
148	Actuator rain alarm	Alarm input	Rain alarm input	1.005 DPT_Alarm
149	Actuator ice alarm	Alarm input	Ice alarm input	1.005 DPT_Alarm
150	Actuator automatic mode enabling	Enable/disable	Actuator automatic mode enabling	1.003 DPT_Enable
151	Actuator automatic mode function selection	Sun protection/Thermoreg.	Sun protection or thermoregulation enabling	1.003 DPT_Enable
152	Type of actuator thermoregulation operation	Heating/air conditioning	Automatic thermoregulation functioning type	1.100 DPT_Heat/Cool
153	Actuator sun presence	True/false	Sun presence/absence signalling reception	1.002 DPT_Bool
154	Actuator automatic mode position command	% Value	Modifies and memorises % position with automatic operation active	5.001 DPT_Scaling
155	Actuator automatic mode louvres adjustment command	% Value	Modifies and memorises louvres % position with automatic operation active	5.001 DPT_Scaling
156	Actuator automatic calibration trigger	% Value	Activates device automatic calibration	1.017 DPT_Trigger

➢ *15.1.2 Communication objects with output functions*

The following table contains all the objects with an output function:

Communication object no.						Object name	Object function	Description	Datapoint type
Ch.1	Ch.2	Ch.3	Ch.4	Ch.5	Ch.6				
1	18	35	52	69	86	Ch.x - Switching	On/Off	Sends dimmer on/off commands	1.001 DPT_Switch
1	1	35	35	69	69	Ch.x/y - Switching	On/Off	Sends dimmer on/off commands	1.001 DPT_Switch
1	18	35	52	69	86	Ch.x - Shutter movement	Up/down	Sends shutter up/down movement commands	1.008 DPT_UpDown
1	1	35	35	69	69	Ch.x/y - Shutter movement	Up/down	Sends shutter up/down movement commands	1.008 DPT_UpDown
1	18	35	52	69	86	Ch.x - Scene	Execute/Store	Sends scene memorizing/execution commands	18.001 DPT_SceneControl
1	18	35	52	69	86	Ch.x - A Sequence	On/Off	Sends On/Off commands associated with A object of the sequence	1.001 DPT_Switch
1	18	35	52	69	86	Ch.x - A object 1 bit value	1/0 value	Sends values 1/0 associated with A object	1.002 DPT_Bool
1	18	35	52	69	86	Ch.x - A object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with A object	2.001 DPT_Switch_Control
1	18	35	52	69	86	Ch.x - A object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with A object	5.010 DPT_Value_1_Ucount
1	18	35	52	69	86	Ch.x - A object 1 byte value	Signed value	Sends signed values (-128..127) associated with A object	6.010 DPT_Value_1_Count
1	18	35	52	69	86	Ch.x - A object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with A object	5.001 DPT_Scaling
1	18	35	52	69	86	Ch.x - A object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
1	18	35	52	69	86	Ch.x - A object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with A object	7.001 DPT_Value_2_Ucount

1	18	35	52	69	86	Ch.x - A object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with A object	8.001 DPT_Value_2_Count
1	18	35	52	69	86	Ch.x - A object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with A object	12.001 DPT_Value_4_Ucount
1	18	35	52	69	86	Ch.x - A object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with A object	13.001 DPT_Value_4_Count
1	18	35	52	69	86	Ch.x - A object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
2	19	36	53	70	87	Ch.x - Brightness dimming	Increase/decrease	Sends brightness control commands	3.007 DPT_Control_Dimming
2	2	36	36	70	70	Ch.x/y - Brightness dimming	Increase/decrease	Sends brightness control commands	3.007 DPT_Control_Dimming
2	19	36	53	70	87	Ch.x - Shutter stop/Louvres control	Stop/Step	Sends stop movement/louvres control commands	1.007 DPT_Step
2	2	36	36	70	70	Ch.x/y - Shutter stop/Louvres control	Stop/Step	Sends stop movement/louvres control commands	1.007 DPT_Step
2	19	36	53	70	87	Ch.x - B sequence	On/Off	Sends On/Off commands associated with B object of the sequence	1.001 DPT_Switch
2	19	36	53	70	87	Ch.x - B object 1 bit value	1/0 value	Sends values 1/0 associated with B object	1.002 DPT_Bool
2	19	36	53	70	87	Ch.x - B object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with B object	2.001 DPT_Switch_Control
2	19	36	53	70	87	Ch.x - B object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with B object	5.010 DPT_Value_1_Ucount
2	19	36	53	70	87	Ch.x - B object 1 byte value	Signed value	Sends signed values (-128..127) associated with B object	6.010 DPT_Value_1_Count
2	19	36	53	70	87	Ch.x - B object 1 byte	% Value	Sends the percentage values	5.001 DPT_Scaling

						value		(0%..100%) associated with B object	
2	19	36	53	70	87	Ch.x - B object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
2	19	36	53	70	87	Ch.x - B object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with B object	7.001 DPT_Value_2_Ucount
2	19	36	53	70	87	Ch.x - B object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with B object	8.001 DPT_Value_2_Count
2	19	36	53	70	87	Ch.x - B object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with B object	12.001 DPT_Value_4_Ucount
2	19	36	53	70	87	Ch.x - B object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with B object	13.001 DPT_Value_4_Count
2	19	36	53	70	87	Ch.x - B object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
3	20	37	54	71	88	Ch.x - C sequence	On/Off	Sends On/Off commands associated with C object of the sequence	1.001 DPT_Switch
3	20	37	54	71	88	Ch.x - C object 1 bit value	1/0 value	Sends values 1/0 associated with C object	1.002 DPT_Bool
3	20	37	54	71	88	Ch.x - C object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with C object	2.001 DPT_Switch_Control
3	20	37	54	71	88	Ch.x - C object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with C object	5.010 DPT_Value_1_Ucount
3	20	37	54	71	88	Ch.x - C object 1 byte value	Signed value	Sends signed values (-128..127) associated with C object	6.010 DPT_Value_1_Count
3	20	37	54	71	88	Ch.x - C object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with C object	5.001 DPT_Scaling

3	20	37	54	71	88	Ch.x - C object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
3	20	37	54	71	88	Ch.x - C object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with C object	7.001 DPT_Value_2_Ucount
3	20	37	54	71	88	Ch.x - C object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with C object	8.001 DPT_Value_2_Count
3	20	37	54	71	88	Ch.x - C object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with C object	12.001 DPT_Value_4_Ucount
3	20	37	54	71	88	Ch.x - C object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with C object	13.001 DPT_Value_4_Count
3	20	37	54	71	88	Ch.x - C object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
4	21	38	55	72	89	Ch.x - D sequence	On/Off	Sends On/Off commands associated with D object of the sequence	1.001 DPT_Switch
4	21	38	55	72	89	Ch.x - D object 1 bit value	1/0 value	Sends values 1/0 associated with object D	1.002 DPT_Bool
4	21	38	55	72	89	Ch.x - D object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with object D	2.001 DPT_Switch_Control
4	21	38	55	72	89	Ch.x - D object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with D object	5.010 DPT_Value_1_Ucount
4	21	38	55	72	89	Ch.x - D object 1 byte value	Signed value	Sends signed values (-128..127) associated with D object	6.010 DPT_Value_1_Count
4	21	38	55	72	89	Ch.x - D object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with D object	5.001 DPT_Scaling

4	21	38	55	72	89	Ch.x - D object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
4	21	38	55	72	89	Ch.x - D object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with D object	7.001 DPT_Value_2_Ucount
4	21	38	55	72	89	Ch.x - D object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with D object	8.001 DPT_Value_2_Count
4	21	38	55	72	89	Ch.x - D object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with D object	12.001 DPT_Value_4_Ucount
4	21	38	55	72	89	Ch.x - D object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with D object	13.001 DPT_Value_4_Count
4	21	38	55	72	89	Ch.x - D object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
5	22	39	56	73	90	Ch.x - E sequence	On/Off	Sends On/Off commands associated with E object of the sequence	1.001 DPT_Switch
5	22	39	56	73	90	Ch.x - E object 1 bit value	1/0 value	Sends values 1/0 associated with E object	1.002 DPT_Bool
5	22	39	56	73	90	Ch.x - E object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with E object	2.001 DPT_Switch_Control
5	22	39	56	73	90	Ch.x - E object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with E object	5.010 DPT_Value_1_Ucount
5	22	39	56	73	90	Ch.x - E object 1 byte value	Signed value	Sends signed values (-128..127) associated with E object	6.010 DPT_Value_1_Count
5	22	39	56	73	90	Ch.x - E object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with E object	5.001 DPT_Scaling

5	22	39	56	73	90	Ch.x - E object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
5	22	39	56	73	90	Ch.x - E object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with E object	7.001 DPT_Value_2_Ucount
5	22	39	56	73	90	Ch.x - E object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with E object	8.001 DPT_Value_2_Count
5	22	39	56	73	90	Ch.x - E object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with E object	12.001 DPT_Value_4_Ucount
5	22	39	56	73	90	Ch.x - E object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with E object	13.001 DPT_Value_4_Count
5	22	39	56	73	90	Ch.x - E object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
6	23	40	57	74	91	Ch.x - F sequence	On/Off	Sends On/Off commands associated with F object of the sequence	1.001 DPT_Switch
6	23	40	57	74	91	Ch.x - F object 1 bit value	1/0 value	Sends values 1/0 associated with F object	1.002 DPT_Bool
6	23	40	57	74	91	Ch.x - F object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with F object	2.001 DPT_Switch_Control
6	23	40	57	74	91	Ch.x - F object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with F object	5.010 DPT_Value_1_Ucount
6	23	40	57	74	91	Ch.x - F object 1 byte value	Signed value	Sends signed values (-128..127) associated with F object	6.010 DPT_Value_1_Count
6	23	40	57	74	91	Ch.x - F object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with F object	5.001 DPT_Scaling

6	23	40	57	74	91	Ch.x - F object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
6	23	40	57	74	91	Ch.x - F object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with F object	7.001 DPT_Value_2_Ucount
6	23	40	57	74	91	Ch.x - F object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with F object	8.001 DPT_Value_2_Count
6	23	40	57	74	91	Ch.x - F object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with F object	12.001 DPT_Value_4_Ucount
6	23	40	57	74	91	Ch.x - F object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with F object	13.001 DPT_Value_4_Count
6	23	40	57	74	91	Ch.x - F object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
7	24	41	58	75	92	Ch.x - G sequence	On/Off	Sends On/Off commands associated with G object of the sequence	1.001 DPT_Switch
7	24	41	58	75	92	Ch.x - G object 1 bit value	1/0 value	Sends values 1/0 associated with G object	1.002 DPT_Bool
7	24	41	58	75	92	Ch.x - G object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with G object	2.001 DPT_Switch_Control
7	24	41	58	75	92	Ch.x - G object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with G object	5.010 DPT_Value_1_Ucount
7	24	41	58	75	92	Ch.x - G object 1 byte value	Signed value	Sends signed values (-128..127) associated with G object	6.010 DPT_Value_1_Count
7	24	41	58	75	92	Ch.x - G object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with G object	5.001 DPT_Scaling

7	24	41	58	75	92	Ch.x - G object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
7	24	41	58	75	92	Ch.x - G object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with G object	7.001 DPT_Value_2_Ucount
7	24	41	58	75	92	Ch.x - G object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with G object	8.001 DPT_Value_2_Count
7	24	41	58	75	92	Ch.x - G object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with G object	12.001 DPT_Value_4_Ucount
7	24	41	58	75	92	Ch.x - G object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with G object	13.001 DPT_Value_4_Count
7	24	41	58	75	92	Ch.x - G object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
8	25	42	59	76	93	Ch.x - H sequence	On/Off	Sends On/Off commands associated with H Object of the sequence	1.001 DPT_Switch
8	25	42	59	76	93	Ch.x - H object 1 bit value	1/0 value	Sends values 1/0 associated with H object	1.002 DPT_Bool
8	25	42	59	76	93	Ch.x - H object 2 bits value	On/Off forced positioning	Sends values 1/0 associated with H object	2.001 DPT_Switch_Control
8	25	42	59	76	93	Ch.x - H object 1 byte value	Unsigned value	Sends unsigned values (0..255) associated with H object	5.010 DPT_Value_1_Ucount
8	25	42	59	76	93	Ch.x - H object 1 byte value	Signed value	Sends signed values (-128..127) associated with H object	6.010 DPT_Value_1_Count
8	25	42	59	76	93	Ch.x - H object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with H object	5.001 DPT_Scaling

8	25	42	59	76	93	Ch.x - H object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/precomfort /economy/off)	20.102 DPT_HVACMode
8	25	42	59	76	93	Ch.x - H object 2 bytes value	Unsigned value	Sends unsigned values (0..65535) associated with H object	7.001 DPT_Value_2_Ucount
8	25	42	59	76	93	Ch.x - H object 2 bytes value	Signed value	Sends signed values (-32768..32767) associated with H object	8.001 DPT_Value_2_Count
8	25	42	59	76	93	Ch.x - H object 4 bytes value	Unsigned value	Sends unsigned values (0.. 4294967295) associated with H object	12.001 DPT_Value_4_Ucount
8	25	42	59	76	93	Ch.x - H object 4 bytes value	Signed value	Sends signed values (-2147483648.. 2147483647) associated with H object	13.001 DPT_Value_4_Count
8	25	42	59	76	93	Ch.x - H object 14 bytes value	ISO 8859-1 characters	Sends characters codified with ISO 8859-1 standard	16.001 DPT_String_8859_1
157						Actuator movement signalling	Up/down	Movement in progress signalling	1.008 DPT_UpDown
158						Actuator position signalling	% Value	Percentage position signalling	5.001 DPT_Scaling
159						Actuator louvres position signalling	% Value	Louvres percentage position signalling	5.001 DPT_Scaling

➢ *15.1.3 Communication objects with parameter setting functions*

The following table contains all the objects with a bus parameter setting function:

Communication object no.	Object name	Object function	Description	Datapoint type
160	Actuator travel limits enabling	Enable/disable	Enables/disables load travel limitation	1.003 DPT_Enable
161	Actuator minimum travel limit	% Value	Sets the % position of the upper travel limit	5.001 DPT_Scaling
162	Actuator maximum travel limit	% Value	Sets the % position of the upper travel limit	5.001 DPT_Scaling
163	Actuator scenes storing enabling	Enable/disable	Scene learning enable/disable	1.003 DPT_Enable

Punto di contatto indicato in adempimento ai fini delle direttive e regolamenti UE applicabili:
Contact details according to the relevant European Directives and Regulations:
GEWISS S.p.A. Via A.Volta, 1 IT-24069 Cenate Sotto (BG) Italy tel: +39 035 946 111 E-mail: qualitymarks@gewiss.com

SAT

+39 035 946 111
8.30 - 12.30 / 14.00 - 18.00
lunedì ÷ venerdì - monday ÷ friday

24h **+39 035 946 260**

sat@gewiss.com
www.gewiss.com