KNX 6-channel touch push-button panel module

GW10746

Technical Manual

Contents

1 Introduction

This manual explains the functions for the devices “**KNX 6-channel touch push-button panel module**” (GW10746), and how they are set and configured with the aid of the ETS configuration software.

2 Application

The KNX 6-channel touch push-button panel module is a command device with 6 channels that can be used individually or combined.
The touch detection areas consist of capacitive sensors positioned under the glass surface, one for each implemented channel. The device has a temperature sensor that acts as a temperature probe, and a buzzer to signal touching. Each channel has two light signalling LEDs - one amber and one blue.

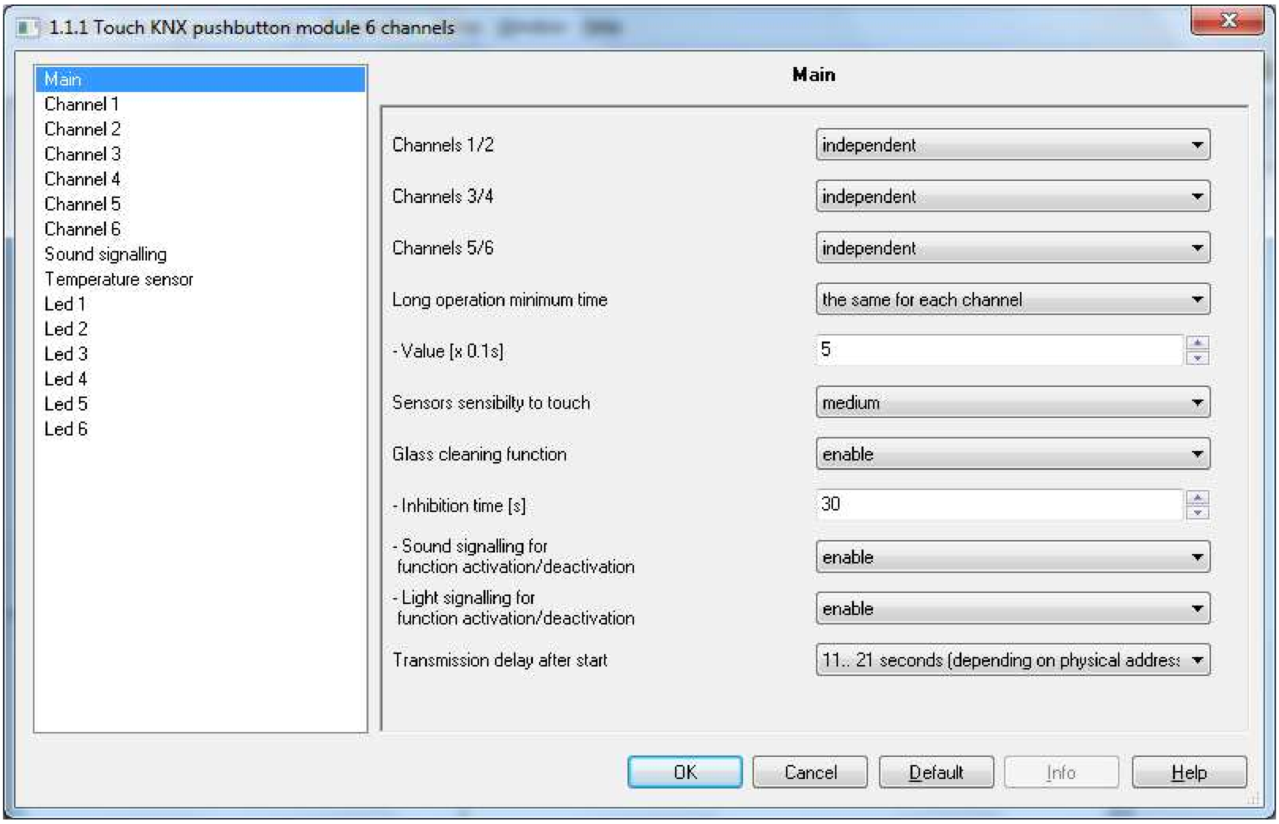
A function can be associated with each channel by means of a specific parameter, as described below.

2.1 Association limits

The maximum number of communication objects available is 143.
The maximum number of associations that the device can store is 212.
The maximum number of group addresses is 212.

3 *“Main”* menu

The **Main** menu contains the application parameters relating to the 6 channels activated from the device (fig. 3.1).



1.1.1 Touch KNX pushbutton module 6 channels

Main

Channels 1/2	independent
Channels 3/4	independent
Channels 5/6	independent
Long operation minimum time	the same for each channel
- Value [x 0.1s]	5
Sensors sensibilty to touch	medium
Glass cleaning function	enable
- Inhibition time [s]	30
- Sound signalling for function activation/deactivation	enable
- Light signalling for function activation/deactivation	enable
Transmission delay after start	11.. 21 seconds (depending on physical addres:

Fig. 3.1

Each of the 6 channels activated from the push-button panel can be autonomously managed (an autonomous function) or combined with another one to produce a single, common function; parameters **“Channels 1/2”**, **“Channels 3/4”** and **“Channels 5/6”** define the channel management. The values that can be set are:

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

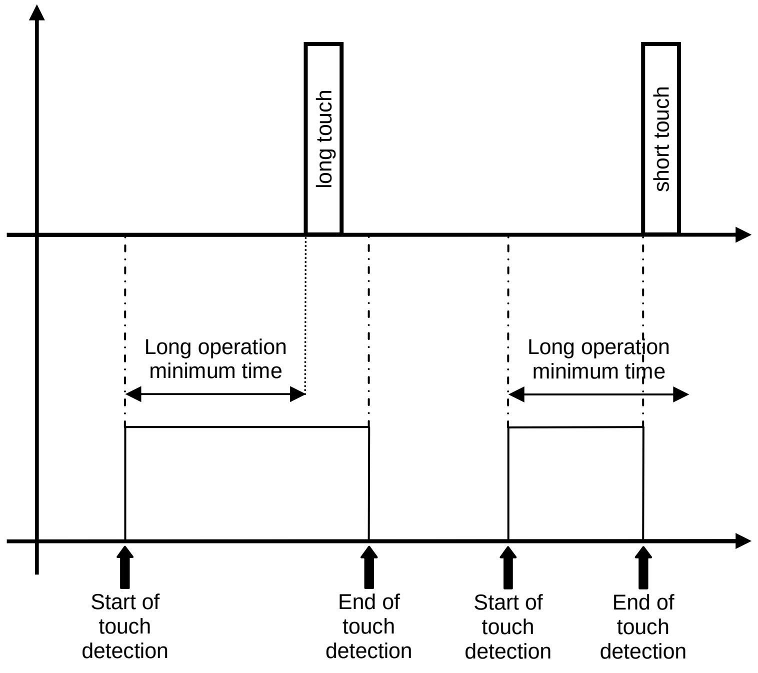
The database structure will vary depending on the values set for the above parameters: an independent setting menu for each channel if the set value is **independent**, or a common menu if the set value is **coupled**.
Many of the functions that the independent or coupled channels can perform require differentiation between short and long operations. With the **“Long operation minimum time”** parameter, you can either define a single duration value for all the channels, or a different one for each channel. The values that can be set are:

- **the same for each channel (default value)**
- diversity each channel

Depending on the selected value, the following parameters will be visualised: **“Value [x 0.1s]”** (if **the same for each channel** value is set) 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 the **diversity every channel** value is set). These parameters establish the minimum effective time in which the capacitive sensor must detect the touch and differentiate between a long or short operation in the capacitive area associated with that channel. The possible values are:

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

The following example shows the meaning of the above parameters



3.1 Parameters

➢ 3.1.1 Sensors sensibility to touch

This parameter defines the level of sensibility of the capacitive sensor associated with the channel. The parameter may assume the following values:

- Low
- **Medium (default value)**
- High

➢ 3.1.2 Glass cleaning function

This parameter enables the inhibition of the capacitive sensors to allow the cleaning of the glass surface. The values that can be set are:

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

If **enable** is selected, the following parameters will be visualised: **“Inhibition time [s]”**, **“Sound signalling for function activation/deactivation”**, **“Light signalling for function activation/deactivation”**, along with the ***Glass cleaning*** communication item.

The **“Inhibition time [s]”** parameter allows you to define the sensor inhibition period. The values that can be set are:

- from 10 to 120, **30 (default value)**

The exact moment of the cleaning function activation and deactivation can be signalled to the user via a special sound effect (used only with this function); the **“Sound signalling for function activation/deactivation”** parameter enables this sound signal. The values that can be set are:

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

In the same way, the exact moment of the cleaning function activation and deactivation can be signalled to the user via a special light effect (used only with this function) that involves the LEDs of all the channels; the

“Light signalling for function activation/deactivation” parameter enables this signal. The values that can be set are:

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

There are 2 ways of activating the glass cleaning function:

- manual activation
  Enabling
  Simultaneous touch of CH2 and CH5 (regardless of the order) for > 3 seconds
  Brief beep
  Simultaneous blinking (blue) of CH2 and CH5 LEDs (1 second ON, 1 second OFF)
  No time-out

  Disabling
  Simultaneous touch of CH2 and CH5 (regardless of the order) for > 3 seconds, or time-out 20 seconds
  Brief beep
  Stopping, blinking, and possible resetting of previous signals

- BUS commands:
  via the ***Glass cleaning*** communication item, it is possible to launch the timed activation of the cleaning function via a BUS command with a logic value of “1” (the function autonomously deactivates when the time set in the **“Inhibition time [s]”** parameter has elapsed) or via a BUS command with a logic value of “0”. The arrival of a BUS command with a logic value of “1” and active timing leads to the resetting of the sensor inhibition time.

➢ *3.1.3 Transmission delay after start*

This allows you to define the time that must pass before the device can transmit the telegrams on the BUS, following a BUS voltage drop/reset.
The values that can be set are:

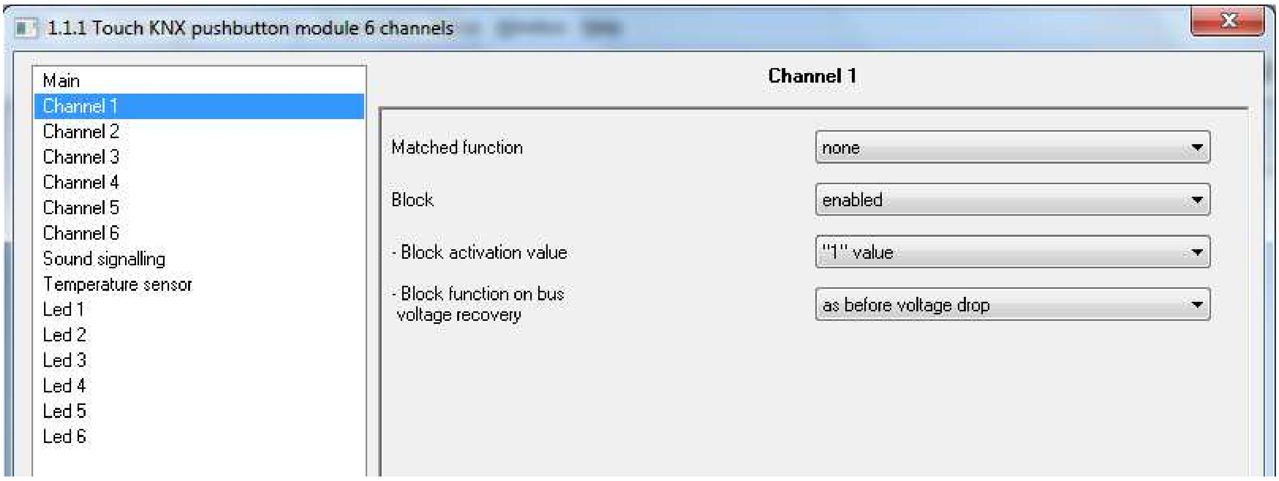
- **11.. 21 seconds (depending on physical address) (default value)**
- 5.. 9 seconds
- 11 seconds
- 13 seconds
- 15 seconds
- 17 seconds
- 19 seconds
- 21 seconds
- no delay

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.

The basic structure of the menu is as follows (fig. 4.1):



Channel 1

Matched function: none

Block: enabled

- Block activation value: "1" value

- Block function on bus voltage recovery: as before voltage drop

Fig. 4.1

4.1 Parameters

4.1.1 Matched function

This determines the function associated to the generic channel x; based on the value set with this parameter, the ***Channel x*** menu will appear differently. The values that can be set are:

- ***none***

No function is associated with the generic channel, therefore it is disabled.

- ***edges/sequence commands***

See chapter 5 - *“**Edges/Sequence commands**”* **function**

- ***1 push button + stop dimmer***

See chapter 6 - *“**1 push button + stop dimmer**”* **function**

- ***cyclic sending 1 push button dimmer***

See chapter 7 - *“**Cyclic sending 1 push button dimmer**”* **function**

- ***1 push button shutter control***

See chapter 8 - *“**1 push button shutter control**”* **function**

- ***scene management***

See chapter 9 - *“**Scene management**”* **function**

- ***switching sequences***

See chapter 10 - *“**Switching sequences**”* **function**

4.1.2 Block

This inhibits the detection of the touch/release of the capacitive sensors, thereby preventing the device from sending - via the BUS - the telegrams associated with those events; if it is activated, any possible status variation will not be interpreted until a block deactivation command is received. The parameter may have the following values:

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

If **enabled** is selected, the following parameters will be visualised: **“Block activation value”** and **“Block function on bus voltage recovery”**, along with the ***Ch.x - Block*** communication item, via which it is possible to activate the function by means of the 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:

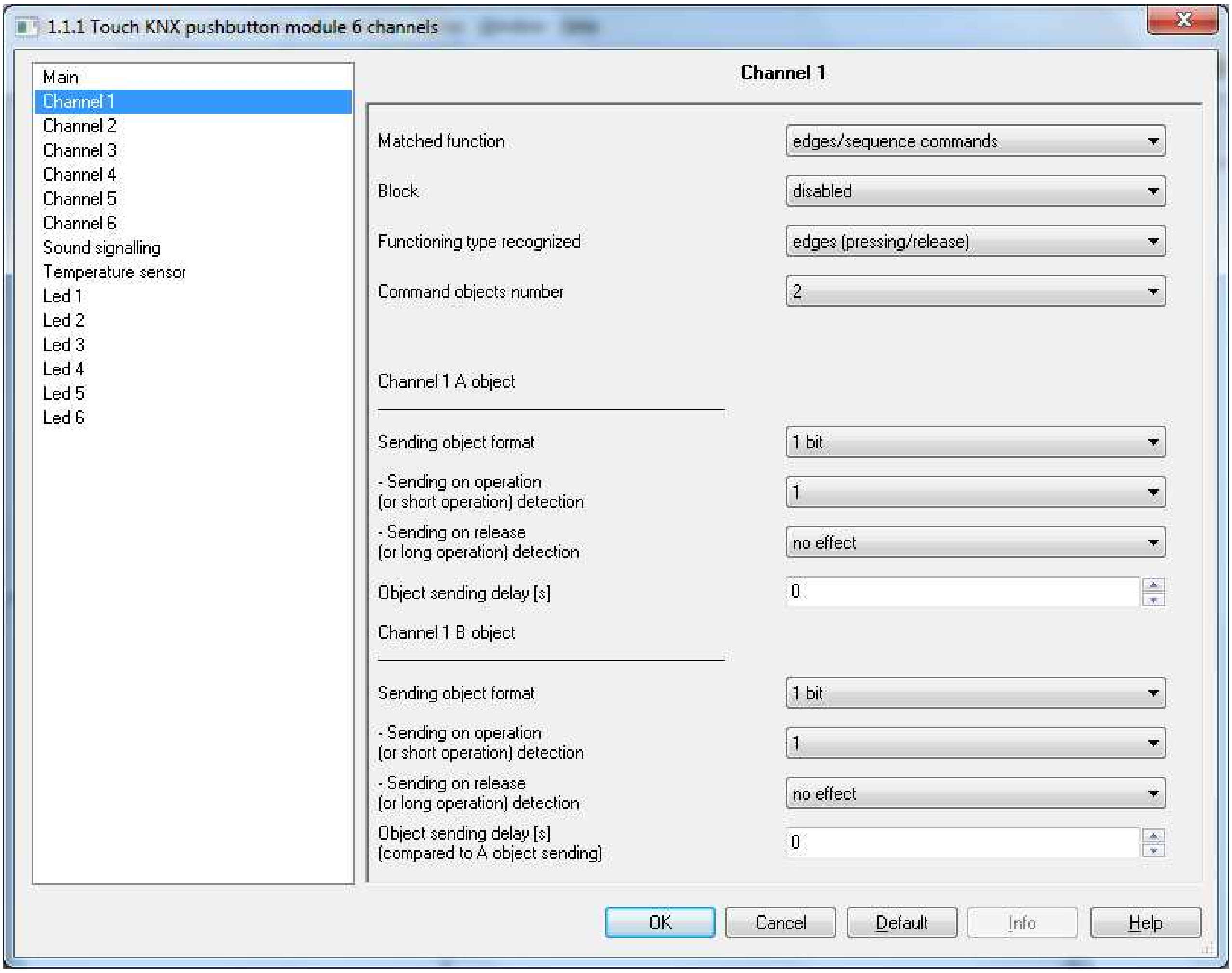
- value “0”
- **value “1” (default value)**

With the **“Block function on bus voltage recovery”** parameter, you can set the status of the block function when the bus voltage is restored. The values that can be set are:

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

5 *"Edges/sequence commands"* function

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 according to the event detected (touch and release), and delay the sending of commands with a settable fixed time.
The basic structure of the menu is as follows (Fig. 5.1):



Channel 1

Matched function: edges/sequence commands
Block: disabled
Functioning type recognized: edges (pressing/release)
Command objects number: 2

Channel 1 A object

Sending object format: 1 bit
- Sending on operation (or short operation) detection: 1
- Sending on release (or long operation) detection: no effect
Object sending delay [s]: 0

Channel 1 B object

Sending object format: 1 bit
- Sending on operation (or short operation) detection: 1
- Sending on release (or long operation) detection: no effect
Object sending delay [s] (compared to A object sending): 0

Fig. 5.1

5.1 Parameters

➢ *5.1.1 Functioning type recognized*

Used to define which type of activation performed on the capacitive sensor must generate the sending of the sequence commands. The values that can be set are:

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

➢ *5.1.2 Command objects number*

Used to define how many communication objects you want to manage with the channel in question. The values that can be set are:

- **1 (default value)**, 2, …, 8

Depending on the value set, for each of the selected objects the following parameters will appear: **"Sending object format"**, **"Sending on operation (or short operation) detection"**, **"Sending on release (or long**

operation) detection" and **"Object sending delay [s]"**, grouped together in the sub-group **Channel x Object z** (z is the index of the object associated with that channel, somewhere between **A** and **H)**.
The **"Sending object format"** parameter 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
- 1 byte signed
- 1 byte percentage
- 1 byte HVAC mode
- 2 bytes unsigned
- 2 bytes signed
- 4 bytes unsigned
- 4 bytes signed
- 14 bytes

The value set for this item will condition the values that can be set for the parameters **"Sending on operation (or short operation) detection"** and **"Sending on release (or long operation) detection"**.
The **"Sending on operation (or short operation) detection"** parameter is used to set the command or value to be sent following the detection of an operation by the capacitive sensor associated with the channel.
The **"Sending on release (or long operation) detection"** parameter is used to set the command or value to be sent following the detection of an operation release by the capacitive sensor associated with the channel.

– 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** **(touch detection default value)**
  - cyclical switching

  Selecting the value **cyclical switching**, the parameter **"Status feedback object"** will be shown, which makes it possible to enable and display the communication object ***Ch.x - z object status feedback***; 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 detection default value)**
  - **activate on (down) forcing** **(touch detection default 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 detection default value)**
  value sending **(touch detection 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 detection default value)**
- **value sending** **(touch detection 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 detection default value)**
- **value sending** **(touch detection 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** **(touch default value)**
- pre-comfort
- economy
- off (building protection)
- cyclical switching (thermostat)
- cyclical switching (chronothermostat)

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 **cyclical switching (thermostat)**, each time the associated event (touch/release) is detected, the device will send a new thermoregulation mode command (HVAC), in the order *Comfort→ Pre-comfort→ Economy→ Off→ Comfort* …; by selecting **cyclical switching (chronothermostat)**, each time the associated event (touch/release) is detected, the device will send a new thermoregulation mode command (HVAC), in the order *Comfort→ Pre-comfort→ 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** **(touch detection default value)**

  By setting **send value**, 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** **(touch 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** **(touch 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** **(touch 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** **(touch detection default value)**

  By setting **value sending**, it is possible to define the value to be sent via the new displayed parameter **“Value (characters ISO 8859-1)”** which can assume the following values:
  - 14 alphanumeric characters with ISO/IEC coding 8859-1

The **“Object sending delay [s] (0.. 255 seconds)”** parameter sets the delay between the detection of the event associated with the sending of the command, and the effective sending of the command/value 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 in these cases is

calculated from the moment when the command/value associated with the object with the previous index (z-1) is sent, not from the moment of detection of the event that generated the sending (touch/release or short/long 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

If a sequence of commands with delays - activated by the detection of a specific event (touch/release or short/long operation) - is being sent, then the detection of the opposite event will cause the termination of the sending of that sequence, only if at least one of the actions associated with the detection of the latter event is different from **no effect**; otherwise, the command/value sequence will be continue to be sent until the last command/value has been sent.

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

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.

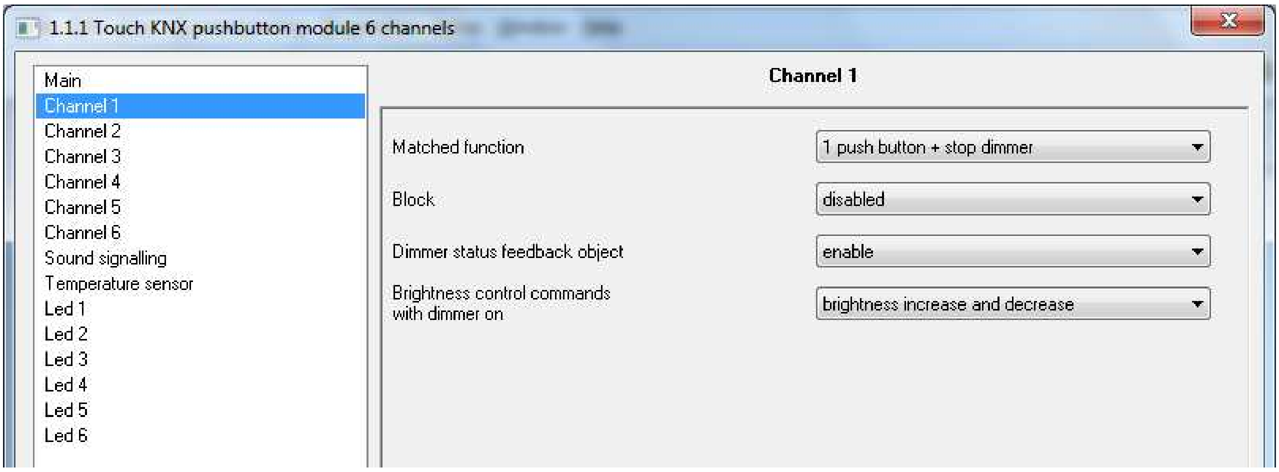
As there is only one button to manage the On/Off and brightness control functions, the operation is managed by differentiating between short operations and long operations:

– 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 at which the brightness reaches the maximum and minimum value does not depend on the commands sent from the touch push-button panel; the latter regulates the brightness itself, stopping the increase/decrease when the required value is reached.

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):



Matched function: 1 push button + stop dimmer
Block: disabled
Dimmer status feedback object: enable
Brightness control commands with dimmer on: brightness increase and decrease

Fig. 6.1

6.1 *Parameters*

The push-button panel makes sure that the command to be sent is the opposite of the last command that was sent, which results as:

- long touch
  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 touch
  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.

➢ *6.1.1 Dimmer status feedback object*

This parameter may have the following values:

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

If **enabled** is selected, the **“Brightness control commands with dimmer on”** parameter is visualised, along with 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.

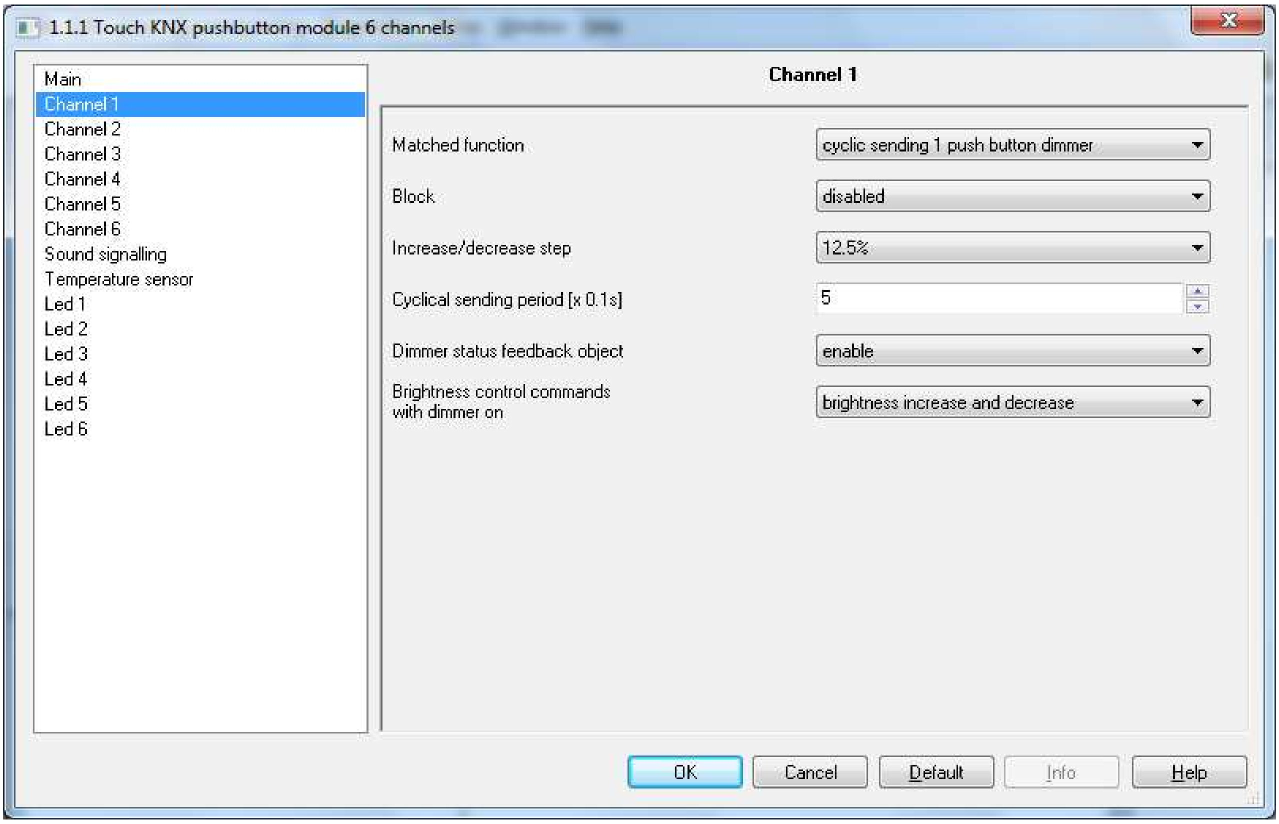
If the feedback object is enabled, each time bus voltage is reset, the device sends a status reading command for this object to update the push-button panel about the status of the connected devices.

7 *“Cyclic sending 1 push button dimmer”* function

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 in the following way: with each touch, the command sent is the opposite of the last one sent. Furthermore, a differentiation is made 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 ***1 push button + stop dimmer***, it is possible to define both the brightness variation steps and the time that must elapse between the sending of one command and another when the long operation is drawn out over time; the sending of the regulation stop telegram at the end of the operation is not therefore necessary, because although the regulation does follow the characteristic power/brightness curve, it is the command 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):



Channel 1

Matched function: cyclic sending 1 push button dimmer
Block: disabled
Increase/decrease step: 12.5%
Cyclical sending period [x 0.1s]: 5
Dimmer status feedback object: enable
Brightness control commands with dimmer on: brightness increase and decrease

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 the operation is maintained, the device sends the command cyclically until it detects the release.

➢ *7.1.2 Cyclical sending period [x 0.1s]*

Used to set the time that must elapse between the sending of one increase/decrease command and another, when the operation 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 for this parameter are:

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

To sum up, when a long operation is detected, the device sends the first increase/decrease command with the set percentage and, if the operation is maintained, it sends the command cyclically until it detects the release.
EXAMPLE: let's suppose we've set a value of ***0.5 sec*** for the item **Long operation minimum time** in the ***Main*** menu, a value of **12.5%** for the **Increase/decrease step** parameter, and a value of **3** (0.3 sec) for the **Cyclical sending period [x 0.1s]** parameter, and the operation is detected:

- 0.5 seconds after the initial detection of the operation, a long operation is detected and so the first 12.5% brightness increase/decrease telegram is sent
- from this moment, for every 0.3 seconds that the operation is maintained, the device sends the 12.5% brightness increase/decrease command again, continuing until it detects the release
- when released, no telegram is sent but the cyclical sending is stopped

➢ *7.1.3 Dimmer status feedback object*

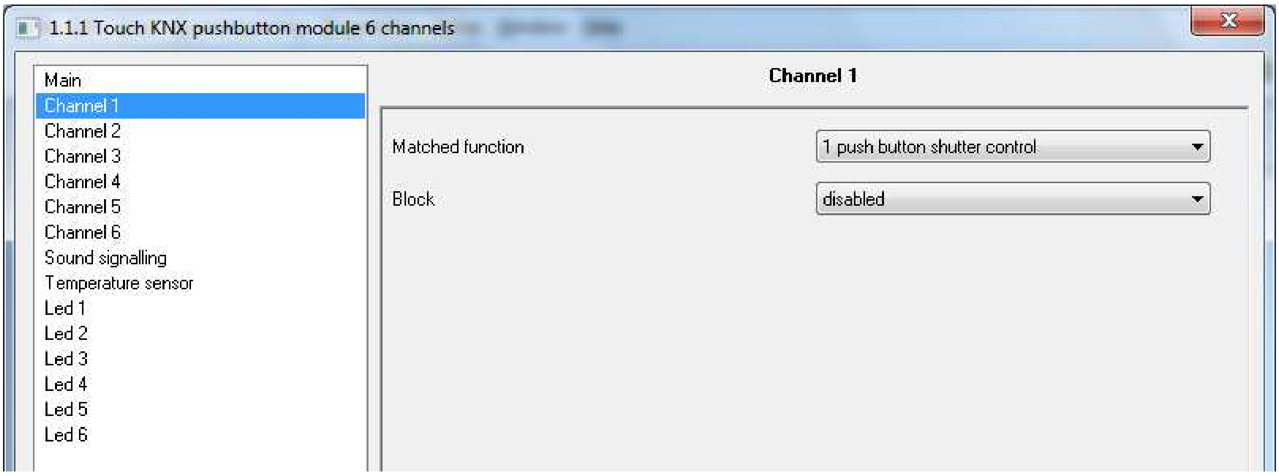
Refer to paragraph 6.1.1 for the settings of this parameter.

8 *“1 push button shutter control”* function

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 louvre 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. There is a difference 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 louvre control command. If the last received movement signal was “up”, the new command will be a closing louvre control command; however, if the last received movement signal was “down”, the new command will be an opening louvre control command. If the shutter is moving, the louvre 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):



Matched function: 1 push button shutter control

Block: disabled

Fig. 8.1

8.1 Parameters

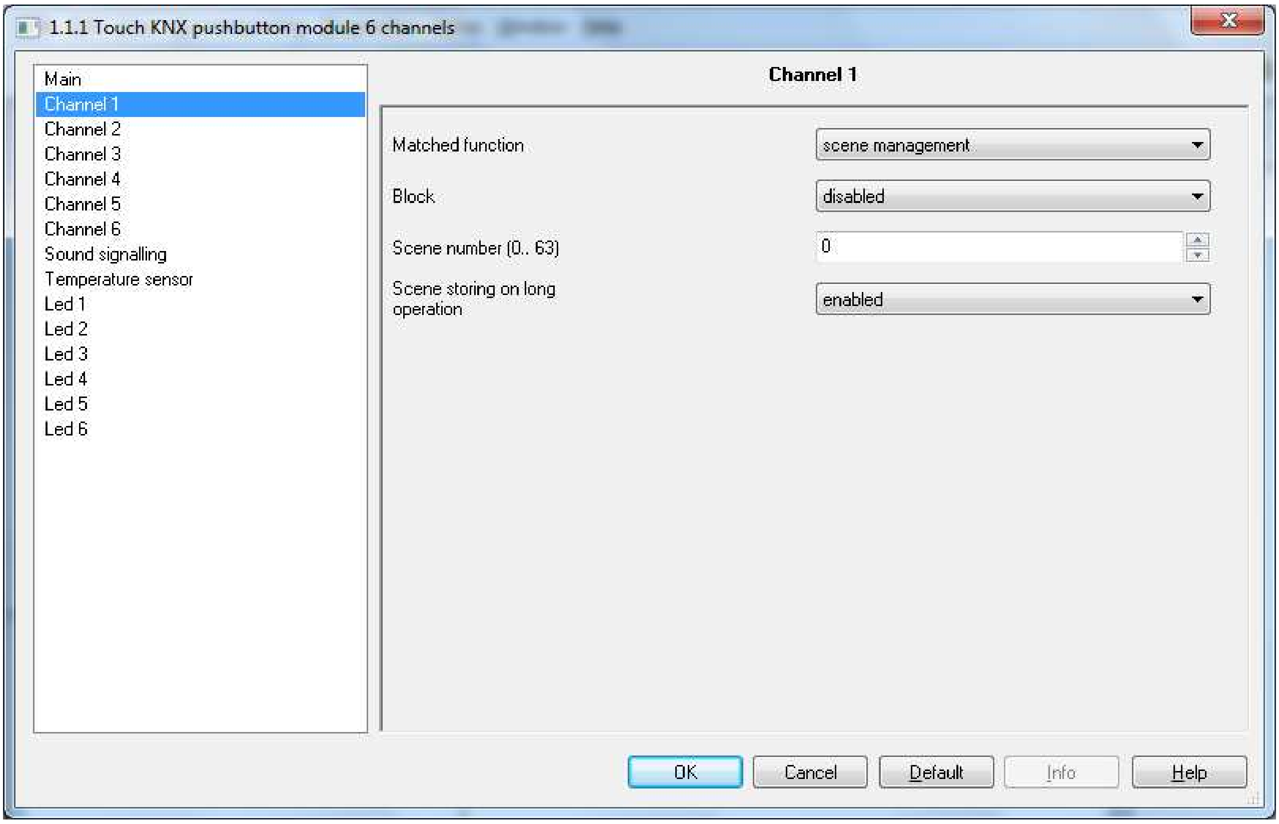
No new parameters are enabled with this function

9 “*Scene management*” function

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

– a long operation is transformed into a scene storing command.
– a short 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***.
The structure of the menu is as follows (Fig. 9.1):



Channel 1

Matched function: scene management
Block: disabled
Scene number (0.. 63): 0
Scene storing on long operation: enabled

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 on long operation

Used to enable the sending of the scene storing command when a long operation is detected. The values that can be set are:

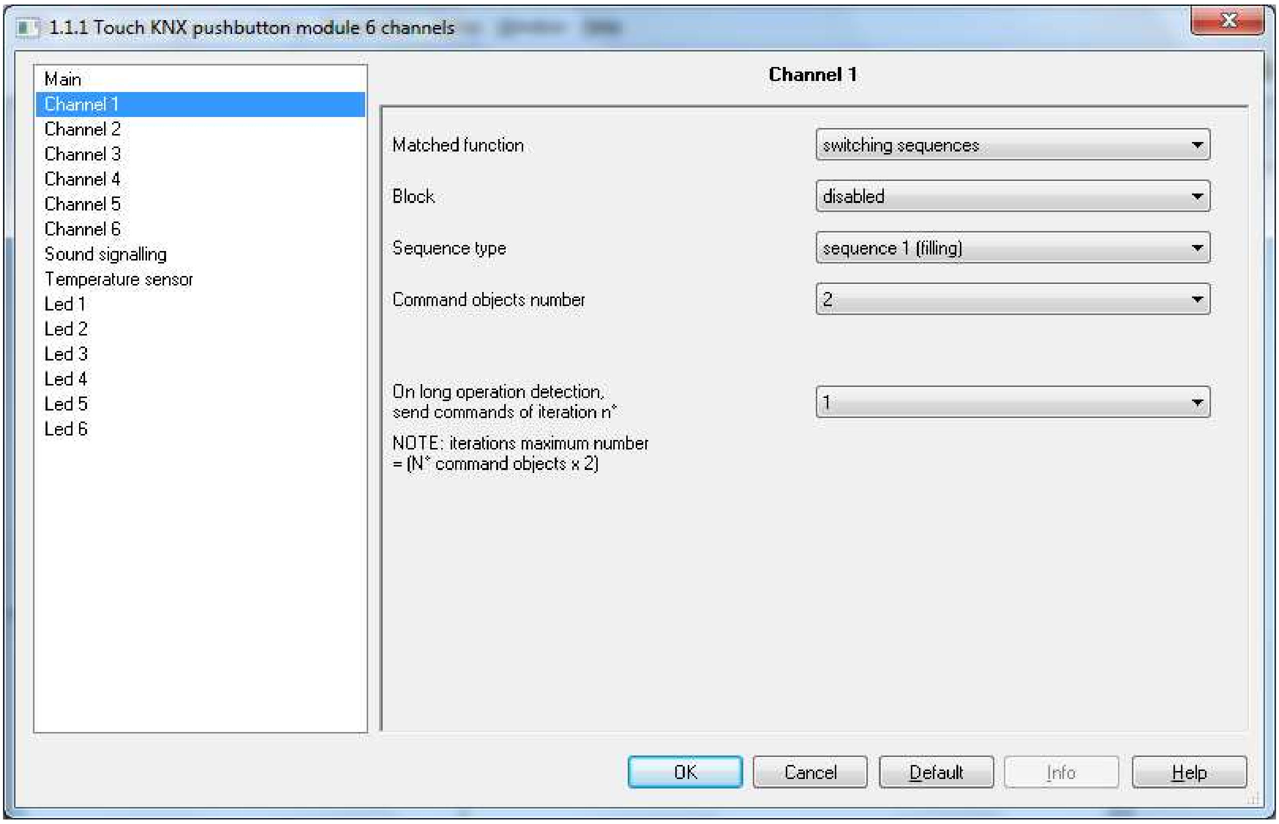
- disabled
- **enabled (default value)**

If **enabled** is selected, the device will send the scene storing command when a long operation is detected; if **disabled** is selected, a long operation is not recognised and only causes the sending of the scene execution command (like the short operation).

Regardless of the value set for the parameter above, it is possible to indirectly generate the sending of the scene storing command following the arrival of a BUS telegram on the object **Ch.x - Scene storing trigger** (both with a value of “1” and with a value of “0”); each time the device receives a telegram on this object, the scene storing telegram will immediately be sent.

10 *“Switching sequences”* function

Used to send a sequence of commands following the detection of a specific operation.
The structure of the menu is as follows (Fig. 9.1):



Matched function: switching sequences
Block: disabled
Sequence type: sequence 1 (filling)
Command objects number: 2
On long operation detection, send commands of iteration n°: 1
NOTE: iterations maximum number = (N° command objects x 2)

Fig. 10.1

10.1 Parameters

➢ 10.1.1 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):
with each operation (edge) detected, the device sends a sequence to the communication object enabled. This sequence follows the filling progress. It 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):
with each operation (edge) detected, the device sends a sequence to the communication object enabled. This sequence follows the sum progress. 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 again from the beginning

The table shows how the trend of the sent commands depends on the count of the detected edge; it begins with the binary code of value 1 and moves up (in this specific case) to the code of value 7 then, from the next edge, the count begins again. 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***.

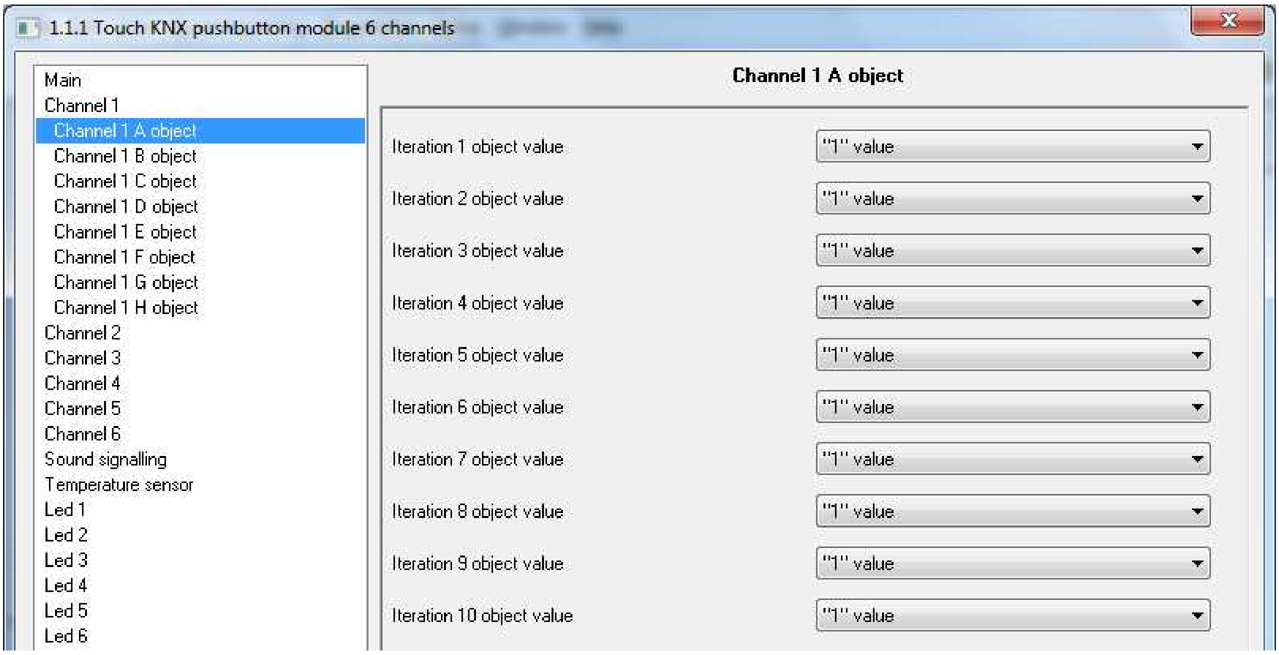
Sequence 3 (random) allows the user to directly set the value for each command for each set edge; this setting enables the parameter **"Number of iterations of the sequence"** and the configuration menu **z object channel x** (one for each enabled command). The parameter **"Number of iterations of the sequence"** 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 **Channel x z object** menu is as follows:



1.1.1 Touch KNX pushbutton module 6 channels

Channel 1 A object

Iteration 1 object value	"1" value
Iteration 2 object value	"1" value
Iteration 3 object value	"1" value
Iteration 4 object value	"1" value
Iteration 5 object value	"1" value
Iteration 6 object value	"1" value
Iteration 7 object value	"1" value
Iteration 8 object value	"1" value
Iteration 9 object value	"1" value
Iteration 10 object value	"1" value

Fig. 10.2

➢ *10.1.2 Commands 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

➢ *10.1.3 On long operation detection, send commands of iteration n°*

Used to define which sequence iteration to send if a long operation is detected. The values that can be set are:

- from 1 to 256, **1 (default value)**

EXAMPLE: with reference to the above tables, let's suppose 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

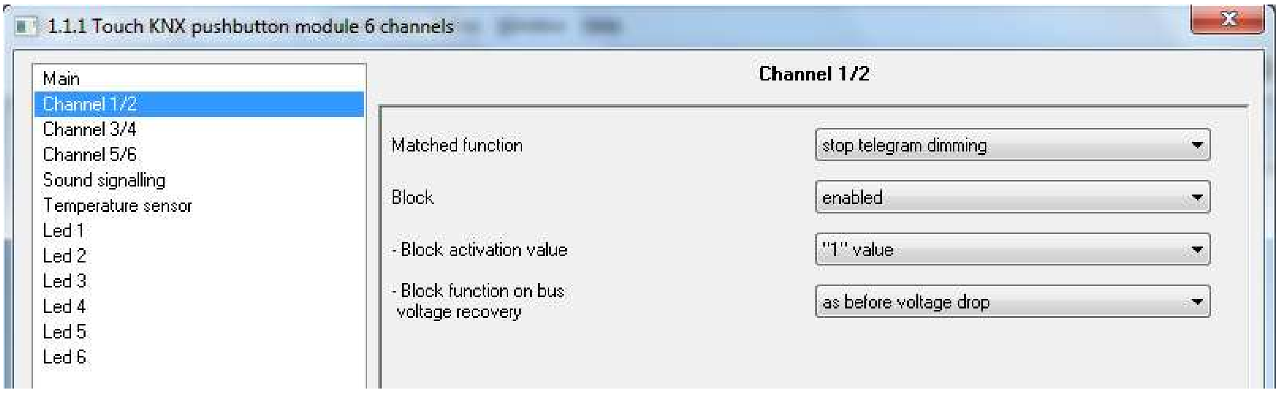
Edge no.	Value sent on *Ch.x - C sequence*	Value sent on *Ch.x - B sequence*	Value sent on *Ch.x - A sequence*
1st edge	0	0	1
2nd edge	0	1	0
3rd edge	0	1	1
4th edge	1	0	0
5th edge	1	0	1
6th edge	1	1	0
7th edge	1	1	1
8th edge	0	0	0

"Sum" sequence

Once a long operation has been detected and the sequence relating to the set iteration has been sent, then when the next short operation is detected, the sequence relating to the iteration immediately after the one associated with the long operation will be sent (in the example given here, the sequence associated with iteration no. 4 will be sent).
To sum up, the value set for the **"On long operation detection, send commands of iteration n°"** parameter defines both the sequence to be sent and the value with which to initialise the iterations counter when a long operation is detected.

11 *"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.
Figure 11.1 shows the basic structure of the menu:



Matched function: stop telegram dimming
Block: enabled
- Block activation value: "1" value
- Block function on bus voltage recovery: as before voltage drop

Fig. 11.1

11.1 Parameters

➢ *11.1.1 Matched function*

Used to determine the function associated with the general x/y channels; depending on the value set with this parameter, the ***Channel x/y*** menu will appear differently. The values that can be set are:

- ***Stop telegram dimming***

See chapter 6 - "***1 push button + stop dimmer***" **function**

- ***Cyclic telegram dimming***

See chapter 7 - "**Cyclic sending 1 push button dimmer**" **function**

- ***Shutter control***

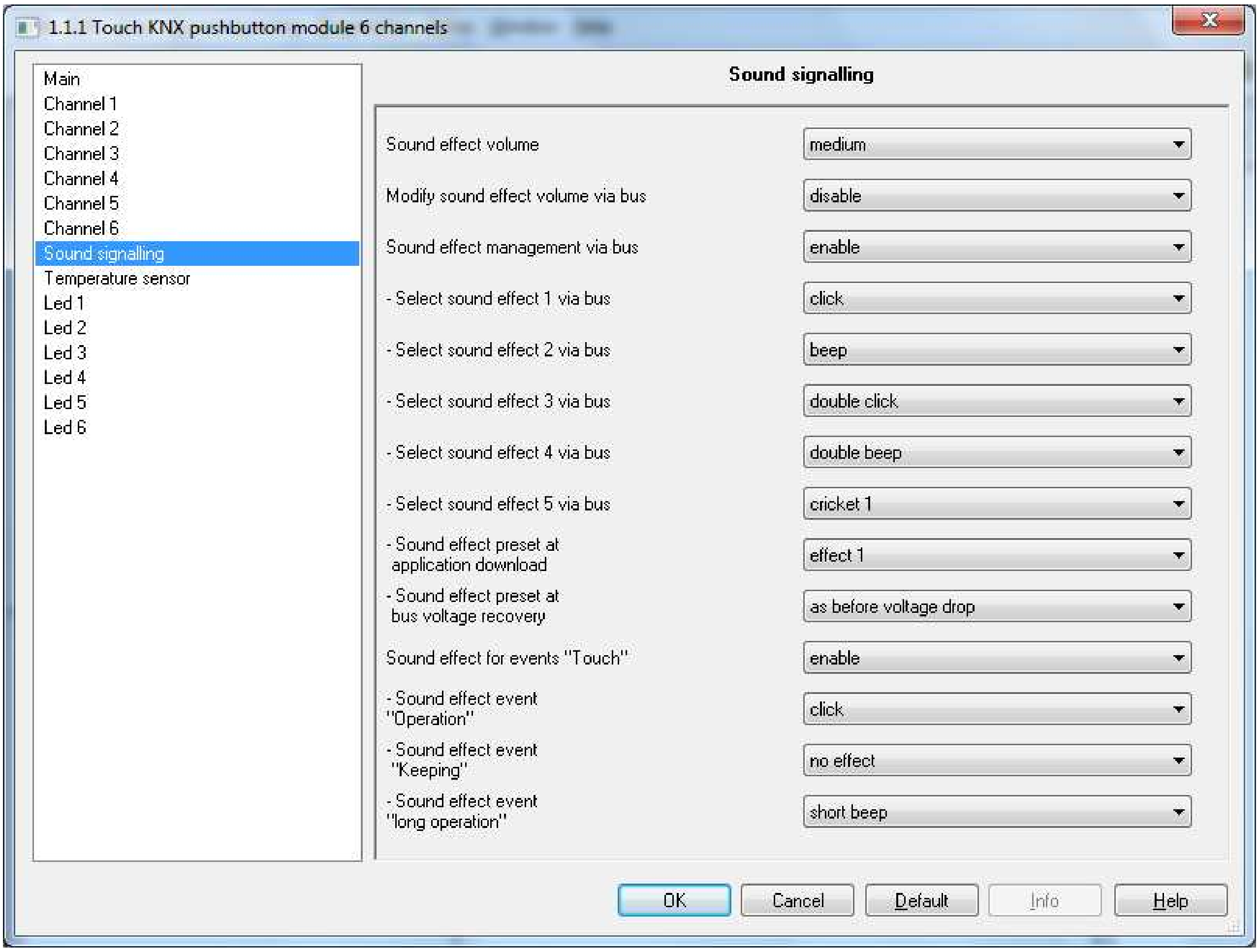
  See chapter 8 - *“**1 push button shutter control**”* **function**

➢ *11.1.2 Block*

Refer to paragraph 4.1.2 for the settings of this parameter.

12 *“Sound signalling”* menu

The device is equipped with an electronic buzzer that can be configured to emit sound signals following BUS commands or touch events. With regards touch events, the configuration of this function is valid for all six channels; it is not possible to differentiate the sound effects associated with each individual channel.
Figure 12.1 shows the basic structure of the menu:



Sound signalling

Sound effect volume	medium
Modify sound effect volume via bus	disable
Sound effect management via bus	enable
- Select sound effect 1 via bus	click
- Select sound effect 2 via bus	beep
- Select sound effect 3 via bus	double click
- Select sound effect 4 via bus	double beep
- Select sound effect 5 via bus	cricket 1
- Sound effect preset at application download	effect 1
- Sound effect preset at bus voltage recovery	as before voltage drop
Sound effect for events "Touch"	enable
- Sound effect event "Operation"	click
- Sound effect event "Keeping"	no effect
- Sound effect event "long operation"	short beep

Fig. 12.1

12.1 Parameters

➢ *12.1.1 Sound effect volume*

Used to define the sound power level. The values that can be set are:

- off
- low
- **medium (default value)**
- high

➢ *12.1.2 Modify sound effect volume via BUS*

Used to enable the ***Volume*** communication object, via which the sound volume can be modified with the BUS command. The values that can be set are:

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

The ***Volume*** communication object receives values between 0% and 100%; given that it is impossible to distinguish clearly volume variations of around 5%, the values received via this object are processed in the following way:

- 0% = off
- from 1% to 35% = low
- from 36% to 70% = medium
- from 71% to 100% = high

➢ *12.1.3 Sound effect management via BUS*

Used to enable the ***Sound effect*** and ***Modify sound effect*** communication objects. The first activates/deactivates the sound effect, and the second jumps forward to the next sound effect (upon receipt of value “1”) or backward to the previous one (upon receipt of value “0”); there are 5 sound effects that can be activated. The values that can be set are:

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

If **enabled** is selected, the two communication objects indicated above are made visible, along with the parameters **“Select sound effect 1 via bus”**, **“Select sound effect 2 via bus”**, **“Select sound effect 3 via bus”**, **“Select sound effect 4 via bus”**, **“Select sound effect 5 via bus”**, **“Sound effect pre-set at application download”** and **“Sound effect pre-set at bus voltage recovery”**.

The parameters **“Select sound effect 1 via bus”**, **“Select sound effect 2 via bus”**, **“Select sound effect 3 via bus”**, **“Select sound effect 4 via bus”** and **“Select sound effect 5 via bus”** allow you to define which sound effects should be activated via bus, thanks to the relevant communication object. The values that can be set are:

- **click (default value)**
- beep
- double click
- double beep
- cricket 1
- cricket 2
- cyclic click
- cyclic beep
- cyclic double click
- cyclic double beep
- cyclic cricket 1
- cyclic cricket 2

If the sound effect is activated via the relevant communication object, then by jumping to the sound effect immediately before or after the current one, this new sound effect will be reproduced without having to receive any new activation commands; this remains true even if the sound effect being reproduced is not cyclically repeated.

The **“Sound effect pre-set at application download”** parameter is used to define the sound effect set by default each time the application parameters are downloaded from ETS; if no telegram is received on the ***Modify sound effect*** object after the downloading of the application parameters, then when the sound signalling is activated from the BUS via the ***Sound effect*** object, the effect reproduced will be the one set in the parameter in question. The values that can be set are:

- **effect 1 (default value)**
- effect 2
- effect 3
- effect 4
- effect 5

The **"Sound effect pre-set at bus voltage recovery"** parameter is used to define the sound effect set by default each time the KNX BUS supply voltage is restored following a blackout; if no telegram is received on the ***Modify sound effect*** object after the BUS voltage has been restored, then when the sound signalling is activated from the BUS via the ***Sound effect*** object, the effect reproduced will be the one set in the parameter in question. The values that can be set are:

- effect 1
- effect 2
- effect 3
- effect 4
- effect 5
- **as before voltage drop (default value)**

➢ *12.1.4 Sound effects for "Touch" events*

Used to enable and visualise the parameters for an association between the sound effect and a touch event detected by the capacitive sensor. The values that can be set are:

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

If **enabled** is selected, the **"Sound effect for event Operation"**, **"Sound effect event Keeping"** and **"Sound effect event long operation"** parameters will be visualised.

The **"Sound effect for event Operation"** parameter allows you to select the sound effect to be associated with the "touch" event detected by the capacitive sensor. The values that can be set are:

- no effect
- **click (default value)**
- beep
- double click
- double beep
- long beep

The **"Sound effect for event Keeping"** parameter allows you to select the sound effect to be associated with the "operation" event detected by the capacitive sensor. The values that can be set are:

- **no effect (default value)**
- click 300 ms
- beep 300 ms
- beep 500 ms
- cricket 1
- cricket 2

The **"Sound effect for event long operation"** parameter allows you to select the sound effect to be associated with the "long operation" event detected by the capacitive sensor. The values that can be set are:

- no effect
- **short beep (default value)**

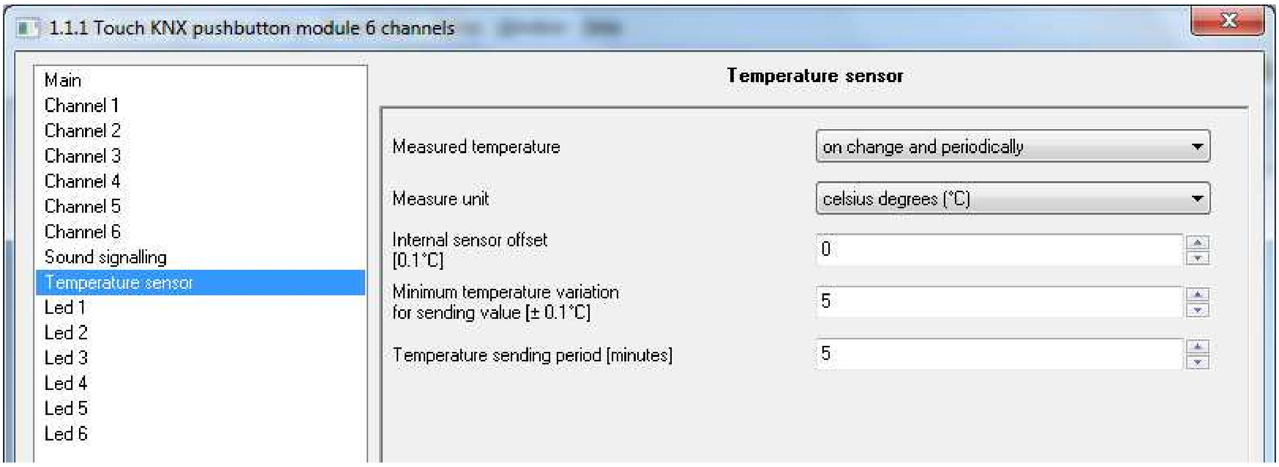
The sound effect associated with Touch events has a higher priority than the active effect (activated by the BUS command) in the moment when the sensor detects the operation; at the end of the reproduction of the effect associated with the current event, a certain time period (not configurable by the user) must pass before the effect activated previously by the BUS command is again reproduced, and continues until it is deactivated by the relative BUS command. The effect activated by the BUS event is therefore “suspended” when a “Touch” event occurs.

The table below shows which sound effects are reproduced, according to the function associated with the channel.

		Touch events		
		Touch	Operation	Long operation
Matched function	No function			
	Edge/command sequence - touch/release	YES		
	Edge/command sequence - short/long operation	YES		YES if at least one action is associated with the long operation
	1 push button + stop dimmer	YES	YES	
	Cyclic sending 1 push button dimmer	YES	YES	
	1 push button shutter control	YES		YES
	Scene management	YES		YES if scene storing is enabled; cannot be deactivated by user
	Switching sequences	YES		YES
	STOP telegram dimming	YES	YES	
	Cyclic telegram dimming	YES	YES	
	Shutter control	YES		YES
	Settable effects	click, beep, double click, double beep, long beep, no effect	click 300 ms, beep 300 ms , beep 500 ms, cricket 1, cricket 2, no effect	short beep, no effect

13 *“Temperature sensor”* menu

This is used to enable and set the conditions for sending the temperature value measured by the sensor located inside the device.
The basic structure of the menu is as follows (Fig. 13.1):



Temperature sensor

Measured temperature: on change and periodically

Measure unit: celsius degrees (°C)

Internal sensor offset [0.1°C]: 0

Minimum temperature variation for sending value [± 0.1°C]: 5

Temperature sending period [minutes]: 5

Fig. 13.1

13.1 Parameters

➢ 13.1.1 Measured temperature

This is used to define the conditions for sending the value of the temperature measured by the probe; The values that can be set are:

- **do not send** **(default value)**
- send on demand only
- send in case of change
- send periodically
- send on change and periodically

By selecting the value **do not send**, no new parameter or communication object will be visible; by selecting any other value, the communication object ***Temperature sensor*** and the parameters **“Measure unit”** and **“Internal sensor offset factor [0.1℃]”** will be displayed. Selecting the value **send in case of change** or **send on change and periodically**, also the parameter **“Minimum temperature variation for sending value [± 0.1℃]”** will be visible, whereas by selecting **send periodically** or **send on change and periodically** the parameter **“Temperature sending period [minutes]”** will be visible.
Selecting the value **send on demand only**, no new parameter will be enabled, as the temperature value is not sent spontaneously by the device; in the case of a status reading request, it sends the requester a telegram in response to the received command, which includes information about the measured temperature value.

➢ 13.1.2 Measure unit

This is used to set the measure unit with which the information will be coded and sent via the communication object ***Temperature sensor***. The values that can be set are:

- **Celsius degrees (℃)** **(default value)**
- Kelvin degrees (°K)

➢ 13.1.3 Internal sensor offset [0.1℃]

This is used to set the correction factor to be applied to the temperature value measured by the probe, to eliminate the contribution of heat generated by the device or the site of installation; The values that can be set are:

- from -20 to + 20, **0 (default value)**

➢ *13.1.4 Minimum temperature variation for sending value [± 0.1℃]*

This is visible if the temperature is sent on change, and is used to define the minimum temperature variation, with respect to the last sent temperature value, which causes the new measured value to be spontaneously sent; the values that can be set are:

- from 1 to 10, **5 (default value)**

➢ *13.1.5 Temperature sending period [minutes]*

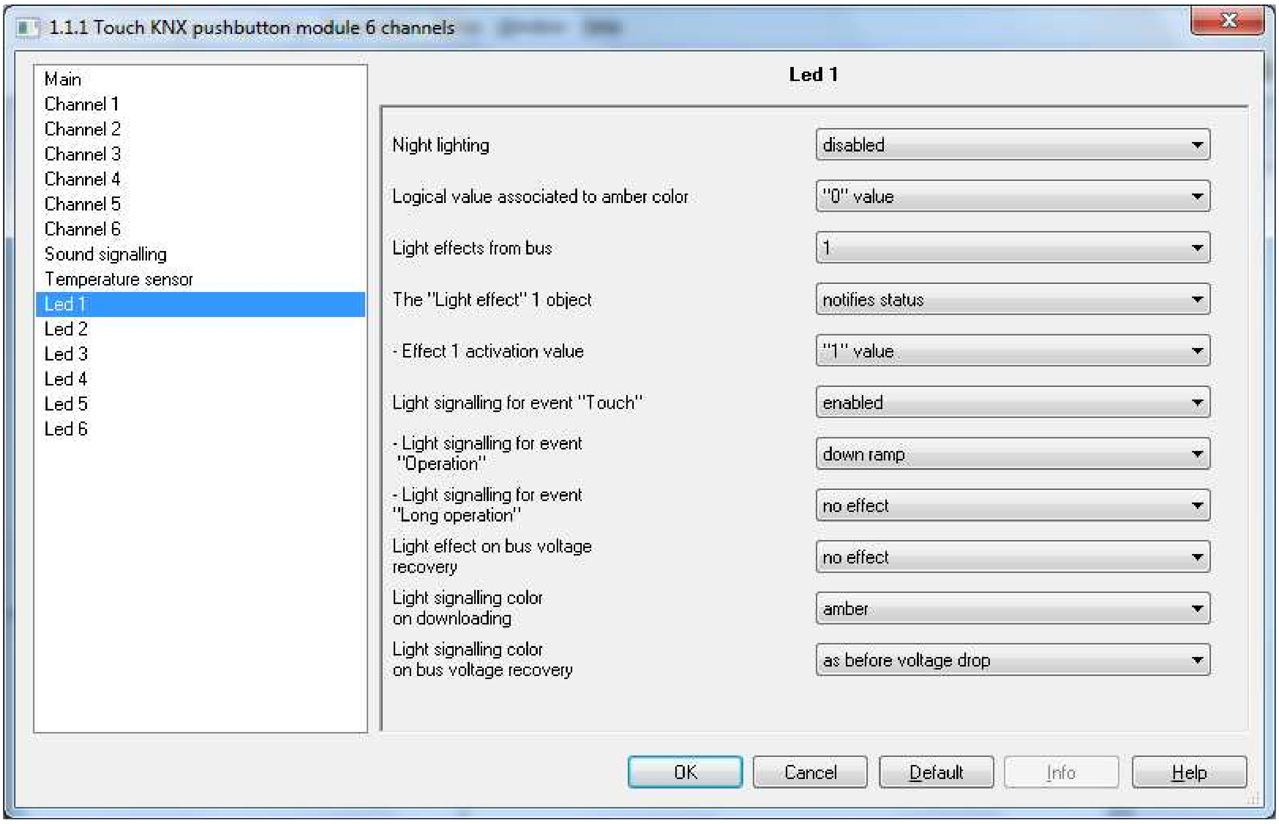
This is visible if the temperature is sent periodically, and is used to define the period after which the measured temperature indication telegrams are sent spontaneously; the values that can be set are:

- from 1 to 255, **5 (default value)**

14 “LED X” menu

This is used to define and personalize the operation of the signalling LEDs associated with the channel.
The signalling LED may be amber or blue. One may be used for the night lighting localisation function, or they may each be managed autonomously by means of the relative communication objects. The "light signalling upon touch/operation/long touch/release" function may be enabled, with different lighting effects that can be preconfigured and selected via the relative configuration parameters. The effects associated with the events detected by the capacitive sensor are performed, even if prior to the touch detection phase (standby) a light effect enabled by a BUS command is active (and is then reactivated at the end of the effect associated with the release). 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***.

Figure 14.1 shows the basic structure of the menu:



Led 1

Night lighting	disabled
Logical value associated to amber color	"0" value
Light effects from bus	1
The "Light effect" 1 object	notifies status
- Effect 1 activation value	"1" value
Light signalling for event "Touch"	enabled
- Light signalling for event "Operation"	down ramp
- Light signalling for event "Long operation"	no effect
Light effect on bus voltage recovery	no effect
Light signalling color on downloading	amber
Light signalling color on bus voltage recovery	as before voltage drop

Fig. 14.1

14.1 Parameters

➢ *14.1.1 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 blue 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"**.

The **“Logical value associated to amber color”** parameter is used to define which logic value received on the ***Led x - color choice*** communication object is associated with the amber color; as a result, the opposite logic value is associated with the blue color. The values that can be set are:

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

➢ *14.1.2 Amber 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)**

➢ *14.1.3 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”**.

Parameters **“The “Light effect” 1 object”**, **“The “Light effect” 2 object”**, **“The “Light effect” 3 object”**, **“The “Light effect” 4 object”** and **“The “Light effect” 5 object”** are used to associate the luminous 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:

- **notified status** **(default value)**
- switches night signalling off (only visible if the blue/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 **personalized effect** is selected, the new configuration menu **Personalize effect y** will be visualised, with 1≤Y≤5 (see paragraph 14.2: “Personalize effect y” menu)

The parameters **“Effect 1 activation value”**, **“Effect 2 activation value”**, **“Effect 3 activation value”**, **“Effect 4 activation value”** and **“Effect 5 activation value”** are used to define which logical 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.

Apart from the lighting effects used for status signalling, it is also possible to associate special effects for signalling the various events detected by the capacitive sensor: *touch*, *operation* (of the touch), *long touch* and *release*. All this allows you to personalize the light signalling so that the various phases listed above can be easily identified directly by the user of the Touch push-button panel. The light effect associated with these events has a higher priority than the effect active at the time when the sensor detects the touch; in addition, these effects do not concern the signalling color change, which remains the same as before the touch detection.

➢ *14.1.4 Light signalling for event “Touch”*

Used to enable the light signals associated with the touch events, and visualise the configuration parameters. The values that can be set are:

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

If **enabled** is selected, the parameters **“Light signalling for event Operation”** and **“Light signalling for event Long operation”** will be visualised.
The **“Light signalling for event Operation”** parameter is used to define the light signalling mode for a “Touch” event. The values that can be set are:

- **down ramp (default value)**
- heartbeat 1
- heartbeat 2
- jellyfish
- blink
- no effect

The **“Light signalling for event Long operation”** parameter is used to define the light signalling mode for a “Long operation” event. The values that can be set are:
- **3 blinkings (default value)**
- no effect

If present, the light effect associated with the “Long operation” effect is only reproduced if the function set on channel x distinguishes between short and long operations; otherwise (no function, edges with sequence commands, switching sequences (sum and filling), local actuator command), this effect will not be reproduced.

The table below shows which light effects are reproduced, according to the function associated with channel x.

		Touch events			
		Touch	Operation	Long operation	Release
Matched function	No function	YES			
	Edge/command sequence - touch/release	YES			YES if at least one action is associated with the release event
	Edge/command sequence - short/long operation	YES		YES if at least one action is associated with the long operation	
	1 push button + stop dimmer	YES	YES		
	Cyclic sending 1 push button dimmer	YES	YES		
	1 push button shutter control	YES		YES	
	Scene management	YES		YES if scene storing is enabled; cannot be deactivated by user	
	Switching sequences	YES		YES	
	STOP telegram dimming	YES	YES		
	Cyclic telegram dimming	YES	YES		
	Shutter control	YES		YES	
	Settable effects	executes the down ramp, executes heartbeat 1, executes heartbeat 2, executes medusa, executes blink, no effect	heartbeat 3, heartbeat 4, quick blinking, slow blinking, very slow blinking, no effect	3 flashes, no effect	executes the down ramp, executes heartbeat 1, executes heartbeat 2, executes medusa, executes blink, no effect

The light effect for signalling the sending of the scene storage command is always reproduced, even if the light signals for Touch events are disabled.

➢ *14.1.5 Light effect on bus voltage recovery*

Makes it possible to set the light signalling effect that is activated with 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 activate, 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.

➢ *14.1.6 Light signalling color on downloading*

This parameter is only visible if night lighting is disabled. It allows you to pre-set the color of the light signalling effect when the application parameters are downloaded via ETS. The values that can be set are:

- **amber (default value)**
- blue

➢ *14.1.7 Light signalling color on bus voltage recovery*

This parameter is only visible if night lighting is disabled. It allows you to pre-set the color of the light signalling effect when the bus voltage is restored. The values that can be set are:

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

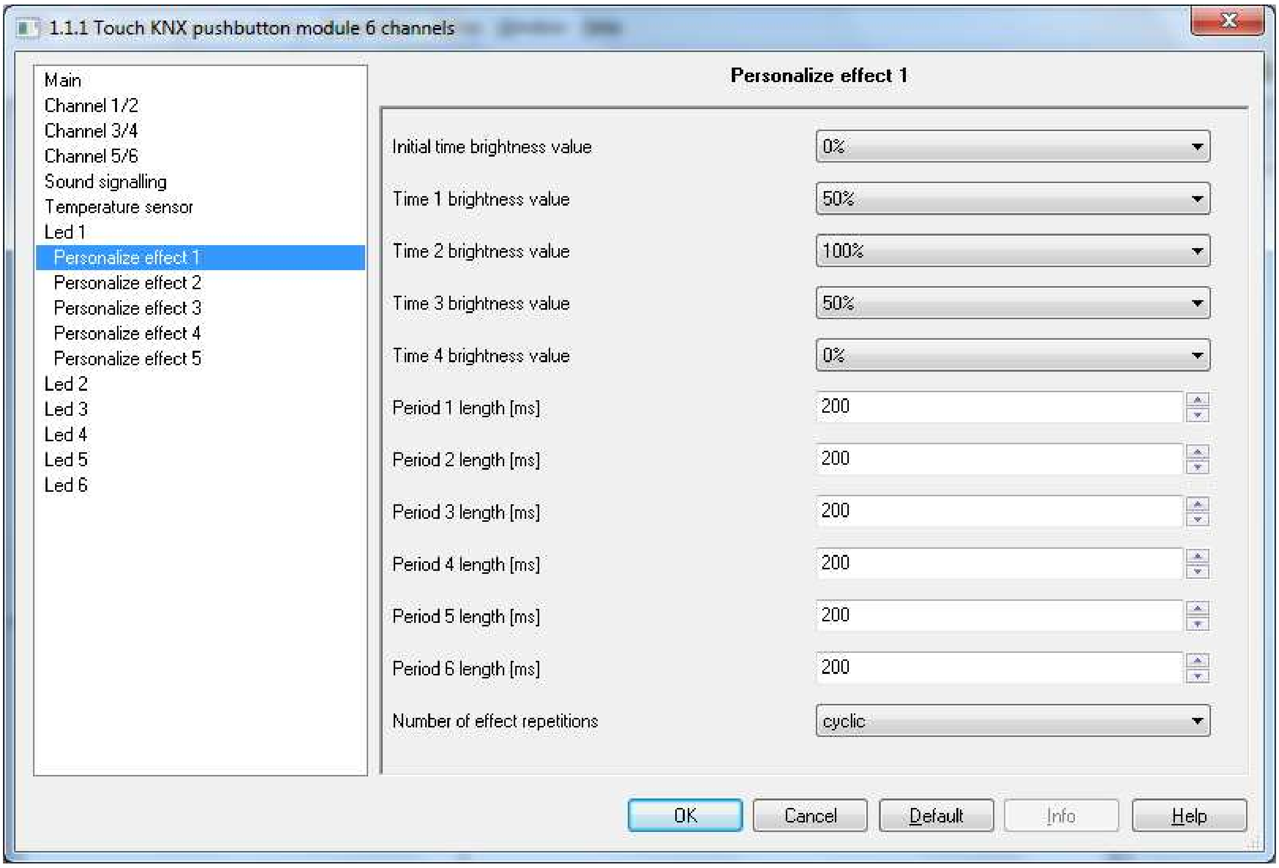
In summary, if night lighting is active, the coloring of the light effects activated by the bus command (via objects ***Led x - Effect 1*** etc.) is associated to a LED not used for signalling; vice versa, if night lighting is disabled, the coloring of these effects depends on the communication object ***Led x - color choice*** and on the parameter value **"Light signalling color on bus voltage recovery"**.
If night lighting is active, in the particular case of an activation command being received during the capacitive sensor detection events (touch, operation, long operation, and release), this effect is not performed until the release phase has ended, but the light signal color immediately becomes the one not associated with night lighting; in the same way, if a light effect is active before the touch detection and is deactivated during one of the events detected by the capacitive sensor, the color of the light signal immediately becomes the one associated with night lighting.

14.2 “Personalize effect y” menu

This menu is visible if the **“Light effect” y object” Px** parameter of the **Led x** menu is set with the **execute personalized effect** value.

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***.

Figure 14.2 shows the basic structure of the menu:

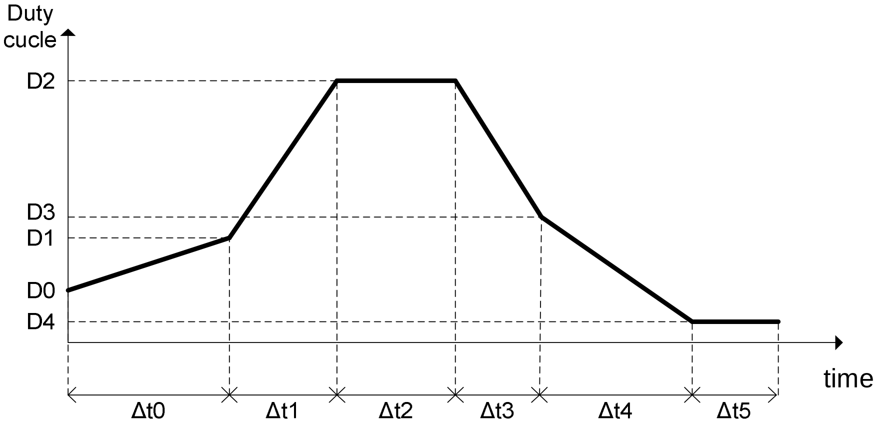


Personalize effect 1

Initial time brightness value	0%
Time 1 brightness value	50%
Time 2 brightness value	100%
Time 3 brightness value	50%
Time 4 brightness value	0%
Period 1 length [ms]	200
Period 2 length [ms]	200
Period 3 length [ms]	200
Period 4 length [ms]	200
Period 5 length [ms]	200
Period 6 length [ms]	200
Number of effect repetitions	cyclic

Fig. 14.2

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:

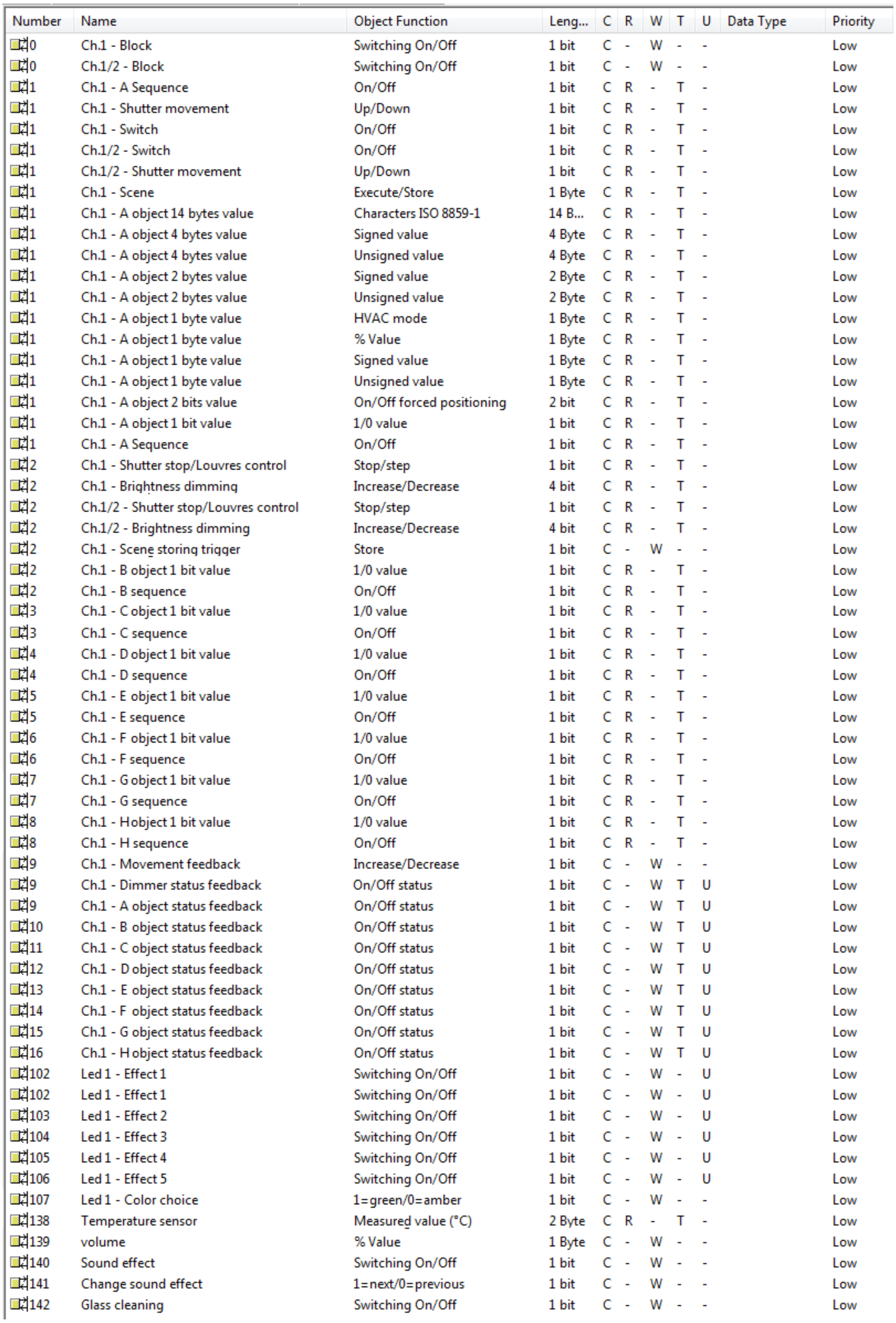
- from 0 to 65535, **200 (default value)**

The **"Number of effect repetitions"** parameter 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)**

15 Communication objects

Fig. 15.1 shows all the communication objects that can be enabled by the ETS database; in this specific case, the image only highlights the objects of channel 1 (independent mode) and channel 1/2 (coupled mode):



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
138	Temperature sensor	Measured value (°C)	2 Byte	C	R	-	T	-		Low
139	volume	% Value	1 Byte	C	-	W	-	-		Low
140	Sound effect	Switching On/Off	1 bit	C	-	W	-	-		Low
141	Change sound effect	1=next/0=previous	1 bit	C	-	W	-	-		Low
142	Glass cleaning	Switching On/Off	1 bit	C	-	W	-	-		Low

Fig. 15.1

15.1 Communication object table

The following tables summarise 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 object	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.1 - 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.1 - 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.1 - 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	108	114	120	126	132	LED X – Effect 1	Switching On /Off	Switching On /Off light effect 1	1.001 DPT_Switch
103	109	115	121	127	133	LED X – Effect 2	Switching On /Off	Switching On /Off light effect 2	1.001 DPT_Switch
104	110	116	122	128	134	LED X – Effect 3	Switching On /Off	Switching On /Off light effect 3	1.001 DPT_Switch
105	111	117	123	129	135	LED X – Effect 4	Switching On /Off	Switching On /Off light effect 4	1.001 DPT_Switch
106	112	118	124	130	136	LED X – Effect 5	Switching On /Off	Switching On /Off light effect 5	1.001 DPT_Switch
107	113	119	125	131	137	LED X - color choice	1=blue/0=amber	Selects the light signalling color	1.001 DPT_Switch
139						Volume	% Value	Receives the volume percentage value of the sound effect	5.001 DPT_Scaling
140						Sound effect	Switching On /Off	Receives the sound effect activation/deactivation commands	1.001 DPT_Switch
141						Change sound effect	1=next/0=previous	Changes the sound effect to be reproduced, jumping to the previous/next effect	1.001 DPT_Switch
142						Glass cleaning	Switching On /Off	Activates/deactivates the sensor inhibition function for cleaning the glass	1.001 DPT_Switch

➢ *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 memorising/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/pre-comfort/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 A 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 A 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 A object	6.010 DPT_Value_1_Count

2	19	36	53	70	87	Ch.X - B object 1 byte value	% Value	Sends the percentage values (0%..100%) associated with A object	5.001 DPT_Scaling
2	19	36	53	70	87	Ch.X - B object 1 byte value	HVAC mode	Sends the HVAC modes (auto/comfort/pre-comfort/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 A 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 A 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 A 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 A 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 A 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 A 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 A 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 A 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/pre-comfort/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 A 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 A 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 A 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 A 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 A object	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 A 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 A 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 A 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/pre-comfort/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 A 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 A 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 A 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 A 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 A 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 A 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 A 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 A 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/pre-comfort/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 A 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 A 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 A 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 A 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 A 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 A 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 A 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 A 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/pre-comfort/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 A 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 A 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 A 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 A 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 A 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 A 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 A 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 A 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/pre-comfort/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 A 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 A 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 A 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 A 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 A 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 A 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 A 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 A 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/pre-comfort/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 A 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 A 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 A 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 A 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
138						Temperature sensor	Measured value (℃)	Sends value measured in ℃	9.001 DPT_Value_Temp
138						Temperature sensor	Measured value (°K)	Sends value measured in °K	9.002 DPT_Value_Tempd